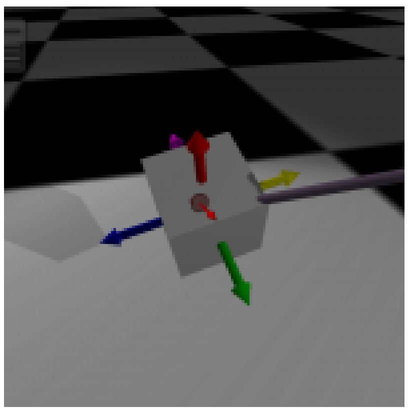
理解四元数

Jeremiah, Wyman

June 25, 2012



—Understanding Quaternions

在这篇文章中我会尝试用简单的方式去解释四元数的概念，我将解释你如何用可视化的方式解释四元数以及几种对四元数的操作。我将把矩阵、欧拉角和四元数放在一起比较，并解释什么时候该用四元数、什么时候该用欧拉角或矩阵。[1]

[1]你不可能在 45 分钟内完全理解四元数。
这篇文章对数学要求很高，不适合胆小的人。

目录

1 简介

在计算机图形学中,我们使用转换矩阵来表示空间中的一个位置 (translation 操作) 以及它的方向 (rotation 操作)。一个可选的转换矩阵还可以表示对一个目标的缩放 (scale) 或错切 (shear) 等。我们可以把转换矩阵想象成一个“基空间 (basis space)”，当你用这个矩阵乘以向量、点（或甚至其它矩阵）后，你就把这些向量、点、矩阵转换进这个矩阵所代表的空间了。

在这篇文章中，我不会讨论转换矩阵的细节。对于描述转换矩阵的细节，你可以查看我前面的名为矩阵文章。

在这篇文章中，我想要用另外一种方式讨论用四元数来描述空间里的物体的方向 (rotation)。

四元数的概念是由爱尔兰数学家William Rowan Hamilton 先生发明的（1843 年，都柏林，爱尔兰）。Hamilton 当时正和他的妻子前往爱尔兰皇家研究院，当他从Brougham 桥通过皇家运河时，他领悟到了一个激动人心的东西，并立刻把它刻在桥的一个石头上：

$$i^2 = j^2 = k^2 = ijk = -1$$

—William Rowan Hamilton Plaque on Broome Bridge on the Royal Canal commemorating his discovery of the fundamental formula for quaternion multiplication.

2 复数

在我们能够完全理解四元数之前，我们必须先知道四元数是怎么来的。四元数的根源是基于复数系统的概念。

除了知名的数集（自然数、整数、实数、有理数）之外，复数系统引入了一个新的数集——虚数。虚数的发明是为了解一些特定的无解的方程，例如：

$$x^2 + 1 = 0$$

要解这个等式，必须让 $x^2 = -1$，这当然是不行的，因为任意实数（正数或负数）的平方都是非负数。

一般而言，数学家是不能忍受一个等式是无解的。于是，一个新的术语被发明了，它就是虚数，一个可以解上面这个等式的数。

虚数有这样的形式：

$$i^2 = -1$$

不要为这个术语较真，因为逻辑上这个数是不存在的。只要知道 i 是一个平方等于 -1 的东西即可。

虚数的集合可以用 $\mathbb{I}$ 来表示。

复数的集合 $\mathbb{C}$ 是一个实数和一个虚数的和，形式如下：

$$z = a + bi \quad a, b \in \mathbb{R}, \quad i^2 = -1$$

可以认为所有实数都是 $b = 0$ 的复数、所有虚数都是 $a = 0$ 的复数。

2.1 复数的加减

复数可以通过加减实部和虚部来加减。

加法：

$$(a_1 + b_1 i) + (a_2 + b_2 i) = (a_1 + a_2) + (b_1 + b_2)i$$

减法：

$$(a_1 + b_1 i) - (a_2 + b_2 i) = (a_1 - a_2) + (b_1 - b_2)i$$

2.2 复数的系数缩放

复数乘以标量，即复数的每一项乘以标量：

$$\lambda(a_1 + b_1 i) = \lambda a_1 + \lambda b_1 i$$

2.3 复数的积

复数也可以通过应用正规代数规则来相乘。

$$\begin{aligned} z_1 &= (a_1 + b_1 i) \\ z_2 &= (a_2 + b_2 i) \\ z_1 z_2 &= (a_1 + b_1 i)(a_2 + b_2 i) \\ &= a_1 a_2 + a_1 b_2 i + b_1 a_2 i + b_1 b_2 i^2 \\ &= (a_1 a_2 - b_1 b_2) + (a_1 b_2 + b_1 a_2) i \end{aligned}$$

2.4 复数的平方

复数也可以通过自身相乘来平方：

$$\begin{aligned} z &= (a + bi) \\ z^2 &= (a + bi)(a + bi) \\ &= (a^2 - b^2) + 2abi \end{aligned}$$

2.5 共轭复数

复数的共轭 (**conjugate**) 就是指把复数的虚数部分变成负的。共轭复数的符号是 $\bar{z}$ 或 z^*。

$$\begin{aligned} z &= (a + bi) \\ z^* &= (a - bi) \end{aligned}$$

复数和它的共轭复数的乘积是：

$$
\begin{aligned}
z &= (a+bi) \\
z^* &= (a-bi) \\
zz^* &= (a+bi)(a-bi) \\
&= a^2 - abi + abi + b^2 \\
&= a^2 + b^2
\end{aligned}
$$

2.6 复数的绝对值

我们可以用复数的共轭 (**conjugate**) 来计算复数的绝对值(或范数 (**norm**) 或幅度 (**magnitude**))。复数的绝对值是复数乘以其共轭 (**conjugate**) 的平方根，表示为 $|z|$:

$$
\begin{aligned}
z &= (a+bi) \\
|z| &= \sqrt{zz^*} \\
&= \sqrt{(a+bi)(a-bi)} \\
&= \sqrt{a^2+b^2}
\end{aligned}
$$

2.7 两复数的商

为了计算两个复数的商，我们将分子和分母乘以分母的复共轭。

$$
\begin{aligned}
z_1 &= (a_1 + b_1 i) \\
z_2 &= (a_2 + b_2 i) \\
\frac{z_1}{z_2} &= \frac{(a_1+b_1 i)}{(a_2+b_2 i)} \\
&= \frac{(a_1+b_1 i)(a_2-b_2 i)}{(a_2+b_2 i)(a_2-b_2 i)} \\
&= \frac{a_1a_2 - a_1b_2 i + b_1a_2 i - b_1b_2 i^2}{a_2^2+b_2^2} \\
&= \frac{a_1a_2+b_1b_2}{a_2^2+b_2^2} + \frac{b_1a_2 - a_1b_2}{a_2^2+b_2^2} i
\end{aligned}
$$

3 i 的幂

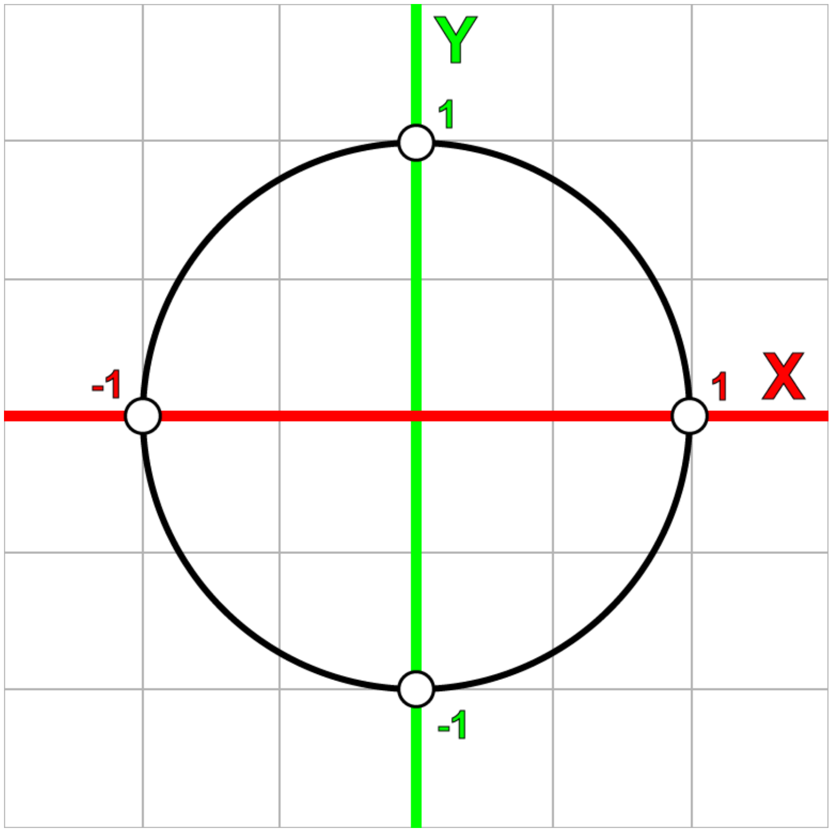
如果我们声明 $i^2=-1$，那么也应该有可能将 i 提升到其他次幂：

$$\begin{array}{lllllll} i^0 & = & & & & & 1 \\ i^1 & = & & & & & i \\ i^2 & = & & & & & -1 \\ i^3 & = & ii^2 & = & & & -i \\ i^4 & = & i^2i^2 & = & & & 1 \\ i^5 & = & ii^4 & = & & & i \\ i^6 & = & ii^5 & = & i^2 & = & -1 \\ i^7 & = & ii^6 & & & = & -i \end{array}$$

如果按照这个顺序写下去，会出现这样一个模式：$(1,i,-1,-i,1,\ldots)$
一个类似的模式也出现在递增的负数幂：

$$\begin{array}{lcr} i^0 & = & 1 \\ i^{-1} & = & -i \\ i^{-2} & = & -1 \\ i^{-3} & = & i \\ i^{-4} & = & 1 \\ i^{-5} & = & -i \\ i^{-6} & = & -1 \\ i^{-7} & = & i \end{array}$$

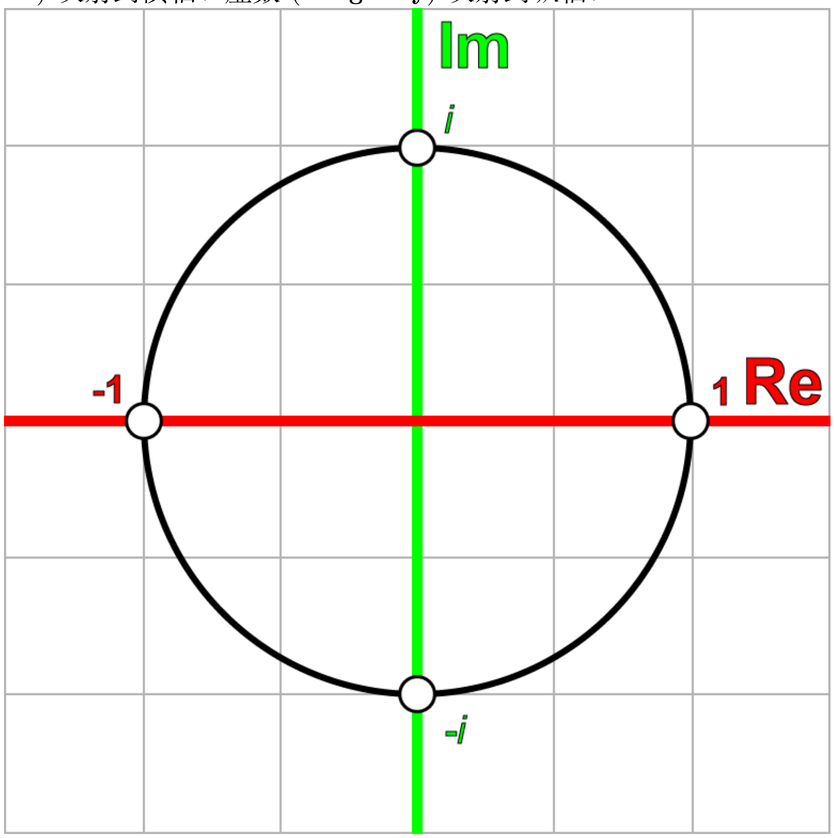
你可能已经在数学里头见过类似的模式，但是是以 $(x,y,-x,-y,x,\ldots)$ 的形式,这是在 2D 笛卡尔平面对一个点逆时针旋转 90° 时生成的;$(x,-y,-x,y,x,\ldots)$ 则是在 2D 笛卡尔平面对一个点顺时针旋转 90° 时生成的。



—Cartesian Plane

4 复数平面

我们也能够把复数映射到一个 2D 网格平面——复数平面，只需要把实数 (**Real**) 映射到横轴、虚数 (**Imaginary**) 映射到纵轴。



—Complex Plane

如前面的序列所示，我们可以认为，对一个复数乘以 i，这个复数就在复数平面上旋转了 90°。

让我们看看这是不是真的。我们随机地在复数平面上取一个绝对点 p:

$$p = 2 + i$$

p 乘以 i 后得到 q:

$$
\begin{aligned}
p &= 2+i \\
q &= pi \\
&= (2+i)i \\
&= 2i+i^2 \\
&= -1+2i
\end{aligned}
$$

q 乘以 i 后得到 r:

$$
\begin{aligned}
q &= -1+2i \\
r &= qi \\
&= (-1+2i)i \\
&= -i+2i^2 \\
&= -2-i
\end{aligned}
$$

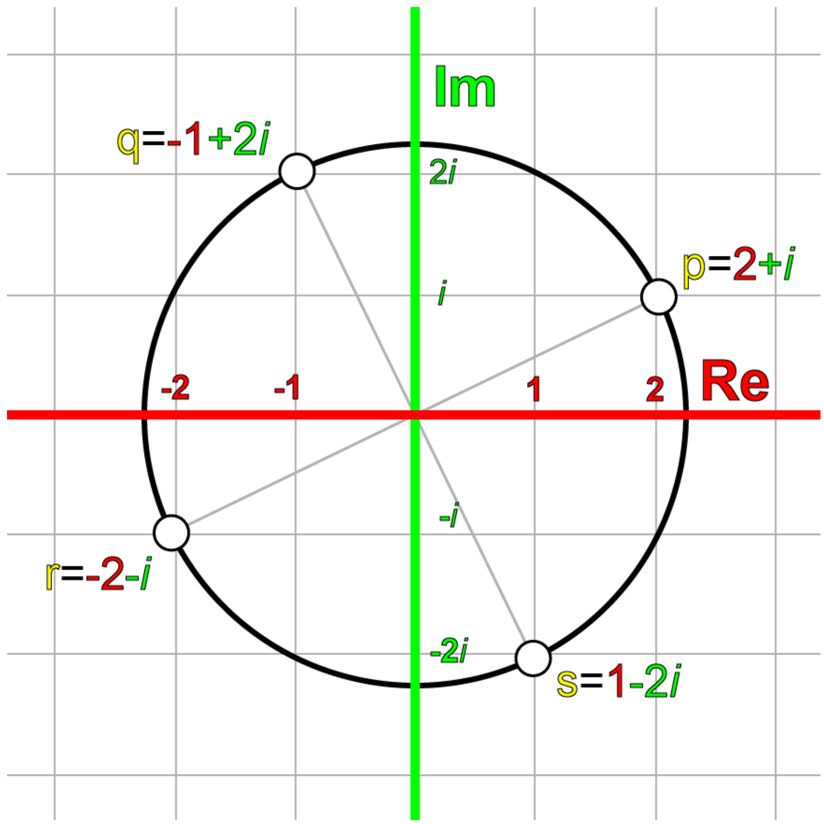
r 乘以 i 后得到 s:

$$
\begin{aligned}
r &= -2-i \\
s &= ri \\
&= (-2-i)i \\
&= -2i-i^2 \\
&= 1-2i
\end{aligned}
$$

s 乘以 i 后得到 t:

$$
\begin{aligned}
s &= 1-2i \\
t &= si \\
&= (1-2i)i \\
&= i-2i^2 \\
&= 2+i
\end{aligned}
$$

t 刚好是开始的 p。如果我们把这些复数放到复数平面上，就得到下面的图:



—Complex Numbers on the Complex Plane

我们也可以按顺时针方向旋转，只需要把上面的乘数 i 改成 $-i$。

4.1 旋转子（Rotors）

我们也可以在复数平面上进行任意角度的旋转，只需要定义下面这个复数：

$$q = \cos\theta + i\sin\theta$$

将任意复数乘以旋转子 q，得出一般公式：

$$\begin{aligned}
p &= a + bi \\
q &= \cos\theta + i\sin\theta \\
pq &= (a + bi)(\cos\theta + i\sin\theta) \\
a' + b'i &= a\cos\theta - b\sin\theta + (a\sin\theta + b\cos\theta)i
\end{aligned}$$

也可以写成矩阵的形式:

$$\begin{bmatrix} a' & -b' \\ b' & a' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} a & -b \\ b & a \end{bmatrix}$$

这也是一个在复数平面绕原点逆时针旋转任意点的方法。(译注:这句话应该是在说旋转矩阵)

5 四元数

利用复数系统和复平面的知识，我们可以把它推广到三维空间，除了 i 之外，我们还可以在数字系统中加入两个虚数。

四元数的一般形式：

$$q = s + xi + yj + zk \quad s, x, y, z \in \mathbb{R}$$

上面的公式是根据 Hamilton 的著名的表达式得到的：

$$i^2 = j^2 = k^2 = ijk = -1$$

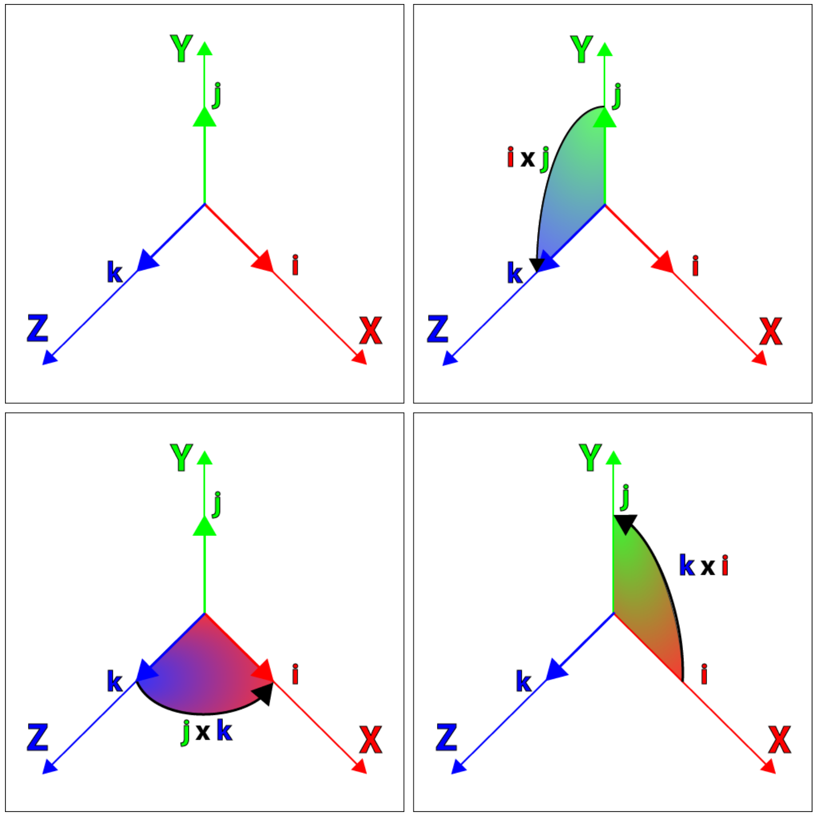
以及：

$$\begin{array}{ccc} ij = k & jk = i & ki = j \\ ji = -k & kj = -i & ik = -j \end{array}$$

你可能已经注意到了，i、j、k 之间的关系非常像笛卡尔坐标系下单位向量的叉积规则：

$$\begin{array}{ccc} \mathbf{x} \times \mathbf{y} = \mathbf{z} & \mathbf{y} \times \mathbf{z} = \mathbf{x} & \mathbf{z} \times \mathbf{x} = \mathbf{y} \\ \mathbf{y} \times \mathbf{x} = -\mathbf{z} & \mathbf{z} \times \mathbf{y} = -\mathbf{x} & \mathbf{x} \times \mathbf{z} = -\mathbf{y} \end{array}$$

Hamilton 自己也发现 i、j、k 虚数可以被用来表达 3 个笛卡尔坐标系单位向量 $\mathbf{i}$、$\mathbf{j}$、$\mathbf{k}$，并且仍然保持有虚数的性质，也即 $\mathbf{i}^2 = \mathbf{j}^2 = \mathbf{k}^2 = -1$。



—Visualizing the Properties of **ij**, **jk**, **ki**

上图展示了如何用 **i**、**j**、**k** 作为笛卡尔坐标系的单位向量。

5.1 作为有序对的四元数

我们可以用有序对的形式来表示四元数：

$$q = [s, \mathbf{v}] \quad s \in R, \mathbf{v} \in \mathbb{R}^3$$

其中的 $\mathbf{v}$，也可以用它各自独立的 3 个分量表示：

$$q = [s, x\mathbf{i} + y\mathbf{j} + z\mathbf{k}] \quad s, x, y, z \in \mathbb{R}$$

使用这种表示法，我们可以更容易地展示四元数和复数之间的相似性。

5.2 四元数的加减

和复数类似，四元数也可以被加减：

$$\begin{aligned} q_a &= [s_a, \mathbf{a}] \\ q_b &= [s_b, \mathbf{b}] \\ q_a + q_b &= [s_a + s_b, \mathbf{a} + \mathbf{b}] \\ q_a - q_b &= [s_a - s_b, \mathbf{a} - \mathbf{b}] \end{aligned}$$

5.3 四元数的积

我们也可以表示四元数的乘积：

$$\begin{aligned} q_a &= [s_a, \mathbf{a}] \\ q_b &= [s_b, \mathbf{b}] \\ q_a q_b &= [s_a, \mathbf{a}][s_b, \mathbf{b}] \\ &= (s_a + x_a i + y_a j + z_a k)(s_b + x_b i + y_b j + z_b k) \\ &= (s_a s_b - x_a x_b - y_a y_b - z_a z_b) + \\ &\quad (x_a s_b + s_a x_b - z_a y_b + y_a z_b) i + \\ &\quad (y_a s_b + z_a x_b + s_a y_b - x_a z_b) j + \\ &\quad (z_a s_b - y_a x_b + x_a y_b + s_a z_b) k \end{aligned}$$

可以看到，四元数的乘积依然还是一个四元数。如果我们把虚数 i、j 、k 替换成有序对：

$$i = [0, \mathbf{i}] \quad j = [0, \mathbf{j}] \quad k = [0, \mathbf{k}]$$

以及还有 $[1, \mathbf{0}] = 1$，将它们代入前面的表达式，就得到了：

$$\begin{aligned} [s_a, \mathbf{a}][s_b, \mathbf{b}] &= (s_a s_b - x_a x_b - y_a y_b - z_a z_b)[1, \mathbf{0}] + \\ &\quad (x_a s_b + s_a x_b - z_a y_b + y_a z_b)[0, \mathbf{i}] + \\ &\quad (y_a s_b + z_a x_b + s_a y_b - x_a z_b)[0, \mathbf{j}] + \\ &\quad (z_a s_b - y_a x_b + x_a y_b + s_a z_b)[0, \mathbf{k}] \end{aligned}$$

再把这个表达式扩展成多个有序对的和：

$$
\begin{aligned}
[s_a, \mathbf{a}][s_b, \mathbf{b}] &= [s_a s_b - x_a x_b - y_a y_b - z_a z_b, \mathbf{0}] + \\
&\quad [\mathbf{0}, (x_a s_b + s_a x_b - z_a y_b + y_a z_b)\mathbf{i}] + \\
&\quad [\mathbf{0}, (y_a s_b + z_a x_b + s_a y_b - x_a z_b)\mathbf{j}] + \\
&\quad [\mathbf{0}, (z_a s_b - y_a x_b + x_a y_b + s_a z_b)\mathbf{k}] +
\end{aligned}
$$

如果我们与四元数单位相乘并提取公共向量分量，我们可以这样重写此方程：

$$
\begin{aligned}
[s_a, \mathbf{a}][s_b, \mathbf{b}] &= [s_a s_b - x_a x_b - y_a y_b - z_a z_b, \mathbf{0}] + \\
&\quad [0, s_a(x_b\mathbf{i} + y_b\mathbf{j} + z_b\mathbf{k}) + s_b(x_a\mathbf{i} + y_a\mathbf{j} + z_a\mathbf{k}) \\
&\quad + (y_a z_b - z_a y_b)\mathbf{i} + (z_a x_b - x_a z_b)\mathbf{j} + (x_a y_b - y_a x_b)\mathbf{k}]
\end{aligned}
$$

这个等式是两个有序对的和。第一个有序对是一个实 (**Real**) 四元数，第 2 个是一个纯 (**Pure**) 四元数。这两个四元数也可以合并成一个：

$$
\begin{aligned}
[s_a, \mathbf{a}][s_b, \mathbf{b}] &= [s_a s_b - x_a x_b - y_a y_b - z_a z_b, \\
&\quad s_a(x_b\mathbf{i} + y_b\mathbf{j} + z_b\mathbf{k}) + s_b(x_a\mathbf{i} + y_a\mathbf{j} + z_a\mathbf{k}) + \\
&\quad (y_a z_b - z_a y_b)\mathbf{i} + (z_a x_b - x_a z_b)\mathbf{j} + (x_a y_b - y_a x_b)\mathbf{k}]
\end{aligned}
$$

如果把下面的表达式代入上面的等式：

$$
\begin{aligned}
\mathbf{a} &= x_a\mathbf{i} + y_a\mathbf{j} + z_a\mathbf{k} \\
\mathbf{b} &= x_b\mathbf{i} + y_b\mathbf{j} + z_b\mathbf{k} \\
\mathbf{a} \cdot \mathbf{b} &= x_a x_b + y_a y_b + z_a z_b \\
\mathbf{a} \times \mathbf{b} &= \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ x_a & y_a & z_a \\ x_b & y_b & z_b \end{vmatrix} \\
&= (y_a z_b - z_a y_b)\mathbf{i} + (z_a x_b - x_a z_b)\mathbf{j} + (x_a y_b - y_a x_b)\mathbf{k}
\end{aligned}
$$

（译注：注意，第三条'$\cdot$' 和第四条'$\times$' 并不是四元数的点乘和叉乘，而是向量的点乘和叉乘。其实按照历史的顺序来说叉乘是在这里被定义的。）

我们就得到了：

$$
[s_a, \mathbf{a}][s_b, \mathbf{b}] = [s_a s_b - \mathbf{a} \cdot \mathbf{b}, s_a\mathbf{b} + s_b\mathbf{a} + \mathbf{a} \times \mathbf{b}]
$$

这就是四元数乘积的一般式。

5.4 实四元数

一个实 (**Real**) 四元数是一个虚部向量为 **0** 向量的四元数:

$$q = [s, \mathbf{0}]$$

两个实 (**Real**) 四元数的乘积是另一个实 (**Real**) 四元数:

$$\begin{aligned} q_a &= [s_a, \mathbf{0}] \\ q_b &= [s_b, \mathbf{0}] \\ q_a q_b &= [s_a, \mathbf{0}][s_b, \mathbf{0}] \\ &= [s_a s_b, \mathbf{0}] \end{aligned}$$

这和两个虚部为 **0** 的复数的乘积几乎一样:

$$\begin{aligned} z_1 &= a_1 + 0i \\ z_2 &= a_2 + 0i \\ z_1 z_2 &= (a_1 + 0i)(a_2 + 0i) \\ &= a_1 a_2 \end{aligned}$$

5.5 四元数的系数缩放

我们也可以用一个系数(实数)去乘四元数:

$$\begin{aligned} q &= [s, \mathbf{v}] \\ \lambda q &= \lambda[s, \mathbf{v}] \\ &= [\lambda s, \lambda \mathbf{v}] \end{aligned}$$

我们可以使用上面所示的乘积或实 (**Real**) 四元数将四元数乘以标量作为实 (**Real**) 四元数来确认这一点:

$$\begin{aligned} q &= [s, \mathbf{v}] \\ \lambda &= [\lambda, \mathbf{0}] \\ \lambda q &= [\lambda, \mathbf{0}][s, \mathbf{v}] \\ &= [\lambda s, \lambda \mathbf{v}] \end{aligned}$$

5.6 纯四元数

与实 (**Real**) 四元数类似，Hamilton 还将纯 (**Pure**) 四元数定义为具有零标量项的四元数：

$$q = [0, \mathbf{v}]$$

也可以写成下面的形式：

$$q = xi + yj + zk$$

然后是两个纯 (**Pure**) 四元数的乘积：

$$\begin{aligned} q_a &= [0, \mathbf{a}] \\ q_b &= [0, \mathbf{b}] \\ q_a q_b &= [0, \mathbf{a}][0, \mathbf{b}] \\ &= [-a \cdot b, a \times b] \end{aligned}$$

5.7 四元数的加法形式

我们可以把四元数写成实 (**Real**) 四元数和纯 (**Pure**) 四元数的和：

$$\begin{aligned} q &= [s, \mathbf{v}] \\ &= [s, \mathbf{0}] + [0, \mathbf{v}] \end{aligned}$$

5.8 单位四元数

给定任意的向量 $\mathbf{v}$，我们可以把这个向量写成一个系数和一个单位方向向量的乘积：

$$\mathbf{v} = v\hat{\mathbf{v}} \quad v = |\mathbf{v}|, |\hat{\mathbf{v}}| = 1$$

将这个定义和纯四元数的定义结合，就得到了：

$$\begin{aligned} q &= [0, \mathbf{v}] \\ &= [0, v\hat{\mathbf{v}}] \\ &= v[0, \hat{\mathbf{v}}] \end{aligned}$$

我们还可以描述一个具有零标量和单位向量的单位四元数：

$$\hat{q} = [0, \hat{\mathbf{v}}]$$

5.9 四元数的二元形式

我们现在可以把单位四元数的定义和四元数的加法形式结合到一起，就创造了一种新的四元数的表示法，这种表示法和复数的表示法形似：

$$\begin{aligned} q &= [s, \mathbf{v}] \\ &= [s, \mathbf{0}] + [0, \mathbf{v}] \\ &= [s, \mathbf{0}] + v[0, \hat{\mathbf{v}}] \\ &= s + v\hat{q} \end{aligned}$$

这就给了我们一种和复数非常相似的四元数表示法：

$$\begin{aligned} z &= a + bi \\ q &= s + v\hat{q} \end{aligned}$$

5.10 共轭四元数

共轭四元数的计算，就是将四元数的虚向量取反：

$$\begin{aligned} q &= [s, \mathbf{v}] \\ q^* &= [s, -\mathbf{v}] \end{aligned}$$

四元数和它的共轭四元数的乘积：

$$\begin{aligned} qq^* &= [s, \mathbf{v}][s, -\mathbf{v}] \\ &= [s^2 - \mathbf{v} \cdot (-\mathbf{v}), -s\mathbf{v} + s\mathbf{v} + \mathbf{v} \times (-\mathbf{v})] \\ &= [s^2 + \mathbf{v} \cdot \mathbf{v}, \mathbf{0}] \\ &= [s^2 + v^2, \mathbf{0}] \end{aligned}$$

5.11 四元数范数

回忆下复数范数的定义：

$$\begin{aligned} |z| &= \sqrt{a^2 + b^2} \\ zz^* &= |z|^2 \end{aligned}$$

类似的，四元数的范数可以这样定义：

$$\begin{aligned} q &= [s, \mathbf{v}] \\ |q| &= \sqrt{s^2 + v^2} \end{aligned}$$

这也让我们可以这样表达四元数范数：

$$qq^* = |q|^2$$

5.12 四元数规范化

利用四元数范数的定义，就可以对四元数进行规范化。要让一个四元数规范化，只需要让这个四元数去除以它的 $|q|$：

$$q = \frac{q}{\sqrt{s^2 + v^2}}$$

举一个例子，让我们规范化下面这个四元数：

$$q = [1, 4\mathbf{i} + 4\mathbf{j} - 4\mathbf{k}]$$

第一步，先计算四元数的范数 (**norm**)：

$$\begin{aligned} |q| &= \sqrt{1^2 + 4^2 + 4^2 + (-4)^2} \\ &= \sqrt{49} \\ &= 7 \end{aligned}$$

然后，我们必须将四元数除以四元数的范数来计算标准化四元数：

$$\begin{aligned} q' &= \frac{q}{|q|} \\ &= \frac{(1 + 4\mathbf{i} + 4\mathbf{j} - 4\mathbf{k})}{7} \\ &= \frac{1}{7} + \frac{4}{7}\mathbf{i} + \frac{4}{7}\mathbf{j} - \frac{4}{7}\mathbf{k} \end{aligned}$$

5.13 四元数的逆

四元数的逆用 q^{-1} 表示。要计算四元数的逆，需要用四元数的共轭四元数去除以四元数的范数的平方：

$$q^{-1} = \frac{q^*}{|q|^2}$$

为了证明这个式子，我们先根据逆的定义，有：

$$qq^{-1} = [1, \mathbf{0}] = 1$$

两边都左乘共轭四元数 q^*：

$$q^*qq^{-1} = q^*$$

通过替换我们得到：

$$\begin{aligned} |q|^2q^{-1} &= q^* \\ q^{-1} &= \frac{q^*}{|q|^2} \end{aligned}$$

对于单位四元数，它的范数是 1，所以可以写成：

$$q^{-1} = q^*$$

5.14 四元数的点积

与矢量点积类似，我们也可以通过乘以相应的标量部分并求和结果来计算两个四元数之间的点积：

$$\begin{aligned} q_1 &= [s_1, x_1\mathbf{i} + y_1\mathbf{j} + z_1\mathbf{k}] \\ q_2 &= [s_2, x_2\mathbf{i} + y_2\mathbf{j} + z_2\mathbf{k}] \\ q_1 \cdot q_2 &= s_1s_2 + x_1x_2 + y_1y_2 + z_1z_2 \end{aligned}$$

我们也可以利用四元数点积，来计算四元数之间的角度差：

$$\cos\theta = \frac{s_1s_2 + x_1x_2 + y_1y_2 + z_1z_2}{|q_1||q_2|}$$

对于单位四元数，我们可以简化上面的等式：

$$\cos\theta = s_1s_2 + x_1x_2 + y_1y_2 + z_1z_2$$

5.15 四元数的指数函数

四元数指数的定义是由绝对收敛级数给出的。

$$e^z = \sum_{k=0}^{\infty} \frac{z^k}{k!}$$

众所周知,根据这个定义,如果 x, y 可交换,我们有 $e^x e^y = e^y e^x = e^{x+y}$。
由于实四元数与所有其他四元数可交换,对于 $a \in \mathbb{R}$,我们有 $e^{a+z} = e^a e^z \quad \forall z \in \mathbb{H}$
所以,如果 $z = a + b\mathbf{i} + c\mathbf{j} + d\mathbf{k} = a + \mathbf{v}$,我们有 $e^z = e^a e^{\mathbf{v}}$,其中 $\mathbf{v}$ 是一个虚(或向量)四元数。现在我们有:

声明 如果 $\mathbf{v} \in \mathbb{H}_p$ 是一个虚的四元数,设 $\theta = |\mathbf{v}|$,我们有:

$$e^{\mathbf{v}} = \cos\theta + \mathbf{v}\frac{\sin\theta}{\theta}$$

证明 我们注意到:

$$\mathbf{v}^2 = (b\mathbf{i} + c\mathbf{j} + d\mathbf{k})(b\mathbf{i} + c\mathbf{j} + d\mathbf{k}) = -b^2 - c^2 - d^2 = -|\mathbf{v}|^2$$

所以:

$$\begin{aligned}
\mathbf{v}^0 &= 1 \\
\mathbf{v}^1 &= \mathbf{v} \\
\mathbf{v}^2 &= -\theta^2 \\
\mathbf{v}^3 &= -\theta^2\mathbf{v} \\
\mathbf{v}^4 &= \theta^4 \\
\mathbf{v}^5 &= \theta^4\mathbf{v} \\
\mathbf{v}^6 &= -\theta^6 \\
\mathbf{v}^7 &= -\theta^6\mathbf{v}
\end{aligned}$$

这个系列变成:

$$
\begin{aligned}
e^{\mathbf{v}} &= \Sigma_{k=0}^{\infty}\frac{\mathbf{v}^k}{k!} \\
&= 1+\frac{\mathbf{v}}{1!}-\frac{\theta^2}{2!}-\frac{\theta^2\mathbf{v}}{3!}+\frac{\theta^4}{4!}+\frac{\theta^4\mathbf{v}}{5!}-\frac{\theta^6}{6!}+\cdots \\
&= 1+\frac{\theta\mathbf{v}}{1!\theta}-\frac{\theta^2}{2!}-\frac{\theta^3\mathbf{v}}{3!\theta}+\frac{\theta^4}{4!}+\frac{\theta^5\mathbf{v}}{5!\theta}-\frac{\theta^6}{6!}+\cdots \\
&= \left(1-\frac{\theta^2}{2!}+\frac{\theta^4}{4!}-\frac{\theta^6}{6!}+\cdots\right)+\frac{\mathbf{v}}{\theta}\left(\frac{\theta}{1!}-\frac{\theta^3}{3!}+\frac{\theta^5}{5!}+\cdots\right) \\
&= \cos\theta+\frac{\mathbf{v}}{\theta}\sin\theta
\end{aligned}
$$

所以四元数的指数函数是:

$$
e^z = e^{a+\mathbf{v}} = e^a\left(\cos|\mathbf{v}|+\frac{\mathbf{v}}{|\mathbf{v}|}\sin|\mathbf{v}|\right)
$$

6 旋转

前面我们定义了一个特殊的复数：旋转子 (**Rotor**)。它是用来旋转 2D 复数平面的点的：

$$q = \cos\theta + i\sin\theta$$

根据四元数和复数的相似性，应该有可能设计一个可以旋转 3D 空间的点的四元数：

$$q = [\cos\theta, \sin\theta\mathbf{v}]$$

让我们测试一下这个理论是否可靠，方法就是计算四元数 q 和向量 $\mathbf{p}$ 的积。第一步，我们把 $\mathbf{p}$ 写成纯 (**Pure**) 四元数的形式：

$$p = [0, \mathbf{p}]$$

以及单位四元数 q：

$$q = [s, \lambda\hat{\mathbf{v}}]$$

从而：

$$\begin{aligned} p' &= qp \\ &= [s, \lambda\hat{\mathbf{v}}][0, \mathbf{p}] \\ &= [-\lambda\hat{\mathbf{v}}\cdot\mathbf{p}, s\mathbf{p} + \lambda\hat{\mathbf{v}}\times\mathbf{p}] \end{aligned}$$

我们可以看到结果是一个同时有系数、有虚向量的四元数。

让我们先考虑特殊的情形：$\mathbf{p}$ 与 $\hat{\mathbf{v}}$ 正交。这种情况下，点乘部分等于 $-\lambda\hat{\mathbf{v}}\cdot\mathbf{p} = 0$。所以上面的四元数就变成了纯 (**Pure**) 四元数：

$$p' = [0, s\mathbf{p} + \lambda\hat{\mathbf{v}}\times\mathbf{p}]$$

这时候，要使 $\mathbf{p}$ 绕 $\hat{\mathbf{v}}$ 旋转，我们只需要代入 $s = \cos\theta$ 和 $\lambda = \sin\theta$ ：

$$p' = [0, \cos\theta\mathbf{p} + \sin\theta\hat{\mathbf{v}}\times\mathbf{p}]$$

现在，让我们找一个例子来测试上面的公式。譬如向量 $\mathbf{p}$ 绕 z 轴 (就是 k 轴) 旋转 45°，那么我们的四元数 q 就变成：

$$
\begin{aligned}
q &= [\cos\theta, \sin\theta\mathbf{k}] \\
&= [\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\mathbf{k}]
\end{aligned}
$$

然后，选一个特殊的向量 $\mathbf{p}$，并且 $\mathbf{p}$ 要和 k 轴正交，譬如把 $\mathbf{p}$ 放到 i 轴上，也就是：

$$
p = [0, 2\mathbf{i}]
$$

好了，现在计算下 qp：

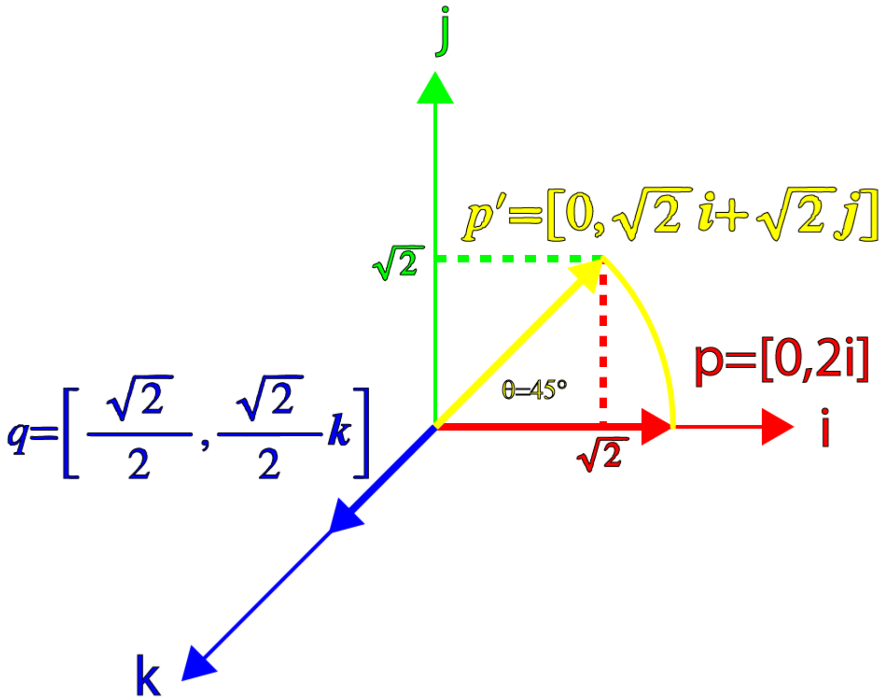
$$
\begin{aligned}
p' &= qp \\
&= \left[\frac{\sqrt{2}}{2}, \frac{\sqrt{2}}{2}\mathbf{k}\right][0, 2\mathbf{i}] \\
&= \left[0, 2\frac{\sqrt{2}}{2}\mathbf{i} + 2\frac{\sqrt{2}}{2}\mathbf{k} \times \mathbf{i}\right] \\
&= [0, \sqrt{2}\mathbf{i} + \sqrt{2}\mathbf{j}]
\end{aligned}
$$

结果是一个绕了 k 轴转了 $45°$ 的纯 (**Pure**) 四元数。我们可以确认这个四元数的向量部分的长度是：

$$
\begin{aligned}
|p'| &= \sqrt{\sqrt{2}^2 + \sqrt{2}^2} \\
&= 2
\end{aligned}
$$

这正是我们所期望的！

我们可以用图像展示旋转过程：



—Quaternion Rotation (1)

现在，让我们考虑更一般化的四元数，即和 $\mathbf{p}$ 不正交的四元数。现在让我们把 $\mathbf{p}$ 的向量部分偏移 45°：

$$\begin{aligned}
\hat{\mathbf{v}} &= \frac{\sqrt{2}}{2}\mathbf{i} + \frac{\sqrt{2}}{2}\mathbf{k} \\
\mathbf{p} &= 2\mathbf{i} \\
q &= [\cos\theta, \sin\theta\hat{\mathbf{v}}] \\
p &= [0, \mathbf{p}]
\end{aligned}$$

把向量 $\mathbf{p}$ 乘以 q，我们得到：

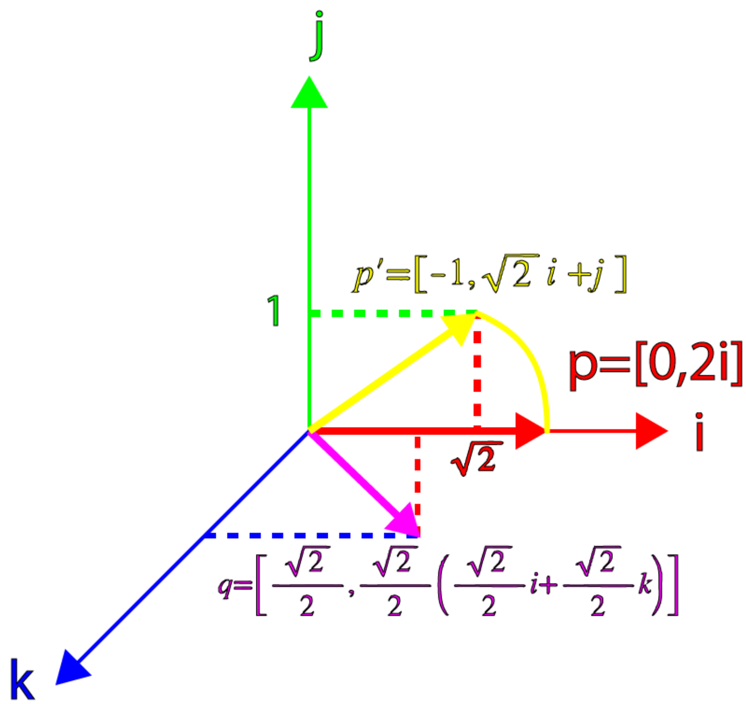
$$\begin{aligned}
p' &= qp \\
&= [\cos\theta, \sin\theta\hat{\mathbf{v}}][0, \mathbf{p}] \\
&= [-\sin\theta\hat{\mathbf{v}}\cdot\mathbf{p}, \cos\theta\mathbf{p} + \sin\theta\mathbf{v}\times\mathbf{p}]
\end{aligned}$$

代入我们设定的 $\hat{\mathbf{v}}$,$\mathbf{p}$，以及 $\theta = 45°$，得到：

$$\begin{aligned} p' &= \left[-\frac{\sqrt{2}}{2}(\frac{\sqrt{2}}{2}\mathbf{i} + \frac{\sqrt{2}}{2}\mathbf{k}) \cdot (2\mathbf{i}), \frac{\sqrt{2}}{2}2\mathbf{i} + \frac{\sqrt{2}}{2}(\frac{\sqrt{2}}{2}\mathbf{i} + \frac{\sqrt{2}}{2}\mathbf{k}) \times 2\mathbf{i} \right] \\ &= [-1, \sqrt{2}\mathbf{i} + \mathbf{j}] \end{aligned}$$

注意，算出来的结果已经不是纯 (**Pure**) 四元数了，并且，它并没有旋转 45°、范数也不再是 2(反而变小了，变成 $\sqrt{3}$)

我们可以用图像展示旋转过程：[2]



—Quaternion Rotation (2)

然而，还有一线生机。Hamilton 发现（但没有正式宣布），如果对 qp 右乘 q 的逆，出来的结果是一个纯 (**Pure**) 四元数，并且四元数向量部分的范数可以保持不变。让我们试试应用在我们的例子里。

首先计算 q^{-1}:

[2]从技术上讲，在三维空间中表示四元数 $p\prime$ 是不正确的，因为它实际上是一个 4D 向量！为了简单起见，我将只可视化四元数的矢量分量。

$$\begin{aligned} q &= [\cos\theta, \sin\theta(\frac{\sqrt{2}}{2}\mathbf{i} + \frac{\sqrt{2}}{2}\mathbf{k})] \\ q^{-1} &= [\cos\theta, -\sin\theta(\frac{\sqrt{2}}{2}\mathbf{i} + \frac{\sqrt{2}}{2}\mathbf{k})] \end{aligned}$$

(译注：这里 $q^{-1} = q^*$ 是因为 q 是单位四元数)

再代入 $\theta = 45°$，得到：

$$\begin{aligned} q^{-1} &= [\frac{\sqrt{2}}{2}, -\frac{\sqrt{2}}{2}(\frac{\sqrt{2}}{2}\mathbf{i} + \frac{\sqrt{2}}{2}\mathbf{k})] \\ &= \frac{1}{2}[\sqrt{2}, -\mathbf{i} - \mathbf{k}] \end{aligned}$$

并结合 qp 和 q^{-1} 的先前值给出：

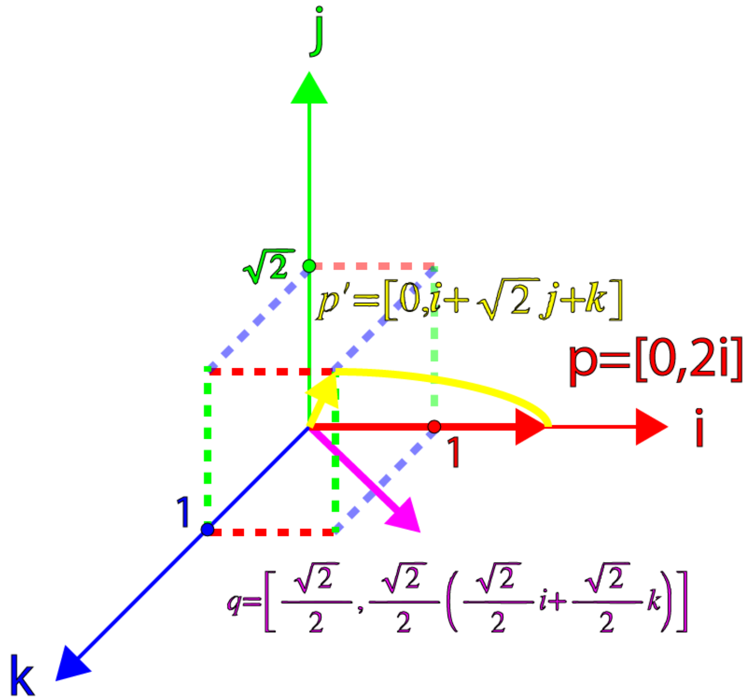
$$\begin{aligned} qp &= [-1, \sqrt{2}\mathbf{i} + \mathbf{j}] \\ qpq^{-1} &= [-1, \sqrt{2}\mathbf{i} + \mathbf{j}]\frac{1}{2}[\sqrt{2}, -\mathbf{i} - \mathbf{k}] \\ &= \frac{1}{2}[-\sqrt{2} - (\sqrt{2}\mathbf{i} + \mathbf{j}) \cdot (-\mathbf{i} - \mathbf{k}), \mathbf{i} + \mathbf{k} + \sqrt{2}(\sqrt{2}\mathbf{i} + \mathbf{j}) - \mathbf{i} + \sqrt{2}\mathbf{j} + \mathbf{k}] \\ &= \frac{1}{2}[-\sqrt{2} + \sqrt{2}, \mathbf{i} + \mathbf{k} + 2\mathbf{i} + \sqrt{2}\mathbf{j} - \mathbf{i} + \sqrt{2}\mathbf{j} + \mathbf{k}] \\ &= [0, \mathbf{i} + \sqrt{2}\mathbf{j} + \mathbf{k}] \end{aligned}$$

这下是纯 (**Pure**) 四元数了，并且它的范数是：

$$\begin{aligned} |p'| &= \sqrt{1^2 + \sqrt{2}^2 + 1^2} \\ &= \sqrt{4} \\ &= 2 \end{aligned}$$

这和原始的 $\mathbf{p}$ 的范数一致。

下面的图像展示了旋转结果：



—Quaternion Rotation (3)

所以我们可以看到，这个结果是一个纯四元数，并且原四元数的向量的范数也保持住了。但是还有一个问题：向量被旋转了 $90°$ 而不是 $45°$。这刚好是我们需要的度数的两倍！为了正确地让一个向量 $\mathbf{p}$ 绕某个轴向量旋转一个角度 θ，我们必须以目标角度的一半来计算。因此，我们构造了下面的四元数：

$$q = [\cos\frac{1}{2}\theta, \sin\frac{1}{2}\theta\hat{\mathbf{v}}]$$

这就是旋转四元数的一般形式！

7 四元数插值

在计算机图形学中使用四元数, 其中一个重要原因是四元数非常适合用来表示空间中的旋转。四元数解决了其他 3 维空间旋转算法会遇到的恼人的问题, 比如使用欧拉角来表示旋转操作时会遇到的万向节锁问题 (Gimbal lock)。

使用四元数, 我们可以定义好几种方案来表示 3 维空间的转动插值。第一种是 **SLERP**, 它被用来把一个点 (物体) 从一个方向平滑地插值到另一个方向。第二个是 **SLERP** 的扩展版本, 被称为 **SQAD**, 它被用来处理用一系列方向定义得到的一条路径的插值。

7.1 SLERP

SLERP 代表 **S**pherical **L**inear Int**erp**olation。**SLERP** 可以在两个方向之间平滑地插值。

第一个方向设为 q_1，第二个方向设为 q_2(请记住，这 2 个指示方向的四元数是单位四元数，不然阅读下文会混乱)。被插值前的点设为 $\mathbf{p}$，插值后的点设为 $\mathbf{p}'$。而插值参数 t，当 $t=0$ 时会把 $\mathbf{p}$ 转到 q_1，当 $t=1$ 时会转到 q_2。

标准的线性插值公式是 (译注：这个公式是笛卡尔坐标系下的，不是指四元数)：

$$\mathbf{p}\prime = \mathbf{p}_1 + t(\mathbf{p}_2 - \mathbf{p}_1)$$

应用这个等式的一般步骤是：

- 计算 $\mathbf{p}_1$ 和 $\mathbf{p}_2$ 之间的差。
- 把差额的小数部分计算出来。
- 通过两点之间的分数差调整原始值。

我们可以使用相同的基本原理在两个四元数方向之间进行插值。

7.1.1 四元数的差

第一步要求我们必须计算 q_1 和 q_2 之间的差。对于四元数，这相当于计算两个四元数之间的角度差 (angular difference)。

$$\Delta q = q_1^{-1} q_2$$

（译注：由 $q_1 \Delta q = q_2$ 推出）

7.1.2 四元数的幂运算

下一步是把差额的小数部分去掉。我们可以把四元数的小数部分提高到数值在范围 $[0 \ldots 1]$ 内的幂。

四元数的幂运算的一般化公式是：

$$q^t = \exp(t \log q)$$

其中，(纯) 四元数的 exp 函数的公式是：

$$\begin{aligned} \exp(q) &= \exp([0, \theta \hat{\mathbf{v}}]) \\ &= [\cos\theta, \sin\theta \hat{\mathbf{v}}] \end{aligned}$$

(纯) 四元数的对数公式是：

$$\begin{aligned} \log q &= \log(\cos\theta + \sin\theta \hat{\mathbf{v}}) \\ &= \log(\exp(\theta \hat{\mathbf{v}})) \\ &= \theta \hat{\mathbf{v}} \\ &= [0, \theta \hat{\mathbf{v}}] \end{aligned}$$

(译注：上述的 2 次公式推导，其实省略了很多证明过程。具体可以参考：第5.15节四元数的指数函数)

对于 $t = 0$，我们有：

$$\begin{aligned} q_0 &= \exp(0 \log q) \\ &= \exp([\cos(0), \sin(0) \hat{\mathbf{v}}]) \\ &= \exp([1, \mathbf{0}]) \\ &= [1, \mathbf{0}] \end{aligned}$$

而对于 $t = 1$，有：

$$\begin{aligned} q_1 &= \exp(\log q) \\ &= q \end{aligned}$$

7.1.3 两个四元数的分数差

对于角旋转的插值计算，我们利用 q_1 和 q_2 的角度分数差来调整原始方向 q_1：

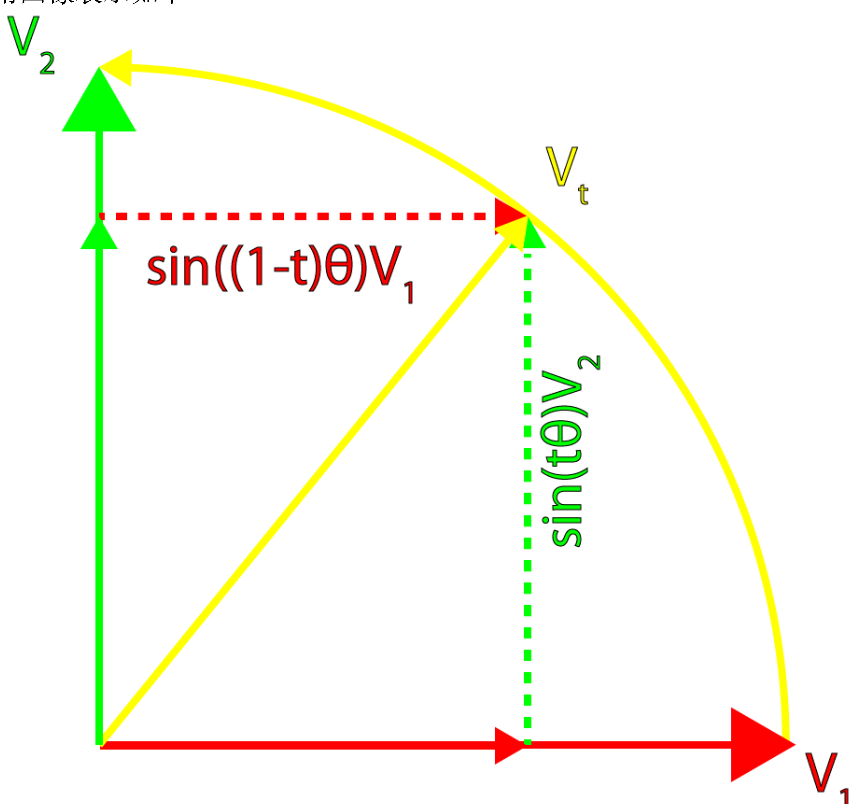
$$q = q_1(q_1^{-1}q_2)^t$$

这也就是使用四元数的球面线性插值的一般形式。然而，这不是 **SLERP** 函数的常用形式。

我们可以应用类似的用于计算向量的球面插值公式，到四元数里。计算向量的球面插值的一般形式定义如下：

$$\mathbf{v}_t = \frac{\sin((1-t)\theta)}{\sin\theta}\mathbf{v}_1 + \frac{\sin(t\theta)}{\sin\theta}\mathbf{v}_2$$

用图像表示如下：



—Quaternion Interpolation

这个公式可以原封不动地应用到四元数：

$$q_t = \frac{\sin((1-t)\theta)}{\sin\theta} q_1 + \frac{\sin(t\theta)}{\sin\theta} q_2$$

但这个公式需要提供角度 θ，我们可以计算 q_1 和 q_2 的点积从而得出角度 θ：

$$\begin{aligned}
\cos\theta &= \frac{q_1 \cdot q_2}{|q1||q2|} \\
&= \frac{s_1 s_2 + x_1 x_2 + y_1 y_2 + z_1 z_2}{|q_1||q_2|} \\
\theta &= \arccos\left(\frac{s_1 s_2 + x_1 x_2 + y_1 y_2 + z_1 z_2}{|q_1||q_2|}\right)
\end{aligned}$$

7.1.4 注意事项

这个方案有两个问题，必须在实现过程中加以考虑。

第一，如果四元数点积的结果是负值，那么后面的插值就会在 4D 球面上绕远路，这并不是我们想要的。为了解决这个问题，我们测试点积的结果，当结果是负值时，我们将两个四元数的其中一个取反，取反它的系数和向量部分，并不会改变它代表的方向。而经过这一步操作，可以保证这个旋转走的是最短路径。

第二，当 q_1 和 q_2 的角度差非常小，小到导致 $\sin\theta$ 趋向 0 时，会出现第二个问题。如果这个情况出现了，当我们除以 $\sin\theta$ 时就会得到一个未定义的结果。在这个情况下，我们可以回退去使用 q_1 和 q_2 的线性插值。

7.2 SQUAD

正如一个 **SLERP** 可以被用来计算四元数之间的插值，一个 **SQUAD** (**S**phericaland **Quad**rangle) 可以被用来对旋转路径进行平滑插值。

如果我们有四元数序列：

$$q_1, q_2, q_3, \ldots, q_{n-2}, q_{n-1}, q_n$$

然后我们再定义一个“辅助”四元数 (s_i)，它是一个中间控制点：

$$s_i = \exp\left(-\frac{\log(q_{i+1}q_i^{-1}) + \log(q_{i-1}q_i^{-1})}{4}\right) q_i$$

所以，沿着子曲线的方向可以定义为：

$$q_{i-1}, q_i, q_{i+1}, q_{i+2}$$

在 t 时刻的方向就是：

$$\mathrm{squad}(q_i, q_{i+1}, s_i, s_{i+1}, t) = \mathrm{slerp}(\mathrm{slerp}(q_i, q_{i+1}, t), \mathrm{slerp}(s_i, s_{i+1}, t), 2t(1-t))$$

8 总结

尽管很难理解，四元数比使用矩阵或欧拉角来表示旋转提供了一些明显的优势。

- 利用 SLERP 和 SQUAD 进行四元数插值，为空间方向间的平滑插值提供了一种方法。
- 使用四元数串联旋转比以矩阵形式表示的组合旋转更快。
- 对于单位范数四元数，通过减去四元数的矢量部分来求其旋转的逆。如果矩阵不是正交的，则计算旋转矩阵的逆要慢得多（如果是，则只是矩阵的转置）。
- 将四元数转换为矩阵的速度略快于欧拉角。
- 四元数只需要 4 个数字（如果它们是标准化的，则需要 3 个数字。实际部分可以在运行时计算）以表示旋转。旋转矩阵至少需要 9 个值。

然而，尽管有利于使用四元数的所有优点，也有一些缺点。

- 由于浮点舍入误差，四元数可能会变得无效，但是这种“误差蠕变”可以通过重新规范化四元数来解决。
- 遏制使用四元数最重要的问题可能是它们很难理解。我希望在阅读了这篇文章之后，这个问题能得到解决。

有几个数学库实现了四元数，其中一些库正确地实现了四元数。以我个人的经验，我发现 GLM(OpenGL Math Library) 是一个很好的数学库，有一个很好的四元数实现。如果你有兴趣在你自己的应用程序中使用四元数，这是我推荐的库。

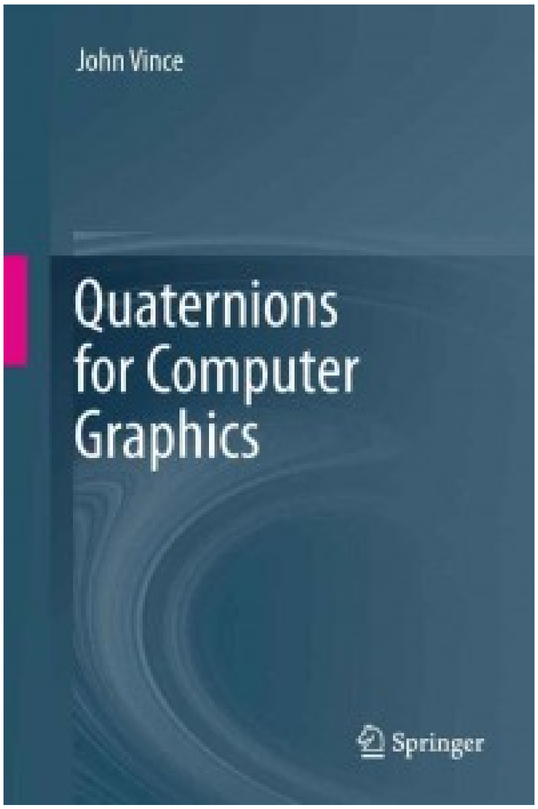
9 下载 Demo

我实现了一个小 demo 来演示一个四元数如何被用来旋转一个 3 维物体。这个 demo 是用Unity 3.5.2 实现的，你可以免费下载它和阅读它的脚本。zip 文件还包含一个 Windows 二进制可执行文件，但是使用 Unity，你还可以生成一个 Mac 应用程序（Unity 4 也引入了 Linux 构建）。

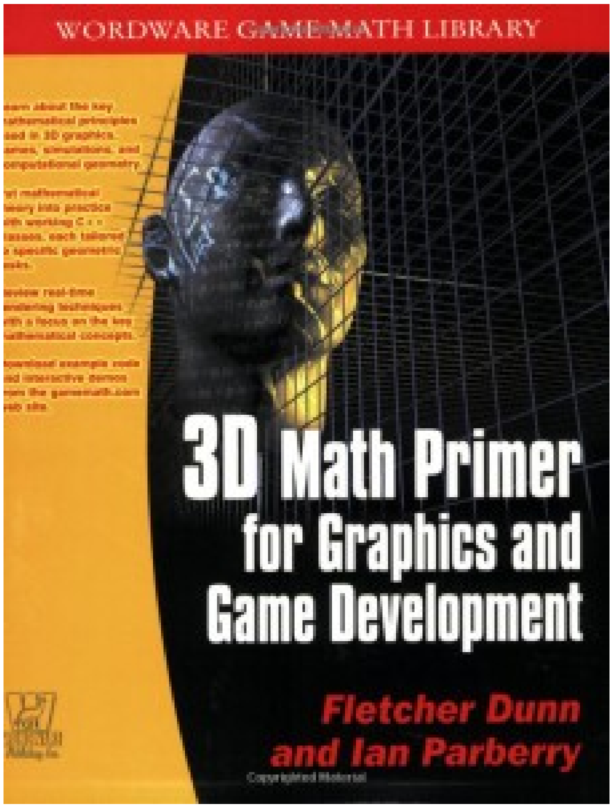
Understanding Quaternions.zip

10 Reference

- Vince, J (2011). Quaternions for Computer Graphics. 1st. ed. London: Springer.



- Dunn, F. and Parberry, I. (2002). 3D Math Primer for Graphics and Game Development. 1st. ed. Plano, Texas: Wordware Publishing, Inc.



- Quaternions –Wikipedia
- Quaternions and Spatial Rotation –Wikipedia
- Exponential Function of Quaternion - Derivation
- 指数函数的泰勒展开公式的证明：MacLaurin series of Exponential function
- cos&sin 三角函数的泰勒展开公式的证明：MacLaurin series of Trigonometric function & Quaternions, Interpolation and Animation

11 后记

- 本文翻译自：Understanding Quaternions。
- 翻译参考了：《理解四元数》一文。主要是此译文有些地方采用了意译的方式，而我认为直译更能体现原作者的精妙原意，特别是简介的第一段关于“基空间 (basis space)”与转换的描述。另外，后来原作者根据读者的反馈又做了一些小调整。所以我重新做了翻译。此外，数学公式中的字体也按照原作者的排版。但是为了便于理解，我对其中一些公式中的项的位置做了调整，特别是在四元数乘法推导过程中的参数。
- 增加了第5.15节四元数的指数函数, 原文来自”Exponential Function of Quaternion - Derivation’’，翻译参考了“四元数公式的补充”。
- 凡有错误，都是我所引入。