Exact Algebraic Solutions to the Quaternion-Based Translational and Rotational Coordinate Matching Problems

Andrew J. Hanson*

School of Informatics, Computing, and Engineering
Indiana University, Bloomington, Indiana, USA

* hansona@indiana.edu

Abstract

We examine the general problem of finding a global rotation that transforms a given set of points and/or coordinate frames into the best possible alignment with a corresponding set of reference data. This "orthogonal Procrustes problem" is often phrased in terms of minimizing a root-mean-square deviation or RMSD corresponding to a distance measure relating the two sets of coordinates. We employ methods based on quaternion eigensystems that have been exploited for decades in several different bodies of literature where they were discovered independently. Our first contribution is to present an exact algebraic solution to the RMSD optimization problem for 3D spatial data collections, a problem that can in principle be solved algebraically, using matrix eigenvalue or symbolic singular-value-decomposition methods, but has proven so impractical in practice that existing usage relies exclusively on numerical methods. Our compact algebraic forms for solutions to the RMSD problem provide heretofore unavailable insights into the structure of the entire eigensystem. Our second result is to exploit quaternions to express the problem of matching sets of 3D coordinate frames (e.g., collections of orthogonal triads representing the spatial orientations of the amino acid residues in a protein), and to present exact algebraic solutions for the chord measure, an attractive approximation to the more rigorous nonlinear arc measure approach to the orientation optimization problem. This pair of solutions, for the 3D spatial data and the 3D orientation data, can be combined to formulate a solution to the full 6DOF combined matching problem. Among ancillary results, we present a compact algebraic solution to the eigenvalue system for arbitrary real 4D matrices, and extend the quaternion context to the 4D spatial and orientation frame matching problems, along with providing some insights into the problem of converting numerical 3D and 4D rotation matrices into their equivalent quaternion forms.

1 Context

We are concerned with the general mathematical problem known variously as the "Orthogonal Procrustes Problem" [1], Wahba's problem (1965) [2], the Kabsch algorithm (1976) [3], and likely a number of others. One is given a $D \times D$ matrix E that typically corresponds to the cross-covariance matrix of a pair (A, B) of N rows of D-dimensional vectors, making $E = A^t \cdot B$, though E could have almost any origin. Then the fundamental mathematical problem is to find the optimal D-dimensional orthogonal matrix R that maximizes $\mathrm{tr}(R \cdot E)$. One solution to this problem in any dimension D is $R_{\mathrm{opt}} = (E^t \cdot E)^{1/2} \cdot E^{-1}$ (see, e.g., [4, 5]). Solutions may also be found using singular-value-decomposition (SVD) methods (see, e.g., [6, 7]), starting with the decomposition $E = U \cdot S \cdot V^t$, and defining $D = \mathrm{diagonal}\,(1, \ldots, 1, \mathrm{sign}\,(\det(U \cdot V^t)))$, to give the result $R_{\mathrm{opt}} = V \cdot D \cdot U^t$. In addition to these general methods based on traditional matrix approaches, a significant literature exists for $D = 3$ that exploits the relationship between 3D rotation matrices and quaternions, and rephrases the task of finding R_{opt} as a quaternion eigensystem problem. This approach notes that, using the quadratic quaternion form $R(q)$ for the rotation matrix, one can rewrite $\mathrm{tr}(R \cdot E) \to q \cdot M(E) \cdot q$, with $M(E)$ a traceless symmetric 4×4 matrix consisting of linear combinations of

the elements of the 3×3 matrix E. Finding the largest eigenvalue $\epsilon_{\rm opt}$ of $M(E)$, usually by numerical Newton's methods, determines the optimal quaternion eigenvector $q_{\rm opt}$, and so $R(q_{\rm opt})$ solves the RMSD problem. At least five authors are known to have published this approach, possibly starting in 1968 with Davenport [8] in the context of Wahba's problem, rediscovered in 1987 by Horn [9], and soon rediscovered again by others, including Diamond (1988) [10], Kearsley (1989) [11], and Kneller (1991) [12]; it appears that none of the later authors cite the work of any earlier author.

While evaluation of the singular-value decompositions and quaternion eigensystems in terms of algebraic symbols for the elements of E is possible in principle, the results from symbolic algebra programs become unintelligible beyond dimension $D = 2$, with individual SVD symbolic matrix elements being on the order of two megabytes for $D = 3$ and fifteen megabytes for $D = 4$, while initial symbolic solutions for the quaternion eigensystem can approach several *gigabytes*. Nearly all applications in the current literature use $D = 3$ Euclidean spatial data and evaluate $R_{\rm opt}$ numerically, using a variety of methods, starting from a numerical 3×3 data matrix E. No explicit compact algebraic solutions are known to appear in the literature for any dimension higher than $D = 2$. In this paper, we exploit the quaternion eigensystem methods to derive and present one-line closed-form algebraic solutions for all the eigenvalues and eigenvectors for the cases that are amenable to quaternion methods, which include the standard case $D = 3$, its extension to $D = 4$, and additional extensions using orientation data in a quaternion context to incorporate 3D frames (6 DOF in the combined problem) and 4D frames (10 DOF in the combined problem).

2 Introduction

We explore the problem of finding global rotations that optimally align pairs of corresponding lists of spatial and/or orientation data. This issue is significant in diverse application domains. Among these are aligning spacecraft (see, e.g., [8, 13, 14]), obtaining correspondence of registration points in aerial imagery (see, e.g., [5, 9, 15–18]), and matching of molecular and biochemical structures (see. e.g, [10–12, 19–24]). Here we critically examine the quaternion eigensystem decomposition approach to studying the rotation matrices appearing in the optimization formula, and present exact compact algebraic solutions to the RMSD optimization problem in 3D and 4D, along with corresponding orientation frame solutions. Our results are of course (as they must be) algebraically equivalent to the results of symbolic eigensystem calculations and the symbolic singular value decomposition (SVD) method; however, our formulas reduce those computable but extremely complex algebraic expressions, which have generally been thought intractable to simplification, to a single line.

Our extension of the quaternion approach to orientation data exploits the fact that 3D and 4D orientation frames can *themselves* be expressed as quaternions, e.g., amino acid 3D orientation frames written as quaternions (see Hanson and Thakur [25]), and we refer to the corresponding matching task as the QRMSD problem. Various proximity measures for such orientation data have been explored in the literature [26–28], and the general consensus is that the most rigorous measure minimizes the sums of squares of angles between pairs of quaternions. This ideal QRMSD proximity measure is highly nonlinear compared to the spatial RMSD measure, but fortunately there is an often-justifiable linearization, the chord angular distance measure; we exhibit two related but distinct exact algebraic solutions to this approximation that closely parallel our RMSD formulation. In addition, we show how to generalize our methods to treat the problem of optimally aligning *combined* 3D spatial and quaternion 3D-frame-triad data. Combined rotational-translational measures similar to the ones we employ have appeared mainly in the molecular entropy literature [29, 30], where, after some confusion, it was recognized that the spatial and rotational measures are dimensionally incompatible, and an arbitrary context-dependent dimensional constant must appear in any combined measure for the RMSD+QRMSD problem.

In the following, we organize our thoughts by first summarizing the fundamentals of quaternions, which will be our main computational tool. We next introduce the spatial and rotational measures that will underlie our studies of the matching problems, and then derive our exact solutions to the 3D spatial matching problem, the 3D frame triad matching problem, and the combined 6 degree-of-freedom matching problem. Appendix A explores some features of the 2D version of the RMSD problem, and in Appendix B we give the details of our approach to the 4D RMSD problem using double quaternions (which, curiously,

was the eigensystem we actually solved first, before obtaining as a special case the elegant 3D solution given in the body of the text). Finally, in Appendix C we summarize Bar-Itzhack's singularity-free 3D matrix-to-quaternion method [31], which is a novel application of the quaternion RMSD framework, and generalize it to 4D orthogonal matrices.

3 Foundations of Quaternions

For the purposes of this paper, we take a quaternion to be a point $q = (q_0, q_1, q_2, q_3) = (q_0, \mathbf{q})$ in 4D Euclidean space with unit norm, $q \cdot q = 1$, and so geometrically it is a point on the unit 3-sphere $\mathbf{S}^3$ (see, e.g., Hanson [32] for further details about quaternions). The last three terms, $\mathbf{q}$, play the role of a generalized imaginary number, and so are treated differently from the first, and in particular the conjugation operation is taken to be $\bar{q} = (q_0, -\mathbf{q})$. Quaternions obey a multiplication operation denoted by $\star$ and defined as follows:

$$q \star p = [Q(q)] \cdot p = \begin{bmatrix} q_0 & -q_1 & -q_2 & -q_3 \\ q_1 & q_0 & -q_3 & q_2 \\ q_2 & q_3 & q_0 & -q_1 \\ q_3 & -q_2 & q_1 & q_0 \end{bmatrix} \cdot \begin{bmatrix} p_0 \\ p_1 \\ p_2 \\ p_3 \end{bmatrix} = (q_0 p_0 - \mathbf{q} \cdot \mathbf{p},\ q_0 \mathbf{p} + p_0 \mathbf{q} + \mathbf{q} \times \mathbf{p}) \ , \qquad (1)$$

where $Q(q)$ is a matrix form of quaternion multiplication that we will find useful.

Choosing exactly one of the three imaginary components in both q and p to be nonzero gives back the classic complex algebra $(q_0 + \mathrm{i}q_1)(p_0 + \mathrm{i}p_1) = (q_0 p_0 - q_1 p_1) + \mathrm{i}\,(q_0 p_1 + p_0 q_1)$, so there are three copies of the complex numbers embedded in the quaternion algebra; the difference is that in general the final term $\mathbf{q} \times \mathbf{p}$ changes sign if one reverses the order, making the quaternion product order-dependent, unlike the complex product. It can be shown that, although a purely imaginary quaternion $(0, \mathbf{q})$ is technically not a vector as Hamilton claimed (see Altmann [33, 34]), the result of a quadratic conjugation by quaternion multiplication is isomorphic to the construction of a 3D Euclidean rotation $R(q)$ generating all possible elements of the orthogonal group $\mathbf{SO}(3)$. If we write

$$q \star (0,\, x,\, y,\, z) \star \bar{q} = R(q) \cdot \mathbf{x} \ , \qquad (2)$$

we find that the result of collecting coefficients is an orthonormal 3D matrix quadratic in the quaternion elements

$$R(q) = \begin{bmatrix} q_0^2 + q_1^2 - q_2^2 - q_3^2 & 2q_1 q_2 - 2q_0 q_3 & 2q_1 q_3 + 2q_0 q_2 \\ 2q_1 q_2 + 2q_0 q_3 & q_0^2 - q_1^2 + q_2^2 - q_3^2 & 2q_2 q_3 - 2q_0 q_1 \\ 2q_1 q_3 - 2q_0 q_2 & 2q_2 q_3 + 2q_0 q_1 & q_0^2 - q_1^2 - q_2^2 + q_3^2 \end{bmatrix} \ . \qquad (3)$$

The formula for $R(q)$ is technically a two-to-one mapping from quaternion space to the 3D rotation group because $R(q) = R(-q)$; changing the sign of the quaternion preserves the rotation matrix. Note also that the identity quaternion $q_{\mathrm{ID}} = (1, 0, 0, 0)$ corresponds to the identity rotation matrix, as does $-q_{\mathrm{ID}} = (-1, 0, 0, 0)$. This 3×3 matrix $R(q)$ is fundamental not only to the quaternion formulation of the spatial RMSD matching problem, but will also be essential to the orientation-frame problem because the *columns* of $R(q)$ are exactly the needed (double-valued) quaternion representation of the *frame triad* describing the orientation of a body in 3D space, i.e., the columns are the vectors of the frame's local x, y, and z axes relative to an initial identity frame.

Multiplying a quaternion p by the quaternion q to get a new quaternion $p' = q \star p$ simply *rotates* the frame corresponding to p by the matrix Eq. (3) written in terms of q. This is non-trivial, and tells us that quaternion multiplication corresponds exactly to multiplication of two *independent* 3×3 orthogonal rotation matrices, and in fact $R(q) \cdot R(p) = R(q \star p)$.

If we choose the following specific 3-variable parameterization of the quaternion q preserving $q \cdot q = 1$,

$$q = (\cos(\theta/2),\, \hat{n}_1 \sin(\theta/2),\, \hat{n}_2 \sin(\theta/2),\, \hat{n}_3 \sin(\theta/2)) \qquad (4)$$

(with $\hat{\mathbf{n}} \cdot \hat{\mathbf{n}} = 1$), then $R(q) = R(\theta, \hat{\mathbf{n}})$ is precisely the "axis-angle" 3D spatial rotation by an angle θ leaving the direction $\hat{\mathbf{n}}$ fixed, so $\hat{\mathbf{n}}$ is the lone real eigenvector of $R(q)$.

Relationships among quaternions can be studied using the *slerp*, or "spherical linear interpolation" [35], that smoothly parameterizes the points on the shortest quaternion path between two constant quaternions, Q_0 and Q_1, as

$$slerp(Q_0, Q_1, s) \equiv q(s)[Q_0, Q_1] \quad = \quad Q_0 \frac{\sin((1-s)\phi)}{\sin\phi} + Q_1 \frac{\sin(s\,\phi)}{\sin\phi} \ . \tag{5}$$

Here $\cos\phi = Q_0 \cdot Q_1$ defines the angle ϕ between the two given quaternions, while $q(s=0) = Q_0$ and $q(s=1) = Q_1$. For small ϕ, this reduces to the standard linear interpolation $(1-s)\,Q_0 + s\,Q_1$. The unit norm $q(s) \cdot q(s) = 1$ is preserved for all s, so $q(s)$ is always a valid quaternion and $R(q(s))$ defined by Eq. (3) is always a valid 3D rotation matrix.

In the following we will make little further use of the quaternion's algebraic properties, but we will extensively exploit Eq. (3) to formulate elegant approaches to RMSD problems, along with employing Eq. (5) to study the behavior of our data under smooth variations of rotation matrices.

Remark on 4D. A less frequently explored property of quaternions is the extension of Eq. (3) to four Euclidean dimensions by choosing two *distinct* quaternions in Eq. (2), producing a 4D Euclidean rotation matrix. Analogously to 3D, the columns of this matrix correspond to the axes of a 4D Euclidean orientation frame. The 4D extension of the quaternion RMSD eigensystem is in fact what we used to derive the 3D exact solutions that are the main results of the paper. For those interested in these details, the particulars of the 4D approach for both spatial and orientation-frame data are given in Appendix B.

4 The 3D Spatial Matching Problem

We review the basic ideas of spatial data matching, and then specialize to 3D, where we obtain the exact algebraic solutions of the entire four-part eigensystem, including the maximal eigenvalue whose quaternion eigenvector gives the optimal global rotation solving the RMSD problem.

4.1 Matching Data in Euclidean Space.

We begin with the general RMSD problem, taking one column with N rows of D-dimensional points $\{y_k\}$ as the *reference* structure, and a second column of N rows of points $\{x_k\}$ as the *test* structure that must be rotated in space by an $\mathbf{SO}(D)$ rotation matrix R_D to achieve the minimum value of the cumulative quadratic distance

$$\mathrm{RMSD}_D^{\,2} \to \mathbf{S}_D^{\,2} = \sum_{k=1}^{N} \|R_D \cdot x_k - y_k\|^2 \ , \tag{6}$$

where we assume that any overall translational components have been eliminated. When this measure is minimized with respect to the rotation R_D, the optimal R_D will rotate the set $\{x_k\}$ to be as close as possible to the set $\{y_k\}$. Here we will focus on 2D, 3D, and 4D data sets because those are the dimensions that are easily adaptable to our targeted quaternion approach. One could in fact start with the quaternion methods for 4D data described in Appendix B, and then study the lower dimensions by taking appropriate limits starting from the more general 4D solution, or even avoid quaternions altogether using singular value decomposition or other linear algebra approaches (see, e.g., [4–7, 19, 20]).

Expanding the measure given in Eq. (6) (see e.g., [8, 9, 11–17, 19–24]), we can show that the RMSD minimization problem is equivalent to maximizing the cross-term expression

$$\Delta_D = \sum_{k=1}^{N} (R_D \cdot x_k) \cdot y_k = \sum_{a=1,b=1}^{D} {R_D}^{ba} E_{ab} \ = \ \mathrm{tr}\, R_D \cdot E, \tag{7}$$

where

$$E_{ab} = \sum_{k=1}^{N} {x_k}^a {y_k}^b = \left[\mathbf{X}^t \cdot \mathbf{Y}\right]_{ab} \ , \tag{8}$$

is the *cross-covariance matrix* of the data, and the range of (a, b) is the dimension D. (We refer to the original literature for the treatment of issues such as center-of-mass alignment, scaling, and point weighting, which, though essential in some problems, provide no additional insights into our current arguments.)

3D Case. We now restrict our attention to the case of 3D data. The key step is to substitute Eq. (3) for $R(q)$ into Eq. (7), and pull apart the pairs of $q_i q_j$ terms so that the 3D expression is transformed remarkably into the 4×4 *profile matrix* $M(E)$ sandwiched between two identical quaternions (not a conjugate pair), that is

$$\Delta(q) = (q_0, q_1, q_2, q_3) \cdot M(E) \cdot (q_0, q_1, q_2, q_3)^t \equiv q \cdot M(E) \cdot q \ , \tag{9}$$

where the traceless, symmetric profile matrix is

$$M(E) = \begin{bmatrix} E_{xx} + E_{yy} + E_{zz} & E_{yz} - E_{zy} & E_{zx} - E_{xz} & E_{xy} - E_{yx} \\ E_{yz} - E_{zy} & E_{xx} - E_{yy} - E_{zz} & E_{xy} + E_{yx} & E_{zx} + E_{xz} \\ E_{zx} - E_{xz} & E_{xy} + E_{yx} & -E_{xx} + E_{yy} - E_{zz} & E_{yz} + E_{zy} \\ E_{xy} - E_{yx} & E_{zx} + E_{xz} & E_{yz} + E_{zy} & -E_{xx} - E_{yy} + E_{zz} \end{bmatrix} . \tag{10}$$

The bottom line is that if one decomposes Eq. (9) into its eigensystem, it is maximized when the unit-length vector q is the eigenvector of $M(E)$'s largest eigenvalue [5, 9, 11, 12, 15–17, 19–24]. The RMSD optimization problem thus reduces to finding the maximal eigenvalue ϵ_{opt} of $M(E)$ (which depends only on the numerical data) and plugging the corresponding eigenvector q_{opt} into Eq. (3) to obtain the rotation matrix $R(q_{\text{opt}})$ that solves the RMSD problem. The resulting proximity measure relating $\{x_k\}$ and $\{y_k\}$ is simply

$$\left.\begin{array}{rcl} \Delta_{\text{opt}} & = & q_{\text{opt}} \cdot M \cdot q_{\text{opt}} \\ & = & q_{\text{opt}} \cdot \left(\epsilon_{\text{opt}}\, q_{\text{opt}}\right) \\ & = & \epsilon_{\text{opt}} \end{array}\right\} , \tag{11}$$

and does not require us to actually compute q_{opt} or $R(q_{\text{opt}})$ explicitly.

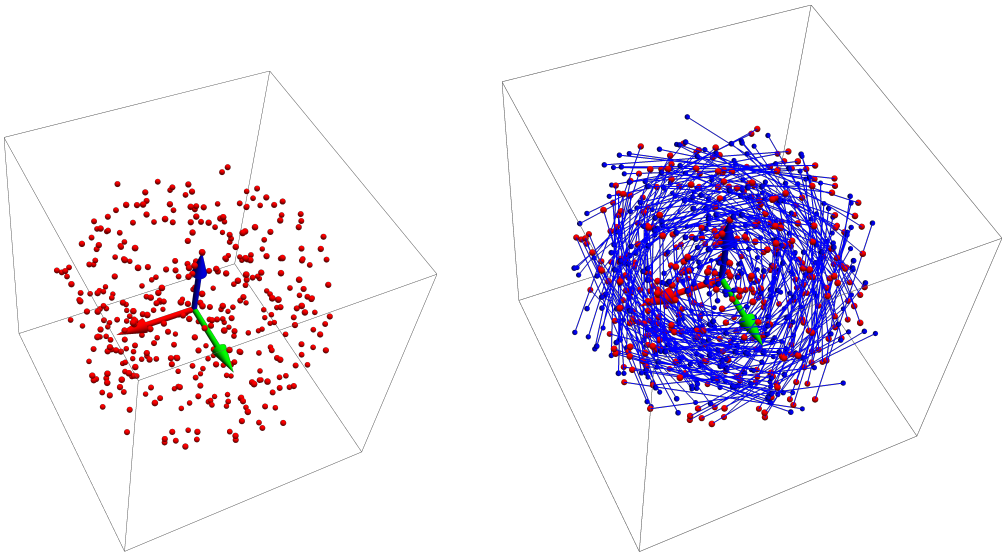
Illustrative Example. In Fig 1(A), we show a simulated 3D reference data set lying inside a spherical boundary, with Fig 1(B) illustrating the displacements of the noisy test data from the reference data due to a global rotation; Eq. (6) corresponds to the sums of the squares of the lengths of the blue line segments. After solving for q_{opt}, we can use the *slerp* of Eq. (5) to interpolate smoothly from the initial state with R in Eqs. (6) and (7) being the identity matrix produced by the identity quaternion q_{ID} (corresponding to Fig 1(B)) to the optimal alignment at $R(q_{\text{opt}})$. Applying

$$q(t) = \mathit{slerp}(q_{\text{ID}}, q_{\text{opt}}, t)$$

to the normalized profile matrix to get a smoothly changing measure $\Delta(t) = q(t) \cdot M(E)/\epsilon_{\text{opt}} \cdot q(t)$ generates a sequence of figures like Fig 2 and 3 . Here Fig 2 shows the alignment midway in the process of evolving from the initial misaligned state at $t = 0$ shown in Fig 1(B) to the final best-possible alignment at $t = 1$ in Fig 3. Note that if we continue on to $t = 2$, we are essentially in the mirror-image state of the initial identity matrix state.

4.2 Solving the 3D Spatial RMSD Problem

We now solve the 3D spatial RMSD optimization problem that is the subject of extensive literature in aeronautical, molecular, and photogrammetric matching. Given the data for the 3D test and reference variables, we know from Eq. (8) that there are nine (distinct, not symmetric) components of the 3×3 cross-covariance matrix E_{ab} that enter into our optimization measure Δ; expanding the measure in terms of the coefficients of quaternions using Eq. (3) for $R(q)$ allowed us to rewrite our fundamental form in terms of the profile matrix $M(E)$ defined in Eq. (10). We now proceed to work out the exact algebraic solution to the 3D RMSD problem.



(A) (B)

Fig 1. fig3DRef.pdf, fig3D2Tgt.pdf. (A) A typical 3D spatial reference data set. (B) The reference data in red alongside the test data in blue, with the Euclidean distances connecting each test data point with its corresponding reference point. We can actually see the axis of the global rotation that relates the pairs of points in the data sets.

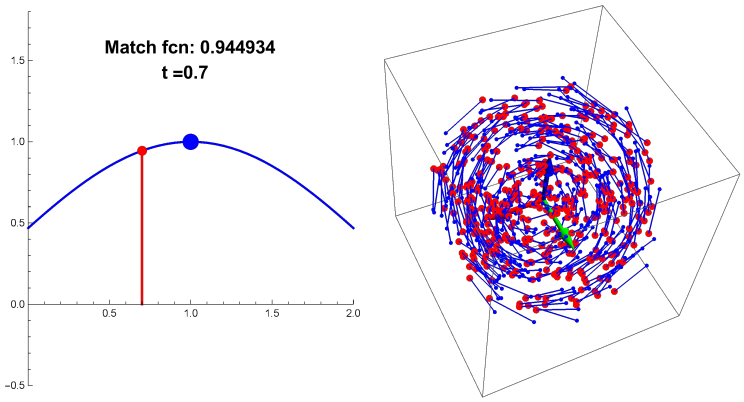


Fig 2. fig3DRotMatch3.pdf. Applying a rotation to the test data that is partway to closest alignment with the reference data.

Eigenvalue Expansions. We begin by writing down the eigenvalue expansion $\det[M - eI_4] = 0$, where e denotes a generic eigenvalue and I_4 is the 4D identity matrix. The 3D RMSD profile matrix is traceless, so we can assume $\mathrm{tr}[M] = 0$ here (both the conditions on tracelessness and symmetry are relaxed in Appendix B). Writing the family of equations to be solved for the unknown variable e in terms of *both* the known components of the matrix M and the unknown eigenvalues ϵ_k, that is

$$e^4 + e^3 p_1 + e^2 p_2 + e p_3 + p_4 = 0 \quad (12)$$

$$(e - \epsilon_1)(e - \epsilon_2)(e - \epsilon_3)(e - \epsilon_4) = 0 \quad (13)$$

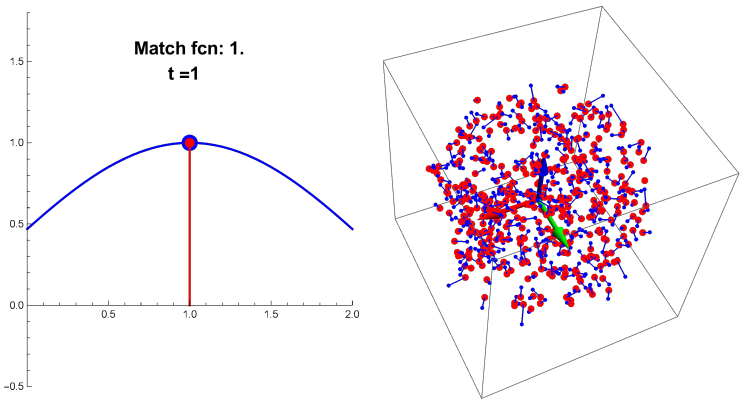


Fig 3. fig3DRotMatch4.pdf. Applying the optimal rotation $R(q_{\rm opt})$ to the test data maximizes the similarity measure and produces an image that clearly shows that this is an optimal alignment.

allows us to eliminate e and write the unknown eigenvalues elegantly in terms of the matrix data as

$$\left.\begin{array}{rcl} p_1(E) &=& -\epsilon_1-\epsilon_2-\epsilon_3-\epsilon_4 \\ p_2(E) &=& \epsilon_1\epsilon_2+\epsilon_1\epsilon_3+\epsilon_2\epsilon_3+\epsilon_1\epsilon_4+\epsilon_2\epsilon_4+\epsilon_3\epsilon_4 \\ p_3(E) &=& -\epsilon_1\epsilon_2\epsilon_3-\epsilon_1\epsilon_2\epsilon_4-\epsilon_1\epsilon_3\epsilon_4-\epsilon_2\epsilon_3\epsilon_4 \\ p_4(E) &=& \epsilon_1\epsilon_2\epsilon_3\epsilon_4 \end{array}\right\} . \tag{14}$$

The coefficients p_k are natural polynomials of degree k in either E or M, and can be written in general as follows:

$$p_1(E) = -\operatorname{tr}[M] = 0 \tag{15}$$

$$\begin{aligned} p_2(E) &= -\frac{1}{2}\operatorname{tr}[M\cdot M] = -2\operatorname{tr}[E\cdot E^t] \\ &= -2\left(E_{xx}^2+E_{xy}^2+E_{xz}^2+E_{yx}^2+E_{yy}^2+E_{yz}^2+E_{zx}^2+E_{zy}^2+E_{zz}^2\right) \end{aligned} \tag{16}$$

$$\begin{aligned} p_3(E) &= -\frac{1}{3}\operatorname{tr}\left[M\cdot M\cdot M\right] = -8\det[E] \\ &= 8\left(E_{xx}E_{yz}E_{zy}+E_{yy}E_{xz}E_{zx}+E_{zz}E_{xy}E_{yx}\right)-8\left(E_{xx}E_{yy}E_{zz}+E_{xy}E_{yz}E_{zx}+E_{xz}E_{zy}E_{yx}\right) \end{aligned} \tag{17}$$

$$p_4(E) = \det[M]\ . \tag{18}$$

The data coefficients $\{p_1(E), p_2(E), p_3(E), p_4(E)\}$ are known: *they are just numbers*. Equation 14 above thus gives us the unknown eigenvalues ϵ_k in terms of the *known data collections*. Therefore our task is to invert Eq. (14) to get $\epsilon_k(p_1,p_2,p_3,p_4) = \epsilon_k(E)$. Unfortunately, this direct approach is intractable, and generally fails to complete in a machine algebra program such as Mathematica before running out of memory. Computing the solution to the equations in the special case $p_1 = 0$ on a large computer runs for nearly 5 hours, and initially produces 24 triples of solutions totaling 8 gigabytes of apparently unsimplifiable algebraic expressions.

To resolve this difficulty, we found it useful to express the eigenvalues of M (which are necessarily all real) in a form that can be made to correspond to a descending magnitude order using this three-parameter representation of a traceless 4D matrix, that is

$$\left.\begin{array}{rcl} \epsilon_1 &=& +\sqrt{X}+\sqrt{Y}+\sqrt{Z} \\ \epsilon_2 &=& +\sqrt{X}-\sqrt{Y}-\sqrt{Z} \\ \epsilon_3 &=& -\sqrt{X}+\sqrt{Y}-\sqrt{Z} \\ \epsilon_4 &=& -\sqrt{X}-\sqrt{Y}+\sqrt{Z} \end{array}\right\} . \tag{19}$$

The next step is to substitute our expression Eq. (19) for ϵ_k in terms of the $\{X, Y, Z\}$ parameters into

Eq. (14), yielding

$$\begin{aligned} p_1 &= 0 && (20)\\ p_2 &= -2\,(X+Y+Z) && (21)\\ p_3 &= -8\,\sqrt{X\,Y\,Z} && (22)\\ p_4 &= X^2+Y^2+Z^2-2\,(YZ+ZX+XY)\ . && (23) \end{aligned}$$

Interesting things now start to happen if we try to determine formulas for $\epsilon_k(E)$ by using Eqs. (21), (22), and (23) to solve for $X(p_2,p_3,p_4)$, $Y(p_2,p_3,p_4)$, and $Z(p_2,p_3,p_4)$ using a machine algebra environment such as Mathematica. We obtain six sets of $\{X,Y,Z\}$ expressions, with one of each set relatively simple, and the rest incomprehensibly complex, which, as in the general case, fail to respond to the standard expression-simplification utilities. Taking the key step of examining what happens when random numbers are substituted for the data matrix elements, we find that the $\{X,Y,Z\}$ each correspond to two of the simple expressions, and the *numerical* values of the other twelve are *also* partitioned among the same values. By rearranging the simple-looking solutions to take their places in the jigsaw puzzle of closed-form solutions, we find a first draft of what the algebraic solutions to the eigenvalue problem for M could look like. After many attempts at reducing the complexity of the solutions and looking for symmetric expressions that could correspond to a single universal formula for X , Y, and Z, an expression emerged that showed that the $\{X,Y,Z\}$ expressions essentially corresponded to different cube roots of unity in otherwise identical expressions, which we examine next.

Algebraic Properties of the Quaternion Eigenvalues in 3D. We now present our closed-form algebraic solution to the quaternion version of the 3D RMSD optimization problem, which has traditionally been examined only numerically. The basic function arising in the solution for all three of the $\{X,Y,Z\}$ determining the eigenvalues in Eq. (19) is only one line long:

$$F_f(p_2,p_3,p_4) = \frac{1}{6}\left(r(p)\cos_f(p)-p_2\right)$$

where

$$\cos_x(p)=\cos\left(\tfrac{\arg(a+\mathrm{i}b)}{3}\right),\ \cos_y(p)=\cos\left(\tfrac{\arg(a+\mathrm{i}b)}{3}-\tfrac{2\pi}{3}\right),\ \cos_z(p)=\cos\left(\tfrac{\arg(a+\mathrm{i}b)}{3}+\tfrac{2\pi}{3}\right)\ . \quad (24)$$

Here $F_f(p)$ corresponds to $X(p)$, $Y(p)$, and $Z(p)$ for $f=\{x,y,z\}$, and, in the notation of the C math library, $\arg(u+\mathrm{i}v)=\mathrm{atan2}(v,u)$ (or ArcTan$[u,v]$ in Mathematica). The utility functions reduce to

$$\left.\begin{aligned} a(p_2,p_3,p_4) &= {p_2}^3+\tfrac{1}{2}\left(27{p_3}^2-72p_2p_4\right)\\ r^2(p_2,p_3,p_4) &= {p_2}^2+12p_4\\ b^2(p_2,p_3,p_4) &= r^6-a^2 \end{aligned}\right\}\ . \quad (25)$$

Accuracy and Performance. Substituting the resulting values for $X(p)$, $Y(p)$, and $Z(p)$ into Eq. (19) produces the four eigenvalues ϵ_k of the matrix M typically in descending numerical order. Taking 100,000 randomized pairs of test and reference data with size $N=10$, generating the corresponding 4×4 profile matrices M, and comparing the 4-tuples of numerical eigenvalues computed by traditional methods, sorted in descending numerical order, to the outcome of plugging the numerical values of M into the algebraic expressions for the eigenvalues in Eq. (19), we find exact matches to approximately machine precision for every single case examined. Out of 400,000 individual (numeric minus algebraic) eigenvalue differences, the maximum difference was 10^{-13} relative to a machine precision of 10^{-16}, and 40,000 were *exactly* zero, with the algebraic solutions matching the machine precision of the numerical solutions with no error whatsoever. We make no claims comparing performance because the efficiency of algebraic expression evaluation is strongly dependent upon idiosyncrasies of individual compilers and their usage. Using compiled Mathematica, the algebraic:numeric timing ratio was about 4:1, but with some analysis one can probably improve that significantly as well as reducing the sources of numerical error in the algebraic evaluation.

Eigenvectors for 3D Data. The eigenvector formulas corresponding to ϵ_k can be generically computed by solving any three rows of $[M \cdot \mathbf{v} - e\mathbf{v}] = 0$ for the elements of $\mathbf{v}$, e.g., $\mathbf{v} = (1,\, v_1,\, v_2,\, v_3)$, as a function of some eigenvalue e (of course, one must account for special cases, e.g., if some element of M is already diagonal). The desired unit quaternion for the optimization problem can then be obtained from the normalized eigenvector

$$q(e, E) = \frac{\mathbf{v}}{\|\mathbf{v}\|} \ . \tag{26}$$

Note that this can often have $q_0 < 0$, and that in those cases any exploration that depends on the sign of q_0, such as a *slerp* from q_{ID}, should choose the sign of Eq. (26) appropriately; in some applications, one may also want an element of statistical randomness, in which case one might randomly pick a sign for q_0. In the general well-behaved case, the form of $\mathbf{v}$ in the eigenvector solution for any eigenvalue $e = \epsilon_k$ may be written explicitly as

$$q(e, E) \quad = \quad \frac{1}{\|\mathbf{v}\|} \times \left[\begin{array}{c} 2ABC + A^2 e_x + B^2 e_y + C^2 e_z - e_x e_y e_z \\ A(aA - bB - cC) - cBe_y - bCe_z - a\, e_y e_z \\ B(bB - cC - aA) - aCe_z - cAe_x - b\, e_z e_x \\ C(cC - aA - bB) - bAe_x - aBe_y - c\, e_x e_y \end{array} \right] \ , \tag{27}$$

where for convenience we define $\{e_x = (e - x + y + z),\, e_y = (e + x - y + z),\, e_z = (e + x + y - z)\}$ with $x = E_{xx}$, cyclic, $a = E_{yz} - E_{zy}$, cyclic, and $A = E_{yz} + E_{zy}$, cyclic. We substitute the maximal eigenvector $q(\epsilon_1, E)$ into Eq. (3) to give the sought-for optimal 3D rotation matrix $R(q(\epsilon_1, E))$ that solves the RMSD problem with $\Delta(q(\epsilon_1, E)) = \epsilon_1$, as we noted in Eq. (11).

5 The 3D Orientation Frame Matching Problem.

We turn next to the orientation-frame problem, assuming that the data are like lists of orientations of roller coaster cars, or lists of residue orientations in a protein, *without* considering any spatial location or nearest-neighbor ordering information. In D-dimensional space, the *columns* of any $\mathbf{SO}(D)$ orthonormal $D \times D$ rotation matrix R_D are what we mean by an orientation frame, since these columns are the directions pointed to by the axes of the identity matrix after rotating something from its defining identity frame to a new attitude; note that no spatial location information whatever is contained in R_D, though one may wish to choose a local center for the rotation if the frame construction involves coordinates, e.g., protein atom locations [25].

In 2D, 3D, and 4D, there exist two-to-one quadratic maps from the topological spaces $\mathbf{S}^1$, $\mathbf{S}^3$, and $\mathbf{S}^3 \times \mathbf{S}^3$ to the rotation matrices R_2, R_3, and R_4. These are the quaternion-related objects that we will use to obtain elegant representations of the frame data-matching problem. In 2D, our frame data element can be expressed as a complex phase, in 3D the frame is a unit quaternion (see [25,32]), and in 4D (see Appendix B), the frame is described by a pair of unit quaternions; higher dimensions may possibly be addressed with Clifford algebras.

What is a Quaternion Frame? We will first present a bit of intuition about coordinate frames that may help some readers with our terminology. If we take the special case of a quaternion representing a rotation in the 2D (x, y) plane, the 3D rotation matrix Eq. (3) reduces to the standard right-handed 2D rotation

$$R_2(\theta) = \left[\begin{array}{cc} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{array} \right] \ . \tag{28}$$

As shown in Fig 4, we can use θ to define a unit direction in the complex plane defined by $z = \exp \mathrm{i}\theta$, and then the *columns* of the matrix $R_2(\theta)$ naturally correspond to a unique associated 2D coordinate frame diad, with an entire collection of points z and their corresponding frame diads depicted in Fig 5.

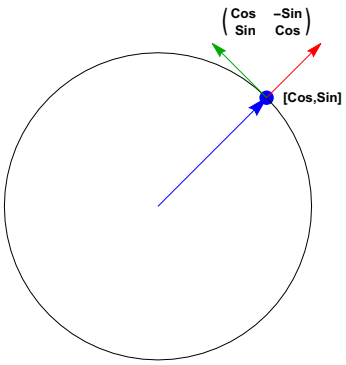


Fig 4. One2DFrame.pdf. A standard 2D coordinate frame corresponds to the columns of ordinary rotation, and is associated to the point $(\cos\theta, \sin\theta)$ on a unit circle.

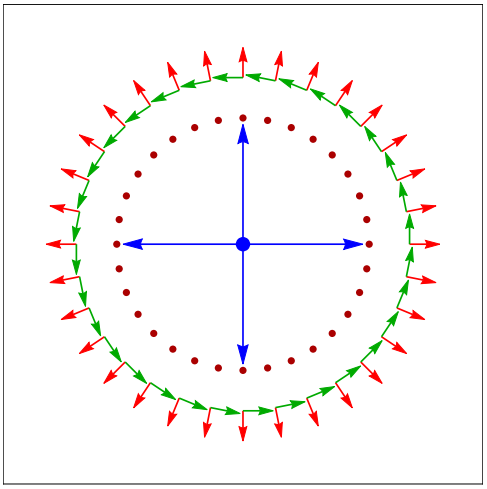


Fig 5. lotsof2DFrames.pdf. The standard 2D coordinate frames associated with a sampling of the entire circle of points $(\cos\theta, \sin\theta)$.

Starting from this context, we can get a clear intuitive picture of what we mean by a "quaternion frame" before diving into the quaternion RMSD problem. The essential step is to look again at Eq. (3) for $n_x = 1$, and write the corresponding quaternion as $(a, b, 0, 0)$ with $a^2 + b^2 = 1$, so this is a "2D quaternion," and is indistinguishable from a complex phase like z that we just introduced. There is one significant difference, however, and that is that Eq. (3) shows us that $R_2(\theta)$ takes a new form, quadratic in a and b,

$$R_2(a,b) = \begin{bmatrix} a^2 - b^2 & -2ab \\ 2ab & a^2 - b^2 \end{bmatrix} . \tag{29}$$

Using either the formula Eq. (4) for $q(\theta, \hat{\mathbf{n}})$ or just exploiting the trigonometric double angle formulas, we see that Eq. (28) and Eq. (29) correspond and that

$$(a,b) = (\cos(\theta/2), \sin(\theta/2)) \tag{30}$$

$$u = (a + \mathrm{i}\, b) = \sqrt{z} = e^{\mathrm{i}\theta/2} . \tag{31}$$

Our simplified 2D quaternion thus describes the *square root* of the usual Euclidean frame given by the columns of $R_2(\theta)$. Thus the pair (a, b) (the reduced quaternion) itself corresponds to a frame. In Fig 6, we show how a given "quaternion frame," i.e., the columns of $R_2(a, b)$, corresponds to a point $u = a + \mathrm{i}\, b$ in the complex plane. Diametrically *opposite* points (a, b) and $(-a, -b)$ now correspond to the *same* frame! Fig 7 shows the corresponding frames for a large collection of points (a, b) in the complex plane, and we see the new and unfamiliar feature that the frames make *two* full rotations on the complex circle instead of just one as in Fig 5.

This is what we have to keep in mind as we now pass to using a full quaternion to represent an arbitrary 3D frame triad via Eq. (3). The last step is to notice that in Fig 7 we can represent the set of frames in one half of the complex circle, $a \geq 0$ shown in magenta, as distinct from those in the other half, $a < 0$ shown in dark blue, where all opposite pairs have the same value of the b coordinate. In the quaternion case, we will display quaternion frames inside one single sphere, like displaying only the b coordinates in Fig 7 on a line, realizing that if one knows the opposite-sign coloring, we can determine both the magnitude of the dependent variable $a = \pm\sqrt{1 - b^2}$ as well as its sign. The same holds true in the general case: if we display only the 3-vector part $\mathbf{q} = (q_x, q_y, q_z)$ along with a color specifying the sign of q_0, we implicitly know both the magnitude and sign of $q_0 = \pm\sqrt{1 - {q_x}^2 - {q_y}^2 - {q_z}^2}$, and such a plot therefore accurately depicts *any* quaternion.

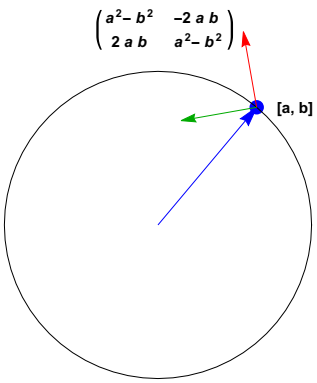


Fig 6. One2DabFrame.pdf. The quaternion point (a, b), in contrast, corresponds via the double-angle formula to coordinate frames that rotate twice as rapidly as (a, b) progresses around the unit circle that is a simplified version of quaternion space.

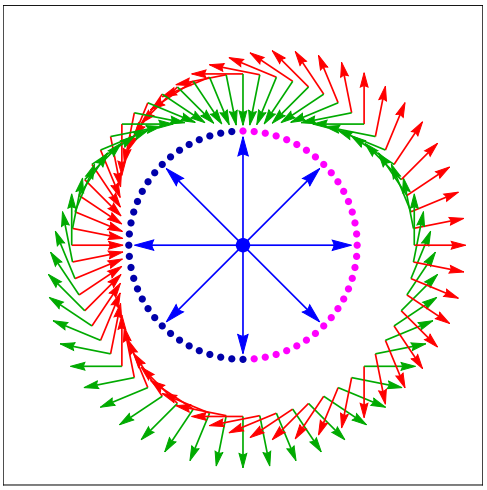
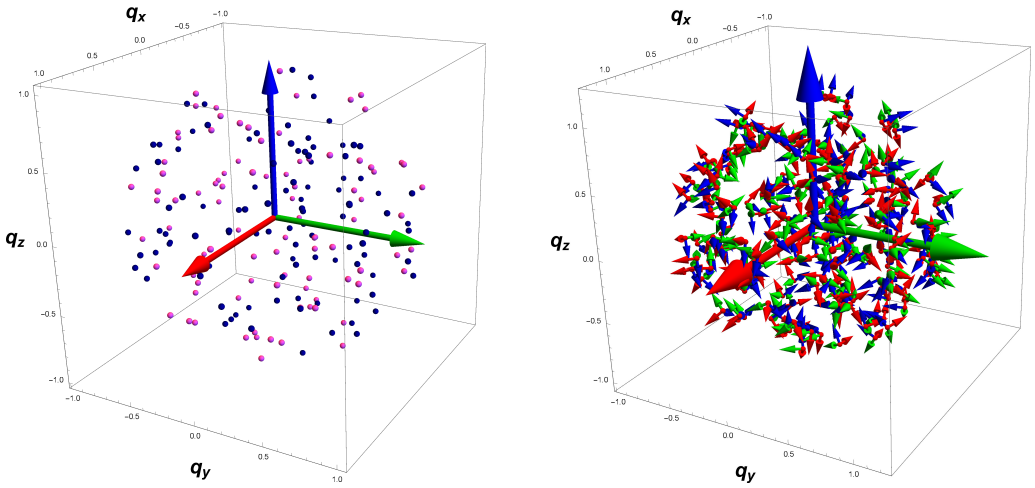


Fig 7. lotsof2DabFrames.pdf. The set of 2D frames associated with the entire circle of quaternion points (a, b); each diametrically opposite point corresponds to an identical frame. For later use in displaying full quaternions, we show how color coding can be used to encode the sign of one of the coordinates on the circle.

Example. We illustrate all this in Fig 8(A), which shows a typical collection of quaternion reference-frame data displaying only the **q** components; the $q_0 \geq 0$ data are mixed with the $q_0 < 0$ data, but are distinguished by the color coding. In Fig 8(B), we show the frame triads resulting from applying Eq. (3) to each quaternion point and plotting it at the associated point in the display.



(A) (B)

Fig 8. fig3DFrmAxes1.pdf, fig3DFrmRef2.pdf. (A) The 3D portions of the quaternion reference-frame data $q = (q_0, q_x, q_y, q_z)$, using different colors for $q_0 \geq 0$ and $q_0 < 0$ in the unseen direction. Since $|q_0| = \sqrt{{q_x}^2 + {q_y}^2 + {q_z}^2}$, the complete quaternion can in principle be determined from the figure. (B) The 3D orientation frame triads for each reference point (q_0, q_x, q_y, q_z) displayed at their associated $\mathbf{q} = (q_x, q_y, q_z)$.

The Arc-Length Distance. We focus now on the 3D orientation-frame case. We might assume we could just define the "QRMSD problem" by converting our list of frame matrices to quaternions (see [25, 32] and also Appendix C), and writing down the quaternion equivalents of the RMSD treatment in Eq. (6) and Eq. (7). However, there are some complications related to the fact that, while the Euclidean center of mass of a cluster of (possibly weighted) points is linear and is easily solved by computing the average, the preferred quaternion equivalent of computing the center of mass technically requires a *non-linear* minimization of the sums of geodesic arc-lengths connecting the points on the hypersphere $\mathbf{S}^3$, and its non-triviality is the subject of its own branch of the literature [36, 37]. The rigorous form of the arc-length measure problem (see, e.g., [26–28]) can be expressed by minimization of the sums of squared arc-length differences,

$$\mathrm{QRMSD}^2 \to \mathbf{S}^{\,2}_{\mathrm{arclength}} = \sum_{k=1}^{N} \left(2 \arccos |(q \star p_k) \cdot r_k|\right)^2 \ , \tag{32}$$

where q, $\{p_k\}$, and $\{r_k\}$ are quaternions, and the absolute value ensures that the correct cosine will be chosen regardless of any quaternion sign. q corresponds to the rotation $R(q)$ in Eq. (3), p similarly refers to the test orientation-frame data, r represents the reference frame data, and "$\star$" denotes the quaternion multiplication by q acting on the entire set $\{p_k\}$ to rotate it to a new orientation that we want to align optimally with the reference frames $\{r_k\}$. The task is to find q that minimizes this sum.

We can understand this expression by observing that the alignment of two unit vectors in any dimension is quantified by the Euclidean dot product $a \cdot b = \|a\|\|b\| \cos\theta_{a,b} = \cos\theta_{a,b}$. Thus the arccosine is just the angle between the directions, and Eq. (32) produces the angular measure in dimensionless units such as radians. In the specific quaternion context, one may introduce a factor of 2 that comes formally from the 1/2 in the relation of quaternion-space angles to 3D rotation angles in Eq. (4),
$\arccos q_0 = \arccos\left(\cos(\theta/2)\right) = \theta/2$; this scaling can be omitted. The behavior of Eq. (32) is highly nonlinear for 3D frames, and minimization can typically only be achieved by numerical methods, in parallel to the spherical averaging problem [37], though in 2D space the problem linearizes to an equivalent 1D Euclidean center-of-mass problem and is normally trivial.

The Linear Chord Distance. However, in the common situation that corresponding pairs of frames are "not too far apart," there is another viable option, the *Euclidean chord measure* using the Euclidean distance between the end points, which can be written

$$\mathrm{QRMSD}^2 \to \mathbf{S}^{\,2}_{\mathrm{chord}} = \sum_{k=1}^{N} \|q \star p_k - r_k\|^2 \ , \tag{33}$$

and now looks exactly like Eq. (6). The basic differences between an individual term in the arc-measure distance and the chord-measure distance, including the implications of quaternion sign ambiguity, are illustrated in Fig 9 .

Observing that squared distances resembling single terms in Eq. (33) can be written as

$$\begin{aligned} \|a-b\|^2 &= a\cdot a - 2a\cdot b + b\cdot b \\ &= 2 - 2\cos(\theta) \\ \cos\theta &= \theta^2\left(1 - \frac{\theta^2}{12} + \frac{\theta^4}{360} - \cdots\right) \ , \end{aligned} \tag{34}$$

we see that, to within errors of order $\theta^2/12$, we could thus effectively replace minimizing the cumulative squared geodesic angular differences by maximizing the cumulative cosines related to the angular differences. However, as noted in Fig 9, defining $\cos\theta = a \cdot b$ using quaternions depends on the *signs* of the two quaternions, either of which can change sign without affecting its corresponding 3D frame triad, so we have to do something that corrects for this sign ambiguity. One way that often is sufficient is to modify the signs of the test quaternions individually to force all the local inner products with their corresponding reference

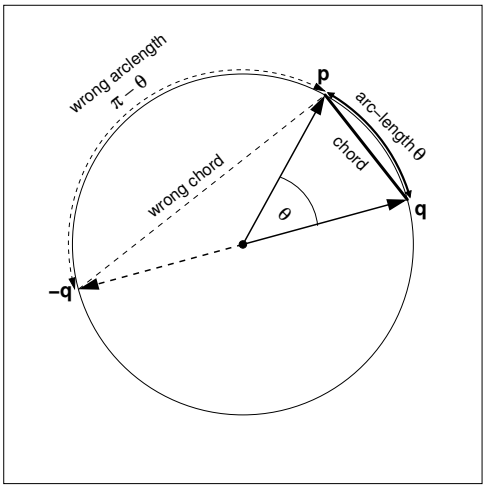


Fig 9. arcvschord.pdf. Because the vectors q and $-q$ give the same rotation matrix, one must choose $|\cos\theta|$ or the *minimum* of the chord distances to get the *right* arc-length or chord measure.

quaternions to be positive, and use only such corrected terms in the distance measure. If there are no sign-ambiguity-induced issues, we can then take our measure to be

$$\begin{aligned}\Delta_{\text{chord}}(q) &= \sum_{k=1}^{N}(q \star p_k)\cdot r_k \\ &= \sum_{a=0,b=0}^{3} Q(q)_{ba}\sum_{k=1}^{N} {p_k}^a {r_k}^b \\ &= \operatorname{tr} Q(q)\cdot W \ , \end{aligned} \tag{35}$$

where $Q(q)$ is defined by Eq. (1), and $W_{ab} = \sum_{k=1}^{N} {p_k}^a {r_k}^b$, in parallel to the Euclidean-space cross-covariance matrix E_{ab}, except that the range of (a,b) is $(0\ldots3)$ instead of $(1\ldots3)$, and the data are unit vectors instead of free vectors. Note also that $Q(q)$ is *linear* in q, while $R(q)$ was *quadratic* in q. Pulling out the (linear) coefficients of (q_0, q_1, q_2, q_3), we find

$$\Delta_{\text{chord}}(q) = q \cdot V \ , \tag{36}$$

where the frame data have collapsed to a single column vector V, analogous to the profile matrix M, of the form

$$V = \begin{bmatrix} +W_{00} + W_{11} + W_{22} + W_{33} \\ +W_{01} - W_{10} + W_{23} - W_{32} \\ +W_{02} - W_{20} + W_{31} - W_{13} \\ +W_{03} - W_{30} + W_{12} - W_{21} \end{bmatrix} \ . \tag{37}$$

With the caveat that we cannot guarantee with absolute certainty the absence of sign issues in Eq. (35), the solution for the optimal unit quaternion trivializes to

$$q_{\text{opt}} = \frac{V}{\|V\|} \ , \tag{38}$$

since that immediately maximizes the value of Δ_{chord} in Eq. (36). This gives the cost at maximum to be

$$\Delta_{\text{chord}}(q_{\text{opt}}) = \|V\| \ , \tag{39}$$

and thus $\|V\|$ is the exact orientation frame analog of the spatial RMSD maximal eigenvalue ϵ_{opt}, except it is far easier to compute.

Geometrically, the relationship of the arc and chord distances for a set of orientation-frame test and reference data can be displayed as in Fig 10 , which depicts both the quaternion arc and chord distances for each point pair in the 4D quaternion space corresponding to the 2D schematic presentation in Fig 9 . In Fig 11 and 12, we show the quaternion-space analog of the progress from the initial state of the QRMSD problem in Fig 10 , corresponding to the identity quaternion in Eq. (35), through an intermediate rotation in Fig 11 , and finally reaching the optimal alignment of test and reference frames in Fig 12 corresponding to q_{opt} from Eq. (38).

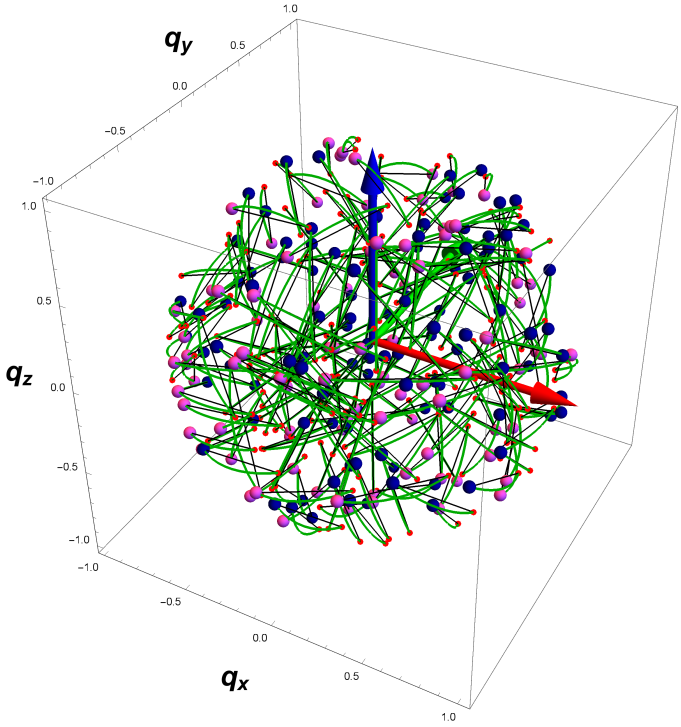


Fig 10. fig3DFrmArcChord3.pdf. Plot of the 3D components of an orientation data set comparing the quaternion arc-length distances between the test and reference points to the approximate chord-length distances.

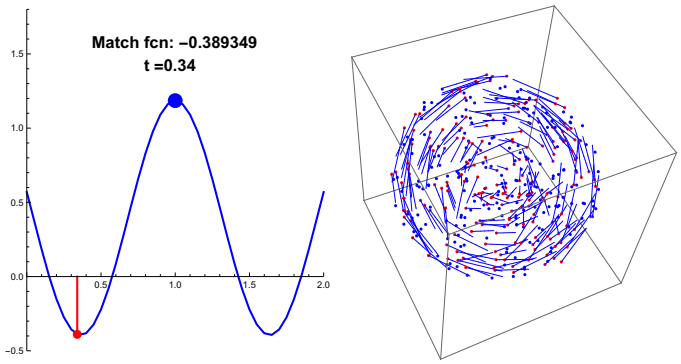


Fig 11. fig3FRotMatch4.pdf. Partial rotation of the test set from the quaternion identity state towards the optimal alignment.

Matrix Form of the Linear Vector Chord Distance. While Eq. (36) does not immediately fit into the eigensystem-based RMSD matrix method used in the previous section, it can in fact be easily

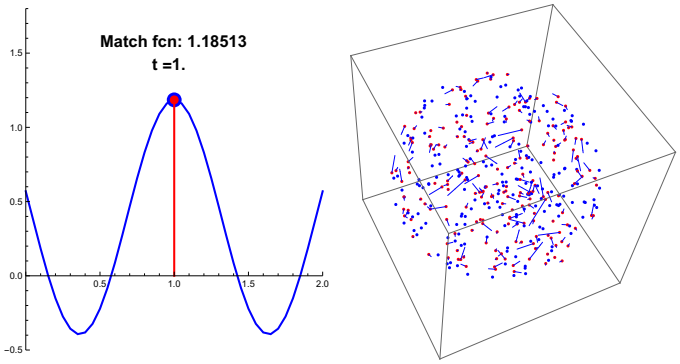


Fig 12. fig3FRotMatch5.pdf. The optimal alignment at $t = 1$ showing the best match of the test quaternion frame set to its reference set of orientation frames.

transformed from a system linear in q to an equivalent matrix system *quadratic* in q. Since any power of optimization measure will yield the same extremal solution, we can simply *square* the right-hand side of Eq. (36) and write the result in the form

$$\begin{aligned} \Delta_{\text{chord-sq}} &= (q \cdot V)(q \cdot V) \\ &= \sum_{a=0,b=0}^{3} q_a \, V_a V_b \, q_b \\ &= q \cdot \Omega \cdot q \ , \end{aligned} \tag{40}$$

where $\Omega_{ab} = V_a V_b$ is a 4×4 symmetric matrix with $\det \Omega = 0$, and $\operatorname{tr} \Omega = \sum_a {V_a}^2 \neq 0$. The eigensystem of Ω can be solved either numerically or algebraically with the extensions of our method to the $\operatorname{tr} \neq 0$ case in Appendix B . The process differs dramatically from what we did with Δ_{chord}, but the eigenvectors are necessarily *identical*. Thus it is in fact possible to merge the Δ_{chord} QRMSD system into the matrix method of the spatial RMSD using Eq. (40) if desired.

Quadratic Rotation Matrix Chord Distance. However, there is another approach that has a very natural way to incorporate manifestly *sign-independent* chord distances into our general context, and which has a very close relationship to Δ_{chord}. The method begins with the observation that full 3D rotation matrices like Eq. (3) can be arranged to rotate the set of frames of the $\{p_k\}$ to be as close as possible to the reference frame $\{r_k\}$ by employing a measure that is a particular product of rotation matrices (see, e.g., [27]). The essence is to notice that the trace of any 3D rotation matrix expressed in axis-angle form (rotation about a fixed axis $\hat{\mathbf{n}}$ by θ) can be expressed in two equivalent forms:

$$\operatorname{tr} R(\theta, \hat{\mathbf{n}}) = 1 + 2\cos\theta \tag{41}$$

$$\operatorname{tr} R(q) = 3{q_0}^2 - {q_1}^2 - {q_2}^2 - {q_3}^2 \ . \tag{42}$$

We thus examine the following alternative, which, remarkably, produces an explicitly symmetric and traceless profile matrix in the quaternions even though there is a constant in the expression Eq. (41). We ignore

constant factors, and begin with this form of the orientation-frame measure [27]:

$$\begin{aligned}
\Delta_{\text{RRR}} &= \sum_{k=1}^{N} \text{tr}\left[R(q)\cdot R(p_k)\cdot R^{-1}(r_k)\right] && (43)\\
&= \sum_{k=1}^{N} \text{tr}\left[R(q)\cdot R(p_k)\cdot R(\bar{r}_k)\right] && (44)\\
&= \sum_{k=1}^{N} \text{tr}\left[R(q)\cdot R(p_k\star\bar{r}_k)\right] && (45)\\
&= \sum_{k=1}^{N} \text{tr}\left[R(q\star p_k\star\bar{r}_k)\right] \ , && (46)
\end{aligned}$$

where $\bar{r}$ denotes the complex conjugate or inverse quaternion. We note that due to the correspondence of Δ_{RRR} with a cosine measure (via Eq. (41)), this must be *maximized* to find the optimal q, so both Δ_{chord} and Δ_{RRR} correspond naturally to the cross-term measure we used for Euclidean point data, which we will now refer to as Δ_x when necessary to distinguish it.

We next observe that the formulas for Δ_{RRR} and the pre-summation arguments of Δ_{chord} are related as follows:

$$\sum_{k=1}^{N} \text{tr}\left[R(q)\cdot R(p_k)\cdot R(\bar{r}_k)\right] = \sum_{k=1}^{N}\left(4\left((q\star p_k)\cdot r_k\right)^2 - (q\cdot q)(p_k\cdot p_k)(r_k\cdot r_k)\right) \ , \qquad (47)$$

where of course the last term is essentially just a constant if one applies the unit-length constraint to all the quaternions, but is algebraically essential to the construction. The odd form of Eq. (47) is not a typographical error: the conjugate $\bar{r}$ of the reference data must be used in the $R\cdot R\cdot R$ expression, and the ordinary r must be used in both terms on the right-hand. We conclude that using the $R\cdot R\cdot R$ measure and replacing the argument of Δ_{chord} by its square *before* summing over k are equivalent maximizing measures that eliminate the quaternion sign dependence.

Choosing Eq. (43) has the remarkable feature of producing, via Eq. (3) for $R(q)$, a symmetric, traceless profile matrix $U(p,r)$ that is *quartic* in the quaternion elements p_k and r_k. This variant of the chord-based QRMSD problem thus falls into the same category as the standard RMSD problem, and permits the application of the same exact solution (or, indeed, the traditional numerical solution method if that is more efficient). The profile matrix equation is unwieldy to write down explicitly in terms of the quaternion elements quartic in $\{p,r\}$, but we actually have several options for expressing the content in a simpler form. One is to write the matrices in abstract canonical 3×3 form, e.g.,

$$R(p) = [P] = \begin{bmatrix} p_{xx} & p_{xy} & p_{xz} \\ p_{yx} & p_{yy} & p_{yz} \\ p_{zx} & p_{zy} & p_{zz} \end{bmatrix} \ , \qquad (48)$$

where the *columns* of this matrix are just the three axes of each data element's frame triad. This is often exactly what our original data look like, for example, if the residue orientation frames of a protein are computed from cross-products of atom-atom vectors [25]. Then we can define for each data element the 3×3 matrix

$$[S_k] = [P_k]\cdot[R_k^{-1}] \equiv R(p_k)\cdot R(\bar{r}_k)] = R(p_k\star\bar{r}_k) = R(s_k) \ ,$$

so we can write S either in terms of a 3×3 matrix like Eq. (48) derived from the actual frame-column data, or in terms of Eq. (3) and the quaternion frame data $s_k = p_k\star\bar{r}_k$. We then may write the frame measure in general as

$$\Delta_{\text{RRR}} = \sum_{k=1}^{N} \text{tr}\left(R(q)\cdot S_k\right) = \sum_{a=1,b=1}^{3} R_{ba}(q)S_{ab} \ , \qquad (49)$$

where the frame-based cross-covariance matrix is simply $S_{ab} = \sum_{k=1}^{N} [S_k]_{ab}$. As before, we can easily expand $R(q)$ using Eq. (3) to convert the measure to a 4D linear algebra problem of the form

$$\Delta_{\text{RRR}} = \sum_{a=0,b=0}^{3} q_a \cdot U_{ab} \cdot q_b = q \cdot U \cdot q \ . \tag{50}$$

Now we can write the profile matrix $U = \sum_k U_k$ appearing in Δ_{RRR} either in terms of the individual k-th components of the numerical 3D rotation matrix $[S] = [P] \cdot [R^{-1}]$ or using the composite quaternion $s = p \star \bar{r}$:

$$\begin{aligned} &U_k([S]) \equiv U(s_k) \\ &= \begin{bmatrix} S_{xx} + S_{yy} + S_{zz} & S_{yz} - S_{zy} & S_{zx} - S_{xz} & S_{xy} - S_{yx} \\ S_{yz} - S_{zy} & S_{xx} - S_{yy} - S_{zz} & S_{xy} + S_{yx} & S_{xz} + S_{zx} \\ S_{zx} - S_{xz} & S_{xy} + S_{yx} & -S_{xx} + S_{yy} - S_{zz} & S_{yz} + S_{zy} \\ S_{xy} - S_{yx} & S_{xz} + S_{zx} & S_{yz} + S_{zy} & -S_{xx} - S_{yy} + S_{zz} \end{bmatrix}_k \end{aligned} \tag{51}$$

$$= \begin{bmatrix} 3{s_0}^2 - {s_1}^2 - {s_2}^2 - {s_3}^2 & -4s_0s_1 & -4s_0s_2 & -4s_0s_3 \\ -4s_0s_1 & -{s_0}^2 + 3{s_1}^2 - {s_2}^2 - {s_3}^2 & 4s_1s_2 & 4s_1s_3 \\ -4s_0s_2 & 4s_1s_2 & -{s_0}^2 - {s_1}^2 + 3{s_2}^2 - {s_3}^2 & 4s_2s_3 \\ -4s_0s_3 & 4s_1s_3 & 4s_2s_3 & -{s_0}^2 - {s_1}^2 - {s_2}^2 + 3{s_3}^2 \end{bmatrix}_k \tag{52}$$

Both Eq. (51) and Eq. (52) are *quartic* (and identical) when expanded in terms of the quaternion data $\{p_k, r_k\}$. To compute the necessary 4×4 numerical profile matrix U, one need only substitute the appropriate 3D frame triads or their corresponding quaternions for the kth frame pair and sum over k . Since the orientation-frame profile matrix U is symmetric and traceless just like the Euclidean profile matrix M, the same solution methods for the optimal quaternion rotation q_{opt} will work without alteration.

Evaluation. The validity of our approximate chord-measures for determining the optimal global frame rotation can be evaluated by comparing their outcomes to the precise angular arc-length measure of Eq. (32). The latter is tricky to optimize, but choosing appropriate techniques, e.g., in the Mathematica `FindMinimum[ ]` utility, it is possible to determine good numerical solutions without writing custom code; in our experiments, fluctuations due to numerical precision limitations were noticeable, but presumably conventional conditioning techniques, which we have not attempted to explore, could improve that significantly. We employed a collection of 1000 simulated quaternion data sets of length 100 for the reference cases, then imposed a normal distribution of random noise on the reference data, followed by a global rotation of all those noisy data points distributed around 45° to produce a corresponding collection of corresponding quaternion test data sets to be matched. (Observe that we do *not* expect the optimal rotation angles to match the exact global rotations, though they will be nearby.)

We then collected the optimal quaternions for the following cases:

(a) **Arc-Length (numerical).** This is the "gold standard," modulo the occasional data pair that seems to challenge the numerical stability of the computation (which was to be expected). We obtained the data set (a) of quaternions that numerically minimized the nonlinear geodesic arc-length-squared measure of Eq. (32); this is in principle the best estimate one can possibly get for the optimal quaternion rotations to align a set of 3D test-frame triads with a corresponding set of reference-frame triads. There is no known way to find this set of optimal quaternions using our linear algebra methods.

(b) **Chord-Length (numerical and algebraic).** This approach determines the data set (b) based on the approximation to Eq. (6) illustrated in Fig 9 , replacing the arc-length by the chord-length, which amounts to removing the arccosine and using the effective cosine to define the measure. The form given in Eq. (33) is a minimization problem that is exactly the quaternion analog of the RMSD problem definition in Eq. (6) for spatial data, with the additional constraint that all the spatial data must be

unit-length 4-vectors (which have only 3 degrees of freedom) instead of arbitrary 3-vectors. In addition, the *sign ambiguity* of quaternions must be resolved for Eq. (33) to be valid: if for some k the sign of r_k relative to p_k is inconsistent with the previous pair, Eq. (33) will become nonsense. In the simulations used in our evaluation study, we have used the reference set as the starting point for moderate pertubations to get the test set; in real life, enforcing this constraint is feasible but not necessarily trivial. Just as Eq. (6) and its cross-term form Eq. (7) give exactly the same results for spatial data when the measures are minimized and maximized, respectively, the orientation-problem equations Eq. (33) and Eq. (35) do the same for the quaternion measure. Finally, the two cross-term forms Eq. (36) and Eq. (40) give the same optimal quaternions, with the interesting fact that Eq. (36) yields the optimal quaternion from a linear equation, and Eq. (40) gives an identical result from a quadratic matrix equation that works the same way as the RMSD matrix optimization, except that the symmetric profile matrix is no longer traceless.

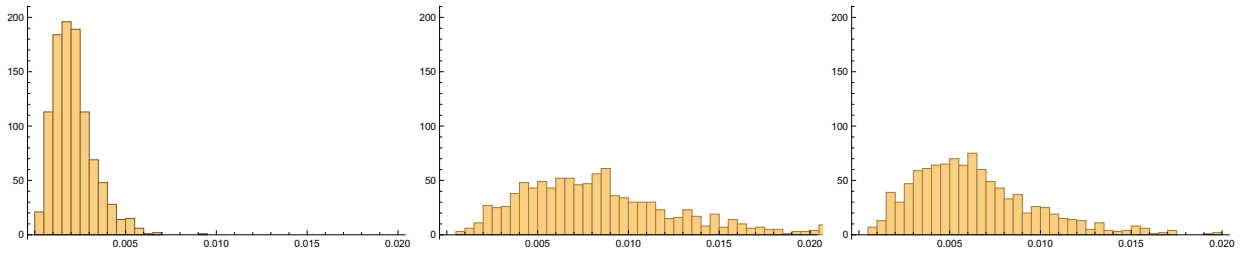
Thus there are in fact four ways of looking at the chord-length measure and obtaining exactly the same optimal quaternions, and we have checked these using two numerical optimizations and two algebraic optimizations. These options are:

- **Minimizing Euclidean Chord-Length Squared.** Here we write the chord-approximation to the QRMSD problem using Eq. (33), which is exactly parallel to the RMSD problem employing Eq. (6), modulo the sign ambiguity issue. We test this by performing a numerical minimization.
- **Maximizing Chord-Length Cross-Term.** Just as the RMSD cross-term maximization problem Eq. (7) is equivalent to the RMSD minimization problem of Eq. (6), we can use maximization of the quaternion cross-term Eq. (35) equivalently with the minimization of the chord-length Eq. (33). We test this by performing a numerical maximization.
- **Linear Reduction of Chord-Length Cross-Term.** Pulling out the linear coefficients of the each quaternion component in Eq. (35) generates Eq. (36), where the 4-vector $V_a(W)$ of Eq. (37) plays the role of the RMSD profile matrix $M_{ab}(E)$ in Eq. (9). Here we test the optimization by algebraically solving the linear expression Eq. (36).
- **Quadratic Equivalent Matrix Form of the Chord-Length Cross-Term.** Finally, there is in fact a maximal matrix eigenvalue problem Eq. (40) that works like Eq. (9) by squaring Eq. (36) to get a matrix problem $q \cdot \Omega \cdot q$ with $\Omega_{ab} = V_a V_b$. Despite the presence of a nonvanishing trace, the maximal quaternion eigenvectors are the same as the other three cases above. This produces the same optimal quaternion solutions as solving the (much, much simpler) linear problem of Eq. (36). This can also be checked algebraically.

(c) $(\mathrm{tr}\,\mathbf{R}(\mathbf{q}) \cdot \mathbf{R}(\mathbf{p}) \cdot \mathbf{R}(\bar{\mathbf{r}}))$ **Chord-Length (algebraic).** Finally, the most rigorous method if consistency of quaternion signs cannot be guaranteed is to use a measure in which algebraic squares occur throughout and enforce rigorous sign-independence. This is our (c) data set. Such measures must of necessity be *quartic* in the quaternion test and reference data, and thus are distinct from the measures of (b) that are *quadratic* in the data elements. This $(\mathrm{tr}\, R(q) \cdot R(p) \cdot R(\bar{r}))$ measure is the form that is most easily integrated into the combined rotational-translational problem treated in the next section, because the combined matrices are both symmetric and traceless like the original RMSD profile matrices. Furthermore, it is obvious from Eq. (47) that this measure is exactly the same as the one obtained from Eq. (35) if we squared *each term in k* before summing the cross-term data elements in option (b). Thus, whichever actual formula we choose, we appear to have exhausted the options for quaternion-sign-independent quartic measures for the orientation data problem.

The task now is simply to evaluate how close the optimal quaternion solutions for the arc-length measure (a) are to the quadratic chord-length measures (b) and the quartic chord-length measures (c). In addition, we would like to know how close the fragile but very elegant quadratic measures (b) are to the rigorously sign-insensitive quartic measures (c); we expect them to be similar, but we do not expect them to be identical.

To quantify the closeness of the measures, we took the magnitude of the inner products between competing optimal quaternions for the same data set, which is essentially a cosine measure, took the



(a:b) (a:c) (b:c)

Fig 13. aab1Histogram.pdf, aac3Histogram.pdf, b1c3Histogram.pdf. Spectrum in degrees of angular differences between optimal quaternion alignment rotations for quaternion frames. (a:b): (a) vs (b), true arc-length vs approximate quadratic chord-length measure. (a:c): (a) vs (c), true arc-length vs approximate quartic chord-length measure. (b:c): (b) vs (c), approximate quadratic vs approximate quartic chord-length measure.

arccosines, and converted to degrees. The results were histogrammed for 1000 random samples consisting of $N = 100$ data points, and are presented in Fig 13. The means and standard deviations of the optimal total rotations relative to the identity frame for the three cases are:

Measure Type	Mean(deg)	Std Dev(deg)
(a) arc-length	44.8062	11.2307
(b) chord quadratic	44.8063	11.2308
(c) chord quartic	44.8065	11.2310

.

One can see that our simulated data set involved a large range of global rotations, and that all three methods produced a set of rotations back to the optimal alignment that are not significantly different statistically. We thus expect very little difference in the histograms of the case-by-case optimal quaternions produced by the three methods. The mean differences illustrated in the Figures are summarized as follows:

Figure:(Pair)	Mean(deg)	Std Dev(deg)
Figure 13 (a:b)	0.0021268	0.0011284
Figure 13 (a:c)	0.0084807	0.0044809
Figure 13 (b:c)	0.0063539	0.0033526

.

We emphasize that these numbers are in degrees for 1000 simulated samples with a distribution of global angles having a standard deviation of 11°. Thus we should have no issues using the chord approximation, though it does seem that the $q \cdot V$ measure is significantly better both in accuracy and simplicity of computation (modulo the ever-present need to resolve sign ambiguity).

6 The 3D Combined Point+Frame Matching Problem.

Since we now have precise matching procedures for both 3D spatial coordinates and 3D frame triad data (using the exact measure for the former and the approximate chord measure for the latter), we can consider the full 6 degree-of-freedom matching problem for combined data from a single structure. In fact this problem can also be solved in closed algebraic form given the eigensystem formulation of the orientation matching problem already presented in the previous section. While there are clearly appropriate domains of this type, e.g., any protein structure in the PDB database can be converted to a list of residue centers and their local frame triads [25], little is known at this time about the potential value of combined matching. To establish the most complete possible picture, we now proceed to describe the details of our solution to the matching problem for combined translational and rotational data, but we remark at the outset that the results of the combined system are not obviously very interesting.

In our treatment, we will assume the Δ_{RRR} measure since its profile matrix is traceless and manifestly independent of any random flipping of the quaternion signs, but there is no obstacle to using $\Delta_{\text{frame-sq}}$ if the data are properly prepared and one prefers the simpler measure. For notational simplicity, we will let Δ_f stand for whatever orientation frame measure we have chosen, corresponding to Δ_x for the spatial measure, and thus we will denote the combined measure by Δ_{xf}.

The Combined Optimization Measure. A significant aspect of establishing a combined measure including the point measure Δ_x and the frame orientation measure Δ_f is the fact that the measures are *dimensionally incompatible*. We *cannot* directly combine the corresponding data minimization measures $\Delta_x(q_x) = \epsilon_{x:\max}$ and $\Delta_f(q_f) = \epsilon_{f:\max}$ because the spatial measure has dimensions of $(\text{length})^2$ and the frame measure is essentially a dimensionless trigonometric function (the arc-distance measure produces $(\text{radians})^2$, which is still incompatible).

While it should be obvious that a combined measure requires an arbitrary, problem-specific, interpolating constant with dimensions of length to produce a compatible measure, there has been some confusion in the molecular entropy literature, where such measures seem first to have been employed. These issues were resolved and dimensionful constants introduced, e.g., in the work of Fogolari, et al. [29,30]. Our approach to defining a valid heuristic combined measure has three components:

- **Normalize the Profiles.** The numerical sizes of the maximal eigenvalues of the Δ_x and the Δ_f systems can easily differ by orders of magnitude. Since scaling the profile matrices changes the eigenvalues *but not the eigenvectors*, it is perfectly legitimate to start by dividing the profiles by their maximal eigenvalues before beginning the combined optimization, since this accomplishes the sensible effect of assigning maximal eigenvalues of exactly unity to both of our scaled profile matrices.
- **Interpolate between the Profiles.** To allow an arbitrary sensible weighting distinguishing between a location-dominated measure and an orientation-dominated measure, we simply incorporate a linear interpolation parameter $t \in [0,1]$, with $t = 0$ singling out Δ_x and the pure (unit eigenvalue) location-based RMSD, and $t = 1$ singling out Δ_f and the pure orientation (unit eigenvalue) QRMSD solution.
- **Scale the Frame Profile.** Finally, we incorporate the mandatory dimensional scaling adjustment by incorporating one additional (nominally dimensional) parameter σ that scales the orientation parameter space described by Δ_f to be more or less important than the "canonical" spatial dimension component Δ_x, which we leave unscaled. That is, with $\sigma = 0$ only the spatial measure survives, with $\sigma = 1$, the normalized measures have equal contributions, and with $\sigma > 1$, the orientation measure dominates (this effectively undoes the original frame profile eigenvalue scaling).

We thus start with a combined spatial-rotational measure of the form

$$\begin{aligned}
\Delta_{\text{initial}} &= (1-t)\sum_{a=1,b=1}^{3} R^{ba}(q)E_{ab} + t\,\sigma \sum_{a=1,b=1}^{3} R^{ba}(q)S_{ab} \\
&= (1-t)\,\text{tr}\,(R(q)\cdot E) + t\,\sigma\,\text{tr}\,(R(q)\cdot S) \\
&= \sum_{a=0,b=0}^{3} q_a\,[(1-t)M_{ab}(E) + t\,\sigma\,U_{ab}(S)]\,q_b \\
&= q\cdot[(1-t)M(E) + t\,\sigma\,U(S)]\cdot q\ , \qquad (53)
\end{aligned}$$

and then impose the unit-eigenvalue normalization on $M(E)$ and $U(S)$, giving our final measure as

$$\Delta_{xf}(t,\sigma) = q\cdot\left[(1-t)\frac{M(E)}{\epsilon_x} + t\,\sigma\frac{U(S)}{\epsilon_f}\right]\cdot q\ . \qquad (54)$$

Because of the dimensional incompatibility of Δ_x and Δ_f, we have to treat the ratio

$$\lambda^2 = \frac{t\sigma}{1-t}$$

as a dimensional constant such as that adopted by Fogolari et al. [30] in their entropy calculations, so if t is dimensionless, then σ carries the dimensional scale information.

From the profile matrix of Eq. (54), we now extract our optimal rotation solution using the same equations as always, Eqs. (24), (27), and (11), that we used to solve the standard RMSD maximal eigenvalue problem. The result is a parameterized eigensystem

$$\left.\begin{array}{l} \epsilon_{\rm opt}(t,\sigma) \\ q_{\rm opt}(t,\sigma) \end{array}\right\} \tag{55}$$

yielding the optimal values $R(q_{\rm opt}(t,\sigma))$, $\Delta_{xf} = \epsilon_{\rm opt}(t,\sigma)$ based on the data $\{E,S\}$ no matter what we take as the values of the two variables (t,σ).

Properties of the Combined Optimization. Substantially different features arise in the solutions depending on how close the optimal rotations were for the initial, separate, systems Δ_x and Δ_f. We now choose a selection of simulated data sets with the following choices of approximate initial global rotations of the test data sets relative to the reference data:

DATA ID	**Initial(Space, Orientation) Offset**	Measured Eigenvector Offset
Data Set 1	$(\mathbf{22}^\circ, -\mathbf{22}^\circ)$	44.60
Data Set 2	$(\mathbf{22}^\circ, -\mathbf{11}^\circ)$	21.98
Data Set 3	$(\mathbf{22}^\circ, \mathbf{0}^\circ)$	11.15
Data Set 4	$(\mathbf{22}^\circ, \mathbf{11}^\circ)$	11.15
Data Set 5	$(\mathbf{22}^\circ, \mathbf{21}^\circ)$	1.20

Table 1. Offsets of sample data for the spatial vs orientation data used in exploring the properties of combined measures.

In Fig 14, we plot the trajectory of the maximal combined similarity measure for Data Set 1 as a function of t, showing the behavior for $\sigma = 1.0, 0.80$, and 1.15. Figure 15 shows a more comprehensive representation of the continuous behavior with σ, and in both figures, we see that the true optima are *at the end points*, $t = 0, 1$, the locations associated with the pure profile eigenvector solutions $q_x(\text{opt})$ and $q_f(\text{opt})$. There is no *better* optimal eigenvector (i.e., global rotation) for any intermediate value of t. In some circumstances, however, it might be argued that it is appropriate to choose the *distinguished value* of t at the minimum of the curve $\Delta_{xf}(t, \sigma = 1)$. As we shall see in a moment, just as in Fig 14 for $\sigma = 1$, this point is generally within a few percent of $t = 0.5$. As the spatial and orientation optima get closer and closer, the curves in t become much flatter and less distinguished, while the variation in σ is qualitatively the same as in Fig 15 .

Finally, we examine one more amusing visualization of the properties of the composite solutions, restricting ourselves to $\sigma = 1$ for simplicity, and examining the "sideways warp" in the quaternion eigenvector $q_{\rm opt}(t, \sigma = 1)$ in Eq. (55). We examine what happens to the combined similarity measure Eq. (54) if we smoothly interpolate from the identity matrix (that is, the quaternion $q_{\rm ID} = (1, 0, 0, 0)$) through the optimal solution for each t and beyond the optimum by the same amount, using the *slerp* interpolation defined in Eq. (5), i.e., $q(s) = slerp(q_{\rm ID}, q_{\rm opt}(t, \sigma = 1), s)$. Figure 16 shows Data Set 1, with the largest relative spatial vs orientation angular differences, Figure 17 corresponds to the intervening Data Sets 2, 3, 4, and 5, with the Data Set parameters in Table 1 ; Data Set 5 in particular is perhaps the most realistic example, having nearly identical spatial and angular rotations, and we see negligible differences between the spatial and angular structures. These graphics also show how the local, non-optimal, neighboring quaternion values peak in s at the optimal ridge going from $t = 0$ to $t = 1$. The red dot is the maximum of Δ_x at $t = 0$, the green dot is the maximum of Δ_f at $t = 1$, and the blue dot, specific to each data set, is the distinguished point at the *minimum* of $\Delta_{xf}(t, \sigma = 1)$ in t, which for our data sets are always within 1% of $t = 0.5$. We observe that for equal and opposite rotations, the midpoint coincides almost exactly with the identity quaternion that occurs at the left and right boundaries of the plot. In other respects, the data in these figures show that we do not have *maxima* in the middle of the interpolation in t, but we do have a distinguished value, always very near $t = 0.5$, that could be used as a baseline for a hybrid translational-rotational rotation choice.

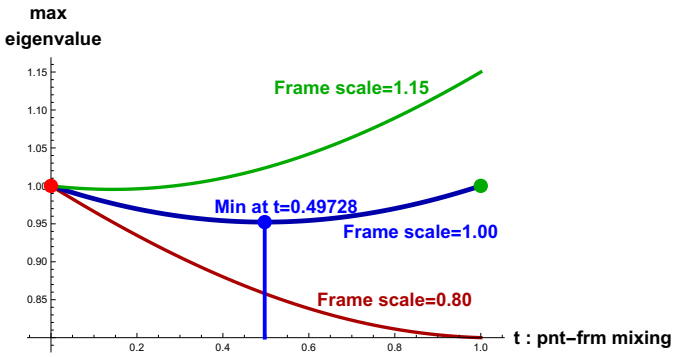


Fig 14. combinedSimpleTcurves1.pdf. The blue curve is the path of the composite eigenvalue for Data Set 1 (the value of the similarity measure $\Delta_{xf}(t, 1)$) in the interpolation variable t with equally weighted space and orientation data, i.e., $\sigma = 1$. It has maxima only at the "pure" extremes at $t = 0, 1$, but there is a minimum that occurs, for these data, not at $t = 1/2$, but very nearby at $t = 0.49728$. Increasing the influence of the spatial data by taking $\sigma = 0.8$ gives the red curve, and increasing the influence of the orientation data by taking $\sigma = 1.15$ gives the green curve.

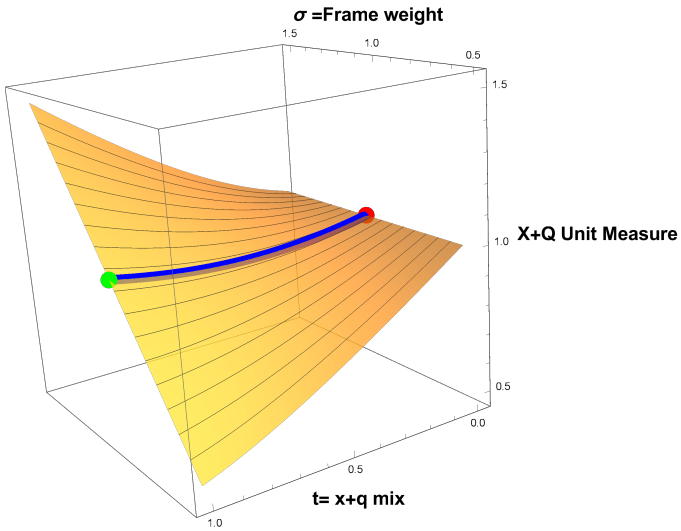
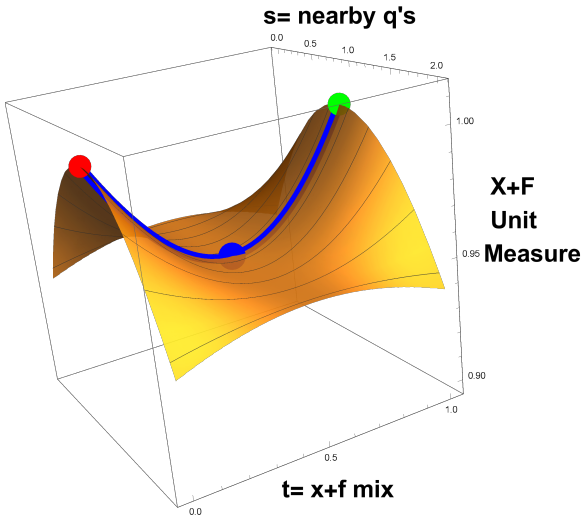
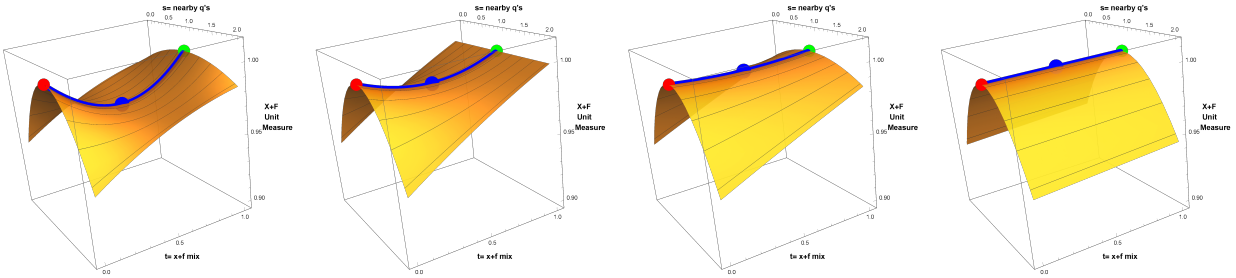


Fig 15. combinedTSigsurf1.pdf. The $\Delta(t, \sigma)$ similarity-measure surface for Data Set 1 as a function of the interpolation parameter t and the relative scaling of the orientation term with σ, with the slightly concave curve at $\sigma = 1$ in the middle. The other data sets look very much like this one.



(Set 1)

Fig 16. combinedTSlerp1.pdf . The $\Delta_{xf}(t, 1)$ similarity-measure surface for Data Set 1, x-angle 22°, f-angle −22°, and fixed $\sigma = 1$ showing the deviation with the quaternion varying perpendicularly around the solution $q(t)$, starting at the identity quaternion at $s = 0$, as a function of the interpolation parameter t. Since $q(t)$ is the maximal eigenvector, all variations in q peak there. Both have distinguished central points at $t \approx 0.5$.



(Set 2) (Set 3) (Set 4) (Set 5)

Fig 17. combinedTS2345.pdf . The $\Delta_{xf}(t, 1)$ similarity-measures with $q(s)$ interpolated from the identity through the optimum for Δ_{xf} and past to the identity-mirror point, for Data Sets 2, 3, 4, and 5, where Data Set 5 has the x-angle and the f-angle only one degree apart, as we might have for real experimental data.

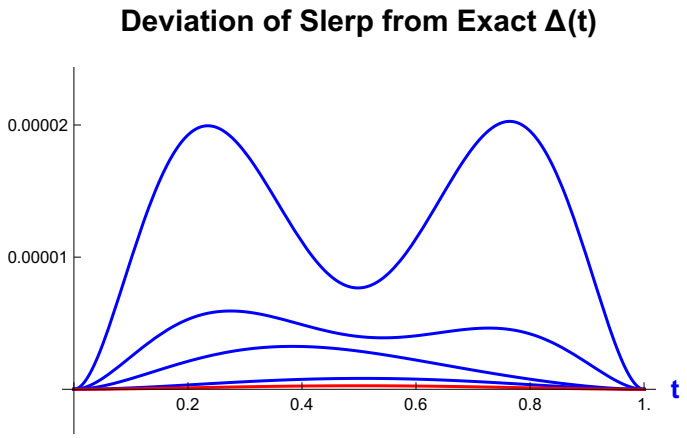


Fig 18. combinedQQSlerpOffset.pdf. Here we see how close a simple *slerp*(t) between the extremal optimal eigenvectors $q_{\text{opt}}(t=0,\sigma=1)=q_x(\text{opt})$ and $q_{\text{opt}}(t=1,\sigma=1)=q_f(\text{opt})$ is to the rigorous result where we optimized $q_{\text{opt}}(t,\sigma=1)$ *for all* t. The differences are *relative to the unit eigenvalue*, and thus are of order thousandths of a percent, decreasing significantly as the global rotations applied to the space and orientation data approach one another. The largest deviation is for Data Set 1, which interestingly has a third minimum near the center in t; for the highly similar data in Data Set 5, the difference shown in red had to be magnified by 100 even to show up on the graph.

The Simple Approximation. Having now observed that it is possible to construct and solve a rigorous combined RMSD-QRMSD problem (with the chord-distance approximation in the angular measure of course), one might ask how that compares to the very simplest idea one might use to interpolate between the measures: what if we take the rigorous combined profile matrix defined by Eq. (54), along with the *slerp* relating the two optimal eigenvectors of the independent spatial and orientation frame problems, that is

$$q(t) = slerp(q_{x:\mathrm{opt}}, q_{f:\mathrm{opt}}, t) \ . \tag{56}$$

Given the individual optimal eigenvectors, if we simply plug this trivial $q(t)$ into Eq. (54) for any t (and $\sigma = 1$), we find *negligible difference* between the results we found above in this Section and Eq. (54) evaluated with $q(t)$ in Eq. (56).

In Fig 18, we plot the continuous differences of the similarity functions, which we recall are scaled to have a maximal eigenvalue equal to unity. These scaled differences are on the order of one thousandth of a percent or less as the global rotations applied to the spatial and rotational data become close to one another. We conclude that for all practical purposes, we might as well use Eq. (56), in the context of Eq. (54), to estimate the combined similarities.

Conclusion

Our objective was to explore quaternion-based treatments of the RMSD data-comparison problem as developed in the work of Davenport [8], Horn [9], Diamond [10], Kearsley [11], and Kneller [12], among others, and to study its exact solutions, as well as extending it to handle wider problems. We studied the intrinsic properties of the RMSD problem for translational, rotational, and rotational-translational data comparison in quaternion-accessible domains, and we obtained exact algebraic solutions for the eigensystems of the 3D and 4D spatial RMSD problem, as well as solutions for a family of attractive approximations to the quaternion form of the corresponding orientation-frame problems (QRMSD). We also found closed-form solutions for the combined translational and orientation-frame RMSD problem, but the results were essentially indistinguishable from those obtained using a simple interpolation between the optimal spatial and the optimal orientation-frame eigenvectors.

Acknowledgments

We thank Sonya M. Hanson for reacquainting us with this problem, providing much useful input and advice, and motivating us to pursue this challenging investigation to its conclusion. We also express our appreciation to Roger Germundsson for very valuable input, and to Randall Bramley for informative discussions about the properties of eigensystems.

Appendices

A Two-Dimensional Limit of 3D Problem

All rotations of the type we have been trying to optimize reduce to a rotation in a 2D plane, which in 3D is defined by the plane perpendicular to the eigenvector $\hat{\mathbf{n}}$ of the rotation matrix Eq. (3). Data sets that are highly linear, determining a robust straight line from least squares, can even circumvent the RMSD problem entirely: a very good rotation matrix can be calculated from the direction $\hat{\mathbf{x}}$ determined by the line fitted to the data set $\{x_i\}$, and the similar direction $\hat{\mathbf{y}}$ corresponding to the reference data set $\{y_i\}$. An optimal rotation matrix in 3D is then simply

$$R(\theta, \hat{\mathbf{n}}) = R(\arccos(\hat{\mathbf{x}} \cdot \hat{\mathbf{y}}), \widehat{\hat{\mathbf{x}} \times \hat{\mathbf{y}}}) \ ,$$

which is easily generalized to any dimension by isolating just the projections of vectors to the plane determined by $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$, and rotating in that 2D basis. Thus we conclude that, in general, if we had access to a prescient preconditioning rotation of the proper form, the entire RMSD problem would reduce to a very simple rotation in some $\{\hat{\mathbf{x}}, \hat{\mathbf{y}}\}$ plane parameterized by a single angle. We can simulate this, giving a massively simpler set of expressions, by assuming the data are coplanar, all having $z = 0$ (or more conditions in higher dimensions) and thus lying in the $\{\hat{\mathbf{x}}, \hat{\mathbf{y}}\}$ plane, for example. This reduces our fundamental RMSD profile matrix Eq. (10) for M to

$$M_{z=0} = \begin{bmatrix} x+y & 0 & 0 & c \\ 0 & x-y & C & 0 \\ 0 & C & -x+y & 0 \\ c & 0 & 0 & -x-y \end{bmatrix} \ , \tag{57}$$

where $x = E_{xx}$, $y = E_{yy}$, $c = E_{xy} - E_{yx}$, and $C = E_{xy} + E_{yx}$. Then $p_2 = -c^2 - C^2 - 2(x^2 + y^2)$, $p_3 = 0$, and $p_4 = (c^2 + (x+y)^2)(C^2 + (x-y)^2)$, and similarly for the other cyclic cases, $x = 0$ and $y = 0$. These are obviously functions of only two variables, $u = c^2 + (x+y)^2$ and $v = C^2 + (x-y)^2$, so we can write in general $p_2 = -u - v$ and $p_4 = uv$. Equation (12) reduces to $e^4 + e^2 p_2 + p_4 = 0$ and the eigenvalues become $\epsilon = (\sqrt{u}, \sqrt{v}, -\sqrt{v}, -\sqrt{u}\,)$, while the normalized (quaternion) eigenvectors become

$$q = \left\{ \begin{bmatrix} \frac{x+y+\sqrt{u}}{\sqrt{c^2+\left(x+y+\sqrt{u}\right)^2}} \\ 0 \\ 0 \\ \frac{c}{\sqrt{c^2+\left(x+y+\sqrt{u}\right)^2}} \end{bmatrix}, \begin{bmatrix} 0 \\ \frac{x-y+\sqrt{v}}{\sqrt{C^2+\left(x-y+\sqrt{v}\right)^2}} \\ \frac{C}{\sqrt{C^2+\left(x-y+\sqrt{v}\right)^2}} \\ 0 \end{bmatrix}, \begin{bmatrix} 0 \\ \frac{x-y-\sqrt{v}}{\sqrt{C^2+\left(x-y-\sqrt{v}\right)^2}} \\ \frac{C}{\sqrt{C^2+\left(x-y-\sqrt{v}\right)^2}} \\ 0 \end{bmatrix}, \begin{bmatrix} \frac{x+y-\sqrt{u}}{\sqrt{c^2+\left(x+y-\sqrt{u}\right)^2}} \\ 0 \\ 0 \\ \frac{c}{\sqrt{c^2+\left(x+y-\sqrt{u}\right)^2}} \end{bmatrix} \right\} . \tag{58}$$

The leading eigenvalue and its eigenvector produce this optimal rotation in the $\{\hat{\mathbf{x}}, \hat{\mathbf{y}}\}$ plane:

$$R_{2\mathrm{D}} = \begin{bmatrix} \frac{\left(x+y+\sqrt{u}\right)^2-c^2}{c^2+\left(x+y+\sqrt{u}\right)^2} & -\frac{2c\left(x+y+\sqrt{u}\right)}{c^2+\left(x+y+\sqrt{u}\right)^2} \\ \frac{2c\left(x+y+\sqrt{u}\right)}{c^2+\left(x+y+\sqrt{u}\right)^2} & \frac{\left(x+y+\sqrt{u}\right)^2-c^2}{c^2+\left(x+y+\sqrt{u}\right)^2} \end{bmatrix} . \tag{59}$$

Yet Another Form. However, we have neglected something. How does this look if we simply go back to the data matrices for 2D? Let us first write down the 2D version of Eq. (6), taking $E_{ab} = \sum_{k=1}^{N} {x_k}^a {y_k}^b$ for $a, b = \{1, 2\}$, so the raw form for the spatial RMSD task is to find the rotation matrix

$$R_2(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix}$$

maximizing

$$\Delta_2 = \sum_{k=1}^{N} (R_2 \cdot x_k) \cdot y_k = \sum_{a=1,b=1}^{2} {R_2}^{ba} E_{ab} \ = \ (E_{xx} + E_{yy}) \cos\theta + (E_{xy} - E_{yx}) \sin\theta \ . \tag{60}$$

We can either differentiate with respect to θ and set $\Delta_2'(\theta) = 0$, or simply observe directly that $\Delta_2(\theta)$ is largest when the vector $(\cos\theta, \sin\theta)$ is parallel to its coefficients; both arguments lead to the solution

$$\tan\theta = \frac{E_{xy} - E_{yx}}{E_{xx} + E_{yy}} = \frac{N}{M} \tag{61}$$

$$(\cos\theta,\, \sin\theta) = \left(\frac{M}{\sqrt{M^2+N^2}}, \frac{N}{\sqrt{M^2+N^2}} \right) . \tag{62}$$

Now we can see that

$$\begin{array}{rcccl} x+y &=& E_{xx}+E_{yy} &=& M \\ c &=& E_{xy}-E_{yx} &=& N \\ u &=& (E_{xx}+E_{yy})^2 + (E_{xy}-E_{yx})^2 &=& M^2+N^2 \\ \epsilon &=& \lambda &=& \sqrt{M^2+N^2}\ , \end{array} \tag{63}$$

and $c^2 + (x+y+\sqrt{u})^2 = 2\lambda(M+\lambda)$. Thus in fact the profile matrix becomes

$$\mathbf{M}_2 = \begin{bmatrix} M & N \\ N & -M \end{bmatrix} \tag{64}$$

and this has eigenvalues exactly $\epsilon = \pm\sqrt{u} = \pm\sqrt{M^2+N^2}$, and the eigenvectors are the first and last columns of Eq. (58) expressed in terms of Eq. (63), so the maximal eigenvector is (a, b), where

$$\begin{array}{rcccccl} a &=& \cos(\theta/2) &=& \dfrac{\lambda+M}{\sqrt{2\lambda(\lambda+M)}} &=& \sqrt{\dfrac{\lambda+M}{2\lambda}} \\ b &=& \sin(\theta/2) &=& \dfrac{N}{\sqrt{2\lambda(\lambda+M)}} &=& \operatorname{sign} N \sqrt{\dfrac{\lambda-M}{2\lambda}}\ . \end{array} \tag{65}$$

(Note the crucial $(\operatorname{sign} N)$ factor.) Going back to our original 2D rotation matrix in Eq. (59) and substituting Eq. (63), we recover our optimal result, namely

$$R_2(\theta) = \begin{bmatrix} \frac{M}{\sqrt{M^2+N^2}} & -\frac{N}{\sqrt{M^2+N^2}} \\ \frac{N}{\sqrt{M^2+N^2}} & \frac{M}{\sqrt{M^2+N^2}} \end{bmatrix} \tag{66}$$

$$= \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} . \tag{67}$$

These results are interesting to study because, despite the complexity of the general solution, the intrinsic algebraic structure of any RMSD problem is entirely characterized by a planar rotation such as that described by Eq. (59) and Eq. (66).

B Double-Quaternion Approach to 4D RMSD Formula

This Appendix presents the nontrivial steps needed to understand and solve the 4D spatial and orientation-frame RMSD optimization problems in the quaternion framework. Here we extend our solutions for the 4×4 symmetric, traceless profile matrix M_3 arising from 3D Euclidean data (see Eqs. (24, 25)) to the case of unconstrained 4×4 matrices M_4, which arise naturally for 4D Euclidean data. While we might expect this solution to allow us to solve the 4D RMSD problems in exactly the same fashion as in 3D, this is, interestingly, false. We will need several stages of analysis to actually find the correct way to apply our solutions to solve the 4D RMSD context.

B.1 Double Quaternions and 4D Rotations.

We start with a more general property of quaternions that we have not seen so far, extending Eq. (3) from three Euclidean dimensions to four Euclidean dimensions by choosing two *distinct* quaternions in Eq. (2), and writing

$$p \star (w,\, x,\, y,\, z) \star \bar{q} = R_4(p,q) \cdot \mathbf{x}_4 \ . \tag{68}$$

Here R_4 turns out to be an orthonormal 4D matrix that is quadratic in the *pair* (p,q) of unit quaternion elements, which together have exactly the six degrees of freedom required for the most general 4D Euclidean rotation in the special orthogonal group $\mathbf{SO}(4)$. The algebraic form of this 4D rotation matrix is

$$R_4(p,q) = \left[\begin{array}{cc} p_0q_0 + p_1q_1 + p_2q_2 + p_3q_3 & -p_1q_0 + p_0q_1 + p_3q_2 - p_2q_3 \\ p_1q_0 - p_0q_1 + p_3q_2 - p_2q_3 & p_0q_0 + p_1q_1 - p_2q_2 - p_3q_3 \\ p_2q_0 - p_3q_1 - p_0q_2 + p_1q_3 & p_3q_0 + p_2q_1 + p_1q_2 + p_0q_3 \\ p_3q_0 + p_2q_1 - p_1q_2 - p_0q_3 & -p_2q_0 + p_3q_1 - p_0q_2 + p_1q_3 \\ -p_2q_0 - p_3q_1 + p_0q_2 + p_1q_3 & -p_3q_0 + p_2q_1 - p_1q_2 + p_0q_3 \\ -p_3q_0 + p_2q_1 + p_1q_2 - p_0q_3 & p_2q_0 + p_3q_1 + p_0q_2 + p_1q_3 \\ p_0q_0 - p_1q_1 + p_2q_2 - p_3q_3 & -p_1q_0 - p_0q_1 + p_3q_2 + p_2q_3 \\ p_1q_0 + p_0q_1 + p_3q_2 + p_2q_3 & p_0q_0 - p_1q_1 - p_2q_2 + p_3q_3 \end{array} \right] . \tag{69}$$

Since this is a quadratic form in p and q, the rotation is unchanged under $(p,q) \to (-p,-q)$, and the quaternions are again a double covering. If we set $p = q$, we recover a matrix that leaves the w component invariant, and is just the rotation Eq. (3) for the $\mathbf{x}_3 = (x,y,z)$ component. Rotations can be composed in quaternion form similarly to the 3D case, with $R_4(p,q) \cdot R_4(p',q') = R_4(p \star p',\, q \star q')$. We observe that the 4D columns of Eq. (69) can be used to define 4D Euclidean orientation frames in the same fashion as the 3D columns of Eq. (3), and we will return to this at the end of the Appendix.

B.2 Starting Point for the 4D RMSD Problem.

The 4D double quaternion matrix Eq. (69) provides the most general quaternion context that we know of for expressing an RMSD problem. This RMSD minimization problem for 4D Euclidean point data is equivalent to maximizing the cross-term expression

$$\Delta_4 = \sum_{i=1}^{N} (R_4 \cdot x_i) \cdot y_i = \sum_{a=0,b=0}^{3} {R_4}^{ba} E_{4:ab} \ = \ \mathrm{tr}\, R_4 \cdot E_4, \tag{70}$$

where as usual

$$E_{4:ab} = \sum_{i=1}^{N} {x_i}^a {y_i}^b = \left[\mathbf{X}^t \cdot \mathbf{Y}\right]_{ab} \ , \tag{71}$$

and we define the range of (a,b) to be $(0,\ldots,3)$.

Using Eq. (69) in Eq. (70) to perform the now-familiar rearrangement of the similarity function, we can rewrite our measure as

$$\Delta_4 = \mathrm{tr}\, R_4 \cdot E_4 \ = \ (p_0,p_1,p_2,p_3) \cdot M_4(E_4) \cdot (q_0,q_1,q_2,q_3)^t \ \equiv \ p \cdot M_4(E_4) \cdot q \ , \tag{72}$$

where the extended profile matrix now becomes

$$M_4(E) = \left[\begin{array}{cc} E_{ww} + E_{xx} + E_{yy} + E_{zz} & +E_{yz} - E_{zy} - E_{wx} + E_{xw} \\ +E_{yz} - E_{zy} + E_{wx} - E_{xw} & E_{ww} + E_{xx} - E_{yy} - E_{zz} \\ +E_{zx} - E_{xz} + E_{wy} - E_{yw} & +E_{xy} + E_{yx} + E_{wz} + E_{zw} \\ +E_{xy} - E_{yx} + E_{wz} - E_{zw} & +E_{zx} + E_{xz} - E_{wy} - E_{yw} \\ +E_{zx} - E_{xz} - E_{wy} + E_{yw} & +E_{xy} - E_{yx} - E_{wz} + E_{zw} \\ +E_{xy} + E_{yx} - E_{wz} - E_{zw} & +E_{zx} + E_{xz} + E_{wy} + E_{yw} \\ E_{ww} - E_{xx} + E_{yy} - E_{zz} & +E_{yz} + E_{zy} - E_{wx} - E_{xw} \\ +E_{yz} + E_{zy} + E_{wx} + E_{xw} & E_{ww} - E_{xx} - E_{yy} + E_{zz} \end{array} \right] . \tag{73}$$

B.3 The 4D Eigensystem

We now study the general algebraic solution to the eigenvalue problem for 4×4 real matrices M_4. In the special case where M_4 is symmetric and traceless, we will find that our results reduce to Eqs. (24) and (25).

We have in the back of our mind that we will be working with a linear algebra problem of the form

$$(p_0, p_1, p_2, p_3) \cdot M_4 \cdot (q_0, q_1, q_2, q_3)^t \equiv p \cdot M_4 \cdot q \ , \tag{74}$$

where there can be both left and right eigenvectors p and q for a single eigenvalue of the profile matrix M_4 if it is completely general, with a trace and no symmetry: q represents eigenvectors of M_4, and p represents eigenvectors of the transpose ${M_4}^t$. Warning: these facts are important, but it will turn out that the eigensystem of M_4 that we will now solve is *insufficient* by itself to solve the 4D RMSD optimization problem; nevertheless, the equations that we obtain *are* highly applicable to the actual solution.

For some types of calculations, we may find it useful to decompose M_4 in a way that isolates particular features using the form

$$M_4(w,x,y,z,\ldots) = \begin{bmatrix} w+x+y+z & a-a_w & b-b_w & c-c_w \\ a+a_w & w+x-y-z & C-C_w & B+B_w \\ b+b_w & C+C_w & w-x+y-z & A-A_w \\ c+c_w & B-B_w & A+A_w & w-x-y+z \end{bmatrix} \ , \tag{75}$$

where $\mathrm{tr}(M_4) = 4w$, the (x, y, z) combinations are manifestly traceless, the symmetric components are (a, b, c) and (A, B, C), and the antisymmetric components (a_w, b_w, c_w) and (A_w, B_w, C_w) of course disappear for the symmetric cases.

We next expand $\det[M_4 - eI_4] = 0$ (where I_4 denotes the 4D identity matrix, and transposing M_4 does not change the problem), along with the corresponding polynomial in the unknown eigenvalues ϵ_k, as a power series in e to obtain our familiar pair of fundamental equations and the result of eliminating e:

$$\left.\begin{array}{rcl} e^4 + e^3 p_1 + e^2 p_2 + e p_3 + p_4 &=& 0 \\ (e-\epsilon_1)(e-\epsilon_2)(e-\epsilon_3)(e-\epsilon_4) &=& 0 \end{array}\right\} \tag{76}$$

$$\left.\begin{array}{rcl} p_1(E) &=& (-\epsilon_1 - \epsilon_2 - \epsilon_3 - \epsilon_4) \\ p_2(E) &=& (\epsilon_1\epsilon_2 + \epsilon_1\epsilon_3 + \epsilon_2\epsilon_3 + \epsilon_1\epsilon_4 + \epsilon_2\epsilon_4 + \epsilon_3\epsilon_4) \\ p_3(E) &=& (-\epsilon_1\epsilon_2\epsilon_3 - \epsilon_1\epsilon_2\epsilon_4 - \epsilon_1\epsilon_3\epsilon_4 - \epsilon_2\epsilon_3\epsilon_4) \\ p_4(E) &=& \epsilon_1\epsilon_2\epsilon_3\epsilon_4 \end{array}\right\} \ . \tag{77}$$

The difference now is that the data coefficients p_k of degree k include a trace term $4w = -p_1$ as well as possible antisymmetric components throughout the new explicit form that follows from the expansion of

$\det[M_4 - eI_4]$:

$$
\begin{aligned}
p_1(E) &= -\operatorname{tr}[M_4] = -4w && (78)\\
p_2(E) &= \frac{1}{2}\left(\operatorname{tr}[M_4]\right)^2 - \frac{1}{2}\operatorname{tr}[M_4]\cdot[M_4] \\
&= 6w^2 - 2(x^2+y^2+z^2) - a^2 - A^2 - b^2 - B^2 - c^2 - C^2 \\
&\quad + {A_w}^2 + {a_w}^2 + {B_w}^2 + {b_w}^2 + {C_w}^2 + {c_w}^2 && (79)\\
p_3(E) &= -\frac{1}{6}\left(\operatorname{tr}[M_4]\right)^3 + \frac{1}{2}(\operatorname{tr}[M_4]\cdot[M_4])\operatorname{tr}[M_4] - \frac{1}{3}\operatorname{tr}[M_4]\cdot[M_4]\cdot[M_4] \\
&= -8xyz + 4w(x^2+y^2+z^2) \\
&\quad - 2ABC - 2Abc - 2aBc - 2abC \\
&\quad + 2A^2x - 2a^2x + 2B^2y - 2b^2y + 2C^2z - 2c^2z \\
&\quad - 2AB_wC_w + 2Ab_wc_w - 2aB_wc_w + 2ab_wC_w \\
&\qquad - 2A_wBC_w + 2a_wBc_w - 2a_wbC_w + 2A_wbc_w \\
&\qquad - 2A_wB_wC + 2a_wb_wC - 2A_wb_wc + 2a_wB_wc \\
&\quad + 2a^2w + 2A^2w - 2A_w^2w - 2A_w^2x - 2a_w^2w + 2a_w^2x \\
&\qquad + 2b^2w + 2B^2w - 2B_w^2w - 2B_w^2y - 2b_w^2w + 2b_w^2y \\
&\qquad + 2c^2w + 2C^2w - 2C_w^2w - 2C_w^2z - 2c_w^2w + 2c_w^2z && (80)\\
p_4(E) &= \det[M_4]\ . && (81)
\end{aligned}
$$

Using a computer algebra tool such as Mathematica to examine these sets of equations, we find that surprising simplifications similar to those found from Eq. (19) persist in 4D when we include the trace component in the Ansatz for the eigenvalues as follows:

$$
\left.
\begin{aligned}
\epsilon_1 &= -\tfrac{p_1}{4} + \sqrt{X} + \sqrt{Y} + \sqrt{Z} \\
\epsilon_2 &= -\tfrac{p_1}{4} + \sqrt{X} - \sqrt{Y} - \sqrt{Z} \\
\epsilon_3 &= -\tfrac{p_1}{4} - \sqrt{X} + \sqrt{Y} - \sqrt{Z} \\
\epsilon_4 &= -\tfrac{p_1}{4} - \sqrt{X} - \sqrt{Y} + \sqrt{Z}
\end{aligned}
\right\}\ . \qquad (82)
$$

From Eq. (77), we see that we can generalize Eq. (22) and write the p_k for the completely general 4D case as

$$
\begin{aligned}
p_1 &= p_1 && (83)\\
p_2 &= \frac{3{p_1}^2}{8} - 2\left(X+Y+Z\right) && (84)\\
p_3 &= \frac{{p_1}^3}{16} - 8\sqrt{X\,Y\,Z} - p_1(X+Y+Z) && (85)\\
p_4 &= \frac{{p_1}^4}{256} + X^2 + Y^2 + Z^2 - 2\left(YZ + ZX + XY\right) - p_1\sqrt{X\,Y\,Z} - \frac{{p_1}^2}{8}(X+Y+Z)\ . && (86)
\end{aligned}
$$

Exploring the structure of Eqs. (83), (84), (85), and (86) as before, we obtained explicit algebraic solutions for ϵ_k in terms of the $X(p)$, $Y(p)$, and $Z(p)$ in the following form:

$$
F_f(p) = \frac{{p_1}^2}{16} - \frac{p_2}{6} - \frac{1}{12}\left(\phi(f)\left(a(p) + \sqrt{-b(p)^2}\right)^{1/3} + \frac{n(p)}{\phi(f)\left(a(p) + \sqrt{-b(p)^2}\right)^{1/3}}\right) \qquad (87)
$$

where $F_f(p)$ with $f = (x, y, z)$ represents $X(p)$, $Y(p)$, or $Z(p)$ corresponding to one of the three values of the cube roots $\phi(f)$ of (-1) given by

$$
\phi(x) \;=\; -1\ , \qquad \phi(y) \;=\; \tfrac{1}{2}\left(1 + \mathrm{i}\sqrt{3}\right)\ , \qquad \phi(z) \;=\; \tfrac{1}{2}\left(1 - \mathrm{i}\sqrt{3}\right)\ . \qquad (88)
$$

The utility functions are defined as

$$\left.\begin{array}{rcl} a(p_1,p_2,p_3,p_4) & = & {p_2}^3 + \frac{1}{2}\left(27\left({p_3}^2 + {p_1}^2 p_4\right) - 9p_2(p_1 p_3 + 8p_4)\right) \\ n(p_1,p_2,p_3,p_4) & = & \sqrt[3]{a^2+b^2} \;=\; {p_2}^2 - 3p_1p_3 + 12p_4 \\ b(p_1,p_2,p_3,p_4)^2 & = & n(p)^3 - a(p)^2 \end{array}\right\} \; . \tag{89}$$

Remark: the traceful symmetric case is simple. There are several situations in which we encounter the intermediate case, where $\mathrm{tr}(M_4) \neq 0$ but M_4 is symmetric, so $w \neq 0$, while (a_w, b_w, c_w) and (A_w, B_w, C_w) all vanish. In this case, provided the p_k are evaluated with the internal inclusion of w, and Eq. (89) is used in place of Eq. (25) to include the value of p_1 explicitly throughout, one can express the eigenvalues in terms of the simpler formula Eq. (24) (with no complex phase issues) that we used in the main text by simply adding the ${p_1}^2/16$ term appearing in Eq. (87).

Discussion. From Eq. (87), we now know the solutions for the list of eigenvalues ϵ_k of M_4, and we can determine their corresponding left and right eigenvectors by solving $M_4 \cdot v_\lambda = \epsilon_k v_\lambda$ or ${M_4}^t \cdot v_\rho = \epsilon_k v_\rho$ as usual. Although it turns out that understanding the 4D RMSD solution requires further work, this is still a very interesting general result in its own right. Equations (82) typically consist of one real maximal eigenvalue, and three others that may be complex, so numerical computations based on Eq. (87) involve complex numbers and choices of signs and phases. Computing ϵ_k from Eq. (87)) using Mathematica and its default phase choices (which are the typically principal roots of the square roots and cube roots when the arguments are less than zero), we found that numerical comparisons of thousands of randomized 4D data sets agreed with all four eigenvalues found by numerical methods. We note that small imaginary numbers near the machine accuracy limit may appear that need to be dropped; a handful of the algebraic results differed from the numerical results by values of the order of 10^{-8}, somewhat larger than the expected machine accuracy limit. We suspect these are due to accidental numerical instabilities of our random matrices; deeper understanding of these rare anomalies is beyond our scope. Data sets in 3D are much better behaved, since symmetric real matrices have strictly real eigenvalues. We presented those 3D solutions involving traceless symmetric 4×4 profile matrices in the main text as a special simplified form of Eq. (87), observing that our algebraic expressions are manifestly real and highly suitable for efficient numerical evaluation, though in practice traditional numerical approximation methods might well be more efficient.

B.4 Issues with the Simplest 4D Approach

To summarize, we previously found that we can maximize $\Delta = \mathrm{tr}(R_3 \cdot E_3)$ over the 3D rotation matrices R_3 by mapping E_3 to the profile matrix M_3, with $\Delta_3 = q \cdot M_3 \cdot q$, solving for the maximal eigenvalue ϵ_{opt}, and choosing $R_{\mathrm{opt}} = R_3(q_{\mathrm{opt}})$ with q_{opt} the normalized quaternion eigenvector corresponding to $\Delta_3(\mathrm{opt}) = \epsilon_{\mathrm{opt}}$. The obvious 4D extension of the 3D quaternion RMSD problem would be to examine $\Delta_4 = \mathrm{tr}\left(R_4 \cdot E_4\right) = q_\lambda \cdot M_4 \cdot q_\rho$ over the 4D rotation matrices R_4, where M_4 turns out no longer to be symmetric, so we must split the eigenvector space into a separate left-quaternion q_λ and right-quaternion q_ρ. Just as in the 3D case, M_4 has a maximal eigenvalue ϵ_{opt}, and we find that the optimal eigenvectors $q_{\lambda:\mathrm{opt}}$ and $q_{\rho:\mathrm{opt}}$ are easily obtained as the corresponding eigenvectors of M_4 and ${M_4}^t$, so we would suppose that

$$\Delta_4(\mathrm{opt}) \stackrel{?}{=} q_{\lambda:\mathrm{opt}} \cdot M_4 \cdot q_{\rho:\mathrm{opt}} = \left(q_{\lambda:\mathrm{opt}} \cdot q_{\rho:\mathrm{opt}}\right)\epsilon_{\mathrm{opt}} \; . \tag{90}$$

Unfortunately, this is wrong. Not only is it typically much smaller than the actual maximum of $\mathrm{tr}(R_4(q_\lambda, q_\rho) \cdot E_4)$ over the space of 4D rotation matrices (or their equivalent representations in terms of q_λ and q_ρ), but even a simple *slerp* through q_{ID} and just beyond the apparent optimal eigenvectors $q_{\lambda:\mathrm{opt}}$ and $q_{\rho:\mathrm{opt}}$ can yield *larger* values of Δ_4! And, to add insult to injury, $q_{\lambda:\mathrm{opt}}$ and $q_{\rho:\mathrm{opt}}$ are in general not even a *basis* for some normalized linear combination that yields the optimal result. What is going wrong, and what is the path to our hoped-for algebraic solution to the 4D RMSD problem, which seems so close to the 3D RMSD problem, but then fails so spectacularly to correspond to the obvious hypothesis?

B.5 Connecting to the Singular Value Decomposition

The quaternion formulations of the 3D RMSD problem and the 4D RMSD problem differ, with 3D being a special case due to the symmetry of the 4×4 profile matrix. In 4D, a more general approach is required to express the optimization of the RMSD in terms of quaternions, and we now approach that in the context of the general singular-value-decomposition (SVD) method associated with Kabsch [3, 19, 20], whose approach minimizes a Fröbenius norm corresponding to our RMSD cross-term measure.

We start with a 4D cross-covariance matrix E_{ab} (noting that the SVD method holds for any dimension), where the SVD decomposition is written as

$$\begin{aligned} \{U,\, S,\, V\} &= \text{SingularValueDecomposition}\,(E_4) && (91)\\ &\text{where} && (92)\\ E_4(U,S,V) &= U \cdot S \cdot V^t && (93)\\ R_{4:\text{opt}}(U,D,V) &= V \cdot D \cdot U^t\ . && (94) \end{aligned}$$

These expressions are related to the decomposition of the symmetric matrices $F = E \cdot E^t$ and $F' = E^t \cdot E$, with the row-wise list $\mathbf{v}$ of eigenvectors of F and the corresponding eigenvectors $\mathbf{v}'$ of F' diagonalizing E_4 as

$$\mathbf{v} \cdot E_4 \cdot \mathbf{v}^t = \tilde{S}\ ,$$

where $\tilde{S}$ is the (signed) list of eigenvalues of E_4; this differs from S, which is the list of positive *square roots* of the eigenvectors of F or F', which is then the list of absolute values of E_4's eigenvalues. In effect, the SVD decomposition changes the signs of relevant rows of the eigenvectors $\mathbf{v}$ and $\mathbf{v}'$ that diagonalize E_4 to transform them to the almost-identical matrices U and V that make all the signs of $\tilde{S}$ positive, transforming it into $S = U^t \cdot E_4 \cdot V$, which is the (positive) list of the square roots of the eigenvalues of the symmetric matrices F or F' built from E_4 and its transpose.

The critical step for our procedure is to encode how this transforms into the quaternion language in 4D, where we know the following map from any data-derived matrix E_4 to the profile matrix $M_4(E_4)$ as given in Eq. (73):

$$\text{tr}(R_4(p,q) \cdot E_4) = p \cdot M_4(E_4) \cdot q\ .$$

We are not aware of a source for an existing general theorem, but it is easy to show symbolically in simple cases, and numerically in general cases, that the largest eigenvalue of $A = M_4 \cdot {M_4}^t$ is just the square of the *trace* of S, so we can write

$$\epsilon_{\text{opt}} = \text{tr}(S) = \sqrt{\text{max eigenvalue}\left(M_4 \cdot {M_4}^t\right)}\ .$$

But using the optimal value of the 4D rotation $R_{4:\text{opt}}(U,D,V) = V \cdot D \cdot U^t$ that is known to maximize the Fröbenius measure of Kabsch, we conclude that, in fact, after decomposing this rotation into its underlying quaternion pair,

$$\text{tr}\left(R_{4:\text{opt}} \cdot E_4\right) = \text{tr}(S) = p_{\text{opt}} \cdot M_4 \cdot q_{\text{opt}} = \epsilon_{\text{opt}}\ .$$

We will complete the argument now by showing how to find p_{opt} and q_{opt} independently using quaternion eigenvalue expressions without explicitly writing down the SVD.

Summary: Solving the 4D RMSD problem. Here now is a summary of how we can rephrase the SVD approach for maximizing the 4D RMSD measure in terms of quaternion methods:

- **Compute the profile matrix.** Using the quaternion decomposition Eq. (69) of the general 4D rotation matrix $R_4(p,q)$, extract the 4D profile matrix $M_4(E_4)$ of Eq. (73) from the initial proximity measure

  $$\Delta_4(\text{opt}) = \text{tr}(R_4(p_{\text{opt}}, q_{\text{opt}}) \cdot E_4) = p_{\text{opt}} \cdot M_4 \cdot q_{\text{opt}}\ . \quad (95)$$

  So far all we know is the numerical value of M_4.

- **Extract the optimal eigenvalue.** The value of $\Delta_4(\text{opt})$ is just the *square root* of the maximal eigenvalue of the 4×4 symmetric matrix $A = M_4 \cdot {M_4}^t$, which is itself easily obtained from Eq. (87), or the symmetric version Eq. (24) extended to handle the non-vanishing trace. If the relative optima for a series of data sets are all we need, we are done,

$$\Delta_4(\text{opt}) = \sqrt{\text{max eigenvalue } A} = \epsilon(\text{opt}) \ .$$

- **Use the eigenvalue to compute the eigenvectors p and q.** We have in fact two distinct symmetric matrices,

$$\begin{aligned} A &= M_4 \cdot {M_4}^t \\ A' &= {M_4}^t \cdot M_4 \end{aligned}$$

that have their own eigenvectors, both corresponding to the maximal eigenvalue $\epsilon(\text{opt})^2$ shared by A and A', so we can easily solve

$$\begin{aligned} \left(A - \epsilon(\text{opt})^2 I_4\right) \cdot p &= 0 \\ \left(A' - \epsilon(\text{opt})^2 I_4\right) \cdot q &= 0 \end{aligned}$$

for p_{opt} and q_{opt}, and this in turn yields the required 4D rotation matrix

$$R_{4:\text{opt}}\left(p_{\text{opt}}, q_{\text{opt}}\right)$$

from Eq. (69). This solves for the explicit rotation matrix in algebraic form, as well as its corresponding quaternion-pair, and also provides a cross-check from Eq. (95),

$$p_{\text{opt}} \cdot M_4 \cdot q_{\text{opt}} = \epsilon(\text{opt}) \ ,$$

which follows from the expansion of $\Delta_4(\text{opt})$. (Note that p_{opt} and q_{opt} are absolutely *not* eigenvectors of M_4, but of A and A^t, which define a distinct eigensystem; of course, since the eigenvectors of M_4 span the space, there is in principle a relationship. We already noted above that, in 4D, the maximal eigenvectors of M_4 itself yield values of Δ_4 that are typically significantly smaller than the maximum we have just computed.)

- **Remark on the solving for the SVD matrices.** It is perhaps significant to note that our results permit the algebraic calculation not just of a single eigenvalue and its associated eigenvector, but of the entire four-part eigensystem. Thus we could even choose to work only with the (nonsymmetric, traceful) 4D cross-covariance matrix E_{ab}, and find the pair of eigensystems corresponding to

$$\begin{aligned} F &= E_4 \cdot {E_4}^t \\ F' &= {E_4}^t \cdot E_4 \ . \end{aligned}$$

Computing the four shared (square root) eigenvalues ϵ_k for F and F', we can solve for the eight corresponding 4D eigenvectors from

$$\begin{aligned} \left(F - {\epsilon_k}^2 I_4\right) \cdot p_k &= 0 \\ \left(F' - {\epsilon_k}^2 I_4\right) \cdot q_k &= 0 \ . \end{aligned}$$

Up to signs, these are simply the vectors $\tilde{U}$ with columns $[p_1, p_2, p_3, p_4]$ and $\tilde{V}$ with columns $[q_1, q_2, q_3, q_4]$ diagonalizing the E_4 matrix

$$\tilde{U}^t \cdot E_4 \cdot \tilde{V} = \tilde{S} \ ,$$

where $\tilde{S}$ has the same elements as the SVD diagonal matrix S except that the elements of S are all positive. By simply changing the signs of the p_k and q_k eigenvectors required to make $\tilde{S} \equiv S$, we achieve the SVD matrices U and V, and thus from the closed-form algebraic solutions for ϵ_k, we can compute a fairly compact algebraic form for the general 4D SVD solution. That information provides an equivalent solution to the 4D RMSD problem via the Kabsch algorithm.

B.6 Remark on 4D Frames

Orientation frames in four dimensions have axes that are the columns of a 4D rotation matrix taking the identity frame to the new orientation frame. Therefore, in parallel with the 3D case, such frames can be represented either as 4D rotation matrices (the action on a 4D identity frame to get a new set of 4 orthogonal axes), or as the pair of quaternions (p, q) used in Eq. (69) to define $R_4(p, q)$. As in the 3D frame case, we will take advantage of the chord-distance linearization of the gold-standard angular measure, and we shall present two alternative approaches to the optimization measure.

Quadratic Form. In 3D, with Eqs. (36) and (37) having a single quaternion involved in the rotation, we were able to write down $\Delta_{\text{chord}} = q \cdot V(W)$ in terms of a simple expression linear in the quaternion q and the cumulative data V, and we observed that a quadratic expression $(q \cdot V)^2$ would also produce the same optimal value $q = V/\|V\|$. The optimal frame problem in 4D, in contrast, already requires a pair of quaternions, and one strategy is to split the analogs of the 3D quadratic expression into two parts, yielding

$$\Delta_{\text{4:chord}} = (q \cdot V)\,(q' \cdot V') = q_a\,(V_a V'_b)\,q'_b = q \cdot \Omega \cdot q' \tag{96}$$

as the generalization from 3D to 4D. Here V is built from $W_{ab} = \sum_{k=1}^{N} {p_k}^a {r_k}^b$, and V' is built from $W'_{ab} = \sum_{k=1}^{N} {p'_k}^a {r'_k}^b$ according to Eq. (37), where each 4D test frame consists of frames denoted by the quaternion pair (p, p'), and each reference frame employs a pair (r, r'). Modulo the usual issue with consistent quaternion signs, now for the 4D frame pairs, the solution for the optimal quaternions must achieve the maximum for *both* elements of the pair, and so we obtain as a solution maximizing Eq. (96)

$$\left.\begin{aligned} q_{\text{opt}} &= \frac{V}{\|V\|} \\ q'_{\text{opt}} &= \frac{V'}{\|V'\|} \\ \Delta_{\text{4:chord}}(\text{opt}) &= \|V\|\|V'\| \end{aligned}\right\} . \tag{97}$$

Remark: Various other polynomials in $(q \cdot V)$ and $(q' \cdot V')$ will produce the same pair of solutions. There is a particular reason to prefer Eq. (96): in the next section, we will see that the *pre-summation* arguments for V and V', gathered together, are *exactly* equal to the 4D triple rotation pre-summation arguments, following the pattern seen in Eq. (47) for the 3D orientation-frame analysis.

Quartic Triple Rotation Form. One can also define a 4D frame similarity measure that is the exact analog of Eq. (46) in 3D as follows:

$$\Delta_{\text{RRR4}} = \sum_{k=1}^{N} \text{tr}\left[R_4(q, q') \cdot R_4(p_k, {p'}_k) \cdot R^{-1}(r_k, {r'}_k)\right] \tag{98}$$

$$= \sum_{k=1}^{N} \text{tr}\left[R(q, q') \cdot R(p_k \star \bar{r}_k,\ {p'}_k \star \bar{r}'_k)\right] \tag{99}$$

$$= \sum_{k=1}^{N} \text{tr}\left[R(q \star p_k \star \bar{r}_k,\ q' \star {p'}_k \star \bar{r}'_k)\right] \tag{100}$$

$$= q \cdot U_4(p, p';\, \bar{r}, \bar{r}') \cdot q' \ . \tag{101}$$

Remarkably, there is a 4D version of the 3D identity Eq. (47) relating the triple rotation measure to the quadratic realizations of the linear quaternion rotation measures, namely

$$\sum_{k=1}^{N} \text{tr}\left[R(q, q') \cdot R(p_k, p'_k) \cdot R(\bar{r}_k, \bar{r}'_k)\right] = 4 \sum_{k=1}^{N} \left((q \star p_k) \cdot r_k\right)\left((q' \star p'_k) \cdot r'_k\right) \ . \tag{102}$$

Thus the pre-summation version of the arguments in the $(q \cdot V)(q \cdot V')$ version of the 4D chord measure turns out to be *exactly* the same as the triple-matrix product measure without the additional trace term that is present in 3D. We now use Eq. (69) for $R(q, q')$ to decompose the measure Eq. (99) into the form

$$\begin{aligned} \Delta_{\rm RRR4} &= \mathrm{tr}\,[R(q,q') \cdot S(p,p';r,r')] & (103) \\ &= q \cdot U(S) \cdot q' \ , & (104) \end{aligned}$$

where $S(p,p';r,r') = \sum_{k=1}^{N} R(p_k \star \bar{r}_k,\ {p'}_k \star \bar{r}'_k)$ and $U(S)$ has the same relationship to S as the 4D profile matrix $M(E)$ in Eq. (73) does to the cross-correlation matrix E. In the next section, Appendix C, we will see that the singleton version of this map is unusually degenerate, with rank one, though that feature does not persist for data sets with $N > 1$.

Now, as in the 4D spatial RMSD analysis, we might naturally assume that we could follow the 3D case by determining the maximal eigenvalue ϵ_0 of U and its left and right eigenvectors q_λ and q_ρ, which would give

$$\Delta_{\rm RRR4} = q_\lambda \cdot U \cdot q_\rho = (q_\lambda \cdot q_\rho)\, \epsilon_0 \ .$$

As before, this is not a maximal value for the measure $\Delta_{\rm RRR4}$ over the possible range of $R(q, q')$. To solve the optimization correctly, we must again be very careful, and work with the maximal eigenvalue of $A = U \cdot U^t$ and $A' = U^t \cdot U$, which we can get from our closed form solution for symmetric 4×4 matrices with a trace, yielding

$$\Delta_{\rm RRR4}(\mathrm{opt}) = \sqrt{\text{max eigenvalue } U \cdot U^t} = \epsilon(\mathrm{RRR4{:}opt}) \ .$$

If we need the actual optimal rotation matrix solving

$$\Delta_{\rm RRR4}(\mathrm{opt}) = \mathrm{tr}\left(R_{4:\mathrm{opt}}(p_{\rm opt}, q_{\rm opt}) \cdot S\right) = p_{\rm opt} \cdot U \cdot q_{\rm opt} = \epsilon(\mathrm{RRR4{:}opt}) \ ,$$

then we just use our optimal eigenvalue to solve

$$\begin{aligned} \left(A - \epsilon(\mathrm{RRR4{:}opt})^2 I_4\right) \cdot p &= 0 \\ \left(A' - \epsilon(\mathrm{RRR4{:}opt})^2 I_4\right) \cdot q &= 0 \end{aligned}$$

for $p_{\rm opt}$ and $q_{\rm opt}$, and that gives the desired 4D rotation matrix explicitly via Eq. (69).

C On Obtaining Quaternions from Rotation Matrices

The quaternion RMSD profile matrix method can be used to implement a singularity-free algorithm to obtain the (sign-ambiguous) quaternions corresponding to numerical 3D and 4D rotation matrices. There are many existing approaches to the 3D problem in the literature (see, e.g., [38], [39], or Section 16.1 of [32]). In contrast to these approaches, Bar-Itzhack [31] has observed, in essence, that if we simply replace the data matrix E_{ab} by a numerical 3D orthogonal rotation matrix R, the optimization problem becomes one of finding the quaternion q that corresponds to the rotation of Eq. (3) that is the inverse of R. That quaternion can then be shown to correspond to the targeted numerical rotation matrix, solving the problem. To see this, we replace the elements E_{ab} in Eq. (10) by a general orthonormal rotation matrix with columns $\mathbf{X} = (x_1, x_2, x_3)$, $\mathbf{Y}$, and $\mathbf{Z}$, scaling by $1/3$, thus obtaining the special 4×4 profile matrix K whose elements in terms of a known numerical matrix $R = [\mathbf{X}|\mathbf{Y}|\mathbf{Z}]$ are

$$K(R) = \frac{1}{3} \begin{bmatrix} x_1 + y_2 + z_3 & z_2 - y_3 & x_3 - z_1 & y_1 - x_2 \\ z_2 - y_3 & x_1 - y_2 - z_3 & x_2 + y_1 & x_3 + z_1 \\ x_3 - z_1 & x_2 + y_1 & -x_1 + y_2 - z_3 & y_3 + z_2 \\ y_1 - x_2 & x_3 + z_1 & y_3 + z_2 & -x_1 - y_2 + z_3 \end{bmatrix} \ . \qquad (105)$$

However, as we know, any orthogonal 3D rotation matrix R can also be expressed as in terms of quaternions via Eq. (3), and this yields an alternate useful algebraic form

$$K(q) = \frac{1}{3}\begin{bmatrix} 3{q_0}^2 - {q_1}^2 - {q_2}^2 - {q_3}^2 & -4q_0q_1 & -4q_0q_2 & -4q_0q_3 \\ -4q_0q_1 & -{q_0}^2 + 3{q_1}^2 - {q_2}^2 - {q_3}^2 & 4q_1q_2 & 4q_1q_3 \\ -4q_0q_2 & 4q_1q_2 & -{q_0}^2 - {q_1}^2 + 3{q_2}^2 - {q_3}^2 & 4q_2q_3 \\ -4q_0q_3 & 4q_1q_3 & 4q_2q_3 & -{q_0}^2 - {q_1}^2 - {q_2}^2 + 3{q_3}^2 \end{bmatrix} . \tag{106}$$

This equation then allows us to quickly prove that K has the correct properties to solve for the appropriate quaternion corresponding to R. First we note that the coefficients p_n of the eigensystem are simply constants,

$$p_1 = 0 \qquad p_2 = -\tfrac{2}{3} \qquad p_3 = -\tfrac{8}{27} \qquad p_4 = \tfrac{1}{27} \quad .$$

Computing the eigenvalues and eigenvectors now using the symbolic quaternion form, we see that the eigenvalues are constant, with maximal eigenvalue exactly one, and the eigenvectors are almost trivial, with the maximal eigenvector being the inverse of the quaternion q that corresponds to the (numerical) rotation matrix:

$$\epsilon = \{1, -\frac{1}{3}, -\frac{1}{3}, -\frac{1}{3}\} \tag{107}$$

$$v = \left\{ \begin{bmatrix} q_0 \\ -q_1 \\ -q_2 \\ -q_3 \end{bmatrix}, \begin{bmatrix} q_1 \\ q_0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} q_2 \\ 0 \\ q_0 \\ 0 \end{bmatrix}, \begin{bmatrix} q_3 \\ 0 \\ 0 \\ q_0 \end{bmatrix} \right\} . \tag{108}$$

Unfortunately, the remainder of our techniques are irrelevant, because the maximal eigenvalue is always known in advance to be unity for any valid rotation matrix; however, as noted by Bar-Itzhack, if there are *errors* in the matrix, then one can use the eigenvalue to determine the *closest* quaternion to an errorful rotation matrix, which can be very useful since the quaternion always produces a valid rotation matrix. It is possible that there are unexploited special properties of the fact that the K matrix is based on an orthonormal matrix, very different from our usual RMSD problem. In any case, the best that one can do seems to be to insert the desired numerical rotation matrix R into Eq. (105) to compute $K = \frac{1}{3}M(R)$, and solve the equations $[K.v = v]$ with unit eigenvalue for $(1, v_1, v_2, v_3)$ (or context-appropriate variants) as usual to get the numerical maximal eigenvector, normalize, and take its inverse by changing the signs of the vector part to obtain the desired quaternion q (up to an overall sign) corresponding to the target rotation matrix. (In some circumstances, one is looking for a uniform statistical distribution of quaternions, in which case the overall sign of q should be chosen randomly.) This solves the problem of extracting the quaternion of an arbitrary 3D rotation matrix in a fashion that involves no singularities and only trivial testing for special cases, thus essentially making the traditional methods obsolete.

Extracting Quaternions from 4D Rotation Matrices. For completeness, we briefly outline the 4D case, starting with Eq. (69), which is the 4D generalization of the 3D quadratic form Eq. (3). First we extend Eq. (105) to correspond to a general 4D orthonormal rotation matrix with columns $\mathbf{W} = (w_0, w_1, w_2, w_3)$, etc., so the matrix takes the form $R_4 = [\mathbf{W}|\mathbf{X}|\mathbf{Y}|\mathbf{Z}]$, producing a numerical data matrix of the form

$$K(R_4) = \frac{1}{4}\begin{bmatrix} w_0 + x_1 + y_2 + z_3 & w_1 - x_0 - y_3 + z_2 & w_2 + x_3 - y_0 - z_1 & w_3 - x_2 + y_1 - z_0 \\ -w_1 + x_0 - y_3 + z_2 & w_0 + x_1 - y_2 - z_3 & -w_3 + x_2 + y_1 - z_0 & w_2 + x_3 + y_0 + z_1 \\ -w_2 + x_3 + y_0 - z_1 & w_3 + x_2 + y_1 + z_0 & w_0 - x_1 + y_2 - z_3 & -w_1 - x_0 + y_3 + z_2 \\ -w_3 - x_2 + y_1 + z_0 & -w_2 + x_3 - y_0 + z_1 & w_1 + x_0 + y_3 + z_2 & w_0 - x_1 - y_2 + z_3 \end{bmatrix} . \tag{109}$$

Now, from Eq. (69), we know that we also have an analog to Eq. (106), and, unexpectedly, for $R_4(p, p')$ this takes the remarkably simple alternate form

$$K(p, p') = \begin{bmatrix} p_0p'_0 & -p_0p'_1 & -p_0p'_2 & -p_0p'_3 \\ -p_1p'_0 & p_1p'_1 & p_1p'_2 & p_1p'_3 \\ -p_2p'_0 & p_2p'_1 & p_2p'_2 & p_2p'_3 \\ -p_3p'_0 & p_3p'_1 & p_3p'_2 & p_3p'_3 \end{bmatrix} . \tag{110}$$

This matrix is exactly the outer product of of $\bar{p}$ and $\bar{p}'$, and it has vanishing determinant, rank 1, and trace $p \cdot p'$, which makes it extremely simple. The eigensystem is just

$$\epsilon = \{p \cdot p',\ 0,\ 0,\ 0\} \tag{111}$$

$$v_{\text{right}} = \left\{ \begin{bmatrix} p_0 \\ -p_1 \\ -p_2 \\ -p_3 \end{bmatrix}, \begin{bmatrix} p'_1 \\ p'_0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} p'_2 \\ 0 \\ p'_0 \\ 0 \end{bmatrix}, \begin{bmatrix} p'_3 \\ 0 \\ 0 \\ p'_0 \end{bmatrix} \right\} \tag{112}$$

$$v_{\text{left}} = \left\{ \begin{bmatrix} p'_0 \\ -p'_1 \\ -p'_2 \\ -p'_3 \end{bmatrix}, \begin{bmatrix} p_1 \\ p_0 \\ 0 \\ 0 \end{bmatrix}, \begin{bmatrix} p_2 \\ 0 \\ p_0 \\ 0 \end{bmatrix}, \begin{bmatrix} p_3 \\ 0 \\ 0 \\ p_0 \end{bmatrix} \right\}, \tag{113}$$

so in fact the sole non-vanishing eigenvalue is just $\epsilon = \operatorname{tr} K(p, p')$. Thus the left and right eigenvectors can be easily computed numerically from Eq. (109) using the eigenvalue extracted from the trace, remembering to conjugate them to get correspondence with $R_4(p, p')$. Again, if a statistical distribution in the double quaternion space is desired, the signs can be chosen randomly, consistent with the sign of $\operatorname{tr} K(p, p') = p \cdot p'$.

References

1. Wikipedia. Orthogonal Procrustes problem — Wikipedia, The Free Encyclopedia; 2018. http://en.wikipedia.org/w/index.php?title=Orthogonal%20Procrustes%20problem&oldid=850916143.
2. Wikipedia. Wahba's problem — Wikipedia, The Free Encyclopedia; 2018. http://en.wikipedia.org/w/index.php?title=Wahba's%20problem&oldid=854401910.
3. Wikipedia. Kabsch algorithm — Wikipedia, The Free Encyclopedia; 2018. http://en.wikipedia.org/w/index.php?title=Kabsch%20algorithm&oldid=838166818.
4. Green WA. The orthogonal approximation of an oblique structure in factor analysis. Psychometrika. 1952;17:429–440.
5. Horn BKP, Hilden HM, Negahdaripour S. Closed-form solution of absolute orientation using orthonormal matrices. J Opt Soc Am A. 1988;5(7):1127–1136.
6. Schönemann PH. A generalized solution of the orthogonal Procrustes problem. Psychometrika. 1966;31:1– 10. doi:https://doi:10.1007/BF02289451.
7. Markley FL. Attitude Determination using Vector observations and the Singular Value Decomposition. Journal of the Astronautical Sciences. 1988;38(2):245–258.
8. Davenport PB. A Vector Approach to the Algebra of Rotations with Applications. Greenbelt, Maryland: NASA: Goddard Space Flight Center; 1968. TN D-4696.
9. Horn BKP. Closed-form solution of absolute orientation using unit quaternions. J Opt Soc Am A. 1987;4:629–642.
10. Diamond R. A note on the rotational superposition problem. Acta Crystallogr. 1988;A44:211–216.
11. Kearsley SK. On the orthogonal transformation used for structural comparisons. Acta Crystallogr. 1989;A45(2):208–210. doi:10.1107/S0108767388010128.
12. Kneller GR. Superposition of Molecular Structures Using Quaternions. Molecular Simulation. 1991;7(1–2):113–119.
13. Wahba G. Problem 65-1, A Least Squares Estimate of Spacecraft Attitude. SIAM Review. 1965;7(3):409.
14. Markley FL, Mortari D. Quaternion Attitude Estimation Using Vector Observations. Journal of the Astronautical Sciences. 2000;48(2):359–380.
15. Huang TS, Blostein SD, Margerum EA. Least-squares estimation of motion parameters from 3D point correspondences. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition. IEEE Computer Society; 1986. p. 24–26.
16. Arun KS, Huang TS, Blostein SD. Least-squares fitting of two 3D point sets. IEEE Trans Pattern Anal Machine Intell. 1987;PAMI-9(5):698–700. doi:10.1109/TPAMI.1987.4767965.
17. Umeyama S. Least-squares estimation of transformation parameters between two point patterns. IEEE Trans Pattern Anal Machine Intell. 1991;13(4):376–380. doi:10.1109/34.88573.
18. Zhang Z. A Flexible New Technique for Camera Calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2000;22:1330–1334.
19. Kabsch W. A solution for the best rotation to relate two sets of vectors. Acta Crystallogr. 1976;A32:922–923.

20. Kabsch W. A discussion of the solution for the best rotation to relate two sets of vectors. Acta Crystallogr. 1978;A34:827–828.

21. MacLachlan AD. Rapid comparison of protein structures. Acta Crystallogr. 1982;A38:871—873. doi:https://doi.org/10.1107/S0567739482001806.

22. Kearsley SK. An algorithm for the simultaneous superposition of a structural series. J Comput Chem. 1990;11:1187–1192.

23. Coutsias EA, Seok C, Dill KA. Using Quaternions to Calculate RMSD. J Comput Chem. 2004;25(15):1849–1857.

24. Theobald D. Rapid calculation of RMSDs using a quaternion-based characteristic polynomial. Acta Crystallogr. 2005;A61:478–480.

25. Hanson AJ, Thakur S. Quaternion maps of global protein structure. Jour Molec Graphics and Modelling. 2012;38:256–278.

26. Park FC, Ravani B. Smooth Invariant Interpolation of Rotations. ACM Trans Graph. 1997;16(3):277–295. doi:10.1145/256157.256160.

27. Huynh DQ. Metrics for 3D Rotations: Comparison and Analysis. J Math Imaging Vis. 2009;35(2):155–164. doi:10.1007/s10851-009-0161-2.

28. Huggins DJ. Comparing Distance Metrics for Rotation Using the k-Nearest Neighbors Algorithm for Entropy Estimation. J Comput Chem. 2014;35:377–385. doi:DOI: 10.1002/jcc.23504.

29. Huggins DJ. Estimating Translational and Orientational Entropies Using the k-Nearest Neighbors Algorithm. J Chem Theory Comput. 2014;10:3617–3625.

30. Fogolari F, Dongmo Foumthuim CJ, Fortuna S, Soler MA, Corazza A, Esposito G. Accurate Estimation of the Entropy of Rotation–Translation Probability Distributions. Journal of Chemical Theory and Computation. 2016;12(1):1–8. doi:10.1021/acs.jctc.5b00731.

31. Bar-Itzhack IY. New Method for Extracting the Quaternion from a Rotation Matrix. Journal of Guidance, Control, and Dynamics. 2000;23(6):1085–1087. doi:https://doi.org/10.2514/2.4654.

32. Hanson AJ. Visualizing Quaternions. Morgan-Kaufmann/Elsevier; 2006.

33. Altmann SL. Rotations, Quaternions, and Double Groups. Oxford University Press; 1986.

34. Altmann SL. Hamilton, Rodrigues, and the Quaternion Scandal. Mathematics Magazine. 1989;62(5):291–308. doi:10.2307/2689481.

35. Shoemake K. Animating Rotation with Quaternion Curves. In: Computer Graphics. vol. 19; 1985. p. 245–254.

36. Brown JL, Worsey AJ. Problems with Defining barycentric coordinates for the sphere. Mathematical Modelling and Numerical analysis. 1992;26:37–49.

37. Buss SR, Fillmore JP. Spherical averages and applications to spherical splines and interpolation. ACM Transactions on Graphics (TOG). 2001;20(2):95–126. doi:http://doi.acm.org/10.1145/502122.502124.

38. Shepperd SW. Quaternion from Rotation Matrix. Journal of Guidance and Control. 1978;1(3):223–224.

39. Shuster MD, Natanson GA. Quaternion computation from a geometric point of view. The Journal of the Astronautical Sciences. 1993;41(4):545–556.