A Quick Study on Quaternions

George K. Francis
Department of Mathematics
University of Illinois
gfrancis@uiuc.edu
http://www.math.uiuc.edu/ gfrancis/

Motivation from Algebra (there are other motivations).

In algebra, to solve $mx = y$ for x we merely divide $x = y/m$. And if asked, what is this m, we again divide: $m = y/x$.[1]

For Hamilton, x and y were forces applied to two different points in space, think of steering a bicycle. In computer graphics, it might be two different places for the camera in space. For the two divisions to make sense in this context, we need to give a meaning to two inverses. (I have used m for "move".) The first one, $x = y/m = (/m)y$, is a no-brainer: $/m$ is the inverse operation, if there is one, for the operation of applying m to x. The second, $m = y/x$, is deeper: what is the reciprocal of a (bound) vector?

Solution by generalizing Complex Numbers.

For 2d-physics (or graphics) the answer resides in the complex numbers which were invented for different purposes long before Hamilton. Complex addition (and scalar multiplication) embodied the "vector" qualities (e.g. translation, as in the parallelogram rule) and complex multiplication took care of rotations.

Hamilton found that if he went to 4D, his generalized complex numbers, the quaternions, had (almost) the same properties. The "almost" included the unexpected wrinkle that multiplication was no longer commutative, $ab! = ba$

[1] We use no special mathematical typography here to encourage readers to adopt a simple, ascii keyboard style of mathematical notation. Thus, no Greek letters, no fat matrices and no arrows over the vectors, super or subscripts. Even $/x$ works for x^{-1}, as multiplicative inverse and inverse function. Of course, there are exceptions. The convention _italic_ for *italic*, and x_i for x_i is replaced by its original in this TeXed document.

in general. In particular, if p and q are quaternions, the "action" of q on p, namely $p \longrightarrow qp/q$, is no longer "trivial", the q's don't cancel out.

Even better, Hamilton noticed that if you split 4D into 1D+3D (i.e. $q = x + uy$, where x and y are real, and u is a 3Dvector unit vector) two things fall into place. First, a unit quaternion (has length 1) can now be written as $\cos t + u \sin t$, where u is the direction of q (a unit vector in 3D) and t is its "declination" angle from the north pole of the 3-sphere S^3 (populated by the unit quaternions) towards u (which lies in the "equatorial" 3D space of our experience.)

Secondly, if p is a "pure" quaternion, i.e. it lies in our (equatorial) 3space (algebraically, if $p = 0 + w$) then q acts on p by rotating it exatly $2t$ degrees around the direction (axis) u. (And vice versa, and much more. Besides, all of these assertions deserve to be demonstrated as being really so.)

In other words, rotations in 3D are now subject to algebra. Where, you might ask, do matrices come in? (Well, Hamilton didn't use them.) Strictly speaking, they come in only at the computational end of things, because nobody can keep 16 numbers in their head. But, since most computer graphicists barely learned linear algebra, and no geometry beyond advanced calculus, it's no wonder that they like matrices. They keep rediscovering (and patenting !) the matricial form of the operation $p \longrightarrow qp/q$. (Half the time they get it wrong as on the website "Gamasutra".)

So here is the mystery in a single sentence: The unit sphere S^3in R^4 is a Lie group under quaternionic multiplication, which acts on our R^3 by the operation $p \longrightarrow qp/q$ as a rotation. The group of 3D rotations, $SO(3)$, is "doubly covered" i.e. $S^3 \longrightarrow SO(3)$ is 2:1 , which, among other things, explains the "plate trick".

Before we solve the harder of Hamilton's riddles, let's demythologize the rotation matrix. First, you need to recall that quaternions multiply by using every vector product you learned in calculus, namely numerical product, dot product, scalar product (twice) and cross product.

$pq = (r + v)(s + w) = (rs - v.w) + rw + vs + v \times w.$

To express qv/q as Mv all you need to remember is that the columns of the orthogonal matrix M are nothing other than $M = [qi/q, qj/q, qk/q]$ where i, j, k are the usual unit vectors. If you need to convince yourself of this fact, recall that the action of a matrix $M = [M_1, M_2, M_3]$ on a vector $v = (x, y, z)$ is just the linear combination,

$Mv = xM_1 + yM_2 + zM_3$. Now apply q and $/q$ to $v = xi + yj + zk$.

Exercise. Given the matrix M of a rotation, find (one of the two) of its quaternions.

Geometrical Motivation and Solution.

Recall that vectors are abstractions that have magnitude and direction. But vectors have two geometrical interpretations. Learning how not to get them confused separates the A's from the C's in Calculus.

A socalled *location vector* is associated with a *position* in space (the arrow comes out of the origin and points to the position). A vector also specifies a point transformation, called a *translation*, which shoves every point along the vector applied to it.

For applied forces as for camera positions, we need a single concept for the pair: location and heading, just as we invented "vector" for the pair magnitude and direction. I propose the word *place.* Thus a place is a position m and a heading H. It isn't enough to think of H as just one unit vector (a direction), because the camera can tilt while pointing in the same direction. So we bite the bullet and think of H as a triplet of mutually perpendicular unit vectors (this mouthful is called an *orthogonal frame*, and one drops the orthogonal thereby causing confusion among the acolytes in computer graphics.)

(What's even worse, $H = [H_x, H_y, H_z]$ in OpenGL coordinates makes $-H_z$ the direction the camera faces, then H_x is to the right, and H_y is up. But one can get used to that too. The alternative makes Hx forward, Hy rightward, makes Hz downwards to keep a right handed orientationl.)

The (H, m) is also interpreted as a displacement (transformation) of places in space thus: $(H, m)(K, n) = (HK, m + Hn)$, where addition and multiplication is from vector algebra. So we can now answer the question of what the fraction of two places might be (here K' means the inverse of K for typographical reasons.)

$$(H, m)/(K, n) = (HK', m - HK'n)$$

Check: $(HK', m - K'n)(K, n) = (HK'K, (m - HK'n) + HK'n) = (H, m)$.

Note that only if the heading is the same, $H = K$, is the Hamilton quotient a pure translation, $(I, m-n)$. Note also how the location vector m in (H, m) acts as a translation vector. Such a pair is called an *affine transformation* in geometry, and in OpenGl it is called a *modelling matrix*, or just Matrix.

Note that for orthogonal transformations (orthonormal matrices) the inverse is just the transpose. In OpenGL one permits also non-uniform but still orthogonal scaling. This operation is also easily reversed, and H' is not hard to keep track of computationally for more general Matrices.

Notes for next time:

To steer an illliView real-time interactive computer animation (RTICA)[2] we multiply, A, (the affine matrix "aff" in the illiView RTICAs) on the right by small displacements (as polled from the mouse or wand.) Why do we displace A on the right and not on the left?

illiView OpenGL answer:

We want to displace all the objects we can see in the camera uniformly the same way. Since every object is displaced on the left by A, instead of applying a series of small displacements to the left of each of the objects, we do it once and for all on the right of A.

Conventional OpenGl answer:

Here everything has a place in the world, including the kamera, kall its place $k = (H_k, m_k)$. To see the world through this camera, you need to premultiply every object matrix by k', i.e. $A = k'$. Now notice how a displacement dk of the camera, takes $k \to dkk$, hence, $A \to (dkk)' = k'dk' = AdA$.

This (finally) reconciles the two approaches (almost.)

[2] *illiView* is the identifier for a large number of RTICAs written by and for my students.