Accurate Computation of
Quaternions from Rotation Matrices

Soheil Sarabandi and Federico Thomas

Institut de Robòtica i Informàtica Industrial (CSIC-UPC)
Llorens Artigas 4-6, 08028 Barcelona, Spain
soheil.sarabandi@gmail.com, fthomas@iri.upc.edu

Abstract. The main non-singular alternative to 3×3 proper orthogonal matrices, for representing rotations in $\mathbb{R}^3$, is quaternions. Thus, it is important to have reliable methods to pass from one representation to the other. While passing from a quaternion to the corresponding rotation matrix is given by Euler-Rodrigues formula, the other way round can be performed in many different ways. Although all of them are algebraically equivalent, their numerical behavior can be quite different. In 1978, Shepperd proposed a method for computing the quaternion corresponding to a rotation matrix which is considered the most reliable method to date. Shepperd's method, thanks to a voting scheme between four possible solutions, always works far from formulation singularities. In this paper, we propose a new method which outperforms Shepperd's method without increasing the computational cost.

Keywords: Quaternions, Rotation matrices

1   Introduction

Arbitrary rotations in $\mathbb{R}^3$ can be represented using proper orthogonal 3×3 matrices (*rotation matrices* for short) of the form:

$$\mathbf{R} = \begin{pmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{pmatrix}. \tag{1}$$

Nevertheless, according to Euler rotation theorem, every rotation in three dimensions is defined by its axis, given by a unit vector $\mathbf{n} = (n_x, n_y, n_z)$, and its angle, the amount of rotation about that axis, given by θ. Clearly, $\mathbf{n}$ and θ provide a much more compact and meaningful information about a rotation than the 9 entries of a rotation matrix.

The information about the axis and angle of rotation is usually organized as a quaternion so that the product of two quaternions gives us the quaternion corresponding to the composition of the rotations that the two matrices represent, in the same way as the product of two rotation matrices gives us the matrix

representing the rotation corresponding to the composition of the two rotations the matrices represent. The four elements of the quaternion are given by

$$q_1 = \cos\frac{\theta}{2}, \qquad q_2 = n_x \sin\frac{\theta}{2}, \qquad q_3 = n_y \sin\frac{\theta}{2}, \qquad q_4 = n_z \sin\frac{\theta}{2}, \qquad (2)$$

which are commonly known as Euler parameters. These quantities are not independent, they are related through the following equation:

$$q_1^2 + q_2^2 + q_3^2 + q_4^2 = 1. \qquad (3)$$

As a consequence, when operating with quaternions, we need to be sure that (3) is always satisfied.

Because both, quaternions and rotation matrices, are useful for certain kinematics calculations, the need arises to convert between these representations. It is well known [1] that a $3{\times}3$ proper orthogonal rotation matrix can be expressed in terms of the quaternion components as:

$$\mathbf{R} = \begin{pmatrix} q_1^2 + q_2^2 - q_3^2 - q_4^2 & 2(q_2q_3 - q_1q_4) & 2(q_2q_4 + q_1q_3) \\ 2(q_2q_3 + q_1q_4) & q_1^2 - q_2^2 + q_3^2 - q_4^2 & 2(q_2q_4 - q_1q_2) \\ 2(q_2q_4 - q_1q_3) & 2(q_3q_4 + q_1q_2) & q_1^2 - q_2^2 - q_3^2 + q_4^2 \end{pmatrix}. \qquad (4)$$

In contrast, there are many methods to obtain the quaternion corresponding to a given rotation matrix. They include algebraic, trigonometric, and numerical methods (see [2] for a recent survey). If we limit our analysis to algebraic methods, that is, to those methods consisting in solving the system of nonlinear equations resulting from equating the matrices in (1) and (4), Shepperd's algorithm is considered as the best one from the numerical point of view. This paper is devoted to the derivation of an alternative method which is shown to be numerically better conditioned than Shepperd's method.

By observing (4), it is easy to conclude that (q_1, q_2, q_3, q_4) and $(-q_1, -q_2, -q_3, -q_4)$ lead to the same rotation matrix. In mathematical terms, it is said that quaternions provide a double covering of the space of rotations. Since this is a 2-to-1 map, it cannot be smoothly inverted. This has important consequences in all methods that compute the quaternion corresponding to a rotation matrix: they all have to deal with singularities, and they all give the same solution within an undetermined but physically unimportant overall sign.

This paper is organized as follows. In the next section, we briefly summarize Shepperd's method. In Section 3 we present an alternative method which, in Section 4, is shown to be numerically more accurate than Shepperd's method. Finally, Section 5 summarizes the main points.

2   Shepperds's method

Since it was first proposed in [3], Shepperd's method remains as one of the most popular methods for computing the quaternion corresponding to a rotation

matrix. It improves on Hughes' method [5] via a voting scheme in which the quaternion is computed without numerical instabilities.

It is worth to mention that, although the method is associated with Shepperd, its origins can be traced back at least to a book by E. Study published in 1910 (see [4] for details).

The equations resulting from equating the matrices in (1) and (4) can be expressed as:

$$4q_1^2 = 1 + r_{11} + r_{22} + r_{33} \tag{5}$$

$$4q_2^2 = 1 + r_{11} - r_{22} - r_{33} \tag{6}$$

$$4q_3^2 = 1 - r_{11} + r_{22} - r_{33} \tag{7}$$

$$4q_3^2 = 1 - r_{11} - r_{22} + r_{33} \tag{8}$$

$$4q_3 q_4 = r_{23} + r_{32} \tag{9}$$

$$4q_2 q_4 = r_{31} + r_{13} \tag{10}$$

$$4q_2 q_3 = r_{12} + r_{21} \tag{11}$$

$$4q_1 q_2 = r_{32} - r_{23} \tag{12}$$

$$4q_1 q_3 = r_{13} - r_{31} \tag{13}$$

$$4q_1 q_4 = r_{21} - r_{12} \tag{14}$$

In Hughes' method, q_1 is calculated first and then it is treated very differently from the remaining three parameters. Since we can solve the system of equations (5)-(14) for any of the four Euler parameters first, there are four different formulas for computing the quaternion as a function of the entries of the rotation matrix, all of them formally equivalent. Numerically, however, these four formulas are not identical and, depending on the rotation matrix, one of them is numerically better conditioned than the others. From the system of equations (5)-(14), we arrive at these four different solutions:

$$\mathbf{u}_1 = \frac{1}{2} \begin{pmatrix} (1+r_{11}+r_{22}+r_{33})^{\frac{1}{2}} \\ (r_{32}-r_{23})/(1+r_{11}+r_{22}+r_{33})^{\frac{1}{2}} \\ (r_{13}-r_{31})/(1+r_{11}+r_{22}+r_{33})^{\frac{1}{2}} \\ (r_{21}-r_{12})/(1+r_{11}+r_{22}+r_{33})^{\frac{1}{2}} \end{pmatrix}, \tag{15}$$

$$\mathbf{u}_2 = \frac{1}{2} \begin{pmatrix} (r_{32}-r_{23})/(1+r_{11}-r_{22}-r_{33})^{\frac{1}{2}} \\ (1+r_{11}-r_{22}-r_{33})^{\frac{1}{2}} \\ (r_{12}+r_{21})/(1+r_{11}-r_{22}-r_{33})^{\frac{1}{2}} \\ (r_{31}+r_{13})/(1+r_{11}-r_{22}-r_{33})^{\frac{1}{2}} \end{pmatrix}, \tag{16}$$

$$\mathbf{u}_3 = \frac{1}{2} \begin{pmatrix} (r_{13}-r_{31})/(1-r_{11}+r_{22}-r_{33})^{\frac{1}{2}} \\ (r_{12}+r_{21})/(1-r_{11}+r_{22}-r_{33})^{\frac{1}{2}} \\ (1-r_{11}+r_{22}-r_{33})^{\frac{1}{2}} \\ (r_{23}+r_{32})/(1-r_{11}+r_{22}-r_{33})^{\frac{1}{2}} \end{pmatrix}, \tag{17}$$

$$\mathbf{u}_4 = \frac{1}{2} \begin{pmatrix} (r_{21}-r_{12})/(1-r_{11}-r_{22}+r_{33})^{\frac{1}{2}} \\ (r_{31}+r_{13})/(1-r_{11}-r_{22}+r_{33})^{\frac{1}{2}} \\ (r_{32}+r_{23})/(1-r_{11}-r_{22}+r_{33})^{\frac{1}{2}} \\ (1-r_{11}-r_{22}+r_{33})^{\frac{1}{2}} \end{pmatrix}. \tag{18}$$

When computing any of the above solutions, numerical issues arise when square rooting, or when dividing by, very small numbers [6]. To obtain the better conditioned solution for each case, the ordinal number i of the largest element in the following vector is determined:

$$\begin{pmatrix} r_{11}+r_{22}+r_{33} \\ r_{11} \\ r_{22} \\ r_{33} \end{pmatrix}. \tag{19}$$

Then, the best solution, from the numerical point of view, is considered to be $\mathbf{u}_i$.

This four-fold multiplicity of the solution arises in other methods. For example, the one presented in [8], based on geometric arguments, was shown to be equivalent to this method.

3   Our method

Let us suppose we only want to compute q_1. Then, from (5), we have that

$$q_1 = \frac{1}{2}\sqrt{1 + r_{11} + r_{22} + r_{33}}. \tag{20}$$

The term inside the above square root lies in the interval $[0,4]$. Indeed, observe that $\text{Trace}(\mathbf{R}) = r_{11} + r_{22} + r_{33} = 2\cos\theta + 1$ [9, Section 2.3]. Unfortunately, numerical problems arise when this term gets close to zero. In practice, it can even become negative due to rounding errors [2]. Since this term coincides with $2 + 2\cos\theta$, it becomes ill-conditioned when $\theta \to \pi$. Observe that (20) only takes into account the diagonal entries of $\mathbf{R}$. To obtain an alternative formula involving all the elements of the rotation matrix, let us substitute in (3) the values of q_1^2, q_2^2, q_3^2, and q_4^2 obtained from (5), (12), (13), and (14), respectively. The result is:

$$\frac{1+r_{11}+r_{22}+r_{33}}{4} + \left(\frac{r_{32}-r_{23}}{4q_1}\right)^2 + \left(\frac{r_{13}-r_{31}}{4q_1}\right)^2 + \left(\frac{r_{21}-r_{12}}{4q_1}\right)^2 = 1. \tag{21}$$

Solving the above equation for q_1, we obtain

$$q_1 = \frac{1}{2}\sqrt{\frac{(r_{32}-r_{23})^2+(r_{13}-r_{31})^2+(r_{21}-r_{12})^2}{3-r_{11}-r_{22}-r_{33}}}. \tag{22}$$

Now, the term in the denominator of (22) also lies in the interval $[0,4]$. Since this denominator coincides with $2 - 2\cos\theta$, (22) is ill-conditioned for $\theta \to 0$.

When this happens, the diagonal of $\mathbf{R}$ is dominant and, as a consequence, the numerator in (22) tends also to be small. Thus, (20) and (22) can be seen as complementary. Therefore, it is reasonable to establish a threshold for the trace of $\mathbf{R}$ —to be determined for optimal results— above which it is preferable to use (22) instead of (20). In other words, we have that

$$
q_1 = \begin{cases} \frac{1}{2}\sqrt{1+r_{11}+r_{22}+r_{33}}, & \text{if } r_{11}+r_{22}+r_{33} > \eta \\[2ex] \frac{1}{2}\sqrt{\dfrac{(r_{32}-r_{23})^2+(r_{13}-r_{31})^2+(r_{21}-r_{12})^2}{3-r_{11}-r_{22}-r_{33}}}, & \text{otherwise} \end{cases}
\tag{23}
$$

where η has to be determined. Extending this reasoning to the computation of the other elements of the quaternion, the result is:

$$
q_2 = \begin{cases} \frac{1}{2}\sqrt{1+r_{11}-r_{22}-r_{33}}, & \text{if } r_{11}-r_{22}-r_{33} > \eta \\[2ex] \frac{1}{2}\sqrt{\dfrac{(r_{32}-r_{23})^2+(r_{12}+r_{21})^2+(r_{31}+r_{13})^2}{3-r_{11}+r_{22}+r_{33}}}, & \text{otherwise} \end{cases}
\tag{24}
$$

$$
q_3 = \begin{cases} \frac{1}{2}\sqrt{1-r_{11}+r_{22}-r_{33}}, & \text{if } -r_{11}+r_{22}-r_{33} > \eta \\[2ex] \frac{1}{2}\sqrt{\dfrac{(r_{13}-r_{31})^2+(r_{12}+r_{21})^2+(r_{23}+r_{32})^2}{3+r_{11}-r_{22}+r_{33}}}, & \text{otherwise} \end{cases}
\tag{25}
$$

$$
q_4 = \begin{cases} \frac{1}{2}\sqrt{1-r_{11}-r_{22}+r_{33}}, & \text{if } -r_{11}-r_{22}+r_{33} > \eta \\[2ex] \frac{1}{2}\sqrt{\dfrac{(r_{21}-r_{12})^2+(r_{31}+r_{13})^2+(r_{32}+r_{23})^2}{3+r_{11}+r_{22}-r_{33}}}, & \text{otherwise} \end{cases}
\tag{26}
$$

Due to the presence of square roots, the signs of q_i, $i = 1, \ldots, 4$ are undefined. As in other methods where these signs are undefined [2], if we assume that q_1 is positive, then, according to (12), (13), and (14), we have to assign the signs of $r_{32}-r_{23}$, $r_{13}-r_{31}$, and $r_{21}-r_{12}$, to q_2, q_3, and q_4, respectively.

4   Computation of the optimal threshold and comparison with Shepperd's method

To find the optimal threshold in our formulation, we here perform a statistical analysis of the proposed method using single-precision floating-point arithmetics in MATLAB$^{\circledR}$. To this end, we generate 10^6 random unit quaternions using the algorithm described in [10] (it actually generates uniformly distributed points in $\mathbb{S}^4$). For each generated quaternion, we obtain the corresponding rotation matrix using (4), and then recover the original quaternion using the proposed method

with different thresholds. The error committed is evaluated as the norm of the vector difference between the original and recovered quaternions. In general, this is not a good way to compute the distance between two quaternions. Nevertheless, since in our case the error is assumed to be very small, the length of the vector connecting both orientations in $\mathbb{S}^4$ is going to coincide with the value of the angle formed by them if seen from the center of $\mathbb{S}^4$ (and this angle can be taken as a distance between any two elements of the 3D rotation group $SO(3)$ [11]).

The results of this statistical analysis, for different values of the threshold, are compiled in Table 1. In this table, the first four columns refer to the error performance. The first one shows the percentage of cases in which the original quaternion is recovered without error. The other three correspond to the error committed in the worst-case, the average error, and the standard deviation of the error, respectively. Finally, we have two columns with the time performance. The first column gives the average time required to compute a quaternion from a rotation matrix; and the second column, the time required in the best of the cases. These results have been obtained for a MATLAB® implementation running on an Intel® Core™i7 with 32 GB of RAM.

Table 1: Performance of the proposed method, for different thresholds, and that of Shepperd's method

Threshold (η)	Quaternions recovered without error	Worst-case error $\times 10^{-6}$	Average error $\times 10^{-6}$	Standard deviation $\times 10^{-6}$	Average time (μs)	Best-case time (μs)
-0.75	19.00%	16.066	0.0332	2.4190	6.93	5.85
-0.50	25.43%	0.136	0.0248	0.0346	7.27	5.36
-0.25	27.61%	0.119	0.0230	0.0329	6.56	5.36
0.00	28.00%	0.123	0.0227	0.0325	6.52	5.36
0.25	27.67%	0.146	0.0254	0.0370	6.36	5.36
0.50	26.63%	0.149	0.0243	0.0342	6.36	5.36
1.00	24.05%	0.189	0.0279	0.0384	6.33	5.36
1.50	21.72%	0.248	0.0323	0.0434	6.35	5.36
2.00	19.20%	0.304	0.0387	0.0510	6.41	5.36
3.00	15.56%	11.164	0.0715	0.6980	6.97	5.56
Shepperd's method	24.40%	0.170	0.0304	0.0410	7.27	6.10

The last row in Table 1 contains the performance figures for Shepperd's method. From this analysis, we conclude that, although the choice of the threshold is not critical, the best results are obtained at about $\eta = 0$ for which the proposed method clearly outperforms Shepperd's method. Indeed, for $\eta = 0$, the proposed method gives the lowest value for the worse-case error, average error, and standard deviation. Moreover, its computational cost is not higher than the celebrated Shepperd's method.
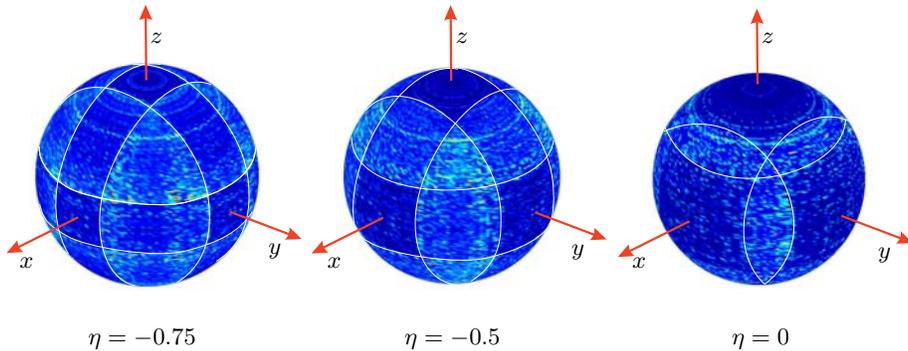
$$\eta = -0.75 \qquad\qquad \eta = -0.5 \qquad\qquad \eta = 0$$

Fig. 1: A discretization of $\mathbb{S}^3$ permits to visualize which orientations can lead to the highest errors. As the threshold in increased from $\eta = -0.75$ to $\eta = 0$, the spherical triangle region centered at each octant is reduced till it vanishes.

To get a deeper insight into the behavior of the proposed method as the threshold varies, let us now explore a visual analysis. To this end, let us discretize the unit sphere $\mathbb{S}^3$. Each sample gives us an orientation for the rotation axis (n_x, n_y, n_z). Then, we discretize the rotated angle in the interval $[0, \pi]$. For each of these samples, we have a quaternion which is converted into a rotation matrix and recovered back using the proposed method. For all the samples of the rotated angle, we only keep the obtained maximum error. Thus, for each orientation in $\mathbb{R}^3$, we have an associated maximum error to which we can assign a color. The result of this experiment is shown in Fig. 1 for the three different thresholds, where brighter colors correspond to higher errors. For $\eta = -0.75$, 25 regions are clearly visible. The lowest errors are obtained for rotation axes corresponding to points contained in the spherical quadrangles centered around the coordinate axes. The highest errors are obtained for rotation axes corresponding to points contained in the spherical triangles centered in each octant. Now, if we increase the threshold, these spherical triangles, where the highest errors are obtained, are reduced. For $\eta = 0$, they have almost completely disappeared. This gives a visual justification of why the optimal threshold is located at $\eta = 0$.

The origin of the pattern of circles centered around the $z-$axis, which do not appear around the other two axes, despite all formulas are symmetric with respect to the three coordinate axes, seem to be an artifact consequence of discretizing the sphere following circles of constant longitude.

5   Conclusion

A new method for computing the quaternion corresponding to a given rotation matrix has been presented. The key idea of this new method is that, instead of generating four alternative solutions for the whole quaternion, as in Shepperd's method, it works with two alternative solutions for each element of the quaternion. This means that the method implicitly works with up to 16 alterna-

tive solutions for the whole quaternion, which accounts for its improved global numerical behavior.

If we take a quaternion at random and we compute the corresponding rotation matrix, the probability of recovering exactly the original quaternion from this matrix using Shepperd's method is about 24%, while using our method this probability is increased to 28%, without an increment in the computational cost. It is certainly a modest improvement, but a large number of methods have been proposed to solve the problem [2] and Shepherd's method has remained the best one till now.

Acknowledgments

References

1. H. Goldstein, *Classical Mechanics*, Addison-Wesley, Cambridge, MA, 1951.
2. S. Sarabandi and F. Thomas, "A survey on the computation of Euler parameters from rotation matrices," submitted for publication.
3. S.W. Shepperd, "Quaternion from rotation matrix," *Journal of Guidance and Control*, Vol. 1, No. 3, pp. 223-224, 1978.
4. M.L. Husty, M. Pfurner, H.-P. Schröcker, and K. Brunnthaler, "Algebraic methods in mechanism analysis and synthesis," *Robotica*, Vol. 25, No. 6, pp. 661-675, 2007.
5. P.C. Hughes, *Spacecraft Attitude Dynamics*, John Wiley & Sons, Inc, 1986.
6. D. Goldberg, "What every computer scientist should know about floating-point arithmetic," *ACM Computing Surveys,* Vol. 23, No. 1, pp. 5-48, 1991.
7. F. Thomas, "Approaching dual quaternions from matrix algebra," *IEEE Transactions on Robotics*, Vol. 30, No. 5, pp. 1037-1048, 2014.
8. M.D. Shuster and G.A. Natanson, "Quaternion computation from a geometric point of view," *The Journal of the Astronautical Sciences*, Vol. 41, No. 4, pp. 545-556, 1993.
9. J. Angeles, *Fundamentals of Robotic Mechanical Systems. Theory, Methods, and Algorithms*, Springer, 1997.
10. G. Marsaglia, "Choosing a point from the surface of a sphere," *Annals of Mathematical Statistics*, Vol. 43, pp. 645-646, 1972.
11. D.Q. Huynh, "Metrics for 3D rotations: comparison and analysis," *Journal of Mathematical Imaging and Vision*, Vol. 35, No. 2, pp. 155-164, 2009.