



NUMERICALLY STABLE SOLUTION TO THE 6R PROBLEM OF INVERSE KINEMATICS

XIN CAO^{✉1} EVANGELOS A. COUTSIAS^{✉*1,2} SARA POLLOCK^{✉3}

¹Department of Applied Mathematics and Statistics, Stony Brook University,
Stony Brook, NY 11794, United States

²Laufer Center for Physical and Quantitative Biology, Stony Brook University,
Stony Brook, NY 11794, United States

³Department of Mathematics, University of Florida,
Gainesville, FL 32611, United States

(Communicated by Annalisa Quaini)

ABSTRACT. In this paper, a stable and accurate algorithm to compute all solutions of the inverse kinematics problem of a 6 revolute manipulator chain is presented. A system of equations is constructed based on the fundamental closure conditions, leading to a closed algebraic system of 20 equations involving 16 quantities, composed of trigonometric functions of five among the six unknown joint angles. Two among these five are stably eliminated using singular value decomposition (SVD) avoiding the need to consider special cases. The resulting system of equations involving three unknowns is solved by conversion to a generalized eigenvalue problem. The remaining three unknown angles are obtained using the previously computed pseudoinverse. In this formulation we exploit the inherently complex form of the system reducing it to 10 complex equations in 9 quantities, which substantially accelerates the SVD computation. The method's robustness is demonstrated through a comparison to current methods and several examples including known problematic cases where some axis or link lengths vanish, or some joint angles are 180 degrees, as well as cases where multiple eigenvalues arise.

1. Introduction. The Inverse Kinematics (**IK**) problem of a linkage, that is positioning the end effector (Arm) given a fixed (Reference) position of the base, is central in the theory of mechanisms and has been extensively studied, both in robotics where the accurate control of robot arms is critical in diverse areas such as medicine, industry and the International Space Station, see e.g. [2] and references therein, as well as in computational structural biology where loop closure analysis has been used in the study of protein loops [9, 19, 20, 6, 16] and large flexible multicyclic molecules [27, 5]. The workspace of a linkage is the set of positions and orientations of the end effector attainable through variation of the linkage degrees of freedom (DoF). It is well known that at least six DoF are required to achieve

2020 *Mathematics Subject Classification.* Primary: 65H14; Secondary: 65H10.

Key words and phrases. 6R manipulator, Inverse Kinematics, Complex singular value decomposition, Multiple eigenvalues, Molecular chains.

The first two authors are supported by NIH RM1GM135136, NSF DMS2054251.

*Corresponding author: Evangelos Coutsias.

a workspace that forms a (union of) open neighborhood(s) in $\mathbb{R}^3 \times SO(3)$. The problem may be realized without loss of generality as a chain of rigid bodies connected through revolute pairs (**R**-pairs) between fixed end positions. In this paper we consider the IK problem for a linkage formed by a kinematic chain of $N \geq 6$ rigid objects connected by revolute joints. The linkage may be a loop with fixed base and end effector or a closed ring (Fig. 1).

For simplicity we focus on the $N = 6$ ring problem, while the extension to redundant linkages of $N > 6$ joints and molecular chains is given in Appendix A. The 6 joint angles are the variables defining the linkage configuration. Together with the twist angles between successive axes and the joint and axis lengths of the resulting orthogonal system, they constitute the Denavit-Hartenberg (**DH**) coordinates [8] of the linkage. The transformation of an open **6R** robot arm to a closed system is accomplished by introducing a virtual link as depicted in the right of Fig. 1.

The first fully algebraic solution of the **6R** problem was given by Lee and Liang [14, 13]. Subsequently, several authors ([18], [21]) have commented on the algebraic complexity of the Lee-Liang method and proposed simplifications. All methods are based on the idea that certain of the variables may be eliminated, eventually resulting in a generalized eigenproblem or a univariate polynomial in one of the variables. Key differences among methods are in the details of the elimination process and the precise method of arriving at a univariate system.

As noted by Angeles [2], current algorithms share some robustness issues under certain conditions, such as the occurrence of infinite eigenvalues associated with π angles. In fact, robustness problems may arise at different stages in a typical method: (1) The first stage of the algorithm requires the elimination of some of the variables. The choice of the specific equations to be solved may lead to the inversion of singular matrices. (2) Assuming successful elimination, the solution of the resulting generalized eigenvalue problem may lead to infinite or multiple eigenvalues. (3) The algebraic manipulations for deriving the remaining DoF, once the eigenproblem has been solved may also introduce singularities.

The Lee-Liang method in its original form suffered from its somewhat arbitrary choices for eliminating or back-substituting variables that in special cases lead to singular matrices or vanishing denominators. Specifically, the 16 among the 20 equations chosen for the initial elimination ([13], eq.(50)) are not guaranteed to result in a nonsingular matrix inversion. Moreover, when solving for the remaining variables, vanishing lengths of either axis 3 or axis 4 results in the vanishing of the denominator of expressions F9 and F10 in [13], causing the method to fail. Although most current methods avoid some of the pitfalls in the original algorithm by Lee and Liang, they may still fail for special combinations of the problem parameters which nevertheless do correspond to well posed configurations. As discussed in [2], the method by Raghavan and Roth [21] will fail if the first link length vanishes. Although this problem may be addressed by reindexing generically, this will fail if all link lengths vanish. The method by Renaud [22] achieves efficiency by utilizing a specific linear combination of the equations for the inversion, which again fails for vanishing link lengths. In the Kohli and Osvatic approach [12], the elimination step involves the inverse of a 6×6 matrix whose singularity is not determined, for instance that matrix becomes singular when the twist angle at the sixth link is π in a closed linkage. Liu and Zhu [15] follow a similar approach but switch order of solution if singularity is detected.

In addition, the strategy followed by various methods to address possible singularities could deteriorate performance. For example, the methods by [18, 15] use singular value decomposition (SVD) to detect ill conditioning of various inversions, either at the elimination stage or for converting the generalized eigenproblem to an ordinary one, and reformulate in various ways until well conditioned matrices are found. Moreover, some methods arrive at eigenvalue problems of higher dimensionality, again leading to deterioration in performance. The method by Manocha and Canny [18] results in a generalized eigenproblem of size 24×24 , while as was proven by [13] the system may be reduced to optimal size 16×16 , as also obtained by [21, 12, 22, 15]. More recently, Husty et al. [10] added clarity to the geometrical structure of the problem, and proposed an elegant algorithm. However the eventual reduction to a univariate polynomial gives a high degree resultant of size 48×48 whose resolution substantially increases execution timings. Moreover, as the authors also state, a reformulation may be required until a non-degenerate system may be found, which could further hinder performance. A comparative analysis of the implementation of [10] to that in [15] performed by Angerer and Hofbaur [3] establishes that the former is indeed substantially more robust, but at a high operation time cost, roughly an order of magnitude slower compared to the latter.

The goal of this paper is to formulate a numerically robust algorithm for the solution of the **6R** IK problem. Although an earlier version of the algorithm was implemented as part of the BRIKARD molecular modeling suite [5], implementation details were never published. The version of R6B6 presented here incorporates substantial improvements and it is essentially a new algorithm. Contrasted to previous algorithms, R6B6 includes modifications in the elimination and solution process that optimally stabilize the method, avoiding well known pitfalls. Starting with the standard set of equations [2], complex SVD is applied to perform the first elimination. Writing the original real-valued 20×16 system as a complex-valued 10×8 system accelerates the computation by a factor of 50%. The resultant is then obtained efficiently by multiplication with the left null vectors. By properly accounting for the projective character of the matrix pair formulation in the resulting generalized eigenproblem, infinite eigenvalues are dealt with directly, while the resulting process is augmented to handle cases of multiple eigenvalues. When computing the remaining variables, the singular vectors from the SVD and some symbolic expressions are applied. Utilizing the pseudoinverse as opposed to inverting a subset of the equations lends stability to the computation and solves the issue of accidental singularities in the first elimination, which may require costly reformulations to resolve. We must note here that for **6R** systems exhibiting flexibility so that a continuum of solutions may exist, all methods (including the one presented here) designed to determine a finite number of solutions will fail. Such cases require further geometrical analysis, as can be found in [4], [23].

The remainder of the paper is organized as follows: In Sec. 2 we give a brief overview of the key steps in the algorithm, with details provided in the following subsections. Sec. 2.1 presents the fundamental closure equations and introduces the main formalism used throughout the paper. Sec. 2.2 presents the details of the first elimination via complex SVD. In Sec. 2.3 we discuss the resultant of the 4 polynomial equations in the half-tangents of the three remaining variables, extended as 8 equations in the same three variables and then posed as a 16×16 generalized eigenvalue problem. In Sec. 2.4 the procedure for extracting the remaining variables and giving the complete solution is presented. In Sec. 3 we discuss implementation

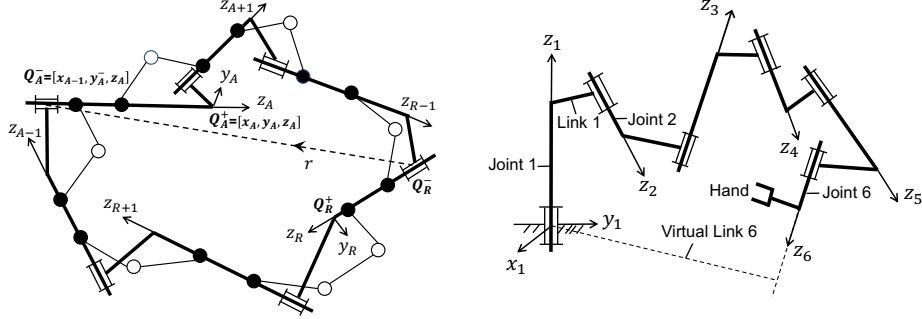


FIGURE 1. The transformation from cyclic molecule (left) and open manipulator (right) to closed **6R** ring. In the left, the atoms in the selected axes are filled and the coordinate systems of complete system show the relationship between reference (R) and arm (A) frames.

details of the solution process, with focus on special cases, such as infinities in the half-tangent variables and multiple eigenvalues. Sec. 4 discusses numerical examples that demonstrate code performance in various exceptional cases, as well as singular cases that may lead to failures. Sec. 5 gives a summary and concluding remarks. In the Appendices we present the details of the DH transformation App. A, give the details of the system of equations App. B, discuss the relation between real and complex SVD App. C and discuss the treatment necessary in the presence of multiple eigenvalues App. D.

2. The loop closure equations: algebraic solution. As preparation for our algorithm, an open chain is firstly converted to a closed **6R** chain. In a redundant chain of $N > 6$ rotors, six rotors are selected (associated with unknown joint angles) and the intervening structure between two consecutive rotors is rigidified by fixing all intervening joint angles (or torsions in a molecular chain) and then replaced by a common normal link of the two axes of rotors. The DH transformation places a Cartesian frame at either end of each joint (Fig. 2). For an open **6R** robot arm, we only need to add one virtual link between the sixth and first joints as shown in the right figure of Fig. 1. This results in a closed **6R** DH system as shown in the left figure of Fig. 1. The details can be found in App. A. The closed kinematic chain of six rotors is decomposed into two subchains, in which two joints are selected, separated by two joints along either subchain. We set one as the reference, denoted by a subscript R , and the second as the arm, denoted by a subscript A . Here R can be assigned to any joint $1, \dots, 6$ in the chain with the others assigned in cyclic succession. The six unknown joint angles are $\theta_i, i = R, R+1, A-1, A, A+1, R-1$. Applying the notation of [21, 26], we denote $c_i = \cos \theta_i, s_i = \sin \theta_i$ and half-tangent $t_i = \tan(\theta_i/2)$.

We then can formulate closure conditions for the closed DH system and solve the problem accordingly. A set of fourteen closure conditions (Sec. 9.2.2 in [2]), involving the 10 quantities given in Table 1 each evaluated in terms of the unknown θ_A , or more precisely its half-tangent t_A , and the sines and cosines of $\theta_{A+1}, \theta_{R-1}$

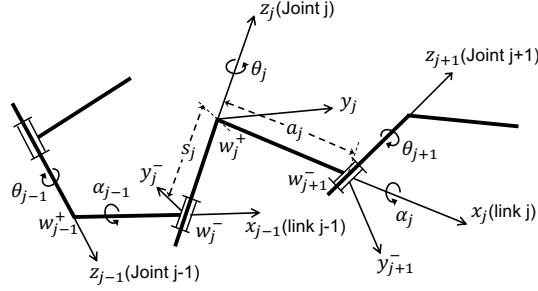


FIGURE 2. Denavit-Hartenberg coordinate frames are at $w_j^\pm$, θ_j is the joint angle and α_j is the twist angle.

E_1	$\langle \mathbf{z}_R, \mathbf{x}_{A-1} + i\mathbf{y}_A^- \rangle$
E_2	$\langle \frac{1}{2} \langle \mathbf{r}, \mathbf{r} \rangle \mathbf{z}_R - \langle \mathbf{r}, \mathbf{z}_R \rangle \mathbf{r}, \mathbf{x}_{A-1} + i\mathbf{y}_A^- \rangle$
E_3	$\langle \mathbf{z}_R \times \mathbf{r}, \mathbf{x}_{A-1} + i\mathbf{y}_A^- \rangle$
E_4	$\langle \mathbf{r}, \mathbf{x}_{A-1} + i\mathbf{y}_A^- \rangle$
E_5	$\langle \mathbf{z}_R, \mathbf{z}_A \rangle$
E_6	$\langle \frac{1}{2} \langle \mathbf{r}, \mathbf{r} \rangle \mathbf{z}_R - \langle \mathbf{r}, \mathbf{z}_R \rangle \mathbf{r}, \mathbf{z}_A \rangle$
E_7	$\langle \mathbf{z}_R \times \mathbf{r}, \mathbf{z}_A \rangle$
E_8	$\langle \mathbf{r}, \mathbf{z}_A \rangle$
E_9	$\langle \mathbf{r}, \mathbf{z}_R \rangle$
E_{10}	$\langle \mathbf{r}, \mathbf{r} \rangle / 2$

TABLE 1. The fourteen quantities used in closure conditions, presented in complexified form as 4 complex and 6 real expressions. Evaluating these along both subchains joining reference R and arm A results in the fundamental closure equations. Eqs. $E_5 - E_{10}$ are duplicated after multiplication by t_A , resulting in 20 real equations.

and $\theta_{R+1}, \theta_{A-1}$, are extended into a set of twenty equations with respect to the unknowns. The closure conditions are set up so the unknown θ_R is absent from all evaluations, and is eliminated by default. Then we can solve the system by following three more steps. The overview of the procedure is as follows and further description of every step will be given in the following subsections:

A. Formulation of the system of equations. Based on the closure conditions, we can construct and arrange the twenty equations through symbolic preprocessing as

$$B\mathbf{b} = P\mathbf{c} + \mathbf{q}t_A + \mathbf{r} = \begin{pmatrix} P_\alpha t_A + P_\beta \\ 1 \end{pmatrix} \begin{pmatrix} \mathbf{c}' \\ 1 \end{pmatrix}, \text{ where } \mathbf{b} = \begin{pmatrix} t_A \mathbf{b}' \\ \mathbf{b}' \end{pmatrix}, \mathbf{c} = \begin{pmatrix} t_A \mathbf{c}' \\ \mathbf{c}' \end{pmatrix}$$

where $B, P, \mathbf{q}, \mathbf{r}, P_\alpha$ and P_β are coefficient matrices: B, P are 20×16 , $\mathbf{q}, \mathbf{r}$ are 20×1 and P_α, P_β are 20×9 . The components in $\mathbf{b}', \mathbf{c}'$ are given by

$$\mathbf{b}' = [c_{R+1}c_{A-1}, s_{R+1}c_{A-1}, c_{A-1}, c_{R+1}s_{A-1}, s_{R+1}s_{A-1}, s_{A-1}, c_{R+1}, s_{R+1}]^T, \quad (1)$$

$$\mathbf{c}' = [c_{A+1}c_{R-1}, s_{A+1}c_{R-1}, c_{R-1}, c_{A+1}s_{R-1}, s_{A+1}s_{R-1}, s_{R-1}, c_{A+1}, s_{A+1}]^T. \quad (2)$$

B. First elimination using SVD. The SVD of B is calculated as $B = U\Sigma V^T$, and the transpose of the left null-vectors of B (denoted as U_N^T which is 4×20) is multiplied on both sides of the system to eliminate the terms on the left side completely which contain the variables θ_{R+1} and θ_{A-1} . Then the resulting right side has four equations and can be arranged as follows:

$$U_N^T B \mathbf{b} = 0 = (U_N^T P_\alpha t_A + U_N^T P_\beta) \begin{pmatrix} \mathbf{c}' \\ 1 \end{pmatrix} = (P_\alpha^U t_A + P_\beta^U) \begin{pmatrix} \mathbf{c}' \\ 1 \end{pmatrix},$$

where P_α^U, P_β^U are coefficients matrices with dimension 4×9 . We note that the SVD is performed in complex form, resulting in a 50% speedup, and the results are converted back into real form.

C. Formulation of generalized eigenvalue problem. The sines and cosines of θ_{R-1} and θ_{A+1} in the resulting 4×9 system above are firstly converted by the half-tangent identities $s_j = 2t_j/(1+t_j^2)$ and $c_j = (1-t_j^2)/(1+t_j^2)$. Multiplying this first set of equations by t_{R-1} yields a second set which is cubic in t_{R-1} , quadratic in t_{A+1} and linear in t_A . Then the expanded system is sorted into three coefficient matrices by powers of t_{A+1} as $(t_{A+1}^2 H_2 + t_{A+1} H_1 + H_0) \mathbf{v} = 0$, where H_2, H_1, H_0 are 8×8 and

$$\mathbf{v} = (t_A \mathbf{h}, \mathbf{h})^T; \mathbf{h} = [t_{R-1}^3, t_{R-1}^2, t_{R-1}, 1]^T. \quad (3)$$

Then we construct the 16×16 generalized eigenvalue problem with notations $\Lambda := t_{A+1}$, $\mathbf{v}_1 := \mathbf{v}$ and $\mathbf{v}_2 := t_{A+1} \mathbf{v}$,

$$\begin{pmatrix} O & I \\ -H_0 & -H_1 \end{pmatrix} \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{pmatrix} = \Lambda \begin{pmatrix} I & O \\ O & H_2 \end{pmatrix} \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{pmatrix}.$$

where O denotes an 8×8 block of zeros and I is the 8×8 identity block. The variables $\theta_{A+1}, \theta_{R-1}$ and θ_A can be obtained from the real eigenvalues and eigenvectors. We also check for multiple eigenvalues and apply extra treatment when necessary.

D. Solving for remaining variables. The vector $\mathbf{b}$ is acquired by multiplying the pseudoinverse $B^\dagger = \hat{V} \hat{\Sigma}^{-1} \hat{U}^T$ on both sides of the original system:

$$\mathbf{b} = B^\dagger (P_\alpha t_A + P_\beta) \begin{pmatrix} \mathbf{c}' \\ 1 \end{pmatrix}.$$

Then θ_{R+1} and θ_{A-1} can be solved with selection of appropriate components of $\mathbf{b}$ to achieve accurate solutions, and θ_R is computed with symbolic expressions.

2.1. Formulation of closure conditions and system of equations. In this section we describe the formulation of closure conditions for the closed DH system. The arm axis $\mathbf{z}_A$ has two associated frames: the first locates at the head of $\mathbf{z}_A$ connecting to link axis $\mathbf{x}_A$, $Q_A^+ = \{\mathbf{x}_A, \mathbf{y}_A, \mathbf{z}_A\}$ and the second is $Q_A^- = \{\mathbf{x}_{A-1}, \mathbf{y}_A^-, \mathbf{z}_A\}$, where the tail of $\mathbf{z}_A$ connects to link axis $\mathbf{x}_{A-1}$. Let $\mathbf{r}$ be the vector connecting the tail of the reference axis $\mathbf{z}_R$ to the tail of arm axis $\mathbf{z}_A$, so that $\mathbf{r}$ and $\mathbf{z}_R$ are coterminal and $\mathbf{r}$ connects $\mathbf{z}_R$ to the frame origin of Q_A^- (left in Fig. 1).

Let us first consider the rotation about θ_A relating the frame Q_A^- to Q_A^+ , where

$$\begin{aligned}\mathbf{x}_{A-1} &= \mathbf{x}_A \cos \theta_A - \mathbf{y}_A \sin \theta_A, \\ \mathbf{y}_A^- &= \mathbf{x}_A \sin \theta_A + \mathbf{y}_A \cos \theta_A.\end{aligned}$$

Taking the inner product between an appropriate vector $\mathbf{w}$, and the coordinate vectors $\mathbf{x}_{A-1}$, and $\mathbf{y}_A^-$, we obtain

$$\begin{aligned}\langle \mathbf{w}, \mathbf{x}_{A-1} \rangle &= \langle \mathbf{w}, \mathbf{x}_A \cos \theta_A - \mathbf{y}_A \sin \theta_A \rangle, \\ \langle \mathbf{w}, \mathbf{y}_A^- \rangle &= \langle \mathbf{w}, \mathbf{x}_A \sin \theta_A + \mathbf{y}_A \cos \theta_A \rangle.\end{aligned}\tag{4}$$

The sines and cosines of θ_A are next converted to half-tangent relations through the identities for $t_A = \tan(\theta_A/2)$ as $\sin \theta_A = t_A / \cos \theta_A$, and $t_A \sin \theta_A + \cos \theta_A = 1$. Multiplying each of the inner-product relations (4) by t_A , and adding (respectively subtracting) the other replaces the two unknowns $\sin \theta_A$ and $\cos \theta_A$ with single unknown $t_A = \tan(\theta_A/2)$ at the cost of coupling the equations (4), resulting in the pair of relations

$$\begin{aligned}\langle \mathbf{w}, \mathbf{x}_{A-1} t_A - \mathbf{y}_A^- \rangle &= \langle \mathbf{w}, -\mathbf{y}_A - \mathbf{x}_A t_A \rangle, \\ \langle \mathbf{w}, \mathbf{y}_A^- t_A + \mathbf{x}_{A-1} \rangle &= \langle \mathbf{w}, \mathbf{x}_A - \mathbf{y}_A t_A \rangle.\end{aligned}\tag{5}$$

Equations in (5) are equivalently expressed in the matrix form

$$\begin{pmatrix} \langle \mathbf{w}, \mathbf{x}_{A-1} \rangle & -\langle \mathbf{w}, \mathbf{y}_A^- \rangle \\ \langle \mathbf{w}, \mathbf{y}_A^- \rangle & \langle \mathbf{w}, \mathbf{x}_{A-1} \rangle \end{pmatrix} \begin{pmatrix} t_A & -1 \\ 1 & t_A \end{pmatrix} = \begin{pmatrix} -\langle \mathbf{w}, \mathbf{y}_A \rangle & -\langle \mathbf{w}, \mathbf{x}_A \rangle \\ \langle \mathbf{w}, \mathbf{x}_A \rangle & -\langle \mathbf{w}, \mathbf{y}_A \rangle \end{pmatrix} \begin{pmatrix} 1 & -t_A \\ t_A & 1 \end{pmatrix},\tag{6}$$

which is the matrix representation of the complex equation

$$(\langle \mathbf{w}, \mathbf{x}_{A-1} \rangle + i \langle \mathbf{w}, \mathbf{y}_A^- \rangle) (t_A + i) = -(\langle \mathbf{w}, \mathbf{y}_A \rangle - i \langle \mathbf{w}, \mathbf{x}_A \rangle) (1 + i t_A).\tag{7}$$

The observation of the complex form of the coupled equations is used to cut the computation by 50% for the first elimination step where the SVD of the matrix of coefficients corresponding to the left-hand side unknowns is computed as a complex 10×8 matrix rather than a real 20×16 matrix.

The displacement of the arm frame Q_A^- with respect to the reference axis $\mathbf{z}_R$ is fully and redundantly described by 10 quantities in Table 1 and the vectors $\mathbf{x}_{A-1}, \mathbf{y}_A^-, \mathbf{z}_A, \mathbf{z}_R$ and $\mathbf{r}$, relating $\mathbf{z}_R$ and connecting vector $\mathbf{r}$ to the arm frame Q_A^- . Six of the quantities, those without imaginary parts ($E_5 - E_{10}$), are already in suitable form to evaluate along both sides of the loop connecting $\mathbf{z}_R$ to $\mathbf{z}_A$ to form a set of closure conditions posed as a system of 6 equations where the unknowns are partitioned into two sets: the sines and cosines of $\theta_{R+1}, \theta_{A-1}$ and the sines and cosines of $\theta_{A+1}, \theta_{R-1}$.

Equating real and imaginary parts of (7), or the first column of the matrix relation (6) with $\mathbf{w} = \mathbf{z}_R$ for (E_1), $\mathbf{w} = \frac{1}{2} \langle \mathbf{r}, \mathbf{r} \rangle \mathbf{z}_R - \langle \mathbf{r}, \mathbf{z}_R \rangle \mathbf{r}$ for (E_2), $\mathbf{w} = \mathbf{z}_R \times \mathbf{r}$ for (E_3), and $\mathbf{w} = \mathbf{r}$ for (E_4), we can obtain the other set of 8 equations by writing the real and imaginary parts separately as $E_{1,R}, E_{1,I}, \dots, E_{4,R}, E_{4,I}$. Details of all the equations are given in App. B. Then the real-valued relations between $\mathcal{E}^L$, the left-hand expansion into the constants multiplying the sines and cosines of $\theta_{R+1}, \theta_{A-1}$, and $\mathcal{E}^R$, and the right-hand expansion into the sines and cosines of $\theta_{A+1}, \theta_{R-1}$, coupled by the half-tangent of θ_A can be evaluated below. The left evaluation $\mathcal{E}^L$ is formulated by orthogonal transformations through which the vectors $\mathbf{x}_R, \dots, \mathbf{x}_{A-1}$ and $\mathbf{z}_R, \dots, \mathbf{z}_{A-1}$ can be expressed in the coordinates of the arm frame Q_A^- . The

right evaluation $\mathcal{E}^R$ is similarly formulated by expressing the vectors $\mathbf{x}_A, \dots, \mathbf{x}_{R-1}$ and $\mathbf{z}_A, \dots, \mathbf{z}_{R-1}$ in the coordinates of the arm frame Q_A^+ .

The orthogonal transformation between frames connected by a joint axis z_i is given by the rotation $\mathbf{u}_i^- = A_{a,i} \mathbf{u}_i^+$, where $\mathbf{u}_i^- = (\langle \mathbf{u}, \mathbf{x}_{i-1} \rangle, \langle \mathbf{u}, \mathbf{y}_i \rangle, \langle \mathbf{u}, \mathbf{z}_i \rangle)^T$ contains the coordinates of unit vector $\mathbf{u}$ expressed in the frame Q_i^- , and $\mathbf{u}_i^+ = (\langle \mathbf{u}, \mathbf{x}_i \rangle, \langle \mathbf{u}, \mathbf{y}_i \rangle, \langle \mathbf{u}, \mathbf{z}_i \rangle)^T$ contains the coordinates of $\mathbf{u}$ expressed in the frame Q_i^+ . Similarly, $\mathbf{u}_i^+ = A_{a,i}^{-1} \mathbf{u}_i^-$ rotates the opposite direction about an axis, with $A_{a,i}^{-1} = A_{a,i}^T$, the transpose of the forward rotation. The transformations either way about a link $\mathbf{x}_i$ are given by $\mathbf{u}_i^+ = A_{l,i} \mathbf{u}_{i+1}^-$, and $\mathbf{u}_{i+1}^- = A_{l,i}^{-1} \mathbf{u}_i^+$. The orthogonal matrices $A_{a,i}$ and $A_{l,i}$ are the 3×3 rotation matrices

$$A_{a,i} = \begin{pmatrix} c_i & -s_i & 0 \\ s_i & c_i & 0 \\ 0 & 0 & 1 \end{pmatrix}, \quad \text{and} \quad A_{l,i} = \begin{pmatrix} 1 & 0 & 0 \\ 0 & \lambda_i & -\mu_i \\ 0 & \mu_i & \lambda_i \end{pmatrix}, \quad (8)$$

where $\lambda_i = \cos \alpha_i$ and $\mu_i = \sin \alpha_i$ and α_i is the given twist angle.

Composing transformations in (8), $\mathbf{u}_i^- = A_i \mathbf{u}_{i+1}^-$; and transforming the opposite direction around the loop, $\mathbf{u}_{i+1}^- = A_i^{-1} \mathbf{u}_i^-$, with

$$A_i = A_{a,i} A_{l,i} = \begin{pmatrix} c_i & -s_i \lambda_i & s_i \mu_i \\ s_i & c_i \lambda_i & -c_i \mu_i \\ 0 & \mu_i & \lambda_i \end{pmatrix}, \quad \text{and} \quad A_i^{-1} = A_i^T. \quad (9)$$

The left evaluation $\mathcal{E}^L$ proceeds by writing

$$\mathbf{u}_A^- = A_{A-1}^{-1} A_{R+1}^{-1} A_{l,R}^{-1} \mathbf{u}_R^+, \quad (10)$$

$$\mathbf{u}_A^- = A_{A-1}^{-1} A_{R+1}^{-1} \mathbf{u}_{R+1}^-, \quad (11)$$

$$\mathbf{u}_A^- = A_{A-1}^{-1} \mathbf{u}_{A-1}^-. \quad (12)$$

Then the coordinates of $\mathbf{z}_R$ in frame Q_A^- are found by setting $\mathbf{u}_R^+ = [0, 0, 1]^T$ in (10); in other words, in the third column of $A_{A-1}^{-1} A_{R+1}^{-1} A_{l,R}^{-1}$. The connecting vector in terms of the axis lengths S_j and link lengths a_j is given by

$$\mathbf{r} = S_R \mathbf{z}_R + a_R \mathbf{x}_R + S_{R+1} \mathbf{z}_{R+1} + a_{R+1} \mathbf{x}_{R+1} + S_{A-1} \mathbf{z}_{A-1} + a_{A-1} \mathbf{x}_{A-1}.$$

Each term in the expansion is transformed to the coordinates of Q_A^- with $\mathbf{z}_R$ from (10), $\mathbf{x}_R, \mathbf{z}_{R+1}$ from (11) and $\mathbf{x}_{R+1}, \mathbf{z}_{A-1}$ from (12).

The right evaluation $\mathcal{E}^R$ proceeds by writing

$$\mathbf{u}_A^+ = A_{l,A} A_{A+1} A_{R-1} \mathbf{u}_R^-, \quad (13)$$

$$\mathbf{u}_A^+ = A_{l,A} A_{A+1} \mathbf{u}_{R-1}^-, \quad (14)$$

$$\mathbf{u}_A^+ = A_{l,A} \mathbf{u}_{A+1}^-. \quad (15)$$

Then the coordinates of $\mathbf{z}_R$ in the frame Q_A^+ are found in the third column of $A_{l,A} A_{A+1} A_{R-1}$, that is by setting $\mathbf{u}_R^- = [0, 0, 1]^T$ in (13). In the right evaluation, the connection vector $\mathbf{r}$ is written as the vector sum proceeding the other way around the loop

$$\mathbf{r} = -(a_{R-1} \mathbf{x}_{R-1} + S_{R-1} \mathbf{z}_{R-1} + a_{A+1} \mathbf{x}_{A+1} + S_{A+1} \mathbf{z}_{A+1} + a_A \mathbf{x}_A + S_A \mathbf{z}_A).$$

Each term in this expansion is then written in the coordinate frame of Q_A^+ , with $\mathbf{x}_{R-1}$ from (13), $\mathbf{z}_{R-1}, \mathbf{x}_{A+1}$ from (14) and $\mathbf{z}_{A+1}$ from (15).

After deriving the vectors and equations symbolically using *MAPLE*, we seek to eliminate all terms involving monomials in $c_{R+1}, c_{A-1}, s_{R+1}, s_{A-1}$ and their products with t_A . There are a total of 16 such terms and only 14 equations in the system. An additional six equations, $(E_{11}) - (E_{16})$, are formulated by multiplying both sides of equations $(E_5) - (E_{10})$ by t_A . This process does not introduce additional monomials as $(E_5) - (E_{10})$ do not involve t_A . To perform an optimally conditioned first elimination, we compute the SVD [25] of the left coefficient matrix and multiply both sides of the equation by its left null-vectors. We proceed by describing a procedure for computing the null-vectors using a complex version of the SVD, taking advantage of the inherently complex structure of the system.

2.2. Complexifying the first elimination. The entire system of equations $(E_1) - (E_{16})$ is written as a 20×16 matrix equation $B\mathbf{b} = P\mathbf{c} + \mathbf{q}t_A + \mathbf{r}$, given in block form as

$$\begin{pmatrix} B_1 & B_2 \\ B_3 & 0 \\ -B_2 & B_1 \\ 0 & B_3 \end{pmatrix} \begin{pmatrix} t_A \mathbf{b}' \\ \mathbf{b}' \end{pmatrix} = \begin{pmatrix} P_1 & P_2 \\ P_3 & 0 \\ P_2 & -P_1 \\ 0 & P_3 \end{pmatrix} \begin{pmatrix} t_A \mathbf{c}' \\ \mathbf{c}' \end{pmatrix} + \begin{pmatrix} \mathbf{q}_1 \\ \mathbf{q}_2 \\ \mathbf{q}_3 \\ 0 \end{pmatrix} t_A + \begin{pmatrix} \mathbf{r}_1 \\ 0 \\ \mathbf{r}_3 \\ \mathbf{r}_4 \end{pmatrix}, \quad (16)$$

where $\mathbf{b}', \mathbf{c}'$ are given in (1) and (2). The matrix B denoted in block form with 4×8 blocks B_1 and B_2 from the left evaluations of $(E_{1,R}) - (E_{4,R})$; and the 6×8 block B_3 from the left evaluations of $(E_5) - (E_{10})$, each holding the constant coefficients multiplying the vector of unknowns $\mathbf{b}$. Similarly for the right-hand side coefficient matrix P and vector of unknowns $\mathbf{c}$, corresponding to the sines and cosines of $\theta_{A+1}, \theta_{R-1}$. Finally, the 20×1 vector $\mathbf{q}$ holds all coefficients of each equation that multiply only the unknown t_A and $\mathbf{r}$ holds all constants. The four equations $(E_{1,R}) - (E_{4,R})$ are written as $\begin{pmatrix} B_1 & B_2 \end{pmatrix} \mathbf{b} = \begin{pmatrix} P_1 & P_2 \end{pmatrix} \mathbf{c} + \mathbf{q}_1 t_A + \mathbf{r}_1$; the six equations $(E_{11}) - (E_{16})$ are written as $B_3 t_A \mathbf{b}' = P_3 t_A \mathbf{c}' + \mathbf{q}_2 t_A$; the equations $(E_{1,I}) - (E_{4,I})$ are written as the subsequent four rows of the system, $\begin{pmatrix} -B_2 & B_1 \end{pmatrix} \mathbf{b} = \begin{pmatrix} P_2 & -P_1 \end{pmatrix} \mathbf{c} + \mathbf{q}_3 t_A + \mathbf{r}_3$; and the last six rows of (16), namely $B_3 \mathbf{b}' = P_3 \mathbf{c}' + \mathbf{r}_4$ hold equations $(E_5) - (E_{10})$. This particular ordering of equations is chosen to reveal the complex structure of B .

The coefficient matrix B on the left-hand side of (16) is the real representation of the complex matrix C

$$B = \begin{pmatrix} B_1 & B_2 \\ B_3 & 0 \\ -B_2 & B_1 \\ 0 & B_3 \end{pmatrix}, \quad C = \begin{pmatrix} B_1 \\ B_3 \end{pmatrix} - i \begin{pmatrix} B_2 \\ 0 \end{pmatrix}.$$

We proceed by computing the complex SVD of C , then block multiplying both sides of (16) by the left singular vectors of B reconstructed from the left singular vectors of C . This technique is motivated by the observation that the set of singular values of B are the set of singular values of C , each with twice the multiplicity; moreover, the singular vectors of B are rearrangements of the real and imaginary parts of the singular vectors of C . The reduction of a 20×16 to a 10×8 matrix decomposition justifies the use of complex arithmetic, as the calculation of the SVD is one of the computationally intensive steps in the entire algorithm. The full description of the relation between the SVD of complex matrix and that of its real representation is discussed in the App. C.

The SVD of C is obtained for instance by the routine ZGESVD in LAPACK [1]. Assume $C = U_C \Sigma_C V_C^H$, with U_C and V_C complex unitary, and V_C^H the Hermitian conjugate of V_C . Σ_C is the real 10×8 matrix whose upper left-hand block is the diagonal matrix containing the singular values of C in descending order, with remaining entries of zero. Write U_C in explicit complex form as

$$U_C = \begin{pmatrix} U_{R,1} + iU_{I,1} & \cdots & U_{R,10} + iU_{I,10} \end{pmatrix}.$$

Assume $B = U \Sigma V^T$ where U and V are real orthogonal matrices, and Σ is the real 20×16 matrix whose upper left-hand block is the diagonal matrix of singular values of B in descending order, with remaining entries of zero. Then U can be obtained from U_C as

$$U = \begin{pmatrix} U_{R,1} & -U_{I,1} & \cdots & U_{R,10} & -U_{I,10} \\ U_{I,1} & U_{R,1} & \cdots & U_{I,10} & U_{R,10} \end{pmatrix}.$$

The left null-vectors of B are the last 4 columns of U , denoted here as the 20×4 matrix U_N ,

$$U_N = \begin{pmatrix} U_{R,9} & -U_{I,9} & U_{R,10} & -U_{I,10} \\ U_{I,9} & U_{R,9} & U_{I,10} & U_{R,10} \end{pmatrix}.$$

Define the 20×9 coefficient matrix P_α from the first 8 columns of P concatenated with $\mathbf{q}$, and the 20×9 matrix P_β from the last 8 columns of P concatenated with $\mathbf{r}$,

$$P_\alpha = \begin{pmatrix} P_1 & \mathbf{q}_1 \\ P_3 & \mathbf{q}_2 \\ P_2 & \mathbf{q}_3 \\ 0 & 0 \end{pmatrix}, \quad P_\beta = \begin{pmatrix} P_2 & \mathbf{r}_1 \\ 0 & 0 \\ -P_1 & \mathbf{r}_3 \\ P_3 & \mathbf{r}_4 \end{pmatrix}.$$

Equation (16) can now be written more compactly as

$$B\mathbf{b} = \begin{pmatrix} P_\alpha t_A + P_\beta \end{pmatrix} \begin{pmatrix} \mathbf{c}' \\ 1 \end{pmatrix}. \quad (17)$$

Since $U_N^T B = 0$, left-multiplying (17) with U_N^T eliminates the sines and cosines of θ_{R+1} and θ_{A-1} , as contained in $\mathbf{b}$. The resulting 4×9 system

$$\begin{pmatrix} U_N^T P_\alpha t_A + U_N^T P_\beta \end{pmatrix} \begin{pmatrix} \mathbf{c}' \\ 1 \end{pmatrix} = \begin{pmatrix} P_\alpha^U t_A + P_\beta^U \end{pmatrix} \begin{pmatrix} \mathbf{c}' \\ 1 \end{pmatrix} = 0, \quad (18)$$

is optimally conditioned, and the process allows a stable elimination of the first set of variables and computation of the resultant for the remaining ones. $\mathbf{c}'$ was defined in (2).

2.3. Formulation of the generalized eigenvalue problem. The next step in the solution process is the conversion of the sines and cosines of θ_{A+1} and θ_{R-1} to their respective half-tangents in order to reformulate the system as a 16×16 generalized eigenvalue problem for eigenvalue t_{A+1} . The two remaining unknown half-tangents corresponding to (18), t_A and t_{R-1} , are extracted from the corresponding eigenvector for each real eigenvalue. First, the 4×9 system (18) is transformed into an 8×8 polynomial system in the half-tangents of θ_A, θ_{A+1} and θ_{R-1} as follows. The sines and cosines of θ_{A+1} and θ_{R-1} in $\mathbf{c}'$ (2) are converted by the half-tangent identities $s_j = 2t_j/(1+t_j^2)$ and $c_j = (1-t_j^2)/(1+t_j^2)$. Multiplying through by $(1+t_{A+1}^2)(1+t_{R-1}^2)$ to clear denominators transforms the system (18) to a system of four polynomial equations. Each of these is quadratic in t_{A+1} , quadratic in t_{R-1} and linear in t_A . Multiplying this first set of equations by t_{R-1} yields a second set

which is cubic in t_{R-1} , quadratic in t_{A+1} and linear in t_A . The coefficients of (18) are thereby sorted into three coefficient matrices by powers of t_{A+1} , namely the 8×8 matrices H_2 , H_1 and H_0 which satisfy

$$(t_{A+1}^2 H_2 + t_{A+1} H_1 + H_0) \mathbf{v} = 0, \quad (19)$$

with the unknowns held in vector $\mathbf{v}$ given in (3).

The matrix polynomial problem (19) is rewritten as the generalized eigenvalue problem

$$\begin{pmatrix} O & I \\ -H_0 & -H_1 \end{pmatrix} \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{pmatrix} = \Lambda \begin{pmatrix} I & O \\ O & H_2 \end{pmatrix} \begin{pmatrix} \mathbf{v}_1 \\ \mathbf{v}_2 \end{pmatrix}. \quad (20)$$

An in-depth discussion of the solution to the matrix polynomial (19) posed as the generalized eigenvalue problem (20) may be found in [7, 17, 18], and the references therein. In particular, it is established that $\{\Lambda_k\}_{k=1}^{16}$, the set of eigenvalues of (20) are the solutions of (19), and the real eigenvalues correspond to solutions of the physical problem. Equation (20) is solved for instance using the LAPACK routine DGGEV. The structure of (20) with respect to (19) is revealed by expanding (20) by blocks, yielding

$$\mathbf{v}_2 = \Lambda \mathbf{v}_1, \quad (21)$$

$$-H_0 \mathbf{v}_1 - H_1 \mathbf{v}_2 = \Lambda H_2 \mathbf{v}_2. \quad (22)$$

For the eigenvalue $\Lambda = t_{A+1}$, applying (21) to (22), we obtain

$$(t_{A+1}^2 H_2 + t_{A+1} H_1 + H_0) \mathbf{v}_1 = 0, \quad (23)$$

that is, for each real eigenvalue t_{A+1} , the components of $\mathbf{v}_1$ are the unknowns of $\mathbf{v}$ in (3). To obtain the correct normalization, take for instance $t_A = v_{2,4}/v_{2,8}$, and $t_{R-1} = v_{2,7}/v_{2,8}$, where $v_{2,j}$ is the j th component of vector $\mathbf{v}_2 = \Lambda \mathbf{v}_1$. The generalized eigenproblem has the optimal size of 16×16 , and it allows handling of exceptional cases, such as joint angle values equal to π . We also remark that the choice of multiplying t_{R-1} with the first set of equations and expanding the matrix polynomial (19) in terms of t_{A+1} is arbitrary.

2.4. Solving for remaining three variables. The remaining variables θ_{R+1} and θ_{A-1} may be obtained by taking advantage of the previously computed SVD of B in system (17) where the left-singular vectors and right-singular vectors were found. For the last variable θ_R , we will calculate the joint angle among three vectors by using the expressions in (10)-(15). Since the numerical values of the sines and cosines of θ_A , θ_{A+1} and θ_{R-1} are now considered known, they can be used in the following calculations.

To find θ_{R+1} and θ_{A-1} , we first substitute three known variables $t_A, \theta_{A+1}, \theta_{R-1}$ into the system (17) and now the right hand side of (17) is a known 20×16 matrix. We next form the pseudo-inverse $B^\dagger$ of the coefficient matrix $B = U\Sigma V^T$, where the factors U, Σ, V have already been computed in the first elimination step. From the economy SVD, $B = \hat{U}\hat{\Sigma}\hat{V}^T$ where $\hat{U}$ contains the left singular vectors of B , $\hat{V}$ contains the right singular vectors of B , and $\hat{\Sigma}$ is the (square) diagonal matrix containing the singular values of B , the pseudo-inverse $B^\dagger$ is given by $B^\dagger = \hat{V}\hat{\Sigma}^{-1}\hat{U}^T$, and satisfies $B^\dagger B \mathbf{b} = \mathbf{b}$, revealing the remaining unknowns residing on the left-hand side of (17) by

$$\mathbf{b} = B^\dagger (P_\alpha t_A + P_\beta) \begin{pmatrix} \mathbf{c}' \\ 1 \end{pmatrix}. \quad (24)$$

As the right hand side in the equation above is known, we extract the values for sines and cosines of $\theta_{R+1}, \theta_{A-1}$ by selecting appropriate components of $\mathbf{b}$. Applying the arctangent function *atan2*, we then can get the values for $\theta_{R+1}, \theta_{A-1}$. Take for instance $\theta_{R+1} = \arctan(b_{16}, b_{15})$ and $\theta_{A-1} = \arctan(b_{14}, b_{11})$.

The last unknown θ_R is recovered using three vectors $\mathbf{x}_{R-1}, \mathbf{z}_R, \mathbf{x}_R$ expressed in the same frame. From equations (10)-(15), we already have the expressions of the vectors $\mathbf{z}_R, \mathbf{x}_R$ in the frame Q_A^- and $\mathbf{x}_{R-1}^+$, the expression of the vector $\mathbf{x}_{R-1}$ in the frame Q_A^+ . Then we only need to apply one more transformation to $\mathbf{x}_{R-1}$ and express it in the frame Q_A^- : $\mathbf{x}_{R-1}^- := A_{a,A} \mathbf{x}_{R-1}^+$. Now all the three vectors are expressed in the same frame Q_A^- and the joint angle θ_R may be computed among $\mathbf{x}_{R-1}^-, \mathbf{z}_R, \mathbf{x}_R$ and then all 6 of the unknown joint angles may be identified for each real solution of the generalized eigenvalue problem (20).

3. Computational details and discussion of special cases. The stable implementation of the method requires further consideration. Let us start from the first identified variable t_{A+1} . The solutions for θ_{A+1} will be obtained once we get the eigenvalues. After obtaining the eigenvalues t_{A+1} , we will use eigenvectors (which have 16 components) to calculate t_A, t_{R-1} . In the case of a simple eigenvalue the eigenvector $\mathbf{V}$ is comprised of constant multiples of various monomials in these variables:

$$\mathbf{V} = [\mathbf{v}_1 ; \mathbf{v}_2 = t_{A+1} \mathbf{v}_1] , \quad \mathbf{v}_1 = [t_A \mathbf{h} ; \mathbf{h}] , \quad \mathbf{h} = [t_{R-1}^3, t_{R-1}^2, t_{R-1}, 1]^T .$$

The key here is that in the eigenproblem solver the eigenvector will be scaled by the largest component. To obtain the solution as accurately as possible, we will select the octad containing the maximal component. Either the first 8 components $\mathbf{v}_1$ or the last 8 components $\mathbf{v}_2$ will be used.

If $|t_{A+1}| > 1$, the maximum is found among the last 8 components and $\mathbf{v}_2$ will be selected to calculate t_A, t_{R-1} . Otherwise, $\mathbf{v}_1$ will be chosen. This holds also in the limits $|t_{A+1}| \rightarrow \infty$ and $|t_{A+1}| \rightarrow 0$. Here we denote the 8 component vectors as $\mathbf{v}_m, m = 1$ or 2 . The index of the maximal component in $\mathbf{v}_m$ depends on the magnitude of $|t_A|, |t_{R-1}|$ as shown in Table 2 where details of the computation for each case are given. Cases where at least one of the variables $\theta_A, \theta_{A+1}, \theta_{R-1}$ is close to π and the corresponding t_A, t_{A+1}, t_{R-1} goes to ∞ require no additional treatment.

After solving for the first three variables, the pseudoinverse is used to calculate $\theta_{A-1}, \theta_{R+1}$ in (24). Since the vector $\mathbf{b}$ also contains t_A which might be large in magnitude, the maximal component of $\mathbf{b}$ must be determined. If $|t_A| < 1$ and the maximal entry appears among the last 8 components of $\mathbf{b}$, we will calculate $\theta_{R+1}, \theta_{A-1}$ as $\theta_{R+1} = \arctan(b_{16}, b_{15})$ and $\theta_{A-1} = \arctan(b_{14}, b_{11})$. Otherwise we will apply the first 8 components of $\mathbf{b}$. One more test is needed to determine which components are selected. If $|b_3| > |b_6|$ (that is, $|c_{A-1}| > |s_{A-1}|$), then $\theta_{R+1} = \arctan(b_2/b_3, b_1/b_3)$. Otherwise $\theta_{R+1} = \arctan(b_5/b_6, b_4/b_6)$. If $|b_7| > |b_8|$ (that is, $|c_{R+1}| > |s_{R+1}|$) then $\theta_{A-1} = \arctan(b_4/b_7, b_1/b_7)$. Otherwise $\theta_{A-1} = \arctan(b_5/b_8, b_2/b_8)$. The results are summarized in the last four columns of Table 2. The procedures given above will produce the final solutions correctly for all regular cases.

Discussion of the difficulty of the possible singular case where one variable is close to π without further analysis has appeared e.g. in [18, 20, 10], and this case can spontaneously arise in the study of macromolecular chains [6, 16] as was also

Max	$ t_A $	$ t_{R-1} $	t_A	t_{R-1}	$\mathbf{b}$	θ_{R+1}	$\mathbf{b}$	θ_{A-1}
$ v_{m,1} $	≥ 1	≥ 1	$\frac{v_{m,1}}{v_{m,5}}$	$\frac{v_{m,1}}{v_{m,2}}$	$ b_3 > b_6 $	$\text{atan}\left(\frac{b_2}{b_3}, \frac{b_1}{b_3}\right)$	$ b_7 > b_8 $	$\text{atan}\left(\frac{b_4}{b_7}, \frac{b_1}{b_7}\right)$
$ v_{m,4} $	≥ 1	< 1	$\frac{v_{m,4}}{v_{m,8}}$	$\frac{v_{m,3}}{v_{m,4}}$	$ b_3 < b_6 $	$\text{atan}\left(\frac{b_5}{b_6}, \frac{b_4}{b_6}\right)$	$ b_7 < b_8 $	$\text{atan}\left(\frac{b_5}{b_8}, \frac{b_2}{b_8}\right)$
$ v_{m,5} $	< 1	≥ 1	$\frac{v_{m,1}}{v_{m,5}}$	$\frac{v_{m,5}}{v_{m,6}}$		$\text{atan}(b_{16}, b_{15})$		$\text{atan}(b_{14}, b_{11})$
$ v_{m,8} $	< 1	< 1	$\frac{v_{m,4}}{v_{m,8}}$	$\frac{v_{m,7}}{v_{m,8}}$		""		""

TABLE 2. Computing variables. Double vertical lines indicate separate tests, with all cases hinging on the magnitude of $|t_A|$: first, t_A and t_{R-1} found from eigenvector components, depending on the magnitude of $|t_{R-1}|$; then θ_{R+1} and θ_{A-1} are found depending on relative sizes of certain components in $\mathbf{b}$.

pointed out in [11]. Following the method in [12], Kim and Chirikjian [11] introduce a Jacobian-based method to handle this issue, presumably due to failure of the basic method. However, in the present formulation, the case where certain joint angles are equal to π is naturally handled by the general approach.

In contrast, the presence of multiple eigenvalues requires special consideration. It was mentioned in [18] that multiple eigenvalue cases may arise in exceptional situations. The key difficulty, given a stable computation of the invariant subspace of a multiple eigenvalue, is that the corresponding eigenvectors may not follow the algebraic monomial pattern as expected and the formulas above will not produce the correct solutions. One way to address this problem is to change the reference (R) and arm (A) in the 6R6B system: there exist 6 possible choices for the reference and the eigenproblem might be simple for one of them. However, as shown in the examples, special cases can exist for which regardless of reference some of the eigenvalues are still multiple. Then we must resort to the modification procedures as discussed in the Appendix. LAPACK 3.10.0 used in the computations presented here can handle multiple eigenvalues stably, while that was not the case for earlier versions. As a result, it is now possible to take advantage of basic linear space theory to transform the eigenvector basis of the invariant subspace to a basis vector set that is congruent to the required monomial structure from which the other variables may be extracted. In robotics, continuous movement of the system is desired as a result of parameter perturbation. When changes in the system may cause an instance of a multiple crossing of eigenvalues, our method can still maintain accuracy so that different solutions could be tracked continuously through the crossing. By comparison, the Jacobian continuation method discussed in [11] can identify the changing process of specific eigenvalues and correctly follow the movement of the system, however at the price of increased computational cost.

Consider a double eigenvalue $\rho_1 \approx \rho_2$ as an example. One tolerance tol_1 , typically less than $1e-14$, is used to determine the equality of eigenvalues. Numerical accuracy can also be at issue, as the tolerance to identify equality may be dependent on the matrix pencil of the eigenproblem. The solver will produce two basis vectors $\mathbf{V}_1$ and $\mathbf{V}_2$ for the two dimensional subspace, and in general we will need to apply extra treatment in order to uncover the unique basis pair possessing the required algebraic

monomial structure. An additional complication results from the presence of several variables of higher multiplicity or the presence of infinity and zero in t_A, t_{R-1} . Then we need to determine if it is a special case by checking certain components in $\mathbf{V}_1, \mathbf{V}_2$. Here we only show the determination for ‘regular double’ and ‘double-double’ cases where t_A, t_{R-1} are not zero or infinity. The detailed description and pseudocode for the determination of all cases is given in the App. D. The ‘regular double’ means that the eigenvalue t_{A+1} is double and t_{R-1} is not, while ‘double-double’ means t_{R-1} is double as well. Because of our construction, the double eigenvalue with only double t_A is treated as regular double. Double-double cases are identified by checking if certain ratios between components in $\mathbf{V}_1, \mathbf{V}_2$ are within another tolerance tol_2 (typically 5e-12), as follows:

```

! Test if eigenvalues  $\rho_1, \rho_2$  are equal, define  $DIF_{ij} = \left| \frac{V_{1,i}}{V_{1,j}} - \frac{V_{2,i}}{V_{2,j}} \right|$ 
IF  $|\rho_1 - \rho_2| < tol_1$  or  $\left| \frac{1}{\rho_1} - \frac{1}{\rho_2} \right| < tol_1$ 
    IF  $DIF_{78}, DIF_{58}, DIF_{34}, DIF_{14} < tol_2, \left| \frac{V_{1,3}}{V_{1,4}} - \frac{V_{1,7}}{V_{1,8}} \right| < tol_2$  THEN
        apply treatment for double-double case
    ELSE
        apply treatment for regular double cases

```

This method also works for the triple case: tol_1 will be applied to test if ρ_3 is equal to ρ_1 and ρ_2 . In regard to higher multiplicity, we have only considered as far as the triple-double case and provided special treatment. The ‘triple’ in triple-double means the eigenvalue t_{A+1} is triple and ‘double’ means that among the corresponding three solutions for t_{R-1} two have equal values. Our method to handle triple-double case however is different from double-double case above. The treatment for regular triple case will be applied first and three solutions for t_{R-1} will be obtained, denoted as DT_1 . Next we choose any two eigenvectors and apply the regular double treatment to them and get two solutions for t_{R-1} , denoted as DT_2 . If two values of DT_2 are in DT_1 , then it is actually triple-double case, otherwise it is regular triple. The logic introduced to treat triple cases is as follows:

```

! Test if eigenvalues  $\rho_1, \rho_2, \rho_3$  are equal
IF  $|\rho_i - \rho_j| < tol_1$  or  $\left| \frac{1}{\rho_i} - \frac{1}{\rho_j} \right| < tol_1$ , for all  $i < j \leq 3$ 
    apply regular triple treatment, get  $DT_1$ 
    apply regular double treatment to  $\mathbf{V}_i, \mathbf{V}_j$ , get  $DT_2$ 
    IF  $DT_2$  in  $DT_1$  THEN
        apply treatment for triple-double case
    ELSE
        apply treatment for regular triple case

```

We note that cases where all half-tangent variables are simultaneously double – representing a common first-order tangency of the surfaces associated with the underlying polynomial system – can not be treated by this method, as a double eigenvalue would then be defective with only a single eigenvector which cannot be handled by the eigensolver. As a result, defective cases are typically not identifiable by checking the eigenvectors from the solver or solvable by the method. However one special case might be identified through the components in eigenvectors, specifically if t_A, t_{R+1} are both double infinity. Such a case will be discussed in Example 5 below.

4. Numerical results. In our method, we employ the left null vectors of matrix B (16) as obtained from the SVD to efficiently arrive at the resultant (18) while complexifying the SVD increases computational efficiency. The pseudoinverse $B^\dagger$ is used to stably determine the eliminated variables. The 16 dimensional generalized eigenvalue problem as derived in our method as well as in [13, 21, 12, 15, 22] corresponds to the smallest possible dimension of the problem. The solver for the generalized eigenvalue problem is used in our method, as well as in [18, 21]. An alternative approach is to multiply the inverse of the right matrix (like in [12, 15]) to convert into an ordinary eigenvalue problem. The key advantage of the generalized eigenproblem approach is in enabling the correct handling of the case of infinite eigenvalues, corresponding to a joint angle equal to π . In this section, we first compare the performance of our method with the results from two state of the art methods as analyzed in [3]. Then several special cases are designed to show the stability of our method, for instance the presence of π in the solution and the case of multiple eigenvalues. Two special singular cases are discussed at the end.

4.1. General Performance. To benchmark the performance of R6B6 in comparison to what may be considered the state of the art methods, we examine four manipulators discussed in [3]: Raghavan and Roth Manipulator (abbreviated as RRM), IACE Laboratory Manipulator (ILM), Paint Manipulator (PTM) and the Manocha and Canny Manipulator (MCM). Detailed description of manipulators can be found in [3]. In that paper, two methods [10, 15] were compared using 2500 tuples of 6 joint angles. The 6-tuples were first randomly chosen and used to construct the end effectors. Based on the end effectors, the two methods were applied respectively to check if the corresponding 6-tuples of joint angles can be recovered. Here we follow the same logic to check the performance of R6B6 with respect to the above manipulators. We choose 2500 random tuples of joint angles in the range $[-\pi, \pi]$ to construct the end effectors and then solve the inverse kinematics using our algorithm written in *C++*. The program is compiled on Mac using Clang version 12.0.0. LAPACK 3.10.0 routines written in Fortran 90 are compiled by gfortran version GNU 8.2.0.

All the cases tested were solved correctly by our method. The timings are given in Table. 3. The timings for the three major steps are given separately. The timing of conversion contains the transformation between open chain and closed ring and the timing for calculating the remaining variables. The SVD timing is for the SVD function and the eigenvalue step is for the generalized eigenvalue computation. As can be seen in the table, the average total time for all the manipulators is around 0.5 ms and the average errors for all the joint angles are $8.1\text{e-}12$, $1.6\text{e-}13$, $1.3\text{e-}12$, $6.7\text{e-}14$ respectively. The maximum errors (not shown in table) are $1.4\text{e-}8$, $1.1\text{e-}10$, $1.3\text{e-}9$, $5.0\text{e-}11$. To further demonstrate the stability of the algorithm, the closest solution found in each case to the input joint angles was used to compute the closure error, i.e. the difference between the posed end effector and the one found from these angles with a forward kinematic construction. The average Frobenius matrix norm [25] of the difference between the posed and reconstructed end effectors is $1.3\text{e-}12$, $8.6\text{e-}14$, $3.8\text{e-}13$, $3.7\text{e-}14$ (with maximum values at $1.7\text{e-}9$, $3.5\text{e-}11$, $2.2\text{e-}10$, $6.4\text{e-}12$), thus verifying the stability of the solution and reconstruction processes. The computations were performed on a Macbook with Intel Core i5 Dual-Core, 2.7 GHz and 8GB RAM.

According to the results in [3] (performed on a PC with AMD Phenom processor, 3.2 GHz and 4 GB RAM), the timing for the manipulators with the method in [10]

and [15] is more than 10 ms and 1 ms respectively. We note that the error results for the later two methods refer to solved cases, while both methods fail for certain cases. The timings, corresponding smallest errors and failure rate (%) of Liu-Zhu (**LZ**) method and Husty-Pfurner (**HP**) method are included in Table 4. No failures were observed for our method, even after repeating the test for 25000 random 6-tuples. We discovered one intrinsically singular case in MCM, as discussed in example 7 below.

Steps	RRM	ILM	PTM	MCM
Conversion	97.38	97.99	96.87	110.57
SVD	102.56	103.24	103.76	107.71
Eigenvalue	165.80	154.76	170.28	166.16
Total (avg.)	482.85	454.33	491.75	507.67
Total (max.)	2222.0	2878.0	2389.0	2526.0
Total (min.)	405.0	388.0	403.0	403.0

TABLE 3. Computation times (in μs) for the various stages of R6B6 algorithm.

		RRM	ILM	PTM	MCM
Average execution time (ms)	R6B6	0.5	0.5	0.5	0.5
	LZ	1.2	2.4	2.4	1.9
	HP	9.7	9.7	9.6	9.7
Error (rad.)	R6B6	8.1e-12	1.6e-13	1.3e-12	6.7e-14
	LZ	1.7e-11	4.3e-5	2.8e-6	1.6e-8
	HP	1.4e-7	3.9e-8	8.6e-6	2.3e-4
Failure rate (%)	R6B6	0	0	0	0
	LZ	0.1	100	100	0
	HP	0	0.3	0.32	3.7

TABLE 4. Comparative results for R6B6, LZ and HP algorithms. Average computation time (in ms), average errors of solved angles in radians and failure rates are given. The results shown for the other two methods are as reported in [3]. The LZ method failed 100% on the ILM and PTM manipulators, so the error reported corresponds to slightly perturbed manipulator parameters.

We next continue testing for some special examples. The list of seven examples is given in Table 5. The first example, involving $t_{A+1} = \infty$, is shown to follow the general case with no additional treatment, although three of the unknown joint angles in one solution take the value π . We then give four multiple eigenvalues examples, including instances of double and triple eigenvalues with additional multiple variables. The second example, where a double infinite eigenvalue is discussed,

	Example #	Description	# solution
π case	1	$\theta_{A+1}, \theta_A, \theta_{R-1}$ in one solution are π	6
Multiple	2	Double ∞ eigenvalue	8
	3	Special double eigenvalues	8
	4	Special triple eigenvalues	8
	5	Double 0 eigenvalue, adjacent double ∞	4
0 inputs	6	One zero-length parameter	2
	7	Many zero-length parameters	10

TABLE 5. List of information for all special examples.

is shown to be amenable to the method in App. D.1. There follow two examples, the third one with a double and the fourth with a triple eigenvalue, both including additional double variables, for which the modifications given in App. D.2,D.3,D.4 will suffice. In the fifth example, a case is considered where there exists a double eigenvalue $t_{A+1} = 0$ with $t_A = t_{R-1} = \infty$. This case appears to be defective, as discussed in App. D.2. Finally, the sixth and seventh examples discuss the cases of zeros in the input parameters, and possible failures related to singular axes systems. The numerical analysis and discussion of the special features are provided in the corresponding examples.

4.2. Example 1: simple π case. If one of the variables $\theta_A, \theta_{A+1}, \theta_{R-1}$ involved in the eigenproblem through a half-tangent representation is close to π , it is stated in [18] and [11] that the system will be singular and extra iterations are needed to improve the accuracy. This is because the half-tangent of a joint angle approaching π goes to infinity which may introduce errors in the solution process. On the other hand, seen from the perspective of projective space, the point at infinity is just another regular point. In the eigenvalue solver in LAPACK as an example, an eigenvalue is represented by the ratio of a pair of parameters α and β . A vanishing β with nonzero α indicates an infinite eigenvalue, and no additional iterative method is required to obtain an accurate solution. Our method takes advantage of this: as outlined above, we only need to identify appropriate components in the computed eigenvectors to achieve accurate solutions. The input linkage parameters for the first example are shown in Table 6 and solutions are given in Table 7. In the first solution, the 4th, 5th and 6th variables are $-\pi$ which correspond to $A, A+1, R-1$ respectively for reference $R = 1$. This case shows that the eigenvalue goes to ∞ while other two variables obtained from the eigenvector also go to ∞ . For example 1, the 4×4 end effector matrix is

$$\begin{bmatrix} \mathbf{x} & \mathbf{y} & \mathbf{z} & \mathbf{d} \\ 0 & 0 & 0 & 1 \end{bmatrix}, \quad (25)$$

$$\begin{aligned} \mathbf{x} &= [0.935729747639523, -0.104687021946279, 0.336824088833465]^T, \\ \mathbf{y} &= [-0.266206316527123, -0.836082865520739, 0.479687021946279]^T, \\ \mathbf{z} &= [0.231395843574460, -0.538522115997707, -0.810215955259964]^T, \\ \mathbf{d} &= [-4.712089111505835, -10.150881856679735, 8.892349540785121]^T. \end{aligned}$$

Joint i	a_i	S_i	$\alpha_i(\text{deg})$
1	1.00	9.00	90.00
2	4.00	1.00	-90.00
3	8.00	2.00	40.00
4	1.25	3.00	90.00
5	2.00	7.00	30.00
6	3.00	3.00	50.00

TABLE 6. Linkage parameters of example 1.

No.	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
1	80.0000	80.0000	110.0000	180.0000	-180.0000	180.0000
2	-108.4903	-65.5094	-130.2246	-129.6402	-157.3853	-168.0230
3	119.0163	-152.6296	131.2120	7.9542	-148.4491	5.1112
4	146.9569	-80.0075	155.6435	-11.9434	104.8319	-179.3449
5	6.8777	81.0330	-172.4691	165.2536	-9.9985	-61.1845
6	-126.9544	-42.7833	-64.5878	-141.2330	-63.8036	70.0651

TABLE 7. Solutions of joint angles (in degrees) for example 1.

4.3. Examples 2-5: Multiple eigenvalues. The chains in examples 2-4 are closed chains involving 6 rotors with vanishing links. Here the end effector reaches the origin so its orientation is given by a 4×4 identity matrix. We use internal parameters, i.e. bond lengths b_l , bond angles b_a and torsional angles t_a as defined in App. A, to pose these three examples, where the first two parameters define the linkage while the torsional angles are to be determined in the solutions. Example 2 entails a double eigenvalue, demonstrating in addition that an infinite double eigenvalue poses no special difficulties. Examples 3 and 4 correspond to double and triple eigenvalues respectively, both with additional double variables. Example 5 exhibits a case where the variables adjacent to the double eigenvalue are both infinite.

4.3.1. Example 2: Double π case. Example 2 shows a case containing an infinite eigenvalue of multiplicity two. The internal parameters and transformed DH parameters of the system are given in Table 8 where the torsions t_a are left unspecified because they will not be used in the solution process. The solutions are given in Table 9. The system is solved by setting the reference as $R = 1$. The fifth variable in solutions 7 and 8 is π , so that we have a double eigenvalue at infinity. This case can be handled by the treatment for regular double eigenvalues in App. D.1.

4.3.2. Example 3: Special double eigenvalue case. The internal parameters and transformed DH parameters of example 3 are given in Table 10. Solving the system, we get 8 solutions as in Table 11. It may be verified that the second, third, fifth and sixth variables of the solutions have two values that are double. Setting the reference R as any of 1, 2, 4, 5 results in double eigenvalues, while for references 3 or 6 the system can be solved as a regular case. With varying references, the constructed system and the matrix pencil would change, requiring adjustments to the tolerances for determining the equality of eigenvalues. The tolerances for references $R = 1, 2, 4, 5$ are 112e-16, 103e-16, 44e-16, 34e-16, respectively. In fact, this example is a double-double case where two consecutive variables can be both

Index i	b_{l_i}	b_{a_i}	t_{a_i}	a_i	S_i	α_i
1	3.00	90.00	—	0.00	3.00	-90.00
2	2.00	90.00	—	0.00	2.00	-90.00
3	2.00	90.00	—	0.00	2.00	-90.00
4	1.00	90.00	—	0.00	1.00	-90.00
5	3.00	90.00	—	0.00	3.00	-90.00
6	1.00	90.00	—	0.00	1.00	-90.00

TABLE 8. Example 2. Left: Internal parameters; Right: DH parameters.

No.	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
1	99.6641	-64.8363	118.1837	-93.2078	116.6642	-63.2178
2	-99.6641	64.8363	-118.1837	93.2078	-116.6642	63.2178
3	-122.1432	51.4420	35.6079	-100.2152	42.2821	42.5890
4	47.9564	50.6006	-135.6256	77.1011	36.0662	-66.6271
5	122.1432	-51.4420	-35.6079	100.2152	-42.2821	-42.5890
6	-47.9564	-50.6006	135.6256	-77.1011	-36.0662	66.6271
7	0.0000	70.5288	0.0000	-70.5288	180.0000	-38.9424
8	0.0000	-70.5288	0.0000	70.5288	180.0000	38.9424

TABLE 9. Solutions of joint angles for example 2.

Index i	b_{l_i}	b_{a_i}	t_{a_i}	a_i	S_i	α_i
1	3.00	90.00	—	0.00	3.00	-60.00
2	2.00	120.00	—	0.00	2.00	-140.00
3	2.00	40.00	—	0.00	2.00	-60.00
4	3.00	120.00	—	0.00	3.00	-90.00
5	1.00	90.00	—	0.00	1.00	-80.00
6	1.00	100.00	—	0.00	1.00	-90.00

TABLE 10. Example 3. Left: Internal parameters; Right: DH parameters.

double. In Table 11, the double value in 5th variable correspond to double value in the 6th variable. For reference $R = 1$ (resp. $R = 4$), the eigenvalue $(A + 1)$ is the 5th (resp 2nd) variable and the 6th (resp. 3rd) variable, $R - 1$, is obtained from the eigenvectors. In these cases, we need one more tolerance check to decide the double-double case by testing equality of certain components in the two eigenvectors (see Appendix, section D.2). The tolerance is 5e-13 and 4e-14 for references 1 and 4 respectively. The reason is that these cases involve a double value for t_{R-1} which requires special treatment. On the other hand references $R = 2$ and $R = 5$ involve a double value for t_A which is in fact treatable by the regular double approach.

4.3.3. *Example 4: Special triple eigenvalue case.* The parameters of example 4 are given in Table 12 and the 8 solutions are given in Table 13, as computed in reference $R = 1$. Any other reference frame, with the exception of reference 4 (see

No.	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
1	27.7358	-79.8312	79.8312	-27.7358	-87.5627	87.5627
2	-27.7358	79.8312	-79.8312	27.7358	87.5627	-87.5627
3	107.6847	-79.8312	79.8312	-107.6847	87.5627	-87.5627
4	-107.6847	79.8312	-79.8312	107.6847	-87.5627	87.5627
5	111.3982	36.4769	-36.4769	-111.3982	-41.2381	41.2381
6	-111.3982	-36.4769	36.4769	111.3982	41.2381	-41.2381
7	169.2944	36.4769	-36.4769	-169.2944	41.2381	-41.2381
8	-169.2944	-36.4769	36.4769	169.2944	-41.2381	41.2381

TABLE 11. Solutions of joint angles for example 3.

Index i	b_{l_i}	b_{a_i}	t_{a_i}	a_i	S_i	α_i
1	$\sqrt{3}$	30.00	–	0.00	$\sqrt{3}$	-120.00
2	$\sqrt{3}$	60.00	–	0.00	$\sqrt{3}$	-150.00
3	1.00	30.00	–	0.00	1.00	-120.00
4	1.00	60.00	–	0.00	1.00	-120.00
5	1.00	60.00	–	0.00	1.00	-120.00
6	1.00	60.00	–	0.00	1.00	-150.00

TABLE 12. Example 4. Left:Internal parameters; Right: DH parameters.

No.	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
1	-98.8994	98.8994	-34.7781	-145.2219	145.2219	34.7781
2	98.8994	-98.8994	34.7781	145.2219	-145.2219	-34.7781
3	-98.8994	98.8994	-98.8994	145.2219	-145.2219	98.8994
4	98.8994	-98.8994	98.8994	-145.2219	145.2219	-98.8994
5	-98.8994	-34.7781	98.8994	-145.2219	-81.1006	98.8994
6	98.8994	34.7781	-98.8994	145.2219	81.1006	-98.8994
7	34.7781	98.8994	-98.8994	81.1006	145.2219	-98.8994
8	-34.7781	-98.8994	98.8994	-81.1006	-145.2219	98.8994

TABLE 13. Solutions of joint angles for example 4.

below), could have been used, producing solutions of comparable accuracy. From Table 13, we can see that every variable of the solutions has two values that are triple. Consider solution 2, 4 and 6: the first variable 98.8994 is triple, the second variable in solutions 2 and 4 are double, and the sixth variable in solutions 4 and 6 are double as well. The other variables display a similar pattern. We note that no three consecutive variables exhibit simultaneous double values, which as mentioned earlier would result in an ill-posed system. We refer to this as a triple-double case where extra treatment will be applied to obtain the correct solutions as discussed in the App. D.4. There also exists tolerance for deciding equality of eigenvalues as in example 3 above. The tolerances here are 147e-16, 25e-16, 23e-16, 14e-16, 29e-16, 54e-16 for references $R = 1$ through $R = 6$, respectively. Here we must point out that the accuracy of the eigenvalue solver in *C++* is not adequate for $R = 4$, in

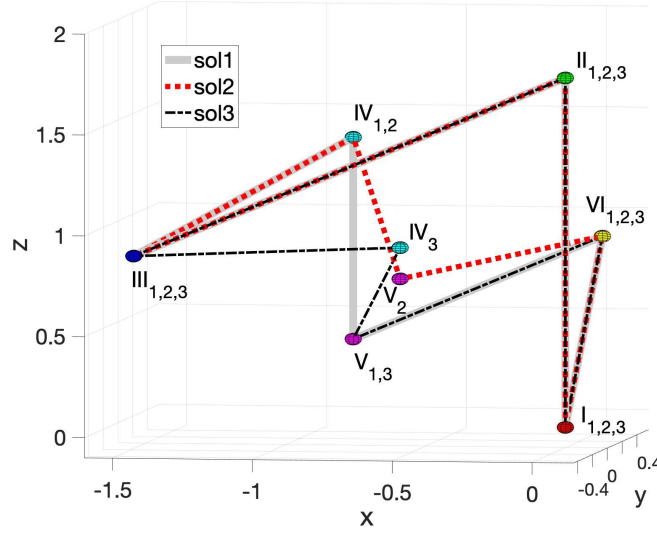


FIGURE 3. Triple multiplicity solutions in example 4. Indices I-VI label the joints, subscripts denote the corresponding solutions.

which some small intermediate values become zero and *NaN* appears in the final solutions. The tolerance under which this issue arises will vary in other systems or software. To illustrate the triple multiplicity solutions, the reconstructed loops corresponding to solutions 2, 4 and 6 from Table 13 are pictured in Fig. 3 with solid grey, dotted red and dot-dashed black lines respectively. In this diagram, indices I-VI label the joints, which are connected by bonds. The first, second and the sixth bonds coincide because the first torsional angles in the illustrated solutions are equal for three rings. The third bond in the first two solutions and the fifth bond in the first and third still coincide because of double multiplicity of the second and sixth respective variables in the solutions.

Next we explore the impact of small perturbations on the equal eigenvalues and the tolerances (discussed in Sec. 3). Setting $R = 1$, the eigenvalues correspond to the fifth variable θ_5 . Perturbing the second bond length in the input parameter, the effect on the eigenvalues corresponding to θ_5 (Table 13, solutions 1, 4 and 7) is shown in Fig. 4. As seen in the figure, the eigenvalues coalesce linearly as the parameter is varied, and the slopes of their trajectories are approximately 4.8, 2.4 and 0. Without perturbations, the condition numbers of these eigenvalues are 4.36, 7.66 and 1125.98 calculated by using the eigenvectors from the solver. For the condition number ν of an eigenvalue Λ we used the expression for regular matrix pairs given in [24] (p. 294), $\nu = \|\mathbf{x}\|_2 \|\mathbf{y}\|_2 / |\mathbf{y}^H \mathbf{x}|$, where $\mathbf{x}$ and $\mathbf{y}$ are the right and left eigenvectors of the matrix pair corresponding to Λ . The condition numbers for the perturbed system with perturbation $1e-10$ are 9.67, 55.39, 48.65, which shows one of the eigenvalues becomes moderately more sensitive as they coalesce. However, the low condition numbers indicate the eigenvectors are stable with respect to this perturbation. Then the method gives an accurate solution for eigenvectors whose components are structured as in (3). As the perturbation increases, the three eigenvalues will diverge and the problem eventually becomes a simple case.

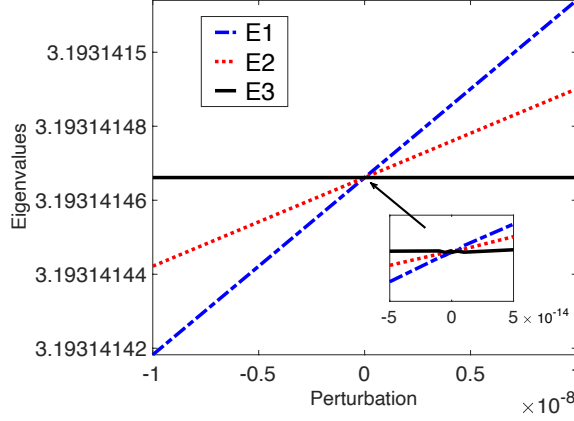


FIGURE 4. Converging process of three equal eigenvalues in example 4.

We manually set a large tolerance to let close eigenvalues be treated as equal and apply the triple case treatment. When the perturbation to the second length is $1e-13$, the tolerance is set to $5e-13$ and the correct solutions can be obtained through the treatment to triple double case, while a small tolerance can not produce closed solutions. As the perturbation is up to $1e-10$, it becomes a simple case where the closed solutions can be obtained without applying extra treatment. However we can set the tolerance as $5e-10$ which will result in the triple case treatment being applied and closed solutions are again found. The difference of the solutions with treatment is on average $1e-5$ from the ones without using the treatment. A similar observation appears for larger perturbations and tolerances. Therefore, a tolerance value on the order of $1e-14$ which works well for the original, pure triple case may be too small to identify equality of eigenvalues for the slightly perturbed system while too large a value will induce unnecessary treatment that slows down the calculation. Setting the tolerance to a larger value $5e-11$ works well for this triple case and the double case above, both in original form as well as under a small perturbation.

4.3.4. Example 5: a special double 0 case. Besides the double-double case with finite values above, we may encounter a special case where the eigenvalue is double and corresponding eigenvector contains double infinity. Given two eigenvectors $\mathbf{V}_1, \mathbf{V}_2$, we may detect this special case from the components in $\mathbf{V}_1, \mathbf{V}_2$ and distinguish two main categories. The first category which is actually a double-double case, contains t_{R-1} as double infinity so that only the first and fifth components in $\mathbf{V}_1, \mathbf{V}_2$ are nonzero and it has three subcases: 1. t_A has two finite values; 2. t_A has one finite and one infinity; 3. t_A has double infinity as well, a defective case as discussed above. When we substitute t_{A+1}, t_{R-1} back into the original system, if we can identify zero in the last two components of vector $B^\dagger P_\alpha(\mathbf{c}', 1)^T$, this implies t_A is double infinity and we assign $t_A = \infty$. Otherwise, t_A has two finite values that can be handled by our modification. However we may not be able to distinguish the last two subcases and we simply set t_A as double infinity, and that presents a limitation of this method. The other category contains t_A as double infinity where the last 4

Joint i	a_i	S_i	α_i
1	0.00	1.00	-90.00
2	1.00	2.00	0.00
3	0.00	1.00	-63.43494882292196
4	1.00	$\sqrt{5}$	116.5650511770780
5	2.00	4.00	90.00
6	3.00	2.00	-180.00

TABLE 14. Linkage parameters of example 5.

No.	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
1	180.0000	180.0000	36.8699	-180.0000	36.8699	180.0000
2	177.5864	98.2537	81.1589	-177.3732	0.5874	179.9358
3	-180.0000	90.0000	90.0000	180.0000	0.0000	-180.0000
4	-180.0000	90.0000	90.0000	180.0000	0.0000	-180.0000

TABLE 15. Solutions of joint angles for example 5.

components in $\mathbf{V}_1, \mathbf{V}_2$ are zero. If t_{R-1} is not double ∞ , this is a regular case that can be handled by choosing different components to calculate t_{R-1} as shown in the App. D.1.

We designed example 5 to demonstrate a case where the eigenvalue is double and there is a single eigenvector containing both t_{R-1}, t_A as double ∞ . The DH system parameters are shown in Table 14. Solutions, using reference $R = 5$, in which the variables involved in the eigensystem are not all simultaneously double, are given in Table 15. When $R = 1$, the fifth variable corresponds to the eigenvalue which has double 0, and corresponding 4th and 6th variables are double π , so this is the first category discussed above. This is actually a defective case: as expected, we can detect the double ∞ in t_{R-1}, t_A by checking components in the eigenvectors and assign ∞ values. The calculation for remaining variables $\theta_{R+1}, \theta_{A-1}$ will be done accordingly, leading to two identical solutions.

4.4. Example 6-7: Zero parameters. In the last two examples, we want to explore the possible singularity when there exist zeros in the input parameters. In example 6, the length of one link is close to zero and the twist angle at this link is π . The parameters for lengths and angles, given in Table 16, were adopted from an instance of the 19-atom molecular ring shown in the left of Fig. 1, here converted to an open chain with end effector constructed with joint angles in degrees by $[150^\circ, 120^\circ, -100^\circ, -130^\circ, -50^\circ, 170^\circ]$ and the end effector is in same format as (25) with

$$\begin{aligned}
\mathbf{x} &= [0.633701650686384, 0.306727455336273, -0.710169336187001]^T, \\
\mathbf{y} &= [0.772570825541157, -0.297759233174306, 0.560779598935362]^T, \\
\mathbf{z} &= [-0.039452977581008, -0.904023067848594, -0.425659200954952]^T, \\
\mathbf{d} &= [1.763571453982034, 6.200325108507339, 8.700810576520498]^T.
\end{aligned}$$

In this singular case, a small modification may cause large deviations in the solutions. With the parameters in Table 16 and the given end effector, the system is not solvable, becoming solvable only when a small positive value is assigned to

Joint i	a_i	S_i	α_i
1	1.00	6.00	90.00
2	5.00	1.00	150.00
3	0.00	0.60	180.00
4	1.00	3.40	80.00
5	1.40	2.40	30.00
6	2.30	3.00	-110.00

TABLE 16. Linkage parameters of example 6.

a_3	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
1e-6	150.0000	120.0000	-100.0000	-130.0000	-50.0000	170.0000
1e-7	150.0000	120.0000	-100.0000	-130.0000	-50.0000	170.0000
1e-8	150.0000	120.0000	-100.0002	-130.0002	-50.0000	170.0000
1e-9	150.0000	120.0000	-100.0013	-130.0013	-50.0000	170.0000
1e-10	150.0000	120.0000	-100.0026	-130.0026	-50.0000	170.0000
1e-11	150.0000	120.0000	-100.0422	-130.0422	-50.0000	170.0000
1e-12	150.0000	120.0000	-104.2283	-134.2283	-50.0000	170.0000
1e-13	150.0000	120.0000	-42.8374	-72.8374	-50.0000	170.0000
1e-14	N/A					

TABLE 17. One solution of joint angles for example 6 as length of a_3 decreases.

the third link. We want to explore the lower limit of the length where the method may fail. Starting from a small positive length 1e-6 for the third link a_3 , we record one solution of the system in Table 17 as the length is decreased. As can be seen, the system is still solvable when the length is 1e-13. The method fails when the length is 1e-14. Moreover, as the length decreases, the third and fourth variables exhibit large changes, while the other four variables remain quite stable. Because the joint angle at the third link is 180° , the axes 3 and 4 become collinear as the length of the third link approaches zero. Geometrically, the joint angles at axes 3 and 4 will combine into one and the system now has actually 5 DoF. As can be seen, the third and fourth joint angles maintain a 30° difference, therefore producing the same net rotation. Since at least 6 DoF are required to achieve a workspace of open neighborhoods, the system will become unsolvable if a small change is made to the end effector in this example. We can expect that the case where a link is close to zero length combined with 0° twist angle is also singular with behavior similar to that shown here.

Example 7 is Manocha-Canny Manipulator from [18] and the parameters are shown in Table. 18. By specifying the values $[22^\circ, 34^\circ, 56^\circ, 90^\circ, 90^\circ, -120^\circ]$ to the joint angles, we can get the end effector. However, the set of angles above cannot be recovered because of singularity. The singularity comes from two consecutive 90° in α_3, θ_4 and θ_5, α_5 , and from consecutive zero lengths S_5, a_5 , causing the vectors of axes s_3, s_6 to be collinear. Then the part between s_3, s_6 will be combined into one continuum unit and only the summation of θ_3, θ_6 can have direct impact on the

Joint i	a_i	S_i	α_i
1	0.3	0.0	90.0
2	1.0	0.0	1.0
3	0.0	0.2	90.0
4	1.5	0.0	1.0
5	0.0	0.0	90.0
6	0.0	0.0	1.0

TABLE 18. Linkage parameters of example 7.

No.	θ_1	θ_2	θ_3	θ_4	θ_5	θ_6
1	22.0000	34.0000	-66.8696	90.0000	90.0000	2.8696
2	22.0000	34.0000	64.7411	90.0000	90.0000	-128.7411

TABLE 19. Two solutions of joint angles for example 7.

end effector. In the Table. 19, we choose two solutions from our method where the fourth and fifth angles are 90° . As can be seen, the summation of the third and sixth joint angles is -64° which is consistent with the initial joint angles above. In fact, any two values whose summation is -64° can reach the proposed end effector. This is the similar situation in example 6, where one DoF is dropped and the manipulator shows flexibility to some extent. In both examples, the matrix pencil in the generalized eigenvalue problem may become singular, the rank of the matrices is reduced by one and one of the eigenvalues is infinite. There may exist higher-order degenerate cases such as Bricard's examples [4], in which the rank of the matrices in generalized eigenvalue problem drops by 2 or more. For these flexible cases, working on the polynomial equations will give more hints and it will be the subject of future research.

5. Conclusion. In this paper, we have presented a novel algorithm for the 6R problem of Inverse Kinematics. It can solve the problem stably even under special conditions which can cause failure in previous methods. As our motivation for this work resulted from the kinematic analysis of arbitrarily long constrained molecular chains such as protein loops and large cyclic molecules, we also included a discussion of the reduction of such chains to DH coordinates. In our method, an arbitrary chain of rotors will be first transformed into a closed chain for which we may easily change the order of solving the 6 variables. The first elimination is performed by computing an SVD of a complex-valued system formed by the full set of closure conditions, and utilizing the pseudoinverse. In addition, the null vectors allow an efficient derivation of the resultant, arriving at an optimally sized 16×16 generalized eigenvalue problem. Three joint angles can be determined from the eigenvalues and eigenvectors, even in cases of infinite or multiple eigenvalues. The remaining three angles are subsequently solved using the SVD information. Two types of special cases were discussed in detail. In the first, at least one variable in one solution is π . The second describes cases of higher multiplicity eigenvalues. Although it may be argued that linkage systems may be carefully designed to avoid cases which could

challenge previous algorithms, such cases do in fact arise for special manipulators, as discussed in the examples.

In order to reach broader audience, three different kinds of inputs are accepted in our code: DH parameters, Cartesian coordinates of the joint junctions or internal parameters, with outputs consistent with the inputs. Since it is developed based on closed chain and accepts different kinds of inputs, our method can find applications to both robotics and conformational sampling of biopolymers. The open source algorithm, *R6B6*, is freely available from the authors. In addition, the *MAPLE* symbolic derivation of the equations as used by our algorithm is also available.

Acknowledgments. XC and EAC acknowledge support from the Laufer Center for Physical and Quantitative Biology at Stony Brook University. We acknowledge helpful conversations with Drs. Robert Lewis and Michael Wester. We thank an anonymous reviewer for pointing out Ref.[3].

Appendix A. Formulating the 6R loop from an arbitrary chain. The algorithm presented here can be applied in the context of transforming an open chain or a redundant chain into closed 6R Denavit-Hartenberg (DH) system or in sampling the alternative conformations of cyclic macromolecules and flexible loop regions of proteins as shown in Fig. 1. That is consistent with the fact that molecular bond lengths and angles are relatively “hard” DoF, while torsional angles among successive tetrads are deformable with low energetic cost. For an arbitrary chain, we need to select six rotor axes as pivots for the solution. The remaining $N - 6$ rotor angles are set to fixed values thus rigidifying the structure and the intervening structure between two selected axes will be replaced by a link orthogonal to both axes as shown in the left of Fig. 5. For an open 6R robot arm we only need to do this once to find the link between the sixth axis and first axis. Following shows the detailed procedures to finish the conversion. As the following discussion is mostly useful for the reduction of long molecular chains to DH coordinates, we shall be referring to the rotor angles as “torsions”, keeping the terms “joint angles” and “twist angles” for the system reduced to DH parameters. The notations in this and in each of the subsequent sections are self-contained, relevant only to each particular section.

Consider a chain of $n - 1$ vectors $\{\mathbf{b}_1, \dots, \mathbf{b}_{n-1}\}$. Denote the vertices of the chain $\{v_1, \dots, v_n\}$ where vector $\mathbf{b}_i = v_i v_{i+1}$ joins vertex v_i to vertex v_{i+1} , $i = 1, \dots, n - 1$. An arbitrary chain may be considered as a closed loop by constructing link $\mathbf{b}_n$ connecting vertex v_n to vertex v_1 , and we have a closed loop with n vectors connecting n vertices. For ease of notation we continue the indices cyclically around the loop so that $v_{n+1} \leftrightarrow v_1$, *etc.* Suppose there are 6 unknown torsions in the chain indexed by i_j , $j = 1, \dots, 6$, and denote the axis vectors $\mathbf{p}_j := \mathbf{b}_{i_j}$. Assume we are given the remaining $n - 6$ torsions $\bar{\tau}_i = \text{sgn}([\mathbf{b}_{i-1}, \mathbf{b}_i, \mathbf{b}_{i+1}])\angle(\mathbf{b}_{i-1} \times \mathbf{b}_i, \mathbf{b}_i \times \mathbf{b}_{i+1})$, $i \notin \{i_j\}_{j=1}^6$ where $[\mathbf{a}, \mathbf{b}, \mathbf{c}] = \mathbf{a} \cdot (\mathbf{b} \times \mathbf{c})$. Also assumed knowns are $\{|\mathbf{b}_1|, \dots, |\mathbf{b}_n|\}$ the lengths of the vectors and $\{\bar{\mathcal{A}}_1, \dots, \bar{\mathcal{A}}_n\}$ the (interior) angles: $\bar{\mathcal{A}}_i = \angle(-\mathbf{b}_{i-1}, \mathbf{b}_i)$. In the context of a molecular chain, the lengths $|\mathbf{b}_i|$ would be the *bond lengths* with $\bar{\mathcal{A}}_i$ being the *bond angles*.

Throughout, we will use the following notation for torsions in terms of the angle between two vectors, $\tau = \angle(\mathbf{a}, \mathbf{b}) = \cos^{-1}((\mathbf{a}/|\mathbf{a}|) \cdot (\mathbf{b}/|\mathbf{b}|)) \in [0, \pi]$, and a sign: $t = \sigma\angle(\mathbf{a}, \mathbf{b}) = \text{atan}(s, c)$ with $\sigma = \pm 1$, $c = ((\mathbf{a}/|\mathbf{a}|) \cdot (\mathbf{b}/|\mathbf{b}|))$ and $s = \sigma\sqrt{1 - c^2}$. Here $\mathbf{a}$, $\mathbf{b}$ are normal vectors on two oriented planes and the torsion refers to one of the dihedrals between the planes. In the typical situation, we have a vector triplet

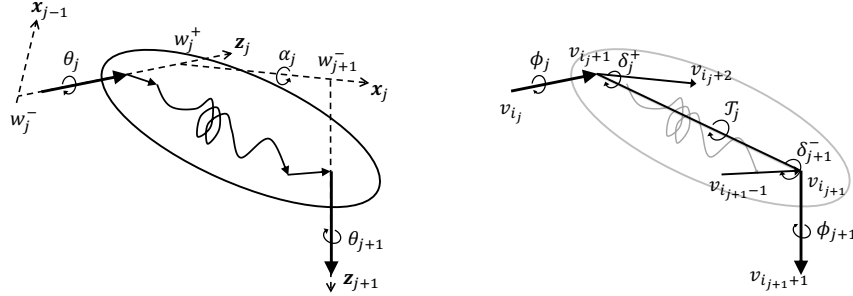


FIGURE 5. Left: Replacing intervening structure by equivalent orthogonal linkage. Right: Simplified loop with a virtual link and offset for angles.

$\mathbf{a}, \mathbf{b}, \mathbf{c}$ and the torsion at $\mathbf{b}$ is computed by a variant of the formula $\text{torsion}(\mathbf{a}, \mathbf{b}, \mathbf{c}) = \text{sgn}([\mathbf{a}, \mathbf{b}, \mathbf{c}]) \text{acos}((\mathbf{a} \times \mathbf{b}) \cdot (\mathbf{b} \times \mathbf{c}) / (|\mathbf{a} \times \mathbf{b}| |\mathbf{b} \times \mathbf{c}|))$.

We proceed by transforming the original loop into a simplified loop with 6 axes, (at most) 6 links. Any existing structure between the vertices v_{i_j+1} and $v_{i_{j+1}}$ (structure between the axes) may be replaced by a “virtual” link $\mathbf{l}_j := v_{i_j+1}v_{i_{j+1}}$ as shown in the right of Fig. 5 so that we now have a loop with at least 6 and at most 12 edges $\{\mathbf{p}_1, \mathbf{l}_1, \dots, \mathbf{p}_6, \mathbf{l}_6\}$ (since some or all of the links might vanish). This loop may be parametrized by the lengths $\{|\mathbf{p}_1|, \dots, |\mathbf{p}_6|, |\mathbf{l}_1|, \dots, |\mathbf{l}_6|\}$, the 6 angles between axes and links, $\mathcal{B}_j = \angle(-\mathbf{p}_j, \mathbf{l}_j)$; the 6 angles between links and axes, $\mathcal{A}_{j+1} = \angle(-\mathbf{l}_j, \mathbf{p}_{j+1})$; the 6 known torsions about the links, $\mathcal{T}_j$; and the 6 unknown torsions about the axes,

$$\begin{aligned} \phi_j &:= \text{sgn}(\nu_j) \angle(\mathbf{l}_{j-1} \times \mathbf{p}_j, \mathbf{p}_j \times \mathbf{l}_j), \\ \nu_j &:= [\mathbf{l}_{j-1}, \mathbf{p}_j, \mathbf{l}_j], \quad j = 1, \dots, 6. \end{aligned}$$

For each of the missing torsions in the original problem there holds

$$\bar{\mathcal{T}}_{i_j} = \phi_j - \delta_j := \phi_j - \delta_j^- - \delta_j^+,$$

where δ_j^- , δ_j^+ are offsets found in the transformation to the 6-axis/6-link loop:

$$\begin{aligned} \delta_j^- &:= \text{sgn}(\gamma_j^-) \angle(\mathbf{l}_{j-1} \times \mathbf{p}_j, \mathbf{b}_{i_{j-1}} \times \mathbf{p}_j), \\ \delta_j^+ &:= \text{sgn}(\gamma_j^+) \angle(\mathbf{b}_{i_{j+1}} \times \mathbf{p}_j, \mathbf{l}_j \times \mathbf{p}_j). \end{aligned}$$

where

$$\gamma_j^- = [\mathbf{l}_{j-1}, \mathbf{b}_{i_{j-1}}, \mathbf{p}_j], \quad \gamma_j^+ = [\mathbf{b}_{i_{j+1}}, \mathbf{l}_j, \mathbf{p}_j].$$

We transform once more to DH coordinates as shown in Fig. 6. Consider the two (in general oblique) lines in space defined by the vectors $\mathbf{p}_j$ and $\mathbf{p}_{j+1}$. We introduce the unit vectors along the $\mathbf{p}_j$: $\hat{\mathbf{p}}_j = \mathbf{p}_j / |\mathbf{p}_j|$. Let μ_j be the mutual moment of the axes (also the reciprocal product of their Plücker coordinates), of sign σ_j , where

$$\mu_j = \hat{\mathbf{p}}_j \cdot \mathbf{l}_j \times \hat{\mathbf{p}}_{j+1}, \quad \sigma_j = \text{sgn}(\mu_j).$$

The torsion at the link $\mathbf{l}_j$ is given by

$$\mathcal{T}_j := \sigma_j \angle(\hat{\mathbf{p}}_j \times \mathbf{l}_j, \mathbf{l}_j \times \hat{\mathbf{p}}_{j+1}).$$

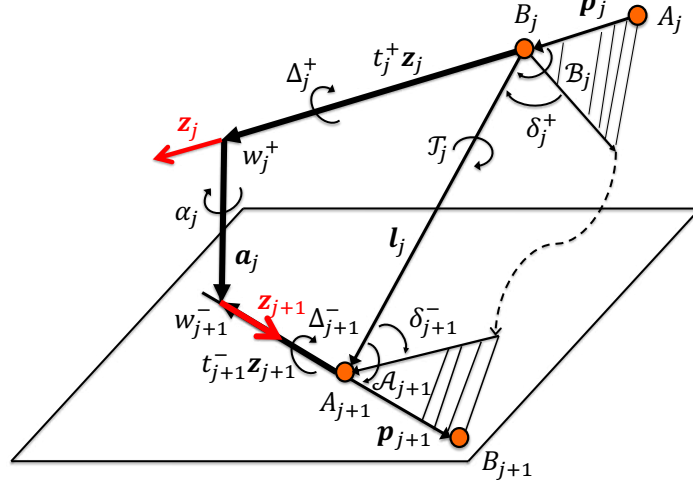


FIGURE 6. Transformation of a simplified loop into the DH parameters at a link.

Each link $\mathbf{l}_j$ is replaced by the unique link orthogonal to both $\mathbf{p}_j$ and $\mathbf{p}_{j+1}$, intersecting an extension of $\mathbf{p}_j$ at w_j^+ and $\mathbf{p}_{j+1}$ at w_{j+1}^- . Define the DH axis vectors $\mathbf{s}_j = w_j^- w_j^+$, the length $s_j = |\mathbf{s}_j|$ and the unit axis vectors $\mathbf{z}_j := \mathbf{s}_j / s_j$, $j = 1, \dots, 6$, collinear with the $\mathbf{p}_j$. Define the DH link-vectors $\mathbf{a}_j = w_j^+ w_{j+1}^-$, the length $a_j = |\mathbf{a}_j|$ and the unit-links $\mathbf{x}_j = \mathbf{a}_j / a_j$. The coordinate systems centered at each w_j^- and w_j^+ are $Q_j^- := \{\mathbf{x}_{j-1}, \mathbf{y}_j^- := \mathbf{z}_j \times \mathbf{x}_{j-1}, \mathbf{z}_j\}$ and $Q_j^+ := \{\mathbf{x}_j, \mathbf{y}_j := \mathbf{z}_j \times \mathbf{x}_j, \mathbf{z}_j\}$ respectively, which are all right-handed orthogonal systems by definition.

We note that the orientation of the axes $\mathbf{z}_j$ of the orthogonal systems may not coincide with the orientation of the axes $\hat{\mathbf{p}}_j$ of the original reduced system since they depend on the placement of both adjacent axes $\hat{\mathbf{p}}_{j\pm 1}$. As a result, the mutual moments of adjacent axes may have different signs in the original and DH systems. On the other hand, the orthogonal links $\mathbf{x}_j$ are unaffected as they depend only on each pair of adjacent axes with no reference to the links adjacent at each side of the pair. The twist angles α_j are the rotations between axes $\mathbf{s}_j$ and $\mathbf{s}_{j+1}$ about link $\mathbf{a}_j$ and the joint angles θ_j are the rotations between links $\mathbf{a}_{j-1}$ and $\mathbf{a}_j$ about axis $\mathbf{s}_j$. As such, these depend on the orientations of the orthogonal axes $\mathbf{z}_j$ so the quadrant in which they fall may only be determined after the entire system of orthogonal links has been constructed. We have

$$\mathbf{z}_j = \zeta_j \hat{\mathbf{p}}_j, \quad \zeta_j = \pm 1$$

so that the mutual moments of the $\mathbf{z}_j$ are

$$\hat{\mu}_j := [\mathbf{z}_j, \mathbf{x}_j, \mathbf{z}_{j+1}] = \zeta_j \zeta_{j+1} \mu_j$$

and with $u_j := \hat{\mathbf{p}}_j \cdot \hat{\mathbf{p}}_{j+1}$ we have

$$\begin{aligned} \cos \alpha_j &:= \mathbf{z}_j \cdot \mathbf{z}_{j+1} = \zeta_j \zeta_{j+1} u_j, \\ \sin \alpha_j &= \zeta_j \zeta_{j+1} \sigma_j \sqrt{1 - u_j^2}, \end{aligned}$$

where ζ_j , the orientations of the $\mathbf{z}_j$ with respect to the $\hat{\mathbf{p}}_j$, need to be determined.

Among the selected 12 atoms/nodes in the reduced linkage, we denote the points by $A_j := v_{i_j}$ and $B_j := v_{i_{j+1}}$. To determine the location of the orthogonal links we introduce quantities t_j^+ (resp. t_j^-), the oriented distances along the $\hat{\mathbf{p}}_j$ from the point B_j (resp. A_j) to the orthogonal link junction w_j^+ (resp. w_j^-). Substitute $B_j w_j^+ = t_j^+ \hat{\mathbf{p}}_j$ and $A_{j+1} w_{j+1}^- = t_{j+1}^- \hat{\mathbf{p}}_{j+1}$ into the equation $B_j w_j^+ + w_j^+ w_{j+1}^- + w_{j+1}^- A_{j+1} = \mathbf{l}_j$ to get expressions in terms of the $\hat{\mathbf{p}}_j$:

$$t_j^+ \hat{\mathbf{p}}_j + \mathbf{a}_{j,j+1} - t_{j+1}^- \hat{\mathbf{p}}_{j+1} = \mathbf{l}_j.$$

Taking dot products with $\hat{\mathbf{p}}_j$ and $\hat{\mathbf{p}}_{j+1}$ and using the orthogonality of the link,

$$\begin{aligned} t_j^+ - u_j t_{j+1}^- &= -l_j \cos \mathcal{B}_j, \\ u_j t_j^+ - t_{j+1}^- &= -l_j \cos \mathcal{A}_{j+1}. \end{aligned}$$

with solutions

$$\begin{aligned} t_j^+ &= l_j (u_j \cos \mathcal{A}_{j+1} - \cos \mathcal{B}_j) / (1 - u_j^2), \\ t_{j+1}^- &= l_j (\cos \mathcal{A}_{j+1} - u_j \cos \mathcal{B}_j) / (1 - u_j^2). \end{aligned}$$

Special care must be taken in case the axis vectors are parallel or antiparallel, when the denominator in the above formulas vanishes. In that case, the orthogonal link may be situated arbitrarily. In general, when the mutual moment vanishes we have three cases: (1) parallel/antiparallel axes; (2) coplanar, non-parallel axes; (3) co-terminal axes. We discuss these special cases below. The idea is to choose a unique convention that allows unique forward and inverse DH transformation. We continue with the generic case discussion and we will follow by detailed analysis of the special cases.

In the generic case discussed above, we have for the signed length of the DH axis vector:

$$S_j = -t_j^- + p_j + t_j^+, \quad \zeta_j = \text{sgn } S_j.$$

Then $\mathbf{s}_j = S_j \hat{\mathbf{p}}_j = \zeta_j s_j \hat{\mathbf{p}}_j$.

The length of the linkage, a_j is found from the invariant triple product as

$$a_j = \mathbf{l}_j \cdot \mathbf{x}_j = (l_j / (1 - u_j^2)) \sqrt{1 - u_j^2 - \cos^2 \mathcal{A}_{j+1} - \cos^2 \mathcal{B}_{j+1} + 2u_j \cos \mathcal{A}_{j+1} \cos \mathcal{B}_j}.$$

Defining $\hat{\sigma}_j = \text{sgn}(\hat{\mu}_j)$ where $\hat{\mu}_j$ is the mutual moment of the axes (also the reciprocal product of their Plücker coordinates) defined above $\hat{\mu}_j := \zeta_j \zeta_{j+1} \mu_j$, we have

$$\mathbf{x}_j = \begin{cases} \hat{\sigma}_j (\mathbf{z}_{j+1} \times \mathbf{z}_j) / |\mathbf{z}_{j+1} \times \mathbf{z}_j|, & \hat{\sigma}_j \neq 0 \\ (\mathbf{z}_{j+1} \times \mathbf{z}_j) / |\mathbf{z}_{j+1} \times \mathbf{z}_j|, & \hat{\sigma}_j = 0, \text{ non-parallel, co-terminal/coplanar} \\ (\mathbf{z}_j \times \mathbf{l}_j) \times \mathbf{z}_j / |(\mathbf{z}_j \times \mathbf{l}_j) \times \mathbf{z}_j|, & \hat{\sigma}_j = 0, \text{ parallel/antiparallel} \end{cases}$$

so that the twist angle at the orthogonal link is

$$\alpha_j = \text{sgn}([\mathbf{z}_j, \mathbf{x}_j, \mathbf{z}_{j+1}]) \angle(\mathbf{z}_j \times \mathbf{x}_j, \mathbf{x}_j \times \mathbf{z}_{j+1}) = -\text{sgn}([\mathbf{z}_j, \mathbf{x}_j, \mathbf{z}_{j+1}]) \angle(\mathbf{y}_j, \mathbf{y}_{j+1}^-).$$

By orthogonality, $\cos \alpha_j = \mathbf{z}_j \cdot \mathbf{z}_{j+1}$, $\sin \alpha_j = \text{sgn}([\mathbf{z}_j, \mathbf{x}_j, \mathbf{z}_{j+1}]) |\mathbf{z}_j \times \mathbf{z}_{j+1}|$. Finally, the (unknown) joint angles about the axes with respect to the orthogonal links,

$$\theta_j = \text{sgn}([\mathbf{x}_{j-1}, \mathbf{z}_j, \mathbf{x}_j]) \angle(\mathbf{x}_{j-1} \times \mathbf{z}_j, \mathbf{z}_j \times \mathbf{x}_j)$$

are related to the torsions about the axes in the reduced linkage system, ϕ_j , by

$$\phi_j = \theta_j - \Delta_j := \theta_j - \Delta_j^- - \Delta_j^+$$

in terms of the offsets $\Delta_j^\pm$ given by:

$$\begin{aligned}\Delta_j^- &= \text{sgn}([\mathbf{x}_{j-1}, \mathbf{l}_{j-1}, \mathbf{z}_j]) \angle (\mathbf{x}_{j-1} \times \mathbf{z}_j, \mathbf{z}_j \times (-\mathbf{l}_{j-1})) , \\ \Delta_j^+ &= \text{sgn}([\mathbf{x}_j, \mathbf{l}_j, \mathbf{z}_j]) \angle (\mathbf{x}_j \times \mathbf{z}_j, \mathbf{z}_j \times (-\mathbf{l}_j)) .\end{aligned}$$

Using the moment equation, it is seen also that

$$\frac{\mathbf{l}_j \cdot \mathbf{x}_j}{|\mathbf{l}_j|} = \sin \mathcal{B}_j \cos \Delta_j^+ = \sin \mathcal{A}_{j+1} \cos \Delta_{j+1}^- .$$

Appendix B. Details of the system of equations. Introducing the expression $\mathbf{g} := \frac{1}{2} \langle \mathbf{r}, \mathbf{r} \rangle \mathbf{z}_R - \langle \mathbf{r}, \mathbf{z}_R \rangle \mathbf{r}$, the equations can be written as

$$\begin{aligned}(E_{1,R}) \quad & \mathcal{E}^L \{ \langle \mathbf{z}_R, \mathbf{x}_{A-1} \rangle t_A - \langle \mathbf{z}_R, \mathbf{y}_A^- \rangle \} = \mathcal{E}^R \{ -\langle \mathbf{z}_R, \mathbf{y}_A \rangle - \langle \mathbf{z}_R, \mathbf{x}_A \rangle t_A \}, \\ (E_{1,I}) \quad & \mathcal{E}^L \{ \langle \mathbf{z}_R, \mathbf{y}_A^- \rangle t_A + \langle \mathbf{z}_R, \mathbf{x}_{A-1} \rangle \} = \mathcal{E}^R \{ \langle \mathbf{z}_R, \mathbf{x}_A \rangle - \langle \mathbf{z}_R, \mathbf{y}_A \rangle t_A \}, \\ (E_{2,R}) \quad & \mathcal{E}^L \{ \langle \mathbf{g}, \mathbf{x}_{A-1} \rangle t_A - \langle \mathbf{g}, \mathbf{y}_A^- \rangle \} = \mathcal{E}^R \{ -\langle \mathbf{g}, \mathbf{y}_A \rangle - \langle \mathbf{g}, \mathbf{x}_A \rangle t_A \}, \\ (E_{2,I}) \quad & \mathcal{E}^L \{ \langle \mathbf{g}, \mathbf{y}_A^- \rangle t_A + \langle \mathbf{g}, \mathbf{x}_{A-1} \rangle \} = \mathcal{E}^R \{ \langle \mathbf{g}, \mathbf{x}_A \rangle - \langle \mathbf{g}, \mathbf{y}_A \rangle t_A \}, \\ (E_{3,R}) \quad & \mathcal{E}^L \{ \langle \mathbf{z}_R \times \mathbf{r}, \mathbf{x}_{A-1} \rangle t_A - \langle \mathbf{z}_R \times \mathbf{r}, \mathbf{y}_A^- \rangle \} = \mathcal{E}^R \{ -\langle \mathbf{z}_R \times \mathbf{r}, \mathbf{y}_A \rangle - \\ & \quad \langle \mathbf{z}_R \times \mathbf{r}, \mathbf{x}_A \rangle t_A \}, \\ (E_{3,I}) \quad & \mathcal{E}^L \{ \langle \mathbf{z}_R \times \mathbf{r}, \mathbf{y}_A^- \rangle t_A + \langle \mathbf{z}_R \times \mathbf{r}, \mathbf{x}_{A-1} \rangle \} = \mathcal{E}^R \{ \langle \mathbf{z}_R \times \mathbf{r}, \mathbf{x}_A \rangle - \\ & \quad \langle \mathbf{z}_R \times \mathbf{r}, \mathbf{y}_A \rangle t_A \}, \\ (E_{4,R}) \quad & \mathcal{E}^L \{ \langle \mathbf{r}, \mathbf{x}_{A-1} \rangle t_A - \langle \mathbf{r}, \mathbf{y}_A^- \rangle \} = \mathcal{E}^R \{ -\langle \mathbf{r}, \mathbf{y}_A \rangle - \langle \mathbf{r}, \mathbf{x}_A \rangle t_A \}, \\ (E_{4,I}) \quad & \mathcal{E}^L \{ \langle \mathbf{r}, \mathbf{y}_A^- \rangle t_A + \langle \mathbf{r}, \mathbf{x}_{A-1} \rangle \} = \mathcal{E}^R \{ \langle \mathbf{r}, \mathbf{x}_A \rangle - \langle \mathbf{r}, \mathbf{y}_A \rangle t_A \}, \\ (E_5) \quad & \mathcal{E}^L \{ \langle \mathbf{z}_R, \mathbf{z}_A \rangle \} = \mathcal{E}^R \{ \langle \mathbf{z}_R, \mathbf{z}_A \rangle \}, \\ (E_6) \quad & \mathcal{E}^L \{ \langle \mathbf{g}, \mathbf{z}_A \rangle \} = \mathcal{E}^R \{ \langle \mathbf{g}, \mathbf{z}_A \rangle \}, \\ (E_7) \quad & \mathcal{E}^L \{ \langle \mathbf{z}_R \times \mathbf{r}, \mathbf{z}_A \rangle \} = \mathcal{E}^R \{ \langle \mathbf{z}_R \times \mathbf{r}, \mathbf{z}_A \rangle \}, \\ (E_8) \quad & \mathcal{E}^L \{ \langle \mathbf{r}, \mathbf{z}_A \rangle \} = \mathcal{E}^R \{ \langle \mathbf{r}, \mathbf{z}_A \rangle \}, \\ (E_9) \quad & \mathcal{E}^L \{ \langle \mathbf{r}, \mathbf{z}_R \rangle \} = \mathcal{E}^R \{ \langle \mathbf{r}, \mathbf{z}_R \rangle \}, \\ (E_{10}) \quad & \mathcal{E}^L \{ \langle \mathbf{r}, \mathbf{r} \rangle / 2 \} = \mathcal{E}^R \{ \langle \mathbf{r}, \mathbf{r} \rangle / 2 \}.\end{aligned}\tag{26}$$

The first eight are combined into 4 complex equations while the remaining six are used as given.

Appendix C. Relation between the SVD of a complex matrix and its real representation. Here we recall the relation between the SVD of a general $n \times m$ complex matrix $B = B_1 + iB_2$ and that of its real representation, the $2n \times 2m$ matrix

$$A_B = \begin{pmatrix} B_1 & -B_2 \\ B_2 & B_1 \end{pmatrix} .\tag{27}$$

The SVD of A_B is the matrix decomposition $A_B = U_A \Sigma_A V_A^T$ where U_A and V_A are real orthogonal matrices, and Σ_A is the real $2n \times 2m$ matrix whose upper left-hand block is the diagonal matrix of singular values of A_B in descending order, with remaining entries of zero. The SVD of $B = U \Sigma V^H$, with U and V complex unitary, and V^H the Hermitian conjugate of V . Σ is the real $n \times m$ matrix whose upper left-hand block is the diagonal matrix containing the singular values of B in descending order, with remaining entries of zero. The singular values of B are the

square roots of the nonzero eigenvalues of BB^H , or equivalently of B^HB , and the singular values of A_B are the square roots of the eigenvalues of $A_B A_B^T$ or $A_B^T A_B$.

To compute the SVD of A_B by means of the smaller computation, the SVD of B , we first apply the similarity transformation

$$A_B = S_n \begin{pmatrix} B & 0 \\ 0 & \bar{B} \end{pmatrix} S_m^H, \quad (28)$$

$$\text{for } S_n = \frac{1}{\sqrt{2}} \begin{pmatrix} I_n & -iI_n \\ -iI_n & I_n \end{pmatrix}, S_m = \frac{1}{\sqrt{2}} \begin{pmatrix} I_m & -iI_m \\ -iI_m & I_m \end{pmatrix},$$

where I_n and I_m are the $n \times n$ and $m \times m$ identity matrix respectively, $\bar{B}$ is the complex conjugate of B , and $S_n^H S_n = I_{2n}$, $S_m^H S_m = I_{2m}$. Applying the SVD of B to (28), noting $\bar{V}^H = V^T$

$$\begin{aligned} A_B &= S_n \begin{pmatrix} U \Sigma V^H & 0 \\ 0 & \bar{U} \Sigma V^T \end{pmatrix} S_m^H \\ &= S_n \begin{pmatrix} U & 0 \\ 0 & \bar{U} \end{pmatrix} \begin{pmatrix} \Sigma & 0 \\ 0 & \Sigma \end{pmatrix} \begin{pmatrix} V^H & 0 \\ 0 & V^T \end{pmatrix} S_m^H \\ &= \left(S_n \begin{pmatrix} U & 0 \\ 0 & \bar{U} \end{pmatrix} S_n^H P_n \right) \left(P_n^T S_n \begin{pmatrix} \Sigma & 0 \\ 0 & \Sigma \end{pmatrix} S_m^H P_m \right) \\ &\quad \times \left(P_m^T S_m \begin{pmatrix} V^H & 0 \\ 0 & V^T \end{pmatrix} S_m^H \right), \end{aligned} \quad (29)$$

where P_n, P_m are permutation matrices, $P_n P_n^T = I_{2n}$, $P_m P_m^T = I_{2m}$. The first term in the right-hand side product of (29) is U_A , the second is Σ_A , and the third is V_A , as verified below. Block multiplying to verify Σ_A

$$\begin{aligned} &P_n^T S_n \begin{pmatrix} \Sigma & 0 \\ 0 & \Sigma \end{pmatrix} S_m^H P_m \\ &= P_n^T \frac{1}{\sqrt{2}} \begin{pmatrix} I_n & -iI_n \\ -iI_n & I_n \end{pmatrix} \begin{pmatrix} \Sigma & 0 \\ 0 & \Sigma \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} I_m & iI_m \\ iI_m & I_m \end{pmatrix} P_m \\ &= P_n^T \begin{pmatrix} \Sigma & 0 \\ 0 & \Sigma \end{pmatrix} P_m = \Sigma_A, \end{aligned}$$

where the permutation P_n, P_m are chosen to rearrange the singular values in descending order with zeros at the end to satisfy the last equality. Block multiplying again to verify U_A , denote the real and imaginary parts of U by $U = U_R + iU_I$.

$$\begin{aligned} S_n \begin{pmatrix} U & 0 \\ 0 & \bar{U} \end{pmatrix} S_n^H P_n &= \frac{1}{\sqrt{2}} \begin{pmatrix} I_n & -iI_n \\ -iI_n & I_n \end{pmatrix} \begin{pmatrix} U & 0 \\ 0 & \bar{U} \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} I_n & iI_n \\ iI_n & I_n \end{pmatrix} P_n \\ &= \frac{1}{2} \begin{pmatrix} U + \bar{U} & i(U - \bar{U}) \\ -i(U - \bar{U}) & U + \bar{U} \end{pmatrix} P_n \\ &= \begin{pmatrix} U_R & -U_I \\ U_I & U_R \end{pmatrix} P_n \\ &= \begin{pmatrix} U_{R,1} & -U_{I,1} & \cdots & U_{R,n} & -U_{I,n} \\ U_{I,1} & U_{R,1} & \cdots & U_{I,n} & U_{R,n} \end{pmatrix} = U_A, \end{aligned}$$

where $U_{R,j}$ and $U_{I,j}$ are indexed in the last line to indicate agreement with the n singular values $\sigma_1, \dots, \sigma_n$, each with two corresponding left-singular vectors. Similarly

for V_A^H , denoting the real and imaginary part $V^H = V_R^H + iV_A^H$ and multiplying by blocks

$$\begin{aligned}
& P_m^T S_m \begin{pmatrix} V^H & 0 \\ 0 & V^T \end{pmatrix} S_m^H \\
&= \frac{1}{\sqrt{2}} P_m^T \begin{pmatrix} I_m & -iI_m \\ -iI_m & I_m \end{pmatrix} \begin{pmatrix} V^H & 0 \\ 0 & \bar{V}^H \end{pmatrix} \frac{1}{\sqrt{2}} \begin{pmatrix} I_m & iI_m \\ iI_m & I_m \end{pmatrix} \\
&= \frac{1}{2} P_m^T \begin{pmatrix} \bar{V}^T + V^T & i(\bar{V}^T - V^T) \\ i(-\bar{V}^T + V^T) & \bar{V}^T + V^T \end{pmatrix} \\
&= P_m^T \begin{pmatrix} V_R^T & V_I^T \\ -V_I^T & V_R^T \end{pmatrix} = V_A,
\end{aligned}$$

where left multiplication reorders the rows in agreement with Σ_A .

Appendix D. The case of multiple eigenvalues. By solving the generalized eigenvalue problem, we can get t_{A+1} from eigenvalues and t_A, t_{R-1} from corresponding eigenvectors, by taking advantage of the special structure, whereas the components are constant multiples of monomials involving the t_A, t_{R-1} . However, when higher multiplicity occurs, the numerically computed eigenvectors may no longer have such structure, although they will still form a basis of the invariant subspace of the multiple eigenvalue t_{A+1} . We present the necessary modifications to extract a basis whose elements have the monomial structure from which the remaining variables can be obtained.

D.1. Regular double case. In the case of one double eigenvalue, suppose two eigenvectors (8 components $\mathbf{v}_m$) from the solver are

$$\mathbf{V}_i = [G_i, F_i, E_i, D_i, C_i, B_i, A_i, L_i]^T, \quad i = 1, 2.$$

Assume two solutions for t_{R-1} are t_1, t_3 , and two corresponding solutions of t_A are t_2, t_4 . Then the two expected eigenvectors are:

$$\hat{\mathbf{V}}_1 = [t_1^3 t_2, t_1^2 t_2, t_1 t_2, t_2, t_1^3, t_1^2, t_1, 1]^T, \quad \hat{\mathbf{V}}_2 = [t_3^3 t_4, t_3^2 t_4, t_3 t_4, t_4, t_3^3, t_3^2, t_3, 1]^T.$$

Since $\mathbf{V}_1, \mathbf{V}_2$ are one set of basis for the two dimensional subspace, we want the vectors $\hat{\mathbf{V}}_1, \hat{\mathbf{V}}_2$ to be the basis for the subspace as well. Assume $t_1 \neq t_3$, while the special case $t_1 = t_3$ will be discussed in next subsection. Suppose there exist four real numbers $\alpha_1, \beta_1, \alpha_2, \beta_2$ satisfying

$$\alpha_1 \hat{\mathbf{V}}_1 + \beta_1 \hat{\mathbf{V}}_2 = \mathbf{V}_1, \quad \alpha_2 \hat{\mathbf{V}}_1 + \beta_2 \hat{\mathbf{V}}_2 = \mathbf{V}_2. \quad (30)$$

Based on the monomial pattern in $\mathbf{V}_1, \mathbf{V}_2$, there are five possibilities for regular double case in total: 1. all t_1, t_2, t_3, t_4 are finite values; 2. one of t_1, t_3 is ∞ and other three will have finite values; 3. one of t_2, t_4 is ∞ ; 4. both of t_1, t_2 or both of t_3, t_4 are ∞ , and other two are finite values; 5. t_1, t_4 are ∞ and t_2, t_3 are finite values, or vice versa. The cases may become more complicated when there exist 0 in the eigenvectors. We will order them by difficulty and show the corresponding treatment for each case below. The corresponding pattern in the expected eigenvectors are in Table. 20 including two special cases, and the case number with an additional letter means the case has extra conditions. The finite values (f) in each case are assumed to be nonzero. In the pseudocode Box 1 below, we summarize the determination of all possible double cases based on the components in $\mathbf{V}_1, \mathbf{V}_2$.

No.	t_1	t_2	t_3	t_4	$\hat{\mathbf{V}}_1$	$\hat{\mathbf{V}}_2$
1	f	f	f	f	$[t_2 \mathbf{h}_1, \mathbf{h}_1]$	$[t_4 \mathbf{h}_3, \mathbf{h}_3]$
2	∞	f	f	f	$[t_2, 0, 0, 0, 1, 0, 0, 0]$	$[t_4 \mathbf{h}_3, \mathbf{h}_3]$
2a	∞	f	0	f	$[t_2, 0, 0, 0, 1, 0, 0, 0]$	$[0, 0, 0, t_4, 0, 0, 0, 1]$
3	f	∞	f	f	$[\mathbf{h}_1, [0, 0, 0, 0]]$	$[t_4 \mathbf{h}_3, \mathbf{h}_3]$
3a	f	∞	0	0	$[\mathbf{h}_1, [0, 0, 0, 0]]$	$[0, 0, 0, 0, 0, 0, 0, 1]$
3b	0	∞	f	0	$[0, 0, 0, 1, 0, 0, 0, 0]$	$[[0, 0, 0, 0], \mathbf{h}_3]$
3c	f	∞	f	0	$[\mathbf{h}_1, [0, 0, 0, 0]]$	$[[0, 0, 0, 0], \mathbf{h}_3]$
4	∞	∞	f	f	$[1, 0, 0, 0, 0, 0, 0, 0]$	$[t_4 \mathbf{h}_3, \mathbf{h}_3]$
4a	∞	∞	0	0	$[1, 0, 0, 0, 0, 0, 0, 0]$	$[0, 0, 0, 0, 0, 0, 0, 1]$
4b	∞	∞	f	0	$[1, 0, 0, 0, 0, 0, 0, 0]$	$[[0, 0, 0, 0], \mathbf{h}_3]$
5	∞	f	f	∞	$[t_2, 0, 0, 0, 1, 0, 0, 0]$	$[\mathbf{h}_3, [0, 0, 0, 0]]$
5a	∞	f	0	∞	$[t_2, 0, 0, 0, 1, 0, 0, 0]$	$[0, 0, 0, 1, 0, 0, 0, 0]$
6	f	f	f	f	$[t_2 \mathbf{h}_1, \mathbf{h}_1]$	$[t_4 \mathbf{h}_1, \mathbf{h}_1]$
6a*	0	n	0	f	$[0, 0, 0, t_2, 0, 0, 0, 1]$	$[0, 0, 0, t_4, 0, 0, 0, 1]$
6b*	∞	n	∞	n	$[t_2, 0, 0, 0, 1, 0, 0, 0]$	$[t_4, 0, 0, 0, 1, 0, 0, 0]$
6c	∞	0	∞	0	defective case	
7	∞	∞	f	∞	$[1, 0, 0, 0, 0, 0, 0, 0]$	$[\mathbf{h}_3, [0, 0, 0, 0]]$

TABLE 20. Possible double cases and monomial patterns. ‘n’ indicates nonzero which can be finite or infinite.

First, we test the 8th components L_1, L_2 . If both are nonzero, then this can be case 1-4. The equality and following equality between terms are determined within one tolerance. Define the ratio $R_{jk}^i := V_{i,j}/V_{i,k}, i = 1, 2$. If $R_{78}^1 = V_{1,7}/V_{1,8} \neq R_{78}^2 = V_{2,7}/V_{2,8}$, then it is case 1 and the treatment is given as follows. We can first pick out the last three rows from either equation in (30)

$$\alpha_i \begin{pmatrix} t_1^2 \\ t_1 \\ 1 \end{pmatrix} + \beta_i \begin{pmatrix} t_3^2 \\ t_3 \\ 1 \end{pmatrix} = \begin{pmatrix} B_i \\ A_i \\ L_i \end{pmatrix}, i = 1, 2.$$

Using the last row to delete the terms of t_1 and t_1^2 , we get

$$\beta_i \begin{pmatrix} t_3^2 - t_1^2 \\ t_3 - t_1 \end{pmatrix} = \begin{pmatrix} B_i - L_i t_1^2 \\ A_i - L_i t_1 \end{pmatrix}, i = 1, 2.$$

Substitute the second row into the first row to remove β_1 and obtain

$$t_3 + t_1 = \frac{B_i - L_i t_1^2}{A_i - L_i t_1}, \text{ and then } t_3 = \frac{B_i - A_i t_1}{A_i - L_i t_1}, i = 1, 2.$$

With two equalities containing two variables, one quadratic equation in terms of t_1 is obtained: $(A_1 L_2 - A_2 L_1) t_1^2 + (B_2 L_1 - B_1 L_2) t_1 + (A_2 B_1 - A_1 B_2) = 0$. Notice that t_3 is also one solution to this equation. Two solutions correspond to t_1, t_3 . With

t_1, t_3 , we could select the third, fourth and last two rows from the first equation in (30) to solve t_2, t_4 :

$$\alpha_1 \begin{pmatrix} t_1 t_2 \\ t_2 \\ t_1 \\ 1 \end{pmatrix} + \beta_1 \begin{pmatrix} t_3 t_4 \\ t_4 \\ t_3 \\ 1 \end{pmatrix} = \begin{pmatrix} E_1 \\ D_1 \\ A_1 \\ L_1 \end{pmatrix}.$$

Then use the last row to delete the term t_2 in the second row and use the second to last row to delete the term $t_1 t_2$ in the first row,

$$\beta_1 \begin{pmatrix} t_3 t_4 - t_3 t_2 \\ t_4 - t_2 \end{pmatrix} = \begin{pmatrix} E_1 - A_1 t_2 \\ D_1 - L_1 t_2 \end{pmatrix}.$$

With these two row, we obtain $t_3 = \frac{E_1 - A_1 t_2}{D_1 - L_1 t_2}$ and then $t_2 = \frac{E_1 - D_1 t_3}{A_1 - L_1 t_3}$. Also, $t_4 = \frac{E_1 - D_1 t_1}{A_1 - L_1 t_1}$.

For the 2nd case, if $R_{78}^1 = R_{78}^2$, we then check the ratios R_{58}^i in $\mathbf{V}_1, \mathbf{V}_2$. If $R_{58}^1 \neq R_{58}^2$, we can expect one of t_1, t_3 goes to ∞ . We set t_1 as ∞ and $t_3 = R_{78}^1$, $t_4 = R_{48}^1$, $t_2 = \frac{G_1 - F_1 t_3}{C_1 - B_1 t_3}$.

For the 3rd case, if $R_{78}^1 = R_{78}^2, R_{58}^1 = R_{58}^2$, we next calculate the ratios R_{34}^i . If $R_{34}^1 \neq R_{34}^2$, we know that one of t_2, t_4 is ∞ and we set it as t_2 and $t_3 = R_{78}^1$. Then we can pick out the second to fourth rows from the either equation in (30) to calculate two values for t_{R-1} as follows:

$$\alpha_i \begin{pmatrix} t_1^2 t_2 \\ t_1 t_2 \\ t_2 \end{pmatrix} + \beta_i \begin{pmatrix} t_3^2 t_4 \\ t_3 t_4 \\ t_4 \end{pmatrix} = \begin{pmatrix} F_i \\ E_i \\ D_i \end{pmatrix}, \quad i = 1, 2$$

$$\beta_i \begin{pmatrix} t_4(t_3^2 - t_1^2) \\ t_4(t_3 - t_1) \end{pmatrix} = \begin{pmatrix} F_i - D_i t_1^2 \\ E_i - D_i t_1 \end{pmatrix}, \quad i = 1, 2.$$

Substitute the second row into the first row above to remove β_i and t_4 ,

$$t_3 + t_1 = \frac{F_i - D_i t_1^2}{E_i - D_i t_1}, \quad \text{and then } t_3 = \frac{F_i - E_i t_1}{E_i - D_i t_1}, \quad i = 1, 2.$$

With two equalities containing two variables, one quadratic equation in terms of t_1 is obtained: $(E_1 D_2 - E_2 D_1) t_1^2 + (D_1 F_2 - D_2 F_1) t_1 + (F_1 E_2 - F_2 E_1) = 0$. Notice that t_3 is also one solution to this equation. Then by solving this equation we can get t_1, t_3 . One of two values will be equal to t_3 and the other is t_1 , then $t_4 = \frac{E_1 - D_1 t_1}{A_1 - L_1 t_1}$.

For the 4th case, if $R_{78}^1 = R_{78}^2, R_{58}^1 = R_{58}^2, R_{34}^1 = R_{34}^2$, we check the ratios R_{14}^i . If they are not equal, we can set t_1, t_2 as ∞ , $t_3 = R_{78}^1$ and $t_4 = R_{48}^1$. Otherwise, we may have special cases as discussed in the next subsection.

In the 5th case, we will see zero in the 8th components. Zeros in the 8th components can also appear in the special double cases discussed in next subsection. If the 8th components are zero, we check the 4th and 5th components. If they are not zero, we then check the ratios R_{34}^i . It is case 5 if they are equal and we set t_1, t_4 as ∞ . If the ratios R_{34}^i are both 0, then $t_3 = 0, t_2 = R_{15}^1$. Otherwise, $t_3 = R_{34}^1$ and $t_2 = \frac{G_1 - F_1 t_3}{C_1 - B_1 t_3}$.

```

! Define  $R_{jk}^i = V_{i,j}/V_{i,k}$ 
IF  $V_{i,8} \neq 0, i = 1, 2$  THEN
  IF  $R_{78}^1 = R_{78}^2$  THEN
    IF  $R_{78}^i = 0, i = 1, 2$  THEN
      IF  $V_{i,5} = 0, i = 1, 2$  THEN
        IF  $V_{i,3} = 0, i = 1, 2$  THEN
          IF  $V_{i,1} = 0, i = 1, 2$  THEN
            case 6a*,  $t_1 = t_3 = 0$ 
          ELSE case 4a,  $t_3 = t_4 = 0$ 
        ELSE case 3a,  $t_3 = t_4 = 0, t_1 = R_{34}^1 (= R_{34}^2)$ 
      ELSE
        case 2a,  $t_3 = 0, t_2 = R_{15}^1 (= R_{15}^2), t_4 = R_{48}^1 (= R_{48}^2)$ 
    ELSE! can be regular case 2, 3, 4
    IF  $R_{58}^1 = R_{58}^2$  THEN
      IF  $V_{i,4} = 0, i = 1, 2$  THEN
        case 4b,  $t_4 = 0, t_3 = R_{78}^1 (= R_{78}^2)$ 
      ELSE
        IF  $R_{34}^1 = R_{34}^2$  THEN
          IF  $R_{34}^i = 0, i = 1, 2$  THEN
            case 3b,  $t_1 = t_4 = 0, t_3 = R_{78}^1 (= R_{78}^2)$ 
          ELSE
            IF  $R_{14}^1 = R_{14}^2$  THEN
              IF  $R_{14}^1 = R_{58}^1$  THEN
                case 6
              ELSE 3c,  $t_4 = 0, t_1 = R_{34}^1 (= R_{34}^2), t_3 = R_{78}^1 (= R_{78}^2)$ 
            ELSE case 4
          ELSE case 3
        ELSE
          IF  $V_{i,1} = 0, i = 1, 2$  THEN
            case 2b,  $t_2 = t_4 = 0, t_3 = R_{78}^1 (= R_{78}^2)$ 
          ELSE case 2
        ELSE case 1
      ELSE
        IF  $V_{i,5} = 0, i = 1, 2$  THEN
          IF  $R_{34}^1 = R_{34}^2$  THEN
            case 7,  $t_1, t_2, t_4 = \infty, t_3 = R_{34}^1 (= R_{34}^2)$ 
          ELSE case 3d,  $t_2, t_4 = \infty$ 
        ELSE
          IF  $V_{i,4} \neq 0, i = 1, 2$  THEN
            IF  $R_{34}^i = 0, i = 1, 2$  THEN
              case 5a,  $t_3 = 0, t_2 = R_{15}^1 (= R_{15}^2)$ 
            ELSE case 5,  $t_3 = R_{34}^1 (= R_{34}^2)$ 
          ELSE
            IF  $V_{i,1} \neq 0, i = 1, 2$  THEN
              case 6b*,  $t_1 = t_3 = \infty$ 
            ELSE case 6c*,  $t_1 = t_3 = \infty, t_2 = t_4 = 0$ 

```

BOX 1: Procedures to determine double eigenvalue cases from the eigenvector components.

D.2. Special double case. Besides the regular double cases, we may face further complications. Based on the check in case 4 of the regular double case above, we need to do more testing on the components in the vectors. Now we have: $R_{78}^1 = R_{78}^2, R_{58}^1 = R_{58}^2, R_{34}^1 = R_{34}^2, R_{14}^1 = R_{14}^2$. We need to distinguish it from the regular case where $t_2 = \infty$ and $t_4 = 0$ by checking the ratios. If $R_{14}^1 \neq R_{58}^1$, we have a regular case 3b where $t_2 = \infty, t_4 = 0, t_1 = R_{34}^1$ and $t_3 = R_{78}^1$. Otherwise it is a double-double case (case 6) and t_1, t_3 are identical while $t_1 = t_3 = R_{78}^1$. t_2, t_4 cannot be obtained correctly because the eigenvalue problem is degenerate. Therefore, by substituting t_{R-1}, t_{A+1} into system (17) and multiplying the pseudoinverse of matrix B from SVD on both sides, we can obtain the system as follows:

$$\mathbf{b} = \begin{pmatrix} t_A \mathbf{b}_1 \\ \mathbf{b}_1 \end{pmatrix} = \mathbf{M}_A t_A + \mathbf{M}_B. \quad (31)$$

Taking advantage of the relationship among the components in the last 8 rows of (31), we have

$$[M_{A,14}t_A + M_{B,14}][M_{A,16}t_A + M_{B,16}] = M_{A,13}t_A + M_{B,13}.$$

Two solutions of this quadratic equation correspond to t_2, t_4 . One more special double-double case is needed to consider. If the 8th components are nonzero, the 7th, 5th, 3rd and 1st components are zero, then this is case 6a where the t_{R-1} has double 0. However, we may not be able to resolve two subcases by checking the components in eigenvectors: 1. t_2, t_4 are finite; 2. $t_2 = \infty, t_4$ is finite. We need to substitute t_{A+1}, t_{R-1} back into the system obtaining (31) containing t_A . If both t_2, t_4 are finite, the modification above can solve it. But if the last two components in vector $\mathbf{M}_A$ are 0, then at least one of t_2, t_4 is infinity and we set both of them as infinity because the finite t_4 in subcase 2 cannot be extracted, representing one limitation of our method.

Based on case 5 of the regular double case, if 4th and 8th components are zero and 5th component is nonzero, then we can identify that both t_1, t_3 will be infinity, and there may exist three subcases (case 6b) as discussed in the main text: 1. both of t_2, t_4 are finite; 2. one of t_2, t_4 is finite and the other is infinity; 3. both of t_2, t_4 are infinity. In this subcase 1, if t_2, t_4 are nonzero, it can be fixed by the modification above for double-double case. However, if t_2, t_4 are both zero (case 6c), then the first component $V_{i,1}$ would be zero leading to a defective case that may not be identified through the eigenvector. Substituting t_{A+1}, t_{R-1} back into system may resolve the problem without a guarantee, another limitation of the method. We can not distinguish subcases 2 and 3 because the last two components in $\mathbf{M}_A$ are zero. Then t_2, t_4 will be set as infinity, indicating a third limitation of the method. On the other hand, if 5th and 8th components are zero, and 4th component is not zero, then we check the ratios R_{34}^i . The equality indicates that it is case 7 where t_2, t_4 are infinity and one of t_1, t_3 is infinity. We set t_1 as infinity and $t_3 = R_{34}^1$. If ratios R_{34}^i are not equal, then we expect t_2, t_4 are infinity and both of t_1, t_3 are finite values. This is case 3d and we use the treatment for case 3 in regular double to calculate t_1, t_3 .

D.3. Regular triple case. For the triple eigenvalue case, one more eigenvector will be given from the solver as $\mathbf{V}_3 = [G_3, F_3, E_3, D_3, C_3, B_3, A_3, L_3]^T$. Assume the third solution for t_{R-1}, t_A is t_5, t_6 and $t_1 \neq t_3 \neq t_5$. The expected third eigenvector is $\hat{\mathbf{V}}_3 = [t_5^3 t_6, t_5^2 t_6, t_5 t_6, t_6, t_5^3, t_5^2, t_5, 1]^T$. Suppose there exist nine real numbers

$\alpha_i, \beta_i, \gamma_i, i = 1, 2, 3$ such that

$$\alpha_i \hat{\mathbf{V}}_1 + \beta_i \hat{\mathbf{V}}_2 + \gamma_i \hat{\mathbf{V}}_3 = \mathbf{V}_i, i = 1, 2, 3. \quad (32)$$

We first pick out last four rows from each of the equations in (32):

$$\alpha_i \begin{pmatrix} t_1^3 \\ t_1^2 \\ t_1 \\ 1 \end{pmatrix} + \beta_i \begin{pmatrix} t_3^3 \\ t_3^2 \\ t_3 \\ 1 \end{pmatrix} + \gamma_i \begin{pmatrix} t_5^3 \\ t_5^2 \\ t_5 \\ 1 \end{pmatrix} = \begin{pmatrix} C_i \\ B_i \\ A_i \\ L_i \end{pmatrix}, i = 1, 2, 3.$$

By deleting the terms t_1, t_1^2, t_1^3 , we get

$$\beta_i \begin{pmatrix} t_3^3 - t_3^2 t_1 \\ t_3^2 - t_3 t_1 \\ t_3 - t_1 \end{pmatrix} + \gamma_i \begin{pmatrix} t_5^3 - t_5^2 t_1 \\ t_5^2 - t_5 t_1 \\ t_5 - t_1 \end{pmatrix} = \begin{pmatrix} C_i - B_i t_1 \\ B_i - L_i t_1^2 \\ A_i - L_i t_1 \end{pmatrix}. \quad (33)$$

Substitute the last equality $\beta_i(t_3 - t_1) = (A_i - L_i t_1) - \gamma_i(t_5 - t_1)$ into the second equality

$$\begin{aligned} [(A_i - L_i t_1) - \gamma_i(t_5 - t_1)](t_3 + t_1) + \gamma_i(t_5 - t_1)(t_5 + t_1) &= B_i - L_i t_1^2, \\ \gamma_i(t_5 - t_1) &= \frac{B_i - L_i t_1^2 - (A_i - L_i t_1)(t_3 + t_1)}{t_5 - t_3}. \end{aligned}$$

Then substituting $\beta_i(t_3 - t_1), \gamma_i(t_5 - t_1)$ into the first equality in (33) and simplifying the expression, we have

$$t_3^2 \frac{A_i t_5 - L_1 t_1 t_5 - B_i + A_i t_1}{t_5 - t_3} + t_5^2 \frac{B_i - A_i t_3 + L_i t_1 t_3 - A_i t_1}{t_5 - t_3} = C_i - B_i t_1.$$

Finally, we simplify equality above as

$$B_i(t_1 + t_3 + t_5) - A_i(t_1 t_3 + t_1 t_5 + t_3 t_5) + L_i t_1 t_3 t_5 = C_i, i = 1, 2, 3.$$

By treating the quantities $t_1 + t_3 + t_5, t_1 t_3 + t_1 t_5 + t_3 t_5, t_1 t_3 t_5$ as three unknowns here, we solve the linear system of three equations and get

$$\begin{aligned} K_1 &= t_1 + t_3 + t_5, \\ K_2 &= t_1 t_3 + t_1 t_5 + t_3 t_5, \\ K_3 &= t_1 t_3 t_5. \end{aligned} \quad (34)$$

where K_1, K_2, K_3 are constants involving $L_i, A_i, B_i, C_i, i = 1, 2, 3$. From system (34) it follows that t_1, t_3, t_5 are the three roots of the cubic

$$t^3 - K_1 t^2 + K_2 t - K_3 = 0.$$

Having obtained t_1, t_3, t_5 , we use them to calculate t_2, t_4, t_6 . Take out six rows containing t_2, t_4, t_6 from the first equation in (32) as follows:

$$\alpha_1 \begin{pmatrix} t_1^2 t_2 \\ t_1 t_2 \\ t_2 \\ t_1^2 \\ t_1 \\ 1 \end{pmatrix} + \beta_1 \begin{pmatrix} t_3^2 t_4 \\ t_3 t_4 \\ t_4 \\ t_3^2 \\ t_3 \\ 1 \end{pmatrix} + \gamma_1 \begin{pmatrix} t_5^2 t_6 \\ t_5 t_6 \\ t_6 \\ t_5^2 \\ t_5 \\ 1 \end{pmatrix} = \begin{pmatrix} F_1 \\ E_1 \\ D_1 \\ B_1 \\ A_1 \\ L_1 \end{pmatrix}.$$

Applying some basic algebraic computations, we have

$$\beta_1 \begin{pmatrix} t_3^2(t_4 - t_2) \\ t_3(t_4 - t_2) \\ t_4 - t_2 \end{pmatrix} + \gamma_1 \begin{pmatrix} t_5^2(t_6 - t_2) \\ t_5(t_6 - t_2) \\ t_6 - t_2 \end{pmatrix} = \begin{pmatrix} F_1 - B_1 t_2 \\ E_1 - A_1 t_2 \\ D_1 - L_1 t_2 \end{pmatrix}.$$

By deleting β_1 , we have

$$\begin{aligned}\gamma_1 t_5^2(t_6 - t_2) - \gamma_1 t_3 t_5(t_6 - t_2) &= F_1 - B_1 t_2 - t_3(E_1 - A_1 t_2), \\ \gamma_1 t_5(t_6 - t_2) - \gamma_1 t_3(t_6 - t_2) &= E_1 - A_1 t_2 - t_3(D_1 - L_1 t_2).\end{aligned}$$

Taking the ratio of the first over the second equality, we obtain

$$\begin{aligned}t_5 &= \frac{F_1 - B_1 t_2 - t_3(E_1 - A_1 t_2)}{E_1 - A_1 t_2 - t_3(D_1 - L_1 t_2)} = \frac{F_1 - E_1 t_3 - (B_1 - A_1 t_3)t_2}{E_1 - D_1 t_3 - (A_1 - L_1 t_3)t_2}, \\ t_2 &= \frac{t_5(E_1 - D_1 t_3) - (F_1 - E_1 t_3)}{(A_1 - L_1 t_3)t_5 - (B_1 - A_1 t_3)}.\end{aligned}$$

Repeating the procedure for t_4, t_6 we find

$$t_4 = \frac{t_1(E_1 - D_1 t_5) - (F_1 - E_1 t_5)}{(A_1 - L_1 t_5)t_1 - (B_1 - A_1 t_5)}, t_6 = \frac{t_3(E_1 - D_1 t_1) - (F_1 - E_1 t_1)}{(A_1 - L_1 t_1)t_3 - (B_1 - A_1 t_1)}. \quad (35)$$

D.4. Special triple case. For the triple eigenvalue case we assumed t_1, t_3, t_5 are not identical, but two among these three roots can be equal, such as $t_1 = t_3$, causing the method above to fail. For this case, following the previous formulation for the general triple case, we can get 3 roots from the cubic equation denoted as DT_1 . We then test if there exist equal roots among them. Using any pair of eigenvectors such as $\mathbf{V}_1, \mathbf{V}_2$ and applying the regular double case treatment, we can get two different solutions denoted as DT_2 . If the two solutions in DT_2 are also in DT_1 , we conclude that two of t_1, t_3, t_5 are equal, indicating a triple-double case; otherwise, the three roots are unequal and we have a regular triple case. For the equal roots, two in DT_2 are the two roots and then we need to identify which root is double by substituting both of the two roots into (18). The root that makes the matrices degenerate is the double value. Once t_1, t_3, t_5 are obtained, we substitute three roots into Eq.(35) above to get t_2, t_4, t_6 , but only one of them t_6 would be correct because we assumed $t_1 = t_3$. Then for the other two roots t_2, t_4 we need one more step to substitute t_{R-1} ($= t_1$) and t_{A+1} into (17) to get two additional solutions similarly as in the special double case.

REFERENCES

- [1] E. Anderson, Z. Bai, C. Bischof, S. Blackford, J. Demmel, J. Dongarra, J. Du Croz, A. Greenbaum, S. Hammarling, A. McKenney and D. Sorensen, *LAPACK Users' Guide*, 3rd edition, Society for Industrial and Applied Mathematics, Philadelphia, PA, 1999.
- [2] J. Angeles, *Fundamentals of Robotic Mechanical Systems: Theory, Methods, and Algorithms*, Fourth edition. Mechanical Engineering Series. Springer, Cham, 2014.
- [3] A. Angerer and M. Hofbaur, *Industrial versatility of inverse kinematics algorithms for general 6R manipulators*, 2013 16th International Conference on Advanced Robotics (ICAR), (2014), 1-7.
- [4] R. Bricard, Mémoire sur la théorie de l'octaèdre articulé, *J. Math. Pures Appl.*, **3** (1897), 113-150.
- [5] E. A. Coutsiyas, K. W. Lexa, M. J. Wester, S. N. Pollock and M. P. Jacobson, *Exhaustive conformational sampling of complex fused-ring macrocycles using inverse kinematics*, *Journal of Chemical Theory and Computation*, **12** (2016), 4674-4687.
- [6] E. A. Coutsiyas, C. Seok, M. P. Jacobson and K. A. Dill, *A kinematic view of loop closure*, *Journal of Computational Chemistry*, **25** (2004), 510-528.
- [7] E. A. Coutsiyas, C. Seok, M. J. Wester and K. A. Dill, *Resultants and loop closure*, *International Journal of Quantum Chemistry*, **106** (2006), 176-189.
- [8] J. Denavit and R. Hartenberg, *A kinematic notation for lower pair mechanisms based on matrices*, *J. Appl. Mech.*, **22** (1955), 215-221.
- [9] N. Go and H. Scheraga, Ring closure and local conformational deformations of chain molecules, *Macromolecules*, **3** (1970), 178-187.

- [10] M. L. Husty, M. Pfurner and H.-P. Schröcker, [A new and efficient algorithm for the inverse kinematics of the general serial 6R manipulator](#), *Mechanism and Machine Theory*, **42** (2007), 66-81.
- [11] J. S. Kim and G. S. Chirikjian, [Inverse kinematic solutions of 6-D.O.F. biopolymer segments](#), *Robotica*, **34** (2016), 1734-1753.
- [12] D. Kohli and M. Osvatic, [Inverse kinematics of general 6R and 5R, P serial manipulators](#), *ASME Journal of Mechanical Design*, **115** (1993), 922-931.
- [13] H.-Y. Lee and C.-G. Liang, [Displacement analysis of the general spatial 7-link 7R mechanism](#), *Mechanism and Machine Theory*, **23** (1988), 219-226.
- [14] H.-Y. Lee and C.-G. Liang, [A new vector theory for the analysis of spatial mechanisms](#), *Mechanism and Machine Theory*, **23** (1988), 209-217.
- [15] S. Liu and S. Zhu, [An optimized real time algorithm for the inverse kinematics of general 6R robots](#), *2007 IEEE International Conference on Control and Automation*, 2080-2084.
- [16] D. J. Mandell, E. A. Coutsiaris and T. Kortemme, [Sub-angstrom accuracy in protein loop reconstruction by robotics-inspired conformational sampling](#), *Nature Methods*, **6** (2009), 551-552.
- [17] D. Manocha, [Solving systems of polynomial equations](#), *IEEE Computer Graphics and Applications*, **14** (1994), 46-55.
- [18] D. Manocha and J. F. Canny, [Efficient inverse kinematics for general 6R manipulators](#), *IEEE Transactions on Robotics and Automation*, **10** (1994), 648-657.
- [19] D. Manocha and Y. Zhu, [Kinematic manipulation of molecular chains subject to rigid constraint](#), in *ISMB*, 1994, 285-293.
- [20] D. Manocha, Y. Zhu and W. Wright, [Conformational analysis of molecular chains using nano-kinematics](#), *Bioinformatics*, **11** (1995), 71-86.
- [21] M. Raghavan and B. Roth, [Inverse kinematics of the general 6R manipulator and related linkages](#), *ASME Journal of Mechanical Design*, **115** (1993), 502-508.
- [22] M. Renaud, [A simplified inverse kinematic model calculation method for all 6R type manipulators](#), in *Current Advances in Mechanical Design and Production VII*, 2000, 15-25.
- [23] C.-Y. Song, Y. Chen and I.-M. Chen, [Kinematic study of the original and revised general line-symmetric Bricard 6R linkages](#), *J. Mechanisms Robotics. Aug*, **6** (2014), 031002.
- [24] G. W. Stewart and J.-G. Sun, *Matrix Perturbation Theory*, Computer Science and Scientific Computing. Academic Press, Inc., Boston, MA, 1990.
- [25] L. N. Trefethen and D. Bau, *Numerical Linear Algebra*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, 1997.
- [26] L. W. Tsai and A. P. Morgan, [Solving the kinematics of the most general six- and five-degree-of-freedom manipulators by continuation methods](#), *ASME Journal of Mechanisms, Transmissions, and Automation in Design*, **107** (1985), 189-200.
- [27] M. G. Wu and M. W. Deem, [Analytical rebridging monte carlo: Application to cis/trans isomerization in proline-containing, cyclic peptides](#), *Journal of Chemical Physics*, **111** (1999), 6625-6632.

Received March 2023; revised May 2023; early access June 2023.