Spherical 4R Linkage Algebraic v_i-v_j Input-output Equations

M. John D. Hayes[1], Mirja Rotzoll[1], and Zachary Copeland[1]

Abstract In this paper the algebraic polynomial equations relating the relative orientations between the six distinct pairs of rigid links in an arbitrary spherical 4R mechanism are derived. First, the forward kinematics transformation matrix of an arbitrary spherical open 4R kinematic chain is computed in terms of its Denavit-Hartenberg parameters, where all angles are converted to their tangent half-angle parameters. This transformation matrix is mapped to its corresponding four non-zero Study soma coordinates. The serial kinematic chain is conceptually closed by equating the forward kinematics transformation to the identity matrix. Gröbner bases and resultants are then used to eliminate the two intermediate joint angle parameters leaving an algebraic polynomial in terms of the selected input and output (IO) joint angle parameters and the four twist angle or link length parameters. This yields six independent algebraic IO Equations. Their utility is demonstrated with two function generator continuous approximate synthesis examples.

Key words: Spherical four-bar linkage, v_i-v_j algebraic input-output equations, continuous approximate synthesis.

1 Introduction

Relative motion between mechanically constrained rigid bodies on the surface of a sphere has fascinated philosophers, mathematicians, and engineers for millennia [3]. The design of predictable motion of a four-bar spherical mechanism appears to have its origins in the development of *universal joints* based on gimbals, which have been investigated since antiquity [10]. Arguably the most significant modern spherical four-bar mechanism is the *Agile Eye* [5], which is used as a camera pointing system. While there is a substantial volume of archival literature regarding spherical 4R

[1] Carleton University, Department of Mechanical and Aerospace Engineering, Ottawa, ON, Canada.

mechanisms, this type of mechanical system still excites the imagination, see [8] for a recent example. Hence, we believe there is sufficient justification to present the work on the derivation of the six v_i-v_j algebraic input-output (IO) equations reported in this paper.

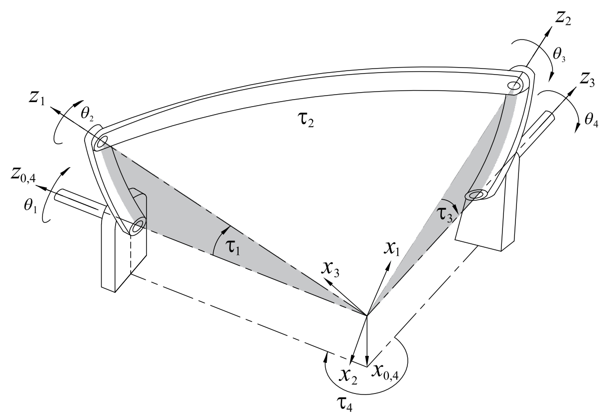


Fig. 1: Spherical 4R DH reference frames and parameters.

Consider the arbitrary spherical 4R linkage illustrated in Fig. 1. The IO equation expresses the implicit functional relationship between the input and output angles, θ_i and θ_j in terms of the constant arc lengths between the four R-pair centres, τ_i. The derivation of the algebraic form of the spherical IO equation [9] makes use of the original Denavit-Hartenberg (DH) parametrisation of the kinematic geometry [6]. It requires that all measures of angle be converted to algebraic parameters using the tangent half-angle substitutions:

$$v_i = \tan\frac{\theta_i}{2}, \quad \cos\theta_i = \frac{1-v_i^2}{1+v_i^2}, \quad \sin\theta_i = \frac{2v_i}{1+v_i^2},$$
$$\alpha_i = \tan\frac{\tau_i}{2}, \quad \cos\tau_i = \frac{1-\alpha_i^2}{1+\alpha_i^2}, \quad \sin\tau_i = \frac{2\alpha_i}{1+\alpha_i^2}.$$

The forward kinematics of an arbitrary kinematic chain is obtained as a linear transformation matrix in terms of the DH parameters. This linear transformation can then be mapped to the corresponding eight Study soma coordinates [9]. For spherical kinematic chains there are only four homogeneous soma coordinates. The ideal generated by the three soma that equate to zero are used to derive the algebraic IO equations relating the six distinct edges of an arbitrary spherical quadrangle.

Making these substitutions, the algebraic form of the IO equation is derived as [9]

$$Av_1^2v_4^2 + Bv_1^2 + Cv_4^2 + 8\alpha_1\alpha_3\left(\alpha_4^2+1\right)\left(\alpha_2^2+1\right)v_1v_4 + D = 0, \qquad (1)$$

where

$$
\begin{aligned}
A &= A_1A_2 = (\alpha_1\alpha_2\alpha_3 - \alpha_1\alpha_2\alpha_4 + \alpha_1\alpha_3\alpha_4 - \alpha_2\alpha_3\alpha_4 + \alpha_1 - \alpha_2 + \alpha_3 - \alpha_4) \\
&\qquad (\alpha_1\alpha_2\alpha_3 - \alpha_1\alpha_2\alpha_4 - \alpha_1\alpha_3\alpha_4 - \alpha_2\alpha_3\alpha_4 - \alpha_1 - \alpha_2 - \alpha_3 + \alpha_4), \\
B &= B_1B_2 = (\alpha_1\alpha_2\alpha_3 + \alpha_1\alpha_2\alpha_4 - \alpha_1\alpha_3\alpha_4 - \alpha_2\alpha_3\alpha_4 + \alpha_1 + \alpha_2 - \alpha_3 - \alpha_4) \\
&\qquad (\alpha_1\alpha_2\alpha_3 + \alpha_1\alpha_2\alpha_4 + \alpha_1\alpha_3\alpha_4 - \alpha_2\alpha_3\alpha_4 - \alpha_1 + \alpha_2 + \alpha_3 + \alpha_4), \\
C &= C_1C_2 = (\alpha_1\alpha_2\alpha_3 - \alpha_1\alpha_2\alpha_4 - \alpha_1\alpha_3\alpha_4 + \alpha_2\alpha_3\alpha_4 - \alpha_1 + \alpha_2 + \alpha_3 - \alpha_4) \\
&\qquad (\alpha_1\alpha_2\alpha_3 - \alpha_1\alpha_2\alpha_4 + \alpha_1\alpha_3\alpha_4 + \alpha_2\alpha_3\alpha_4 + \alpha_1 + \alpha_2 - \alpha_3 + \alpha_4), \\
D &= D_1D_2 = (\alpha_1\alpha_2\alpha_3 + \alpha_1\alpha_2\alpha_4 + \alpha_1\alpha_3\alpha_4 + \alpha_2\alpha_3\alpha_4 - \alpha_1 - \alpha_2 - \alpha_3 - \alpha_4) \\
&\qquad (\alpha_1\alpha_2\alpha_3 + \alpha_1\alpha_2\alpha_4 - \alpha_1\alpha_3\alpha_4 + \alpha_2\alpha_3\alpha_4 + \alpha_1 - \alpha_2 + \alpha_3 + \alpha_4).
\end{aligned}
$$

While the derivation of this algebraised v_1-v_4 IO equation is novel and far from intuitive, the algebraic form of this fourth degree polynomial in the v_1-v_4 IO angle parameters is not. The earliest derivations of similar equations representing manipulatable octahedra, identical in form, is due to R. Bricard in 1897 [2].

2 Derivation of the Six Spherical v_i-v_j IO Equations

Using the eight bi-cubic coefficient definitions from Eq. (1), the remaining five v_i-v_j equations contain all eight of the bi-cubic coefficients, but in different permutations:

$$A_1B_2v_1^2v_2^2 + A_2B_1v_1^2 + C_1D_2v_2^2 + 8\alpha_2\alpha_4\left(\alpha_1^2+1\right)\left(\alpha_3^2+1\right)v_1v_2 + C_2D_1 = 0; \quad (2)$$

$$A_1B_1v_1^2v_3^2 + A_2B_2v_1^2 + C_2D_2v_3^2 + C_1D_1 = 0; \quad (3)$$

$$A_1D_2v_2^2v_3^2 + B_2C_1v_2^2 + B_1C_2v_3^2 - 8\alpha_1\alpha_3\left(\alpha_2^2+1\right)\left(\alpha_4^2+1\right)v_2v_3 + A_2D_1 = 0; \quad (4)$$

$$A_1C_1v_2^2v_4^2 + B_2D_2v_2^2 + A_2C_2v_4^2 + B_1D_1 = 0; \quad (5)$$

$$A_1C_2v_3^2v_4^2 + B_1D_2v_3^2 + A_2C_1v_4^2 + 8\alpha_2\alpha_4\left(\alpha_1^2+1\right)\left(\alpha_3^2+1\right)v_3v_4 + B_2D_1 = 0. \quad (6)$$

The v_1-v_4 IO Equation. To obtain this IO equation from the ideal generated by the three soma coordinates that equate to zero, both v_2 and v_3 are eliminated by first computing the Gröbner bases using the Maple 2021 "`tdeg`" monomial ordering with the list sequence (v_3, v_2, v_4, v_1). This is *graded reverse lexicographic order*, also known as *degrevlex* in the literature [1], with indeterminate ordering $v_3 > v_2 > v_4 > v_1$. This monomial ordering sorts the terms by total degree before breaking ties between terms with identical degree by comparing the smallest indeterminate first and considering a higher degree as smaller in the term ordering. In this case, 12 bases are computed, all functions of all four v_i. We eliminate v_2 and v_3 by computing the bases of these 12 with the reverse monomial ordering by using "`plex`", which is the *pure lexicographic order*, also known as *lex*. This results in 10 new bases, with one that is a function of only v_1 and v_4 and the four α_i, which represents the IO equation we are looking for. This polynomial splits into three factors. The first two are $(1+v_1^2)(1+v_4^2)$, a product that is always greater than zero, and can be safely factored out, leaving us with Eq. (1). This, and some of the other IO equations are computable in one application of the elimination monomial ordering called "`lexdeg`", but the computation time is about 3500 s compared to 120 s for the sequential application of "`tdeg`" and "`plex`" on an Intel Core i7-7700 CPU @ 3.60 GHz.

It is important to note that we are using the standard Denavit-Hartenberg [6] relative joint angle parameters, which are each a measure of the angle a link makes with the previous link in the kinematic chain. This fact enables us to derive the remaining five IO equations such that the same eight bi-cubic coefficient factors characterise all six IO equations. This is generally not the case when vector loop methods are used together with trigonometry, see [7] for a detailed example.

The v_1-v_2 IO Equation. The derivation steps are precisely the same as for the v_1-v_4 IO equation. Eliminating v_3 and v_4 from the same three soma coordinates, the resulting v_1-v_2 IO equation splits into three similar factors. The first two, $(1+v_1^2)(1+v_2^2)$, can be safely factored out, leaving us with Eq. (2).

The v_1-v_3 IO Equation. The derivation steps are precisely the same as for the previous two IO equations. But, after the elimination of v_2 and v_4 from the same three soma coordinates, the resulting v_1-v_3 IO equation splits into five factors. The first two are $(1+v_1^2)(1+v_3^2)$, and can be safely factored out. The next two are

$$(\alpha_2^2\alpha_3^2+2\alpha_2\alpha_3+1)v_3^2+\alpha_2^2\alpha_3^2-2\alpha_2\alpha_3+1, \tag{7}$$

$$(\alpha_2^2-2\alpha_2\alpha_3+\alpha_3^2)v_3^2+\alpha_2^2+2\alpha_2\alpha_3+\alpha_3^2. \tag{8}$$

In order for either, or both, of Eqs. (7) or (8) to be identically zero the arc length parameters α_2 and α_3 must be complex, meaning these two factors may also eliminated, leaving us with Eq. (3).

The v_2-v_3 IO Equation. To derive this IO equation using elimination methods on the three soma coordinates we have been using requires a very different approach. We were successful by first applying grevlex to the three soma coordinates using the list sequence (v_1,v_4,v_2,v_3), then applying *graded lexicographic order* using "`grlex`" to the bases identified with grevlex. After each computation we obtain 12 bases, all in terms of the four α_i and the four v_i, with the exception of one in the graded lexicographic order set of bases, which is in terms of the four α_i, but only v_1, v_2, and v_3, and is used in the elimination steps. Next, resultants are used to eliminate v_4 first, then v_1. We obtain a v_2-v_3 IO equation that splits into nine factors.

The first five of these factors are simple to divide out since they are trivially non-zero: the first is -1; the other four are the squares of a single α_i added to a positive integer. The next three factors are functions of v_2 and v_3, but only α_1, α_2, and α_3:

$$\begin{gathered}(\alpha_1\alpha_2-\alpha_1\alpha_3+\alpha_2\alpha_3+1)^2v_2^2v_3^2+(\alpha_1\alpha_2+\alpha_1\alpha_3-\alpha_2\alpha_3+1)^2v_2^2+\\ 8\alpha_1\alpha_3(\alpha_2^2+1)v_2v_3+(\alpha_1\alpha_2-\alpha_1\alpha_3-\alpha_2\alpha_3-1)^2v_3^2+(\alpha_1\alpha_2+\alpha_1\alpha_3+\alpha_2\alpha_3-1)^2;\end{gathered} \tag{9}$$

$$\begin{gathered}(\alpha_1\alpha_2\alpha_3+\alpha_1-\alpha_2+\alpha_3)^2v_2^2v_3^2+(\alpha_1\alpha_2\alpha_3-\alpha_1+\alpha_2+\alpha_3)^2v_2^2-\\ 8\alpha_1\alpha_3(\alpha_2^2+1)v_2v_3+(\alpha_1\alpha_2\alpha_3+\alpha_1+\alpha_2-\alpha_3)^2v_3^2+(\alpha_1\alpha_2\alpha_3-\alpha_1-\alpha_2-\alpha_3)^2;\end{gathered} \tag{10}$$

$$\alpha_3(\alpha_1\alpha_2+1)(\alpha_1-\alpha_2)v_2^2+2\alpha_1\alpha_3(\alpha_2^2+1)v_2v_3-\alpha_1(\alpha_2\alpha_3+1)(\alpha_2-\alpha_3)v_3^2+ \alpha_2(\alpha_1+\alpha_3)(\alpha_1\alpha_3-1). \tag{11}$$

In order for Eqs. (9), (10), and/or (11) to be identically zero the arc length parameters α_1, α_2, and/or α_3 must be complex numbers, so we may safely divide these three factors out, leaving only Eq. (4) as the desired IO equation.

The v_2-v_4 IO Equation. The derivation steps for the v_2-v_4 IO equation are the same as those for the v_1-v_3 IO equation. The the second set of Gröbner bases computed using the pure lexicographic order with list sequence (v_3, v_1, v_2, v_4) lead to an IO equation that splits into five factors, the first two are trivial.The next two are

$$(\alpha_1^2\alpha_2^2+2\alpha_1\alpha_2+1)v_2^2+\alpha_1^2\alpha_2^2-2\alpha_1\alpha_2+1, \tag{12}$$

$$(\alpha_1^2-2\alpha_1\alpha_2+\alpha_2^2)v_2^2+\alpha_1^2+2\alpha_1\alpha_2+\alpha_2^2. \tag{13}$$

For either, or both of Eqs. (12) and (13) to equate to zero, it requires both α_1 and α_2 to be complex. We can therefore factor both of these out, leaving only the desired v_2-v_4 IO, Eq. (5).

The v_3-v_4 IO Equation. Finally, the derivation steps for the v_3-v_4 IO equation are precisely the same as for the v_1-v_4 and v_1-v_2 IO equations. After the elimination of v_1 and v_2 from the same three soma coordinates, the resulting v_3-v_4 IO equation splits into three factors. The first two are safely divided out, leaving us with Eq. (6).

3 Application

The utility of these six spherical 4R algebraic IO equations is nicely demonstrated by the following two continuous approximate dimensional synthesis examples for function generation.

$v_3 = f(v_1)$ Function Generator. For this example, we will apply a modified form of the continuous approximate syntheses method [4] to function generation for $v_3 = f(v_1)$ using Eq. (3). The prescribed function is

$$v_3 = 2+\tan\left(\frac{v_1^2}{v_1^2+1}\right), \tag{14}$$

over the range $-2 \le v_1 \le 2$.

The v_1-v_3 IO equation relates the input angle parameter to a measure of the transmission angle. We begin by squaring Eq. (3) to eliminate the residual error values that are equal in magnitude yet opposite in sense. We partition the result into a 9x1 array of angle parameters and a 9x1 array of associated link arc length parameter coefficients. The array $\mathbf{s}_{v_1,v_3}$ of angle parameters is used to generate the synthesis equation

$$\mathbf{s}_{v_1,v_3} = [v_1^4 v_3^4, v_1^4 v_3^2, v_1^4, v_1^2 v_3^4, v_1^2 v_3^2, v_1^2, v_3^4, v_3^2, 1]^T, \tag{15}$$

while the corresponding 9x1 array of link arc length parameters $\boldsymbol{\alpha}_{v_1,v_3}$ are the coefficients scaled by the v_1 and v_3 variable elements in $\mathbf{s}_{v_1,v_3}$. Once the v_3 parameters are replaced by the prescribed function $v_3 = f(v_1)$, the array is integrated between the desired bounds of $v_{1_{\min}}$ and $v_{1_{\max}}$, the only synthesis equation required is revealed with the numerical minimisation of an Euclidean inner product, which can be generalised for all six $v_j = f(v_i)$ function generator possibilities:

$$\min_{(\alpha_1,\alpha_2,\alpha_3,\alpha_4)\in\mathbb{R}} \left(\boldsymbol{\alpha}_{v_i,v_j} \cdot \int_{v_{i_{\min}}}^{v_{i_{\max}}} \mathbf{s}_{v_i,f(v_i)} dv_i \right). \tag{16}$$

The numerical integrator and optimiser used in Maple 2021 require initial guesses for the link arc length parameters. Three reasonable initial guesses are those that satisfy the exact synthesis problem after setting $\alpha_4 = 1$ to normalise the equation. For the three precision pairs of $(v_1, v_3) = (-2, 3.0296), (0, 2), (2, 3.0296)$ the standard Maple solver could not identify any real, or non-trivial solutions. We decided instead to use the optimiser, which also requires initial guesses. We arbitrarily selected $(\alpha_1, \alpha_2, \alpha_3) = (3/10, 7/12, 11/10)$: these Maple algorithms are more reliable when rational numbers, ideally with prime numerators, are used for the initial guesses rather than floating point decimals. For the exact synthesis problem, the Maple optimiser returned the floating point numbers listed in Table 1, with $\alpha_4 = 1$. For the integrator initial guesses we used the same as those for the exact synthesis optimiser and the continuous approximate synthesis yielded the results listed in Table 1.

The *structural error* (S.E.) of a function generator is considered to be a useful performance indicator for the utility of the identified linkage. It is defined to be the difference between the prescribed linkage output value and the actual generated output value for a given input value [6]. Using Eq. (3) and the prescribed function equation itself, we can visualise the S.E. by examining the plots of the corresponding v_1-v_3 curves, see Fig. 2a, and by computing the area between the two curves. Since we have algebraic expressions for the three curves generated by the exact and continuous approximate synthesis and the prescribed function, it is a simple matter to re-define the S.E. as the difference of the integrals of the prescribed and generated functions. Comparing two different linkages designed to generate the same function over the same range, the one with the smallest structural error generates the prescribed function with the greatest precision. For the synthesis results listed in Table 1 of the $v_3 = f(v_1)$ function generator, we observe that the linkage identified with our modified continuous approximate synthesis method generates the prescribed function with a S.E. that is an order of magnitude smaller than the S.E. of the linkage identified with the standard exact precision point method.

$v_4 = f(v_1)$ Function Generator. The prescribed $v_4 = f(v_1)$ function for this example is

$$v_4 = 2 + \tan\left(\frac{v_1}{v_1^2+1}\right), \tag{17}$$

over the same range as the previous example, $-2 \le v_1 \le 2$. We will again use the modified continuous approximate synthesis using Eq. (1) after making the substitution $v_4 = f(v_1)$ and integrating the square of the resulting equation over the specified bounds. In this case the 13x1 array $\mathbf{s}_{v_1,v_4}$ of angle parameters is

$$\mathbf{s}_{v_1,v_4} = [v_1^4 v_4^4, v_1^4 v_4^2, v_1^4, v_1^3 v_4^3, v_1^3 v_4, v_1^2 v_4^4, v_1^2 v_4^2, v_1^2, v_1 v_4^3, v_1 v_4, v_4^4, v_4^2, 1]^T, \tag{18}$$

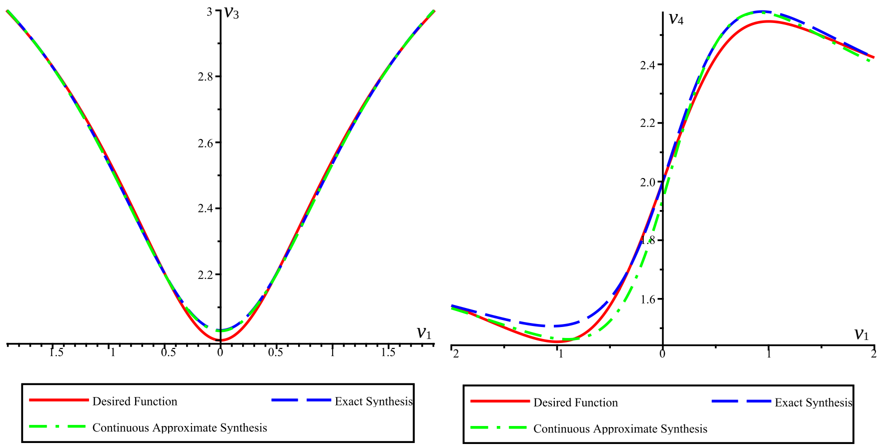
while the corresponding 13x1 array of link arc length parameters $\boldsymbol{\alpha}_{v_1,v_4}$ are the coefficients scaled by the v_1 and v_4 elements in $\mathbf{s}_{v_1,v_4}$. Proceeding as in the previous example, we use as our initial guesses for the integrator the α_i that satisfy the exact synthesis problem by solving the three equations generated by the three IO precision pairs of $(v_1, v_4) = (-2, 1.5772), (0, 2), (2, 2.4228)$. It is important to note that the exact synthesis using the spherical 4R algebraic IO equation can, according to Bezout's theorem, lead to as many as 216 solutions for α_1, α_2, and α_3, all in terms of α_4. In this example there are 32 solutions. We used the first, $\alpha_1 = -0.1083$, $\alpha_2 = 0.5183$, and $\alpha_3 = 1.0432$ for an initial guess in the optimiser. The resulting continuous approximate synthesis results are listed in Table 1. Again, the S.E. for linkage identified with the continuous approach is an order of magnitude smaller than that of the linkage identified with exact synthesis, as illustrated in Fig. 2b.

Table 1: Exact and continuous approximate synthesis results.

Function	Synthesis	α_1	α_2	α_3	α_4	S.E.
$v_3 = 2 + \tan\left(\frac{v_1^2}{v_1^2+1}\right)$	Exact	0.0226	0.2023	1.3459	1	0.0052
	Continuous	0.0372	0.3460	1.3244	0.7998	0.0007
$v_4 = 2 + \tan\left(\frac{v_1}{v_1^2+1}\right)$	Exact	-0.1083	0.5183	1.0432	1	0.1010
	Continuous	-0.1030	0.4920	0.7512	0.6199	0.0165

4 Conclusions

In this paper we have derived the six possible spherical 4R algebraic IO equations that describe the relative input and output angles between different pairs of edges in a spherical quadrangle. The equations were derived using Study's soma coordinates that represent the displacement space of all spherical 4R kinematic chains, and elimination methods to reveal the desired algebraic IO equation. We also illustrated the utility of the equations to function generator synthesis using a novel continuous approximate synthesis approach to implicitly minimise the structural error.



(a) Synthesis results for $v_3 = f(v_1)$.

(b) Synthesis results for $v_4 = f(v_1)$.

Fig. 2: Synthesis results.

References

1. Adams, W., Loustaunau, P.: An Introduction to Gröbner Bases, vol. 3. American Mathematical Society, Graduate Studies in Mathematics (1994)
2. Bricard, R.: Mémoire sur la théorie de l'octaèdre articulé. J. Math. Pures Appl. **3**, 113–148 (1897)
3. Ceccarelli, M. (ed.): Distinguished Figures in Mechanism and Machine Science, Their Contributions and Legacies Part 1. Springer, New York, U.S.A. (2007)
4. Copeland, Z., Rotzoll, M., Hayes, M.J.D.: Concurrent Type and Dimensional Continuous Approximate Function Generator Synthesis for All Planar Four-bar Mechanisms. 11th CCToMM Symposium on Mechanisms, Machines, and Mechatronics, Ontario Tech University, Oshawa, ON, Canada (2021)
5. Gosselin, C.M., Hamel, J.F.: The Agile Eye: a High-performance Three-degree-of-freedom Camera-orienting Device. Proceedings of the 1994 IEEE International Conference on Robotics and Automation (1994)
6. Hartenberg, R., Denavit, J.: Kinematic Synthesis of Linkages. McGraw-Hill, Book Co., New York, N.Y., U.S.A. (1964)
7. Hayes, M.J.D., Rotzoll, M., Iraei, A., Nichol, A., Bucciol, Q.: Algebraic Differential Kinematics of Planar 4R Linkages. 20th International Conference on Advanced Robotics, ICAR 2021, Ljubljana, Slovenia (2021)
8. Moazami, S., Zargarzadeh, H., Palanki, S.: Kinematics of Spherical Robots Rolling Over 3D Terrains. Complexity **2019**, https://doi.org/10.1155/2019/7543,969 (2019)
9. Rotzoll, M., Hayes, M.J.D., Husty, M.L., Pfurner, M.: A General Method for Determining Algebraic Input-output Equations for Planar and Spherical 4R Linkages. pp. 90–97. Advances in Robotic Kinematics 2020, eds. Lenarčič, J. and Parenti-Castelli, V., Springer Nature Switzerland AG, Cham, Switzerland (2020)
10. Willis, R.: Principles of Mechanism, 2^{nd} *edition*. Cambridge University Press, Cambridge, England (1841)