EXACT NUMERICAL COMPUTATION OF THE RATIONAL GENERAL LINEAR TRANSFORMATIONS

COPYRIGHT

ERRATUM

The sign convention in equation 18 is not explicit. The convention is in two parts. Firstly, the integer square root is signed. That is, the positive or negative root $\sqrt{x}$ is chosen so that $\mathrm{sgn}(\sqrt{x}) = \mathrm{sgn}(x)$. Secondly, the radius r is non-negative. Consequently the sign of the denominators p and q of r/p and r/q is chosen so that $\mathrm{sgn}(p) = \mathrm{sgn}(r/p)$ and $\mathrm{sgn}(q) = \mathrm{sgn}(r/q)$.

Exact Numerical Computation of the Rational General Linear Transformations

James A.D.W. Anderson*
Department of Computer Science, The University of Reading, England

1. Abstract

The rational, general-linear transformations can be computed exactly using rational, matrix arithmetic. A subset of these transformations can be expressed in QR form as the product of a rational, orthogonal matrix Q and a rational, triangular matrix R of homogeneous co-ordinates. We present here a derivation of a half-tangent formula that encodes all of the rational rotations. This presentation involves many fewer axioms than in previous, unpublished work and reduces the number of transrational numbers in the total trigonometric functions from three to two. The practical consequence of this is that rotational sensors, such as computer vision cameras, gyroscopes, lidar, radar, and sonar can all be calibrated in terms of rational half-tangents, hence all subsequent, general-linear, numerical computations can be carried out exactly. In this case the only error is sensor error, so computations can be carried out precisely to the physical limits of the sensor.

Keywords: exact arithmetic, transrational arithmetic, Jacobi algorithm.

2. Introduction

The rational, general-linear transformations can be computed exactly using rational, matrix arithmetic. Most commonly rational numbers are represented by variable length bit-strings describing the numerator and denominator. When it is known that multiplication dominates a computation, rational numbers are sometimes represented by variable-length lists of the prime factors of the numerator and denominator. More abstract schemes are also used. But regardless of the computer representation of rational numbers, many practical matrix algorithms require a parameterisation in terms of rotations. The QR factorisation[4] gives a real linear, or real, homogeneous, general-linear transformation in terms of a rotation matrix Q and a triangular matrix R, thereby isolating the rotation terms. In general the QR factorisation of a rational transformation is irrational, but a rational solution lies asymptotically close by. If a sensor is calibrated in terms of rational rotations, then all subsequent, numerical, general-linear computations can be carried out exactly. In this case, the only error is sensor error, so computations can be carried out precisely to the physical limits of the sensor. In some cases the fact that all computations are rational, rather than floating point, brings advantages in terms of the accuracy, speed, size, weight, power consumption, and cost of the computing element of a sensor.

The use of rational arithmetic to solve eigensystems was discussed by the author in internal reports in 1997 and 1998 and was summarised in collaborative work[2] in 1999. A transrational variant of the Jacobi algorithm exploiting the half-tangent parameterisation of rotation was presented at a workshop in 1999. The details of this algorithm were not published, but we give sufficient detail here to re-construct the algorithm. The use of rational rotations to construct exactly unitary, Fourier transformations was published in 2001[3]. All of these works defined total trigonometric functions, so, for example, $\tan(\pm\pi/2) = \pm\infty$ was handled by introducing transrational numbers $\pm 1/0 = \pm\infty$. The degenerate case of computing trigonometric functions of triangles with sides of zero length was handled by defining nullity $\Phi = 0/0$. The concept of a point at nullity, being a point which lies outside projective space, but within the range of projective geometry was presented in[1]. Nullity serves the very practical purpose of being a numerical result which denotes that there is no numerical solution to some geometrical problem in projective geometry and arises naturally by applying the rules of arithmetic syntactically. Analogously the number $\Phi = 0/0$ indicates that a trigonometric ratio of a triangle with sides of zero length does not exist. Thus computational algorithms were developed that had no error states, because division by zero is permitted in transrational arithmetic. However, the development of transrational arithmetic was rather arbitrary. Here we follow the general approach of the earlier work, by defining transrational arithmetic syntactically on the usual rules of rational arithmetic, but we introduce a more natural, semantic, canonical form than in previous work by imposing

* J.A.D.W.Anderson@reading.ac.uk, http://www.cs.reading.ac.uk
Department of Computer Science, The University of Reading, Reading, Berkshire, England, RG6 6AY.

the usual sign conventions of arithmetic on numbers with a zero denominator. This leads naturally to a transrational arithmetic with just two transrational numbers: nullity $\Phi = 0/0$, and infinity $\infty = 1/0$. The existence of a single infinity representing an existential, that is, exact infinity at $1/0$ and a constructive, that is, asymptotic infinity at $-1/0$ is extremely convenient in both trigonometry and projective geometry. As a side effect, it greatly reduces the number of axioms needed to develop transrational arithmetic.

3. Transrational Arithmetic

$\equiv$	$\frac{x}{y}$	$\frac{-x}{y}$	$\frac{x}{-y}$	$\frac{-x}{-y}$	$\frac{0}{y}$	$\frac{0}{-y}$	$\frac{x}{0}$	$\frac{-x}{0}$	$\frac{0}{0}$
$\frac{x}{y}$	T	$F_{1,2}$	$F_{1,2}$	T	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	F_2
$\frac{-x}{y}$	$F_{1,2}$	T	T	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	F_2
$\frac{x}{-y}$	$F_{1,2}$	T	T	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	F_2
$\frac{-x}{-y}$	T	$F_{1,2}$	$F_{1,2}$	T	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	F_2
$\frac{0}{y}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	T	T	$F_{1,2}$	$F_{1,2}$	F_2
$\frac{0}{-y}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	T	T	$F_{1,2}$	$F_{1,2}$	F_2
$\frac{x}{0}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	T	T	F_2
$\frac{-x}{0}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	$F_{1,2}$	T	T	F_2
$\frac{0}{0}$	F_2	F_2	F_2	F_2	F_2	F_2	F_2	F_2	T

Table 1: Equivalence of Transrational Numbers

We make use of the parity and sign functions defined, respectively, on an integer argument:

$$\mathrm{pty}(z) = \begin{cases} 1, z \geq 0 \\ -1, z < 0 \end{cases} \qquad \mathrm{sgn}(z) = \begin{cases} 0, z = 0 \\ \mathrm{pty}(z), z \neq 0 \end{cases}$$

Rational arithmetic is usually developed in terms of tuples of integers (n, d) where n is the numerator and d is the positive denominator. In developing transrational arithmetic we allow a zero or positive denominator in a number reduced to its lowest terms and apply the definitions of rational arithmetic syntactically. Hence tuples of integers, (n_1, d_1) and (n_2, d_2), are said to be equivalent if (Eqn 1) and (Eqn 2) both hold.

$$n_1 d_2 = n_2 d_1 \qquad \text{(Eqn 1)}$$

Table 1 shows the equivalence of fractions x/y with $x, y \in N$. In the body of the table F_1 shows that the equivalence is falsified by (Eqn 1). In order to prevent all fractions with a zero denominator begin equivalent we force all fractions to obey the sign conventions of rational arithmetic, as follows.

$$[(\mathrm{sgn}(n_1) = \mathrm{sgn}(n_2)) \text{ and } (\mathrm{sgn}(d_1) = \mathrm{sgn}(d_2))] \text{ or } [(-\mathrm{sgn}(n_1) = \mathrm{sgn}(n_2)) \text{ and } (-\mathrm{sgn}(d_1) = \mathrm{sgn}(d_2))] \quad \text{(Eqn 2)}$$

In the body of Table 1 F_2 denotes that the equivalence is falsified by (Eqn 2). On examining the third-last and second-last rows and columns we see that $x/0 \equiv -x/0$ and we choose $1/0$ as the fraction in lowest terms. On examining the last row and column we see that $0/0$ is distinct from all other fractions because every entry is false except $0/0 \equiv 0/0$. Thus $0/0$ is a unique fraction which is, therefore, in its lowest terms. The remaining rows and columns in the table describe rational numbers which are reduced to their lowest terms by dividing the numerator and denominator throughout by their positive, greatest common divisor and taking the denominator non-negative. Thus all fractions of integers are reduced to their lowest terms.

We name the fraction $0/0 = \Phi$ *nullity*[1] and the fraction $1/0 = \infty$ *infinity*. We define the set of transrational numbers T as follows.

$$T = Q \cup \{0/0, 1/0\} \quad \text{(Eqn 3)}$$

Sometimes we refer to Φ and ∞ as *strictly transrational numbers* to distinguish them from the rational numbers in the set of transrational numbers T. Note that the lowest terms of the strictly transrational numbers are consistent with the lowest terms of the rational numbers, because the strictly transrational numbers are distinct from the rational numbers.

The following four operations of rational arithmetic are applied syntactically to tuples of integers with the results reduced to lowest terms as defined above.

$$(a, b) + (c, d) = (ad + bc, bd) \quad \text{(Eqn 4)}$$

$$(a, b) \times (c, d) = (ac, bd) \quad \text{(Eqn 5)}$$

$$-(a, b) = (-a, b) \quad \text{(Eqn 6)}$$

$$(a, b)^{-1} = (b, a) \quad \text{(Eqn 7)}$$

These operations are consistent with rational arithmetic because the operations are just the operations of rational arithmetic applied syntactically and because the lowest terms of rational and strictly transrational numbers are consistent. Hence any rational calculation produces an identical result in transrational arithmetic.

We adopt one of the ordering relationships of rational arithmetic applied syntactically and reduced to lowest terms. That is, with $x, y \in T$:

$$x > y \Leftrightarrow x - y > 0 \quad \text{(Eqn 8)}$$

We define an ordering of infinity relative to zero as follows.

$$\infty > 0 \quad \text{(Eqn 9)}$$

An ordering of T follows. Let $q = n/d$, $q \in Q$, then $\infty - q = 1/0 - n/d = (1d - 0n)/(0d) = d/0 = 1/0 = \infty > 0$. That is $\infty > q$. In other words, infinity is a point at infinity on the real number line. But, as we have seen above, nullity is not equal to any other transrational number, so it lies off the real number line augmented by infinity.

4. Principal Range

The classical trigonometric functions are cyclic over a principal range. We now introduce a mapping of the transrational numbers onto a finite range and then define a congruence that makes any function on this range cyclic.

Firstly, we define a mapping *mapq* of the transrational numbers n/d onto the finite range $(-2, 2] \cup \Phi$ as follows. Note that $\text{mapq}(\Phi) = \Phi$, $\text{mapq}(\infty) = 2$, and $\text{mapq}(t)$ is monotonic on $t \in Q \cup \infty$.

$$\text{mapq}(n/d) = \begin{cases} n/d, & |n| \le d \\ 2\,\text{sgn}(n) - d/n, & |n| > d \end{cases} \qquad \text{(Eqn 10)}$$

The inverse *arcmapq* of *mapq* is:

$$\text{arcmapq}(n/d) = \begin{cases} n/d, & |n| \le d \\ -((n/d - 2\,\text{sgn}(n))^{-1}), & |n| > d \end{cases} \qquad \text{(Eqn 11)}$$

We define that $(-2, 2] \cup \Phi$ is the principal range and give a congruence making it cycle over the number line.

$$\text{principal}(n/d) = \begin{cases} \Phi, & n/d = \Phi \\ 2, & n/d = \infty \\ -(((2d - n) \bmod 4d) - 2d)/d, & n/d \in Q \end{cases} \qquad \text{(Eqn 12)}$$

For rational n/d in the range $(-2, 2]$ this is equivalent to the congruence $4k + n/d \equiv n/d, k \in Z$.

5. Transrational Trigonometry

The trigonometric functions of an angle in radians can all be given by infinite series, but only finitely many terms can be evaluated on a digital computer, so some numerical error results when the series is truncated. By contrast an ancient result, available in Euclid's Elements, gives all of the integer lengths of right triangles in terms of all pairs of integers and relates these to the half-tangent. This can be used to define a transrational trigonometry.

Following Euclid, all of the integer roots of $p^2 + q^2 = r^2$ are given in terms of the integers n and d by:

$$p = d^2 - n^2 \qquad q = 2dn \qquad r = d^2 + n^2 \qquad \text{(Eqn 13)}$$

We identify p and q respectively with the x and y Cartesian axes, then, without loss of generality, we let $d \ge 0$ and identify the triangle $p'q'r'$ with all triangles pqr identical over a positive dilatation.

$$p' = p/k \qquad q' = q/k \qquad r' = r/k \qquad \text{(Eqn 14)}$$

where $k = 1$ when $r = 0$ and k is the largest, positive, common divisor of p, q, r when $r \ne 0$. We now see that the transrational numbers n/d encode all p', q', r'.

Firstly, $\Phi = 0/0$ encodes the trivial solution $p' = q' = r' = 0$. Secondly, $\infty = 1/0$ encodes $p' = -1$, $q' = 0$, $r' = 1$. Thirdly, rational n/d gives n and d relatively prime. Now kn, kd with $k \in Z$ gives:

$$\begin{aligned} p &= (kd)^2 - (kn)^2 = k^2(d^2 - n^2) \\ q &= \quad 2(kd)(kn) \quad = k^2(2dn) \\ r &= (kd)^2 + (kn)^2 = k^2(d^2 + n^2) \end{aligned}$$

Here k^2 is a common factor which is eliminated by (Eqn 14). So the pairs of integers with a common factor are of no interest. All pairs of integers have now been considered, so the transrational numbers encode all p', q', r'.

A function *pqr* to generate the rational sides of a right triangle with unit hypotenuse is now defined:

$$\mathrm{pqr}(t) \rightarrow (p', q', r') \qquad \text{(Eqn 15)}$$

Given an arbitrary transrational value t it is reduced to the principal range by (Eqn 12). The pair of integers (n, d) is then obtained from (Eqn 11). Intermediate values p, q, r are obtained from (Eqn 13) and the final values p', q', r' from (Eqn 14).

The inverse function *arcpqr* of *pqr* is:

$$\mathrm{arcpqr}(p', q', r') = \begin{cases} q'/(r' + p'), p \geq 0 \\ 2\mathrm{pty}(q') - q'/(r' - p'), p < 0 \end{cases} \qquad \text{(Eqn 16)}$$

The transrational trigonometric functions corresponding, respectively, to the real trigonometric functions *cosine*, *sine*, *tangent*, *secant*, *cosecant*, and *cotangent* are defined syntactically next, with the results reduced to lowest terms. Here p', q', r' are computed from n/d as described above.

$$\begin{aligned} \mathrm{cosq}(n/d) &= p'/r' & \mathrm{sinq}(n/d) &= q'/r' & \mathrm{tanq}(n/d) &= q'/p' \\ \mathrm{secq}(n/d) &= r'/p' & \mathrm{cscq}(n/d) &= r'/q' & \mathrm{cotq}(n/d) &= p'/q' \end{aligned} \qquad \text{(Eqn 17)}$$

Here *cosq* is pronounced so that it rhymes with “mosque”, *sinq* so that it rhymes with “sink”, and *tanq* so that it rhymes with “tank”. The less frequently used functions might best be referred to as the “trans(rational) secant”, “trans(rational) cosecant”, and “trans(rational) cotangent”.

The inverses of the transrational trigonometric functions are then given in terms of *arcpqr*, (Eqn 16), as follows.

$$\begin{aligned} \mathrm{arccosq}(p'/r') &= \mathrm{arcsecq}(r'/p') = \mathrm{arcpqr}(p', \sqrt{(r'+p')(r'-p')}, r') \\ \mathrm{arcsinq}(q'/r') &= \mathrm{arccscq}(r'/q') = \mathrm{arcpqr}(\sqrt{(r'+q')(r'-q')}, q', r') \\ \mathrm{arctanq}(q'/p') &= \mathrm{arccotq}(p'/q') = \mathrm{arcpqr}(p', q', \sqrt{p'^2 + q'^2}) \end{aligned} \qquad \text{(Eqn 18)}$$

Here the inverse is exact if and only if the integer square root is exact, otherwise we use the integer floor of the square root to give a rational estimate of the irrational quantity.

6. Discussion

We give a very simple parameterisation of the rational rotations in terms of the rational half-tangent. This allows exactly orthogonal computations on rational rotations, such as quaternions, and exactly unitary computations on complex rational rotations, as in the discrete Fourier transform[3]. Exactness might prove useful in computing extremely fine rotational quantities as arise in astronomy and in the reconstruction of images from lidar, radar, and sonar. A theoretical advantage of exactness is that any local improvement in a rotation implies a global improvement, whereas non-exact arithmetics can introduce global, non-rotational, departures from a rotation when making a local improvement. For example, the transrational Jacobi algorithm, described below, always returns exactly orthogonal eigenvectors, which become successively more accurate as the algorithm proceeds. By contrast, floating point implementations of this algorithm generally compute non-orthogonal eigenvectors, thereby contributing a source of error absent from the transrational algorithm.

In the case of a small rotation *sinq* is a close approximation to the radian measure of the rotation, so transrational computations can provide approximate radian measures directly. Of course, one could compute a truncated, rational, series approximation to the inverse of any trigonometric value to obtain a close approximation to the radian measure.

When calibrating a sensor it is helpful to recall that all of the transrational trigonometric functions are defined in terms of integral sides of a right triangle. Hence a sensor can be calibrated by counting wavelengths of light in an interferometer. This, potentially, allows calibration down to the atomic scale in the sensor. Conversely a sensor can often be arranged to transduce its input in terms of the integral sides of a right triangle. A sensor with a physical rotor can sometimes be made to calibrate itself during operation. Here it is important that the radius, r, of the physical or synthetic rotor in the sensor is calibrated in the natural units of the device. It can then be shown that *arcpqr* returns either the exact rational rotation or a close rational approximation to the required irrational rotation. Hence no further conditioning of the sensed data is required.

The Jacobi algorithm[4] recovers the whole eigensystem from a real, symmetric matrix by applying rotations. Substituting transrational trigonometric functions for the corresponding real trigonometric functions yields an algorithm which always generates exactly orthogonal eigenvectors. In the case of an irrational eigensystem the algorithm converges to an asymptote, but in some simple cases a rational eigensystem is recovered exactly. This occurs where a Jacobi rotation is applied during a sweep only if it is exact. Either no exact rotation can be found during a sweep, or else the eigensystem is recovered exactly. It is not known if the transrational Jacobi algorithm terminates when approximate rotations are applied to non-simple, rational eigensystems.

In the classical Jacobi algorithm the required rotation is found by solving, for θ, the following equation in the elements of the symmetric matrix A.

$$\tan(2\theta) = \frac{2a_{ij}}{a_{ii} - a_{jj}}$$

The equivalent transrational calculation of the parameter t is:

$$t = \mathrm{arctanq}\left(\mathrm{arctanq}\left(\frac{2a_{ij}}{a_{ii} - a_{jj}}\right)\right) \qquad \text{(Eqn 19)}$$

The function *arctanq* is transrational, so no special action need be taken when $a_{ii} - a_{jj} = 0$. The required Jacobi rotation is given in all cases by:

$$\begin{bmatrix} \mathrm{cosq}(t) & \mathrm{sinq}(t) \\ -\mathrm{sinq}(t) & \mathrm{cosq}(t) \end{bmatrix} \qquad \text{(Eqn 20)}$$

7. Conclusion

We give an improved and stricter development of transrational numbers and transrational trigonometry than in our previous, unpublished work, and give sufficient detail to re-construct a transrational algorithm for computing eigensystems by the Jacobi method. We note that rotational sensors, such as computer vision cameras, gyroscopes, lidar, radar, and sonar can all be calibrated in terms of rational half-tangents, thereby obtaining exact results in all subsequent, numerical, general-linear computations. In this case the only error is sensor error, so computations can be carried out precisely to the physical limits of the sensor.

8. Acknowledgements

I would especially like to thank three people for their contributions to this paper. Steve Maybank made me aware of the encoding of integral, right triangles in terms of two integers. Peter Sweby helped me write an earlier version of this paper. Paul Jelfs found an error in one of my equations. Any remaining infelicities or errors are entirely my responsibility.

9. References

1 Anderson, J.A.D.W. (1997) "Representing Geometrical Knowledge" Phil. Trans. Roy. Soc. Lond. series B, vol. 352, no. 1358, pp. 1129-1139, Aug. 1997.

2 Anderson, J.A.D.W. & Sweby, P.K. (1999) *Exact computation in numerical linear algebra: the discrete Fourier transform and symmetric eigensystem*. Technical report RUCS/1999/TR/010/A, Department of Computer Science, the University of Reading, England, RG6 6AY.

3 Anderson, J.A.D.W. & Sweby, P.K. (2001) "Exact computation in numerical linear algebra: the discrete Fourier transform" in Torro, E.F. ed. *Godunov Methods Theory and Applications* Kluwer Academic 2001.

4 Golub, G.H. & van Loan, C.F. (1986) *Matrix Computations* North Oxford Academic, 1986.