四元数与三维旋转

Krasjet

（写完之后暂时还没校对过 _(:3」∠)_，所以肯定会存在很多错误．如果你发现有错误，请直接到 Issues 里报告）

这篇文章的主要目的就是简单讨论一下四元数（Quaternion）与三维旋转之间关系．虽然网上四元数相关的资料有很多了，但是我好像一直没找到令我满意的，所以就想自己来写一篇．目前很多资料都使用了比较抽象的方式来解释这一主题，而且在某一些点上讲得不是很清楚．因为 3D 空间还是在我们理解范围之内的，所以四元数与三维旋转的一些关系可以直接使用一些基础的几何学和线性代数的知识来推导和理解．我们在大部分的时间中也会采用这一方式来理解四元数，如果你对更抽象的内容感兴趣，我们也会在最后非常浅显地提到一些，但是它们不是我们的重点．

因为我们主要的侧重点会偏向几何以及计算机图形学中的应用，如果你是因为在学习物理学或者抽象代数来看这篇文章，我不确定这篇文章能否帮助到你．当然，我在最后提到的一些拓展阅读可能还是有一点用的．

我在写这篇文章的时候顺手写了一些用于 Demo 的 MATLAB 代码，你可以在 GitHub 上找到它们：https://github.com/Krasjet/quaternion

1 复数

在介绍四元数与 3D 旋转之间的关系之前，我们先来讨论一下复数（Complex Number）的一些性质以及它与 2D 旋转之间的关系．四元数的很多性质在很多层面上都与复数非常类似，所以理解复数的一些性质会对理解四元数非常有帮助．

因为复数不是我们的重点，下面对复数的讨论不会非常全面．如果你对复数已经非常了解了，也可以直接跳到第二章．

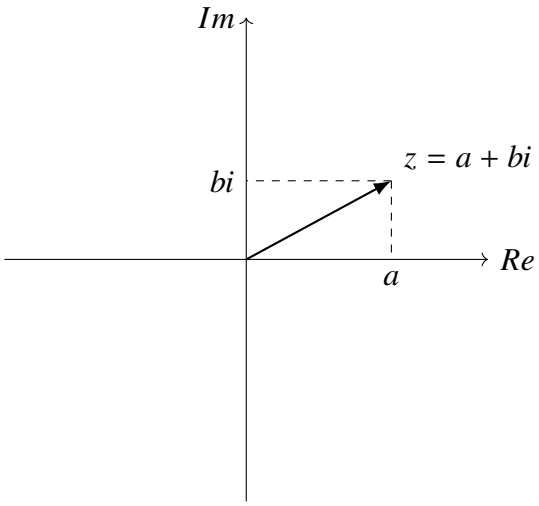
1.1 定义

任意一个复数 $z \in \mathbb{C}$ 都可以表示为 $z = a + bi$ 的形式，其中 $a, b \in \mathbb{R}$ 而且 $i^2 = -1$．我们将 a 称之为这个复数的实部（Real Part），b 称之为这个复数的虚部（Imaginary Part）．

因为 $z = a + bi$ 其实就是对于 $\{1, i\}$ 这个基（Basis）的线性组合（Linear Combination），我们也可以用向量来表示一个复数：

$$z = \begin{bmatrix} a \\ b \end{bmatrix}$$

因为这个向量有两个元素，我们可以使用复平面上的一个点来表示一个复数，复平面的横坐标 Re 代表它的实部，纵坐标 Im 代表它的虚部：



1.2 性质

1.2.1 复数加减法

如果我们有两个复数 $z_1 = a + bi, z_2 = c + di$，它们的和就是分量相加的结果：

$$z_1 + z_2 = (a + c) + (b + d)i$$

同理，如果要对它们相减，直接将分量相减就可以了：

$$z_1 - z_2 = (a - c) + (b - d)i$$

虽然复数的加减法还有很多其它的性质，但是加减法并不是我们的重点，我们需要关注的是复数的乘法.

1.2.2 复数乘法

如果有两个复数 $z_1 = a + bi, z_2 = c + di$，我们可以使用分配律来计算它们的乘积：

$$\begin{aligned} z_1 z_2 &= (a + bi)(c + di) \\ &= ac + adi + bci + bdi^2 \end{aligned}$$

因为 $i^2 = -1$，这可以进一步化简为

$$\begin{aligned} z_1 z_2 &= ac - bd + adi + bci \\ &= ac - bd+ \\ &(bc + ad)i \end{aligned}$$

如果仔细观察你就能发现，这个结果其实就是一个矩阵与向量相乘的结果，也就是说：

$$\begin{aligned} z_1 z_2 &= ac - bd+ \\ &(bc + ad)i \\ &= \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \begin{bmatrix} c \\ d \end{bmatrix} \end{aligned}$$

右侧的 $\begin{bmatrix} c \\ d \end{bmatrix}$ 是用向量的形式来表示的 z_2，而左侧的 $\begin{bmatrix} a & -b \\ b & a \end{bmatrix}$ 则是 z_1 的**矩阵形式**. 我们可以发现，复数相乘这个**运算**，其实是与 $\begin{bmatrix} a & -b \\ b & a \end{bmatrix}$ 这个矩阵所代表的**变换**是等价的（虽然复数与这个矩阵的关系远远不止这些，但是我们现在只将这个矩阵所代表的变换与复数的乘法联系起来）.

那么，在矩阵形式下，复数与复数的相乘则可以表示为矩阵的相乘，如果我们有两

个复数 $z_1 = a + bi, z_2 = c + di$，那么与 $z_1 z_2$ 所代表的变换则可以表示为：

$$\begin{aligned} z_1 z_2 &= \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \begin{bmatrix} c & -d \\ d & c \end{bmatrix} \\ &= \begin{bmatrix} ac - bd & -(bc + ad) \\ bc + ad & ac - bd \end{bmatrix} \end{aligned}$$

注意，复数的相乘是满足交换律的，如果你自己尝试一下，就会发现 $z_1 z_2$ 与 $z_2 z_1$ 是等价的：

$$\begin{aligned} z_2 z_1 &= \begin{bmatrix} c & -d \\ d & c \end{bmatrix} \begin{bmatrix} a & -b \\ b & a \end{bmatrix} \\ &= \begin{bmatrix} ac - bd & -(bc + ad) \\ bc + ad & ac - bd \end{bmatrix} = z_1 z_2 \end{aligned}$$

除此之外，我们来看一下一些特殊复数的矩阵形式：

$$1 = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} = I \qquad (a = 1, b = 0)$$

$$i = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \qquad (a = 0, b = 1)$$

可以看到，实数单位 1 与单位矩阵是等价的，而虚数单位 i 则等价于 $\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$. 如果我们尝试对它进行平方，就可以发现：

$$i^2 = i \cdot i = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix} = \begin{bmatrix} -1 & 0 \\ 0 & -1 \end{bmatrix} = -I = -1$$

所以，即便是在矩阵形式下，i^2 与 -1 其实也是等价的，这进一步展示了复数与这一矩阵形式的关联.

1.2.3 复数的模长与共轭

在进行下一步的讨论之前，我们先定义一下复数的模长（Magnitude）. 如果 $z = a+bi$，那么它的模长为：

$$\|z\| = \sqrt{a^2 + b^2}$$

接下来，我们定义复数的共轭（Conjugate）. 如果 $z = a + bi$，那么它的共轭为：

$$\overline{z} = a - bi$$

我们只需要翻转 z 虚部的符号就能得到它的共轭了. 如果我们尝试计算 $z\overline{z}$，我们会发现：

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

所以，一个复数的模长又可以通过乘积的方式进行计算：

$$\|z\| = \sqrt{z\overline{z}}$$

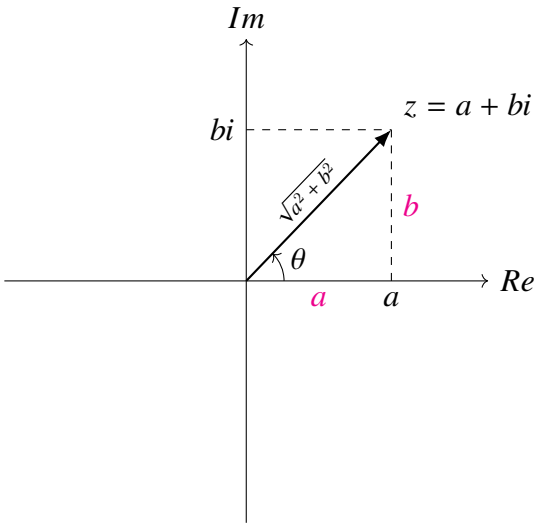
1.3 复数相乘与 2D 旋转

现在，我们回到之前的话题，既然与复数的相乘代表着 $\begin{bmatrix} a & -b \\ b & a \end{bmatrix}$ 矩阵所作出的变换，那这种变换代表着什么呢？

实际上，如果我们对这个矩阵进行一些变形，这个问题就很容易解决了：

$$\begin{bmatrix} a & -b \\ b & a \end{bmatrix} = \sqrt{a^2 + b^2} \begin{bmatrix} \frac{a}{\sqrt{a^2+b^2}} & \frac{-b}{\sqrt{a^2+b^2}} \\ \frac{b}{\sqrt{a^2+b^2}} & \frac{a}{\sqrt{a^2+b^2}} \end{bmatrix}$$

我们将矩阵中每一个元素都除以了模长 $\|z\|\sqrt{a^2 + b^2}$，并将这一项提到了矩阵外面. 经过这一变换，我们只需要观察一下复平面就能够解答之前的问题了：



可以看到，$\|z\|$ 正是复数 z 与坐标轴所形成的三角形的斜边长，而 a, b 分别为三角形的两个直角边．如果将这个三角形与实数轴 Re 的夹角记为 θ 的话，按照三角函数的定义可以得出 $\frac{a}{\sqrt{a^2+b^2}} = \cos(\theta)$，且 $\frac{b}{\sqrt{a^2+b^2}} = \sin(\theta)$，这个角度 θ 其实就是 $\tan^{-1}(\frac{b}{a})$. 知道了这些，原矩阵就可以变形为：

$$
\begin{aligned}
\begin{bmatrix} a & -b \\ b & a \end{bmatrix} &= \sqrt{a^2+b^2} \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \\
&= \|z\| \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \\
&= \|z\| \cdot I \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \qquad \text{(对任意方形矩阵 } A = IA\text{)} \\
&= \begin{bmatrix} \|z\| & 0 \\ 0 & \|z\| \end{bmatrix} \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}
\end{aligned}
$$

我们将原本的矩阵变形为了两个变换矩阵的复合，其中左边的 $\begin{bmatrix} \|z\| & 0 \\ 0 & \|z\| \end{bmatrix}$ 是缩放矩阵，而右边的 $\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$ 则是我们熟悉的 2D 旋转矩阵.

即便你不认识后面的那个旋转矩阵也没有关系，我们可以看看这个矩阵对两个基

$\begin{bmatrix}1\\0\end{bmatrix}$ 和 $\begin{bmatrix}0\\1\end{bmatrix}$ 的变换效果. 首先是 $\begin{bmatrix}1\\0\end{bmatrix}$:

$$\begin{aligned}\begin{bmatrix}a & -b\\ b & a\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix} &= \begin{bmatrix}\|z\| & 0\\ 0 & \|z\|\end{bmatrix}\begin{bmatrix}\cos(\theta) & -\sin(\theta)\\ \sin(\theta) & \cos(\theta)\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix}\\ &= \begin{bmatrix}\|z\| & 0\\ 0 & \|z\|\end{bmatrix}\begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix}\end{aligned}$$

第一步首先将 $\begin{bmatrix}1\\0\end{bmatrix}$ 变换到了 $\begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix}$ 的位置，也就是逆时针旋转了 θ 度，接下来:

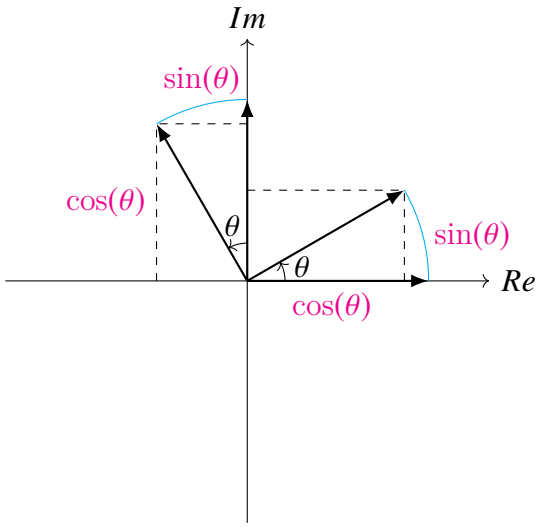
$$\begin{bmatrix}\|z\| & 0\\ 0 & \|z\|\end{bmatrix}\begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix} = \begin{bmatrix}\|z\|\cos(\theta)\\ \|z\|\sin(\theta)\end{bmatrix}$$

缩放矩阵将 $\begin{bmatrix}\cos(\theta)\\ \sin(\theta)\end{bmatrix}$ 缩放了 $\|z\|$ 倍，变为 $\begin{bmatrix}\|z\|\cos(\theta)\\ \|z\|\sin(\theta)\end{bmatrix}$. 总的来说，就是对 $\begin{bmatrix}1\\0\end{bmatrix}$ 逆时针旋转了 θ 度，并缩放了 $\|z\|$ 倍.

接下来是 $\begin{bmatrix}0\\1\end{bmatrix}$:

$$\begin{aligned}\begin{bmatrix}a & -b\\ b & a\end{bmatrix}\begin{bmatrix}0\\1\end{bmatrix} &= \begin{bmatrix}\|z\| & 0\\ 0 & \|z\|\end{bmatrix}\begin{bmatrix}\cos(\theta) & -\sin(\theta)\\ \sin(\theta) & \cos(\theta)\end{bmatrix}\begin{bmatrix}1\\0\end{bmatrix}\\ &= \begin{bmatrix}\|z\| & 0\\ 0 & \|z\|\end{bmatrix}\begin{bmatrix}-\sin(\theta)\\ \cos(\theta)\end{bmatrix}\end{aligned}$$

这里，第一步将 $\begin{bmatrix}0\\1\end{bmatrix}$ 变换到了 $\begin{bmatrix}-\sin(\theta)\\ \cos(\theta)\end{bmatrix}$ 的位置，这同样是逆时针旋转了 θ 度，见下图:



第二步的变换同样会将 $\begin{bmatrix} -\sin(\theta) \\ \cos(\theta) \end{bmatrix}$ 缩放为 $\begin{bmatrix} -\|z\|\sin(\theta) \\ \|z\|\cos(\theta) \end{bmatrix}$. 这样等于是将整个坐标系逆时针旋转了 θ 角，并缩放了 $\|z\|$ 倍.

所以，复数的相乘其实是**旋转与缩放的复合**. 如果有一个复数 $z = a + bi$，那么 z 与任意一个复数 c 相乘都会将 c 逆时针旋转 $\theta = \tan^{-1}(\frac{a}{b})$ 度，并将其缩放 $\|z\| = \sqrt{a^2 + b^2}$ 倍.

如果 $\|z\| = 1$，也就是说 $a^2 + b^2 = 1$，即这个复数可以用一个单位向量来表示，那么这个复数所代表的几何意义就完全只有旋转了. 所留下来的部分就只有纯粹的旋转矩阵:

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

如果我们想让 2D 空间中任意一个向量 $\mathbf{v}$ 旋转 θ 度，那么我们就可以使用这个矩阵对 $\mathbf{v}$ 进行变换:

> Theorem 1: 2D 旋转公式（矩阵型）
>
> $$\mathbf{v}' = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \mathbf{v}$$

注意，其实 $\begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix}$ 这个旋转矩阵如果写成复数形式的话就是 $\cos(\theta) +$

$i\sin(\theta)$.

如果我们将向量 $\mathbf{v} = \begin{bmatrix} x \\ y \end{bmatrix}$ 看作是一个复数 v，其中实部为 x，虚部为 y．那么，我们可以构造一个复数 $z = \cos(\theta) + i\sin(\theta)$，并将它与 v 相乘来进行旋转．旋转 θ 度之后的向量 v' 可以用等价的复数乘法来表示：

Theorem 2: 2D 旋转公式（复数积型）

$$\begin{aligned} v' &= zv \\ &= (\cos(\theta) + i\sin(\theta))v \end{aligned}$$

1.3.1 复数的极坐标型

$\cos(\theta) + i\sin(\theta)$ 还可以进行下一步的变形．根据欧拉公式（Euler's Formula），

$$\cos(\theta) + i\sin(\theta) = e^{i\theta}$$

欧拉公式的证明在微积分的课程中应该都会涉及到，这里就不作证明了．有了这一个等式，我们能将复数表示为，

$$\begin{aligned} z &= \|z\| \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \\ &= \|z\|(\cos(\theta) + i\sin(\theta)) \\ &= \|z\|e^{i\theta} \end{aligned}$$

如果我们定义 $r = \|z\|$，我们就得到了复数的极坐标形式：

$$z = re^{i\theta}$$

现在复数的定义就与实部与虚部的两个分量 a, b 无关了，我们可以使用一个缩放因子 r 和旋转角度 θ 的形式来定义任意一个复数，而且它旋转与缩放的性质仍然存在．如果我们想对 2D 空间中向量 $\mathbf{v} = \begin{bmatrix} x \\ y \end{bmatrix}$ 进行旋转并缩放，我们可以类似地将这个向量

看作是一个复数 $v = x + yi$. 那么，经过旋转 θ 度，缩放 r 倍之后的向量 v' 就可以这样计算：

$$v' = re^{i\theta}v$$

如果仅需要旋转 θ 度的话，可以令缩放因子 $r = 1$，那么变换后的结果就是：

Theorem 3: 2D 旋转公式（指数型）

$$v' = e^{i\theta}v$$

这三种 2D 旋转公式其实都是等价的，根据不同的需求我们可以使用旋转的不同形态. 在我们之后讨论四元数的时候，我们也会看到非常类似的公式.

1.4 旋转的复合

如果我们有两个代表 2D 旋转的单位复数 $z_1 = \cos(\theta) + i\sin(\theta)$，$z_1 = \cos(\phi) + i\sin(\phi)$ 以及一个向量 $v = x + yi$，我们可以先对 v 进行 z_1 的旋转：

$$v' = z_1 v$$

在此基础上，我们对 v' 进行 z_2 的旋转：

$$\begin{aligned} v'' &= z_2(z_1 v) \\ &= (z_2 z_1)v \end{aligned}$$

如果我们将这两次旋转所作出的等效变换称之为 z_{net}，那么

$$v'' = (z_2 z_1)v = z_{net}v$$

$$z_{net} = z_2 z_1$$

我们之前讨论过，因为复数的相乘遵守交换律，所以

$$z_{net} = z_2 z_1 = z_1 z_2$$

如果我们尝试计算一下 z_{net}，我们会发现：

$$\begin{aligned} z_{net} &= (\cos(\theta) + i\sin(\theta))(\cos(\phi) + i\sin(\phi)) \\ &= \cos(\theta)\cos(\phi) + i(\cos(\theta)\sin(\phi)) + i(\sin(\theta)\cos(\phi)) - \sin(\theta)\sin(\phi) \\ &= (\cos(\theta)\cos(\phi) - \sin(\theta)\sin(\phi)) + (\cos(\theta)\sin(\phi) + \sin(\theta)\cos(\phi))i \end{aligned}$$

这个式子利用三角恒等式可以化简为

$$\begin{aligned} z_{net} &= (\cos(\theta) + i\sin(\theta))(\cos(\phi) + i\sin(\phi)) \\ &= \cos(\theta + \phi) + i\sin(\theta + \phi) \end{aligned}$$

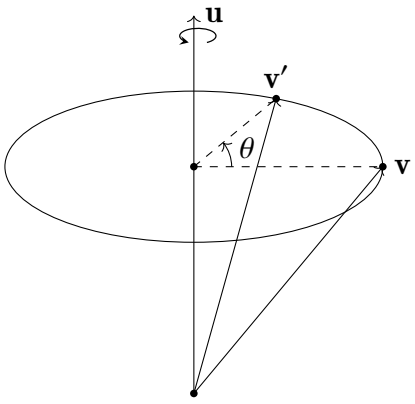
所以，当我们对两个 2D 旋转进行复合时，所得到的变换 z_{net} 仍是一个旋转，而且与施加的次序无关．这个等效变换的旋转角是 z_1 与 z_2 旋转角之和．

2 三维空间中的旋转

在讨论四元数之前，我们先来研究一下三维空间中的旋转．

表示三维空间中旋转的方法有很多种，但我们这里关注的是轴角式 (Axis-angle) 的旋转．虽然使用欧拉角的旋转很常用，但是我们知道欧拉角的的表示方法不仅会导致 Gimbal Lock 而且依赖于三个坐标轴的选定．我们使用四元数正是为了解决这个问题．我们这里所要讨论的轴角式旋转是旋转更加普遍的情况：

假设我们有一个过原点的（如果旋转轴不经过原点我们可以先将旋转轴平移到原点，进行旋转，再平移回原处）旋转轴 $\mathbf{u} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$，我们希望将一个向量 $\mathbf{v}$，沿着这个旋转轴旋转 θ 度，变换到 $\mathbf{v}'$：



注意，在这篇教程中我们将使用右手坐标系统，也就是说旋转的正方向是用右手螺旋定则定义的．你可以用**右手**拇指指向旋转轴 $\mathbf{u}$ 的正方向，其它四个手指弯曲的方向即为旋转的方向．在上图中即为逆时针方向．

在轴角的表示方法中，一个旋转的定义需要使用到四个变量：旋转轴 $\mathbf{u}$ 的 x, y, z 坐标，以及一个旋转角 θ，也就是我们一共有四个自由度 (Degree of Freedom)．这很明显是多余欧拉角的三个自由度的．实际上，任何三维中的旋转只需要三个自由度就能定义了．那么这个多余的自由度是什么呢?

其实，在我们定义旋转轴 $\mathbf{u}$ 的 x, y, z 坐标的同时，我们就定义了 $\mathbf{u}$ 的模长（长度）．然而，旋转轴 $\mathbf{u}$ 其实是个**方向**，并不是空间中一个点（向量），它并没有长度．定义三维空间中的一个**方向**只需要用到两个量就可以了（与任意两个坐标轴之间的夹角）．最简单的例子就是地球的经纬度，我们仅仅使用经度和纬度两个自由度就可以定义地球上任意一个方位．而如果我们要表示某一个方位上的特定一个点，则还需要添加海拔这个自由度．

但是，很明显旋转轴 $\mathbf{u}$ 是一个方向，不是一个点．为了消除这个额外的自由度，我们可以规定旋转轴 $\mathbf{u}$ 的模长 $\|\mathbf{u}\| = \sqrt{x^2 + y^2 + z^2} = 1$，也就是说 $\mathbf{u}$ 是一个单位向量．这样子一来，只要给出 $\mathbf{u}$ 的任意两个坐标，我们都能求出第三个坐标．比如，我们规定了 $\mathbf{u}$ 的 x, z 坐标，那么就能得到 $y = \sqrt{1 - x^2 - z^2}$．其实我们可以将模长规定为任意的常量，但是规定 $\|\mathbf{u}\| = 1$ 能为我们之后的推导带来很多的便利，这也是数学和物理中定义方向的惯例．

2.1 旋转的分解

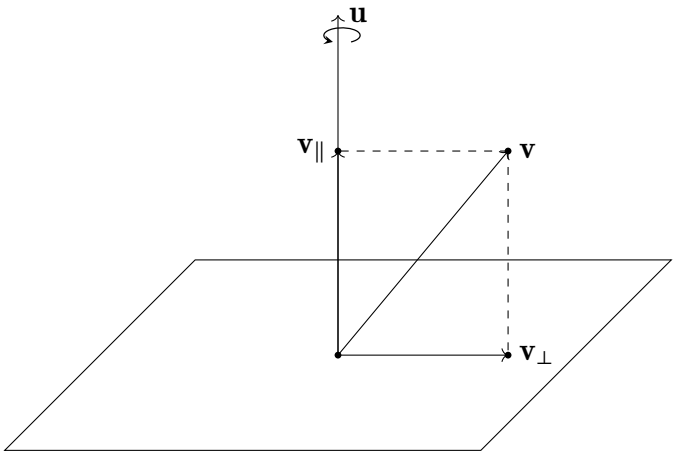
有了这个约束，我们现在就可以开始思考怎么样进行这个旋转了．首先，我们可以将 $\mathbf{v}$ 分解为平行于 $\mathbf{u}$ 以及正交（垂直）于 $\mathbf{u}$ 的两个分量，$\mathbf{v}_{\parallel}$ 和 $\mathbf{v}_{\perp}$，即：

$$\mathbf{v} = \mathbf{v}_{\parallel} + \mathbf{v}_{\perp}$$

那么旋转后的向量 $\mathbf{v}'$ 可以写为：

$$\mathbf{v}' = \mathbf{v}'_{\parallel} + \mathbf{v}'_{\perp}$$

我们可以看一下分解的示意图：



可以看到，$\mathbf{v}_{\parallel}$ 其实就是 $\mathbf{v}$ 在 $\mathbf{u}$ 上的正交投影 (Orthogonal Projection)，根据正交投影的公式，我们可以得出：

$$\begin{aligned}
\mathbf{v}_{\parallel} &= proj_{\mathbf{u}}(\mathbf{v}) \\
&= \frac{\mathbf{u}\cdot\mathbf{v}}{\mathbf{u}\cdot\mathbf{u}}\mathbf{u} \\
&= \frac{\mathbf{u}\cdot\mathbf{v}}{\|\mathbf{u}\|^2}\mathbf{u} && (\|\mathbf{u}\|^2 = \mathbf{u}\cdot\mathbf{u}) \\
&= (\mathbf{u}\cdot\mathbf{v})\mathbf{u} && (\|\mathbf{u}\| = 1)
\end{aligned}$$

因为 $\mathbf{v} = \mathbf{v}_{\parallel} + \mathbf{v}_{\perp}$，我们可以得到：

$$\begin{aligned}
\mathbf{v}_{\perp} &= \mathbf{v} - \mathbf{v}_{\parallel} \\
&= \mathbf{v} - (\mathbf{u}\cdot\mathbf{v})\mathbf{u}
\end{aligned}$$

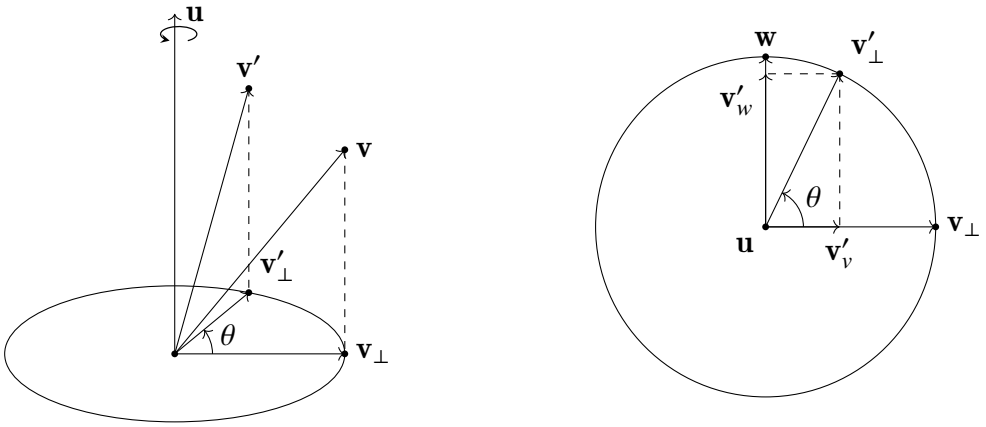
2.2 $\mathbf{v}_{\parallel}$ 的旋转

首先，我们来看一下平行于 $\mathbf{u}$ 的 $\mathbf{v}_{\parallel}$．这种情况其实非常简单，从之前的图示中就可以看到，$\mathbf{v}_{\parallel}$ 其实根本就没有被旋转，所以：

> Theorem 4: 3D 旋转公式（向量型，平行情况）
>
> 当 $\mathbf{v}_{\parallel}$ 平行于旋转轴 $\mathbf{u}$ 时，旋转 θ 角度之后的 $\mathbf{v}'_{\parallel}$ 为：
>
> $$\mathbf{v}'_{\parallel} = \mathbf{v}_{\parallel}$$

2.3 $\mathbf{v}_{\perp}$ 的旋转

接下来，我们需要处理正交于 $\mathbf{u}$ 的 $\mathbf{v}_{\perp}$．因为这两个向量是正交的，这个旋转可以看做是平面内的一个旋转．因为旋转不改变 $\mathbf{v}_{\perp}$ 的长度，所以路径是一个圆．下面是这个旋转的示意图，右侧的为顶视图：



现在，3D 的旋转就被我们转化为了 2D 平面上的旋转．由于在这个平面上我们只有一个向量 $\mathbf{v}_{\perp}$，用它来表示一个旋转是不够的，我们还需要构造一个同时正交于 $\mathbf{u}$ 和 $\mathbf{v}_{\perp}$ 的向量 $\mathbf{w}$，这个可以通过叉乘来获得：

$$\mathbf{w} = \mathbf{u} \times \mathbf{v}_{\perp}$$

注意叉乘的顺序，按照右手定则你可以发现这个新的向量 $\mathbf{w}$ 指向 $\mathbf{v}_{\perp}$ 旋转 $\pi/2$ 后的

方向，并且和 $\mathbf{v}_\perp$ 一样也处于正交于 $\mathbf{u}$ 的平面内．因为 $\|\mathbf{u}\| = 1$，我们可以发现，

$$\begin{aligned}\|\mathbf{w}\| &= \|\mathbf{u} \times \mathbf{v}_\perp\| \\ &= \|\mathbf{u}\| \cdot \|\mathbf{v}_\perp\| \cdot \sin(\pi/2) && (\pi/2 \text{ 是 } \mathbf{u} \text{ 与 } \mathbf{v}_\perp \text{ 的夹角}) \\ &= \|\mathbf{v}_\perp\|\end{aligned}$$

也就是说，$\mathbf{w}$ 和 $\mathbf{v}_\perp$ 的模长是相同的，所以，$\mathbf{w}$ 也位于圆上．有了这个新的向量 $\mathbf{w}$，就相当于我们在平面内有了两个坐标轴．我们现在可以把 $\mathbf{v}'_\perp$ 投影到 $\mathbf{w}$ 和 $\mathbf{v}_\perp$ 上，将其分解为 $\mathbf{v}'_v$ 和 $\mathbf{v}'_w$．使用一点三角学的知识我们就能得到：

$$\begin{aligned}\mathbf{v}'_\perp &= \mathbf{v}'_v + \mathbf{v}'_w \\ &= \cos(\theta)\mathbf{v}_\perp + \sin(\theta)\mathbf{w} \\ &= \cos(\theta)\mathbf{v}_\perp + \sin(\theta)(\mathbf{u} \times \mathbf{v}_\perp)\end{aligned}$$

这也就完成了旋转的第二步，我们可以得到这样一个定理：

> Theorem 5: 3D 旋转公式（向量型，正交情况）
>
> 当 $\mathbf{v}_\perp$ 正交于旋转轴 $\mathbf{u}$ 时，旋转 θ 角度之后的 $\mathbf{v}'_\perp$ 为：
>
> $$\mathbf{v}'_\perp = \cos(\theta)\mathbf{v}_\perp + \sin(\theta)(\mathbf{u} \times \mathbf{v}_\perp)$$

2.4 $\mathbf{v}$ 的旋转

将上面的两个结果组合就可以获得：

$$\begin{aligned}\mathbf{v}' &= \mathbf{v}'_\parallel + \mathbf{v}'_\perp \\ &= \mathbf{v}_\parallel + \cos(\theta)\mathbf{v}_\perp + \sin(\theta)(\mathbf{u} \times \mathbf{v}_\perp)\end{aligned}$$

因为叉乘遵守分配律，

$$\begin{aligned}\mathbf{u} \times \mathbf{v}_\perp &= \mathbf{u} \times (\mathbf{v} - \mathbf{v}_\parallel) \\ &= \mathbf{u} \times \mathbf{v} - \mathbf{u} \times \mathbf{v}_\parallel \\ &= \mathbf{u} \times \mathbf{v} && (\text{当 } \mathbf{a} \text{ 平行于 } \mathbf{b} \text{ 时，} \mathbf{a} \times \mathbf{b} = \mathbf{0})\end{aligned}$$

最后，将 $\mathbf{v}_{\parallel} = (\mathbf{u}\cdot\mathbf{v})\mathbf{u}$，$\mathbf{v}_{\perp} = \mathbf{v} - (\mathbf{u}\cdot\mathbf{v})\mathbf{u}$ 代入：

$$\begin{aligned}\mathbf{v}' &= (\mathbf{u}\cdot\mathbf{v})\mathbf{u} + \cos(\theta)(\mathbf{v} - (\mathbf{u}\cdot\mathbf{v})\mathbf{u}) + \sin(\theta)(\mathbf{u}\times\mathbf{v}) \\ &= \cos(\theta)\mathbf{v} + (1-\cos(\theta))(\mathbf{u}\cdot\mathbf{v})\mathbf{u} + \sin(\theta)(\mathbf{u}\times\mathbf{v})\end{aligned}$$

这样我们就得到了一般形式的旋转公式：

> Theorem 6: 3D 旋转公式（向量型，一般情况，也叫做「Rodrigues' Rotation Formula」）
> 3D 空间中任意一个 $\mathbf{v}$ 沿着单位向量 $\mathbf{u}$ 旋转 θ 角度之后的 $\mathbf{v}'$ 为：
>
> $$\mathbf{v}' = \cos(\theta)\mathbf{v} + (1-\cos(\theta))(\mathbf{u}\cdot\mathbf{v})\mathbf{u} + \sin(\theta)(\mathbf{u}\times\mathbf{v})$$

虽然最后结果看起来很复杂，但我们很快就能看到四元数与上面这些公式之间的联系.

3 四元数

终于，我们可以开始讨论四元数与旋转之间的关系了．四元数的定义和复数非常类似，唯一的区别就是四元数一共有三个虚部，而复数只有一个．所有的四元数 $q \in \mathbb{H}$（H 代表四元数的发现者 William Rowan Hamilton）都可以写成下面这种形式：

$$q = a + bi + cj + dk \quad (a,b,c,d \in \mathbb{R})$$

其中：

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

上面这个看似简单的公式就决定了四元数的一切性质.

与复数类似，因为四元数其实就是对于基 $\{1,i,j,k\}$ 的线性组合，四元数也可以写成

向量的形式：

$$q = \begin{bmatrix} a \\ b \\ c \\ d \end{bmatrix}$$

除此之外，我们经常将四元数的实部与虚部分开，并用一个三维的向量来表示虚部，将它表示为标量和向量的有序对形式：

$$q = [s, \mathbf{v}] \quad \left(\mathbf{v} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}, s, x, y, z \in \mathbb{R}\right)$$

3.1 性质

3.1.1 模长（范数）

仿照复数的定义，我们可以暂时将一个四元数 $q = a + bi + cj + dk$ 的模长（或者说范数）定义为：

$$\|q\| = \sqrt{a^2 + b^2 + c^2 + d^2}$$

如果用标量向量有序对的形式进行表示的话，$q = [s, \mathbf{v}]$ 的模长为：

$$\begin{aligned} \|q\| &= \sqrt{s^2 + \|\mathbf{v}\|^2} \\ &= \sqrt{s^2 + \mathbf{v} \cdot \mathbf{v}} \qquad (\mathbf{v} \cdot \mathbf{v} = \|\mathbf{v}\|^2) \end{aligned}$$

显然，四元数的模长很难用几何的方法来进行展示，因为它代表的是一个四维的长度．但是，和高维向量的模长一样，这只是类比复数模长进行衍生定义的结果，你只需要将它理解为一个定义就可以了．

3.1.2 四元数加法和减法

与复数类似，四元数的加法只需要将分量相加就可以了．如果我们有两个四元数 $q_1 = a + bi + cj + dk$，$q_2 = e + fi + gj + hk$，那么它们的和为：

$$\begin{aligned} q_1 + q_2 &= a + bi + cj + dk + e + fi + gj + hk \\ &= (a + e) + (b + f)i + (c + g)j + (d + h)k \end{aligned}$$

减法也是同理，只要将加号改为减号就可以了：

$$q_1 + q_2 = (a - e) + (b - f)i + (c - g)j + (d - h)k$$

如果四元数是以标量向量有序对形式定义的，比如说 $q_1 = [s, \mathbf{v}]$，$q_2 = [t, \mathbf{u}]$，那么：

$$q_1 \pm q_2 = [s \pm t, \mathbf{v} \pm \mathbf{u}]$$

3.1.3 标量乘法

如果我们有一个四元数 $q = a + bi + cj + dk$ 和一个标量 s，那么它们的乘积为：

$$\begin{aligned} sq &= s(a + bi + cj + dk) \\ &= sa + sbi + scj + sdk \end{aligned}$$

四元数与标量的乘法是遵守交换律的，也就是说 $sq = qs$．

3.1.4 四元数乘法

四元数之间的乘法比较特殊，它们是不遵守交换律的，也就是说一般情况下 $q_1q_2 \neq q_2q_1$．这也就有了左乘和右乘的区别．如果是 q_1q_2，那么我们就说「q_2 左乘以 q_1」，如果是 q_2q_1，那我们就说「q_2 右乘以 q_1」．除了交换律之外，我们经常使用的结合律和分配律在四元数内都是成立的．

那么，如果有两个四元数 $q_1 = a + bi + cj + dk$ 和 $q_2 = e + fi + gj + hk$，那么它们的

乘积为：

$$\begin{aligned} q_1q_2 &= (a+bi+cj+dk)(e+fi+gj+hk) \\ &= ae+afi+agj+ahk+ \\ &\quad bei+bfi^2+bgij+bhik+ \\ &\quad cej+cfji+cgj^2+chjk+ \\ &\quad dek+dfki+dgkj+dhk^2 \end{aligned}$$

这样乘法最终的结果显然非常凌乱，但是我们可以根据 $i^2=j^2=k^2=ijk=-1$ 这个公式来化简．通过这个公式我们可以知道：

$$\begin{aligned} ijk &= -1 \\ iijk &= -i && \text{（等式两边同时左乘以 } i\text{）} \\ -jk &= -i && (ii=i^2=-1) \\ jk &= i \end{aligned}$$

同理，

$$\begin{aligned} ijk &= -1 \\ ijkk &= -k && \text{（等式两边同时右乘以 } i\text{）} \\ -ij &= -k \\ ij &= k \end{aligned}$$

利用 $ij=k$ 这个公式，我们可以继续推导：

$$\begin{aligned} ij &= k \\ ijj &= kj && \text{（等式两边同时右乘以 } j\text{）} \\ -i &= kj \\ kj &= -i \end{aligned}$$

从这里我们就已经可以发现交换律不成立了，因为 $jk\neq kj$．按照类似的步骤进行推导，我们可以得出这样一个表格：

×	1	i	j	k
1	1	i	j	k
i	i	-1	k	$-j$
j	j	$-k$	-1	i
k	k	j	$-i$	-1

表格最左列中一个元素右乘以顶行中一个元素的结果就位于这两个元素行列的交叉处．比如说 $ji = -k$．用颜色标记的格子代表着乘法交换律不成立．

利用这个表格，我们就能进一步化简四元数乘积的结果：

$$\begin{aligned} q_1 q_2 = {} & ae + afi + agj + ahk + \\ & bei - bf + bgk - bhj + \\ & cej - cfk - cg + chi + \\ & dek + dfj - dgi - dh \\ = {} & (ae - bf - cg - dh) + \\ & (be + af - dg + ch)i \\ & (ce + df + ag - bh)j \\ & (de - cf + bg + ah)k \end{aligned}$$

3.1.5 矩阵形式

可以看到，四元数的相乘其实也是一个线性组合，我们也可以写成矩阵的形式：

$$q_1 q_2 = \begin{bmatrix} a & -b & -c & -d \\ b & a & -d & c \\ c & d & a & -b \\ d & -c & b & a \end{bmatrix} \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix}$$

注意，这个矩阵所做出的变换等价于**左乘** q_1．因为四元数不符合交换律，所以**右乘** q_1 的变换是一个不同的矩阵，它可以使用完全相同的方法推导而得，这里我直接给

出结果．下面这个矩阵所做出的变换等价于**右乘** q_1：

$$q_2 q_1 = \begin{bmatrix} a & -b & -c & -d \\ b & a & d & -c \\ c & -d & a & b \\ d & c & -b & a \end{bmatrix} \begin{bmatrix} e \\ f \\ g \\ h \end{bmatrix}$$

3.1.6 Graßmann 积

回到之前的结果，我重新整理了一下乘积中的每一项，观察一下，看看能不能发现什么规律：

$$\begin{aligned} q_1 q_2 = (ae - (bf + cg + dh)) + \\ (be + af + ch - dg)i \\ (ce + ag + df - bh)j \\ (de + ah + bg - cf)k \end{aligned}$$

注意，如果令 $\mathbf{v} = \begin{bmatrix} b \\ c \\ d \end{bmatrix}, \mathbf{u} = \begin{bmatrix} f \\ g \\ h \end{bmatrix}$，那么

$$\begin{aligned} \mathbf{v} \cdot \mathbf{u} &= bf + cg + dh \\ \mathbf{v} \times \mathbf{u} &= \begin{vmatrix} \mathbf{i} & \mathbf{j} & \mathbf{k} \\ b & c & d \\ f & g & h \end{vmatrix} \\ &= (ch - bg)\mathbf{i} - (bh - df)\mathbf{j} + (bg - cf)\mathbf{k} \end{aligned}$$

现在结果应该就非常清楚了，如果使用标量向量有序对形式来表示，$q_1 q_2$ 的结果可以用向量点乘和叉乘的形式表示出来（其实按照历史的顺序来说叉乘是在这里被定义的）：

$$q_1 q_2 = [ae - \mathbf{v} \cdot \mathbf{u}, a\mathbf{u} + e\mathbf{v} + \mathbf{v} \times \mathbf{u}]$$

这个结果也被叫做 Graßmann 积 (Graßmann Product)，一般来说：

> Theorem 7: Graßmann 积
>
> 对任意四元数 $q_1 = [s, \mathbf{v}], q_2 = [t, \mathbf{u}]$，$q_1 q_2$ 的结果是
>
> $$q_1 q_2 = [st - \mathbf{v} \cdot \mathbf{u}, s\mathbf{u} + t\mathbf{v} + \mathbf{v} \times \mathbf{u}]$$

如果你还记得之前推导 3D 旋转公式时的结果，你应该就能注意到上面这个定理会成为将四元数与旋转联系起来的关键.

3.1.7 纯四元数

在我们正式进入四元数的讨论之前，我们还需要定义一种特殊的四元数. 如果一个四元数能写成这样的形式：

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

那我们则称 v 为一个纯四元数 (Pure Quaternion)，即仅有虚部的四元数. 因为纯四元数仅由虚部的 3D 向量决定，我们可以将任意的 3D 向量转换为纯四元数. 在本文中，如果一个 3D 向量为 $\mathbf{v}$，那么不加粗、没有上标的 v 则为它对应的纯四元数 $v = [0, \mathbf{v}]$.

纯四元数有一个非常棒的特性，如果有两个纯四元数 $v = [0, \mathbf{v}], u = [0, \mathbf{u}]$，那么

$$\begin{aligned} vu &= [0 - \mathbf{v} \cdot \mathbf{u}, 0 + \mathbf{v} \times \mathbf{u}] \\ &= [-\mathbf{v} \cdot \mathbf{u}, \mathbf{v} \times \mathbf{u}] \end{aligned}$$

我们在之后还需要证明更多有关纯四元数的定理，但这一条暂时就够用了.

3.1.8 逆和转置

因为四元数是不遵守交换律的，我们通常不会将两个四元数相除写为 $\frac{p}{q}$ 的形式. 取而代之的是将乘法的逆运算定义为 pq^{-1} 或者 $q^{-1}p$，注意它们的结果一般是不同的.

其中，q^{-1} 是 q 的逆，我们规定：

$$qq^{-1} = q^{-1}q = 1 \quad (q \neq 0)$$

这也就是说：

$$(pq)q^{-1} = p(qq^{-1}) = p \cdot 1 = p$$

$$q^{-1}(qp) = (q^{-1}q)p = 1 \cdot p = p$$

所以，右乘 q 的逆运算为右乘 q^{-1}，左乘 q 的逆运算为左乘 q^{-1}，这个与矩阵的性质非常相似.

显然，要在无数的四元数中寻找一个满足 $qq^{-1} = q^{-1}q = 1$ 的 q^{-1} 是非常困难的，但是实际上我们可以使用四元数共轭的一些性质来获得 q^{-1}.

我们定义，一个四元数 $q = a + bi + ck + dj$ 的共轭为 $q^* = a - bi - cj - dk$. 如果用标量向量有序对的形式来定义的话，$q = [s, \mathbf{v}]$ 的共轭为 $q^* = [s, -\mathbf{v}]$. 共轭四元数的一个非常有用的性质就是：

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

可以看到，这最终的结果是一个实数，而这个实数正是四元数模长的平方：

$$\begin{aligned} qq^* &= [s^2 + \mathbf{v} \cdot \mathbf{v}, \mathbf{0}] \\ &= s^2 + x^2 + y^2 + z^2 \\ &= \|q\|^2 \end{aligned}$$

因为 $(q^*)^* = [s, -(-\mathbf{v})] = [s, \mathbf{v}] = q$,

$$\begin{aligned} q^*q &= (q^*)(q^*)^* \\ &= \|q^*\|^2 \\ &= s^2 + x^2 + y^2 + z^2 \\ &= \|q\|^2 \\ &= qq^* \end{aligned}$$

所以我们得到，$q^*q = qq^*$. 这个特殊的乘法是遵守交换律的.

如果你还记得之前四元数逆的定义的话：

$$
\begin{aligned}
qq^{-1} &= 1 \\
q^*qq^{-1} &= q^* && \text{（等式两边同时左乘以 } q^*\text{）} \\
(q^*q)q^{-1} &= q^* \\
\|q\|^2 \cdot q^{-1} &= q^* && \text{（} q^*q = \|q\|^2\text{）} \\
q^{-1} &= \frac{q^*}{\|q\|^2}
\end{aligned}
$$

用这种办法寻找一个四元数的逆会非常高效，我们只需要将一个四元数的虚部改变符号，除以它的模长就能获得这个四元数的逆了．如果 $\|q\| = 1$，也就是说 q 是一个单位四元数 (Unit Quaternion)，那么：

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

3.2 四元数与 3D 旋转

有了这些知识，我们就能开始讨论四元数与 3D 旋转之间的关联了．

回忆一下我们之前讨论过的内容：如果我们需要将一个向量 $\mathbf{v}$ 沿着一个用单位向量所定义的旋转轴 $\mathbf{u}$ 旋转 θ 度，那么我们可以将这个向量 $\mathbf{v}$ 拆分为正交于旋转轴的 $\mathbf{v}_\perp$ 以及平行于旋转轴的 $\mathbf{v}_\parallel$．我们可以对这两个分向量分别进行旋转，获得 $\mathbf{v}'_\perp$ 和 $\mathbf{v}'_\parallel$．将它们相加就是 $\mathbf{v}$ 旋转之后的结果 $\mathbf{v}' = \mathbf{v}'_\parallel + \mathbf{v}'_\perp$．

我们可以将这些向量定义为纯四元数：

$$
\begin{aligned}
v &= [0, \mathbf{v}] & v' &= [0, \mathbf{v}'] \\
v_\perp &= [0, \mathbf{v}_\perp] & v'_\perp &= [0, \mathbf{v}'_\perp] \\
v_\parallel &= [0, \mathbf{v}_\parallel] & v'_\parallel &= [0, \mathbf{v}'_\parallel] \\
u &= [0, \mathbf{u}]
\end{aligned}
$$

那么我们就能得到：

$$
v = v_\parallel + v_\perp \qquad\qquad v' = v'_\parallel + v'_\perp
$$

和之前一样，我们在这里也分开讨论 $v_\perp$ 和 $v_\parallel$ 的情况．

3.2.1 $v_\perp$ 的旋转

我们首先讨论正交于旋转轴的 $v_\perp$．我们之前推导过，如果一个向量 $\mathbf{v}_\perp$ 正交于旋转轴 $\mathbf{u}$，那么：

$$\mathbf{v}'_\perp = \cos(\theta)\mathbf{v}_\perp + \sin(\theta)(\mathbf{u} \times \mathbf{v}_\perp)$$

我们可以很容易地将前面的 $\mathbf{v}'_\perp$ 和 $\mathbf{v}_\perp$ 替换为 $v'_\perp$ 和 $v_\perp$，但是我们仍遗留下来了一个 $\mathbf{u} \times \mathbf{v}_\perp$．幸运的是，利用四元数的性质，我们可以将它写成四元数积的形式．我们之前推导过，如果有两个纯四元数 $v = [0, \mathbf{v}], u = [0, \mathbf{u}]$，那么 $vu = [-\mathbf{v} \cdot \mathbf{u}, \mathbf{v} \times \mathbf{u}]$．类似地，

$$uv_\perp = [-\mathbf{u} \cdot \mathbf{v}_\perp, \mathbf{u} \times \mathbf{v}_\perp]$$

因为 $\mathbf{v}_\perp$ 正交于 $\mathbf{u}$，所以 $\mathbf{u} \cdot \mathbf{v}_\perp = 0$，也就是说

$$\begin{aligned} uv_\perp &= [0, \mathbf{u} \times \mathbf{v}_\perp] \\ &= \mathbf{u} \times \mathbf{v}_\perp \end{aligned}$$

注意，$uv_\perp$ 同样是一个纯四元数．我们离最终的结论已经很近了！将这个等式以及之前定义的纯四元数代入，我们就能获得：

$$v'_\perp = \cos(\theta)v_\perp + \sin(\theta)(uv_\perp)$$

因为四元数的乘法遵守分配律，我们可以继续变换这个等式：

$$\begin{aligned} v'_\perp &= \cos(\theta)v_\perp + \sin(\theta)(uv_\perp) \\ &= (\cos(\theta) + \sin(\theta)u)v_\perp \end{aligned}$$

你应该可以注意到，$(\cos(\theta) + \sin(\theta)u)$ 其实也是一个四元数．到此为止，我们已经将旋转与四元数的积联系起来了．如果令 $q = \cos(\theta) + \sin(\theta)u$，我们能得到：

$$v'_\perp = qv_\perp$$

如果我们能构造一个 q，那么我们就能完成这个旋转了．我们可以对 q 继续进行变

形：

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

也就是说，如果旋转轴 $\mathbf{u}$ 的坐标为 $\begin{bmatrix} u_x \\ u_y \\ u_z \end{bmatrix}$，旋转角为 θ，那么完成这一旋转所需要的四元数 q 可以构造为：

$$q = \cos(\theta) + \sin(\theta)u_x + \sin(\theta)u_y + \sin(\theta)u_z$$

这样我们就完成了对 $v_\perp$ 的旋转，我们可以得到下面这个定理：

> Theorem 8: 3D 旋转公式（四元数型，正交情况）
>
> 当 $\mathbf{v}_\perp$ 正交于旋转轴 $\mathbf{u}$ 时，旋转 θ 角度之后的 $\mathbf{v}'_\perp$ 可以使用四元数乘法来获得．令 $v_\perp = [0, \mathbf{v}_\perp]$，$q = [\cos(\theta), \sin(\theta)\mathbf{u}]$，那么：
>
> $$v'_\perp = qv_\perp$$

这个四元数 q 其实还有一个性质，我们可以注意到：

$$\begin{aligned} \|q\| &= \sqrt{\cos^2(\theta) + (\sin(\theta)\mathbf{u} \cdot \sin(\theta)\mathbf{u})} \\ &= \sqrt{\cos^2(\theta) + \sin^2(\theta)(\mathbf{u} \cdot \mathbf{u})} \\ &= \sqrt{\cos^2(\theta) + \sin^2(\theta)(\|\mathbf{u}\|^2)} && (\mathbf{u} \cdot \mathbf{u} = \|\mathbf{u}\|^2) \\ &= \sqrt{\cos^2(\theta) + \sin^2(\theta)} && (\|\mathbf{u}\| = 1) \\ &= 1 && (\text{三角恒等式}) \end{aligned}$$

所以，我们构造出来的这个 q 其实是一个单位四元数．我们之前讨论过，复数的乘法的几何意义可以理解为缩放与旋转的复合．这个性质可以类比到四元数，而因为 $\|q\| = 1$，它所代表的变换并不会对原向量进行缩放，是一个纯旋转．

3.2.2 $v_{\parallel}$ 的旋转

接下来是平行于旋转轴的 $v_{\parallel}$．我们之前讨论过，如果一个向量 $\mathbf{v}_{\parallel}$ 平行于 $\mathbf{u}$，那么旋转不会对它作出任何的变换，也就是说：

> Theorem 9: 3D 旋转公式（四元数型，平行情况）
>
> 当 $\mathbf{v}_{\parallel}$ 平行于旋转轴 $\mathbf{u}$ 时，旋转 θ 角度之后的 $\mathbf{v}'_{\parallel}$ 用四元数可以写为：
>
> $$v'_{\parallel} = v_{\parallel}$$

3.2.3 v 的旋转

有了这些知识，我们能够获得一般情况下 v' 的结果了：

$$\begin{aligned} v' &= v'_{\parallel} + v'_{\perp} \\ &= v_{\parallel} + qv_{\perp} \qquad \text{（其中 } q = [\cos(\theta), \sin(\theta)\mathbf{u}]\text{）} \end{aligned}$$

我们当然可以像以前那样将 $v_{\parallel}$ 和 $v_{\perp}$ 拆开，继续进行化简．但是这里我们不会这么做，因为我们有更好的办法来进一步化简它．

在进一步化简之前，我们需要证明几个引理：

> Lemma 1
>
> 如果 $q = [\cos(\theta), \sin(\theta)\mathbf{u}]$，而且 $\mathbf{u}$ 为单位向量，那么 $q^2 = qq = [\cos(2\theta), \sin(2\theta)\mathbf{u}]$.

Proof. 这个引理的证明很简单，只需要使用 Graßmann 积和一些三角恒等式就可以了：

$$\begin{aligned} q^2 &= [\cos(\theta), \sin(\theta)\mathbf{u}] \cdot [\cos(\theta), \sin(\theta)\mathbf{u}] \\ &= [\cos^2(\theta) - (\sin(\theta)\mathbf{u} \cdot \sin(\theta)\mathbf{u}), (\cos(\theta)\sin(\theta) + \sin(\theta)\cos(\theta))\mathbf{u} + (\sin(\theta)\mathbf{u} \times \sin(\theta)\mathbf{u})] \\ &= [\cos^2(\theta) - \sin^2(\theta)\|\mathbf{u}\|^2, 2\sin(\theta)\cos(\theta)\mathbf{u} + \mathbf{0}] \\ &= [\cos^2(\theta) - \sin^2(\theta), 2\sin(\theta)\cos(\theta)\mathbf{u}] \\ &= [\cos(2\theta), \sin(2\theta)\mathbf{u}] \end{aligned}$$

□

其实这个引理的几何意义就是，如果绕着同一个轴 $\mathbf{u}$ 连续旋转 θ 度两次，那么所做出的变换等同于直接绕着 $\mathbf{u}$ 旋转 2θ 度.

有了这个引理，我们再来回忆一下四元数逆的定义：

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

现在，我们就能够对原本的旋转公式进行变形了：

$$\begin{aligned} v' &= v_{\parallel} + qv_{\perp} && (q = [\cos(\theta), \sin(\theta)\mathbf{u}]) \\ &= 1 \cdot v_{\parallel} + qv_{\perp} \\ &= pp^{-1}v_{\parallel} + ppv_{\perp} && (\text{令 } q = p^2, \text{ 则 } p = [\cos(\frac{1}{2}\theta), \sin(\frac{1}{2}\theta)\mathbf{u}]) \end{aligned}$$

在这里，我们引入了一个新的四元数 $p = [\cos(\frac{1}{2}\theta), \sin(\frac{1}{2}\theta)\mathbf{u}]$. 根据刚刚证明的引理，我们可以验证：

$$\begin{aligned} pp &= p^2 \\ &= [\cos(2 \cdot \frac{1}{2}\theta), \sin(2 \cdot \frac{1}{2}\theta)\mathbf{u}] \\ &= [\cos(\theta), \sin(\theta)\mathbf{u}] = q \end{aligned}$$

你应该能够注意到，和 q 一样，$\|p\| = 1$，p 也是一个单位四元数，也就是说：

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

将这个结果代入之前的等式中：

$$\begin{aligned} v' &= pp^{-1}v_{\parallel} + ppv_{\perp} \\ &= pp^*v_{\parallel} + ppv_{\perp} \end{aligned}$$

这个公式还能进行进一步的化简，但在继续之前我们还需要再证明两个引理：

Lemma 2

假设 $v_{\parallel} = [0, \mathbf{v}_{\parallel}]$ 是一个纯四元数, 而 $q = [\alpha, \beta\mathbf{u}]$, 其中 $\mathbf{u}$ 是一个单位向量, $\alpha, \beta \in \mathbb{R}$. 在这种条件下，如果 $\mathbf{v}_{\parallel}$ 平行于 $\mathbf{u}$，那么 $qv_{\parallel} = v_{\parallel}q$

Proof. 这个引理的证明同样需要用到 Graßmann 积. 我们先计算等式左边:

$$
\begin{aligned}
LHS &= qv_{\parallel} \\
&= [\alpha, \beta\mathbf{u}] \cdot [0, \mathbf{v}_{\parallel}] \\
&= [0 - \beta\mathbf{u} \cdot \mathbf{v}_{\parallel},\ \alpha\mathbf{v}_{\parallel} + \mathbf{0} + \beta\mathbf{u} \times \mathbf{v}_{\parallel}] \\
&= [-\beta\mathbf{u} \cdot \mathbf{v}_{\parallel},\ \alpha\mathbf{v}_{\parallel}] && (\mathbf{v}_{\parallel} \text{ 平行于 } \mathbf{u},\ \text{所以 } \beta\mathbf{u} \times \mathbf{v}_{\parallel} = \mathbf{0})
\end{aligned}
$$

接下来计算等式的右边:

$$
\begin{aligned}
RHS &= v_{\parallel}q \\
&= [0, \mathbf{v}_{\parallel}] \cdot [\alpha, \beta\mathbf{u}] \\
&= [0 - \mathbf{v}_{\parallel} \cdot \beta\mathbf{u},\ \mathbf{0} + \alpha\mathbf{v}_{\parallel} + \mathbf{v}_{\parallel} \times \beta\mathbf{u}] \\
&= [-\mathbf{v}_{\parallel} \cdot \beta\mathbf{u},\ \alpha\mathbf{v}_{\parallel}] && (\mathbf{v}_{\parallel} \text{ 平行于 } \mathbf{u},\ \text{所以 } \mathbf{v}_{\parallel} \times \beta\mathbf{u} = \mathbf{0}) \\
&= [-\beta\mathbf{u} \cdot \mathbf{v}_{\parallel},\ \alpha\mathbf{v}_{\parallel}] = LHS && (\text{点乘遵守交换律})
\end{aligned}
$$

□

下面是第二个引理:

Lemma 3

假设 $v_{\perp} = [0, \mathbf{v}_{\perp}]$ 是一个纯四元数, 而 $q = [\alpha, \beta\mathbf{u}]$, 其中 $\mathbf{u}$ 是一个单位向量, $\alpha, \beta \in \mathbb{R}$. 在这种条件下, 如果 $\mathbf{v}_{\perp}$ 正交于 $\mathbf{u}$, 那么 $qv_{\perp} = v_{\perp}q^{*}$

Proof. 它的证明与之前的引理完全类似:

$$
\begin{aligned}
LHS &= qv_{\perp} \\
&= [\alpha, \beta\mathbf{u}] \cdot [0, \mathbf{v}_{\perp}] \\
&= [0 - \beta\mathbf{u} \cdot \mathbf{v}_{\perp},\ \alpha\mathbf{v}_{\perp} + \mathbf{0} + \beta\mathbf{u} \times \mathbf{v}_{\perp}] \\
&= [0,\ \alpha\mathbf{v}_{\perp} + \beta\mathbf{u} \times \mathbf{v}_{\perp}] && (\mathbf{v}_{\perp} \text{ 正交于 } \mathbf{u},\ \text{所以 } \beta\mathbf{u} \cdot \mathbf{v}_{\perp} = 0)
\end{aligned}
$$

$$
\begin{aligned}
RHS &= v_\perp q^* \\
&= [0, \mathbf{v}_\perp] \cdot [\alpha, -\beta\mathbf{u}] \\
&= [0 + \mathbf{v}_\perp \cdot \beta\mathbf{u},\ \mathbf{0} + \alpha\mathbf{v}_\perp + \mathbf{v}_\perp \times (-\beta\mathbf{u})] \\
&= [0,\ \alpha\mathbf{v}_\perp + \mathbf{v}_\perp \times (-\beta\mathbf{u})] && (\mathbf{v}_\perp \text{ 正交于 } \mathbf{u},\ \text{所以 } \mathbf{v}_\perp \cdot \beta\mathbf{u} = 0) \\
&= [0,\ \alpha\mathbf{v}_\perp - (-\beta\mathbf{u}) \times \mathbf{v}_\perp] && (\mathbf{a} \times \mathbf{b} = -\mathbf{b} \times \mathbf{a}) \\
&= [0,\ \alpha\mathbf{v}_\perp + \beta\mathbf{u} \times \mathbf{v}_\perp] = LHS
\end{aligned}
$$

□

现在，我们就能对之前的公式做出最后的变形了：

$$
\begin{aligned}
v' &= pp^* v_\parallel + ppv_\perp \\
&= pv_\parallel p^* + pv_\perp p^* \\
&= p(v_\parallel + v_\perp)p^*
\end{aligned}
$$

显然，$(v_\parallel + v_\perp)$ 其实就是 v，所以，

$$
v' = pvp^*
$$

到此为止，我们就解开了四元数与 3D 旋转之间的谜题．虽然我们之前将 v 划分成了两个分量，但是推导的结果其实并没有包含 $v_\parallel$ 和 $v_\perp$．3D 空间中任意一个旋转都能够用三个四元数相乘的形式表达出来．虽然推导的结果可能比较冗长，但是最终的结果非常简短．我们可以总结为一个定理：

> Theorem 10: 3D 旋转公式（四元数型，一般情况）
>
> 任意向量 $\mathbf{v}$ 沿着以单位向量定义的旋转轴 $\mathbf{u}$ 旋转 θ 度之后的 $\mathbf{v}'$ 可以使用四元数乘法来获得．令 $v = [0, \mathbf{v}]$，$q = [\cos(\frac{1}{2}\theta), \sin(\frac{1}{2}\theta)\mathbf{u}]$，那么：
>
> $$v' = qvq^* = qvq^{-1}$$

换句话说，如果我们有 $q = [\cos(\theta), \sin(\theta)\mathbf{u}]$，那么 $v' = qvq^*$ 可以将 $\mathbf{v}$ 沿着 $\mathbf{u}$ 旋转 2θ 度．

虽然这个公式非常简洁，但是它并不是那么直观．如果你想了解它真正的含义的话，还需要将它还原到变形之前的式子：

$$v' = qvq^* = qq^*v_{\parallel} + qqv_{\perp} = v_{\parallel} + q^2 v_{\perp}$$

这也就是说，qvq^* 这个变换，对 v 平行于旋转轴的分量 $v_{\parallel}$ 实施的变换是 qq^*，这两个变换完全抵消了，也就是没有旋转．而对于正交于旋转轴的分量 $v_{\perp}$ 则实施的是两次变换 $q^2 = qq$，将它旋转 $\frac{\theta}{2} + \frac{\theta}{2} = \theta$ 度．

实际上，这个公式也是与上一章推导的「Rodrigues' Rotation Formula」完全等价的．如果你感兴趣的话，可以试试证明下面这个等式，确认我们的推导是没有错误的（提示：证明可能会用到 $\mathbf{a} \times (\mathbf{b} \times \mathbf{c}) = (\mathbf{a} \cdot \mathbf{c})\mathbf{b} - (\mathbf{a} \cdot \mathbf{b})\mathbf{c}$ 这个公式）：

$$qvq^* = [0,\ \cos(\theta)\mathbf{v} + (1 - \cos(\theta))(\mathbf{u} \cdot \mathbf{v})\mathbf{u} + \sin(\theta)(\mathbf{u} \times \mathbf{v})]$$

虽然这个等式的证明可能看起来会非常麻烦，但我仍推荐您去尝试一下．除了一些三角公式，证明所需要的全部知识我们都已经涉及到了．它会让你更清楚地了解变换的整个过程以及为什么变换的结果是一个纯四元数．

因为所有的旋转四元数的实部都只是一个角度的余弦值，假设有一个单位四元数 $q = [a, \mathbf{b}]$，如果我们想要提取它所对应旋转的角度，那么我们可以直接得到：

$$\frac{\theta}{2} = \cos^{-1}(a)$$

如果想要再获得旋转轴，那么只需要将 $\mathbf{b}$ 的每一项都除以 $\sin(\frac{\theta}{2})$ 就可以了：

$$\mathbf{u} = \frac{\mathbf{b}}{\sin(\cos^{-1}(a))}$$

3.3 3D 旋转的矩阵形式

在实际的应用中，我们可能会需要将旋转与平移和缩放进行复合，所以需要用到四元数旋转的矩阵形式．它可以很容易地从上面这个公式推导出来．我们之前讨论过，

左乘一个四元数 $q = a + bi + cj + dj$ 等同于下面这个矩阵：

$$L(q) = \begin{bmatrix} a & -b & -c & -d \\ b & a & -d & c \\ c & d & a & -b \\ d & -c & b & a \end{bmatrix}$$

而右乘 q 等同于这个矩阵：

$$R(q) = \begin{bmatrix} a & -b & -c & -d \\ b & a & d & -c \\ c & -d & a & b \\ d & c & -b & a \end{bmatrix}$$

所以，我们可以利用这两个公式将 $v' = qvq^*$ 写成矩阵形式．假设 $a = \cos(\frac{1}{2}\theta)$, $b = \sin(\frac{1}{2}\theta)u_x$，$c = \sin(\frac{1}{2}\theta)u_y$，$d = \sin(\frac{1}{2}\theta)u_z$，$q = a + bi + cj + dk$，我们就能得到：

$qvq^* = L(q)R(q^*)v$ （或者 $R(q^*)L(q)v$，它们是等价的）

$$= \begin{bmatrix} a & -b & -c & -d \\ b & a & -d & c \\ c & d & a & -b \\ d & -c & b & a \end{bmatrix} \begin{bmatrix} a & b & c & d \\ -b & a & -d & c \\ -c & d & a & -b \\ -d & -c & b & a \end{bmatrix} v \qquad \text{（注意 } R(q^*) = R(q)^T\text{）}$$

$$= \begin{bmatrix} a^2+b^2+c^2+d^2 & ab-ab-cd+cd & ac+bd-ac-bd & ad-bc+bc-ad \\ ab-ab+cd-cd & b^2+a^2-d^2-c^2 & bc-ad-ad+bc & bd+ac+bd+ac \\ ac-bd-ac+bd & bc+ad+ad+bc & c^2-d^2+a^2-b^2 & cd+cd-ab-ab \\ ad+bc-bc-ad & bd-ac+bd-ac & cd+cd+ab+ab & d^2-c^2-b^2+a^2 \end{bmatrix} v$$

因为 $a^2 + b^2 + c^2 + d^2 = 1$，这个式子能化简为

$$qvq^* = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1-2c^2-2d^2 & 2bc-2ad & 2ac+2bd \\ 0 & 2bc+2ad & 1-2b^2-2d^2 & 2cd-2ab \\ 0 & 2bd-2ac & 2ab+2cd & 1-2b^2-2c^2 \end{bmatrix} v$$

这样我们就得到了 3D 旋转的矩阵形式．因为矩阵的最外圈不会对 v 进行任何变换，我们可以将它压缩成 3×3 矩阵：

Theorem 11: 3D 旋转公式（矩阵型）

任意向量 $\mathbf{v}$ 沿着以单位向量定义的旋转轴 $\mathbf{u}$ 旋转 θ 角度之后的 $\mathbf{v}'$ 可以使用矩阵乘法来获得．令 $a = \cos(\frac{1}{2}\theta)$，$b = \sin(\frac{1}{2}\theta)u_x$，$c = \sin(\frac{1}{2}\theta)u_y$，$d = \sin(\frac{1}{2}\theta)u_z$，那么：

$$\mathbf{v}' = \begin{bmatrix} 1-2c^2-2d^2 & 2bc-2ad & 2ac+2bd \\ 2bc+2ad & 1-2b^2-2d^2 & 2cd-2ab \\ 2bd-2ac & 2ab+2cd & 1-2b^2-2c^2 \end{bmatrix} \mathbf{v}$$

虽然 3D 旋转的矩阵形式可能不如四元数形式简单，但是对于大批量的变换，使用**预计算**好的矩阵是比四元数更有效率的．

3.4 旋转的复合

在这一小节里，我们来证明一下使用四元数的旋转是可以复合的．旋转的复合其实在我们之前证明 $q^2 = [\cos(2\theta), \sin(2\theta)\mathbf{u}]$ 的时候就已经涉及到一点了，但是这里我们考虑的是更一般的情况．

假设有两个表示沿着不同轴，不同角度旋转的四元数 q_1, q_2，我们先对 v 进行 q_1 的变换，再进行 q^2 的变换，变换的结果是什么呢？

我们不妨将这两次变换分步进行．首先，我们实施 q_1 的变换，变换之后的 v' 为：

$$v' = q_1 v q_1^*$$

接下来，对 v' 进行 q^2 的变换，得到 v''：

$$\begin{aligned} v'' &= q_2 v' q_2^* \\ &= q_2 q_1 v q_1^* q_2^* \end{aligned}$$

我们需要对这两个变换进行复合，写为一个等价变换的形式：

$$v'' = q_{net} v q_{net}^*$$

为了写成上面这种形式，我们还需要一个引理：

Lemma 4

对任意四元数 $q_1 = [s, \mathbf{v}]$、$q_2 = [t, \mathbf{u}]$：

$$q_1^* q_2^* = (q_2 q_1)^*$$

Proof. 仍然使用与之前类似的方法：

$$\begin{aligned} LHS &= q_1^* q_2^* \\ &= [s, -\mathbf{v}] \cdot [t, -\mathbf{u}] \\ &= [st - (-\mathbf{v}) \cdot (-\mathbf{u}),\ s(-\mathbf{u}) + t(-\mathbf{v}) + (-\mathbf{v}) \times (-\mathbf{u})] \\ &= [st - \mathbf{v} \cdot \mathbf{u}, -s\mathbf{u} - t\mathbf{v} + \mathbf{v} \times \mathbf{u}] \end{aligned}$$

$$\begin{aligned} RHS &= (q_2 q_1)^* \\ &= ([t, -\mathbf{u}] \cdot [s, -\mathbf{v}])^* \\ &= [ts - \mathbf{u} \cdot \mathbf{v},\ t\mathbf{v} + s\mathbf{u} + \mathbf{u} \times \mathbf{v}]^* \\ &= [st - \mathbf{v} \cdot \mathbf{u}, -s\mathbf{u} - t\mathbf{v} - \mathbf{u} \times \mathbf{v}] \\ &= [st - \mathbf{v} \cdot \mathbf{u}, -s\mathbf{u} - t\mathbf{v} + \mathbf{v} \times \mathbf{u}] = LHS \end{aligned}$$

□

所以，我们能得到：

$$\begin{aligned} v'' &= q_2 q_1 v q_1^* q_2^* \\ &= (q_2 q_1) v (q_2 q_1)^* \end{aligned}$$

这也就是说，$q_{net} = q_2 q_1$．注意四元数乘法的顺序，我们先进行的是 q_1 的变换，再进行 q_2 的变换．这和矩阵与函数的复合非常相似，都是从右往左叠加．

要注意的是，q_1 与 q_2 的等价旋转 q_{net} 并不是分别沿着 q_1 和 q_2 的两个旋转轴进行的两次旋转．它是沿着一个新的旋转轴进行的一次等价旋转，仅仅只有旋转的结果相同．

虽然我们讨论的是两个旋转的复合，但是它可以很容易推广到更多个旋转的复合. 比如说我们还需要进行第三个旋转 q_3，那么

$$\begin{aligned} v''' &= q_3(q_2q_1)v(q_2q_1)^*q_3^* \\ &= (q_3q_2q_1)v(q_3q_2q_1)^* \end{aligned}$$

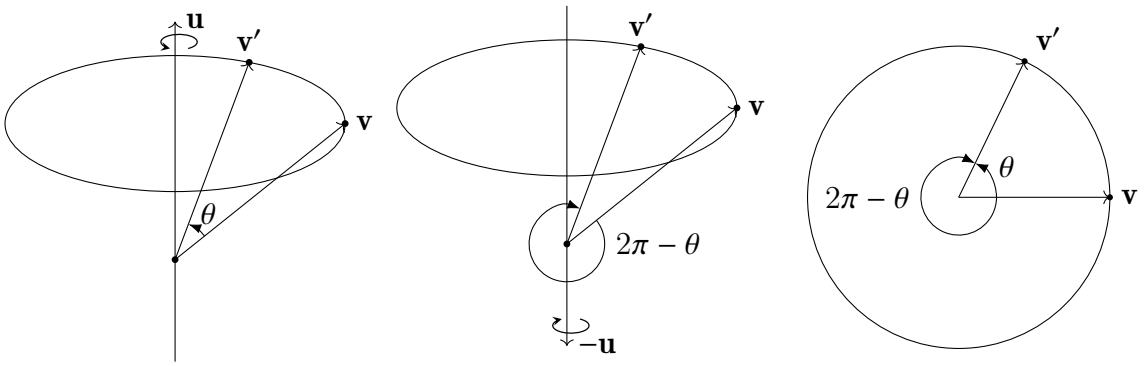
它的等价旋转就是 $q_{net} = q_3q_2q_1$.

3.5 非唯一性

四元数与 3D 旋转的关系并不是一对一的，同一个 3D 旋转可以使用两个不同的四元数来表示. 对任意的单位四元数 $q = [\cos(\frac{1}{2}\theta), \sin(\frac{1}{2}\theta)\mathbf{u}]$，$q$ 与 $-q$ 代表的是同一个旋转. 如果 q 表示的是沿着旋转轴 $\mathbf{u}$ 旋转 θ 度，那么 $-q$ 代表的是沿着相反的旋转轴 $-\mathbf{u}$ 旋转 $(2\pi - \theta)$ 度:

$$\begin{aligned} -q &= [-\cos(\frac{1}{2}\theta), -\sin(\frac{1}{2}\theta)\mathbf{u}] \\ &= [\cos(\pi - \frac{1}{2}\theta), \sin(\pi - \frac{1}{2}\theta)(-\mathbf{u})] \qquad (\cos(\pi-\theta) = -\cos(\theta), \sin(\pi-\theta) = \sin(\theta)) \end{aligned}$$

所以，这个四元数旋转的角度为 $2(\pi - \frac{1}{2}\theta) = 2\pi - \theta$. 从下面的图中我们可以看到，这两个旋转是完全等价的:



其实从四元数的旋转公式中也能推导出相同的结果:

$$(-q)v(-q)^* = (-1)^2qvq^* = qvq^*$$

所以，我们经常说单位四元数与 3D 旋转有一个「2 对 1 满射同态」(2-1 Surjective Homomorphism) 关系，或者说单位四元数双倍覆盖 (Double Cover) 了 3D 旋转. 它的

严格证明会用到一些李群的知识，但我相信这里给出的解释已经足够直观了．我不想让这个教程变得太复杂，所以在这里就不多说了．如果你感兴趣的话，可以到最后的拓展阅读中找到一些资料．

因为这个映射是满射，我们可以说所有的单位四元数都对应着一个 3D 旋转．或者说，一个四维单位超球面（也叫做 $\mathbb{S}^3$）上任意一点所对应的四元数都对应着一个 3D 旋转，这一点会在之后讨论旋转插值的时候会用到．

有一点需要注意的是，虽然 q 与 $-q$ 是两个不同的四元数，但是由于旋转矩阵中的每一项都包含了四元数两个分量的乘积，它们的旋转矩阵是完全相同的．旋转矩阵并不会出现双倍覆盖的问题．

3.6 指数形式

在讨论复数的时候，我们利用欧拉公式将 2D 的旋转写成了 $v' = e^{i\theta}v$ 这样的指数形式．实际上，我们也可以利用四元数将 3D 旋转写成类似的形式．

类似于复数的欧拉公式，四元数也有一个类似的公式．如果 $\mathbf{u}$ 是一个单位向量，那么对于单位四元数 $u = [0, \mathbf{u}]$，有：

$$e^{u\theta} = \cos(\theta) + u\sin(\theta) = \cos(\theta) + \mathbf{u}\sin(\theta)$$

这也就是说，$q = [\cos(\theta), \sin(\theta)\mathbf{u}]$ 可以使用指数表示为 $e^{u\theta}$．这个公式的证明与欧拉公式的证明非常类似，直接使用级数展开就可以了．因为它的证明与主题关系不是很大，我会把它的证明放在附录中．如果你感兴趣的话，可以翻到最后去了解一下．

注意，因为 $\mathbf{u}$ 是一个单位向量，$u^2 = [-\mathbf{u}\cdot\mathbf{u}, \mathbf{0}] = -\|\mathbf{u}\|^2 = -1$．这是与欧拉公式中的 i 是非常类似的．

有了指数型的表示方式，我们就能够将之前四元数的旋转公式改写为指数形式了：

> Theorem 12: 3D 旋转公式（指数型）
>
> 任意向量 $\mathbf{v}$ 沿着以单位向量定义的旋转轴 $\mathbf{u}$ 旋转 θ 角度之后的 $\mathbf{v}'$ 可以使用四元数的指数表示．令 $v = [0, \mathbf{v}], u = [0, \mathbf{u}]$，那么
>
> $$v' = e^{u\frac{\theta}{2}} v e^{-u\frac{\theta}{2}}$$

有了四元数的指数定义，我们就能够定义四元数的更多运算了．首先是自然对数 log，对任意单位四元数 $q = [\cos(\theta), \sin(\theta)\mathbf{u}]$：

$$\log(q) = \log(e^{u\theta}) = [0, \mathbf{u}\theta]$$

接下来是单位四元数的幂运算：

$$q^t = (e^{u\theta})^t = e^{u(t\theta)} = [\cos(t\theta), \sin(t\theta)\mathbf{u}]$$

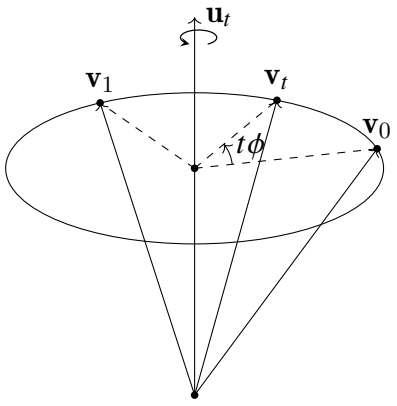
可以看到，一个单位四元数的 t 次幂等同于将它的旋转角度缩放至 t 倍，并且不会改变它的旋转轴（u 必须是单位向量，所以不能与 t 结合）．这些运算会在之后讨论四元数插值时非常有用．

4 四元数插值

有了这么多的前置知识，我们就能开始讨论四元数的插值 (Interpolation) 了．与其它的旋转表示形式不同，四元数的一些性质让它的插值变得非常简单．

假设有两个旋转变换 $q_0 = [\cos(\theta_0), \sin(\theta_0)\mathbf{u}_0]$ 和 $q_1 = [\cos(\theta_1), \sin(\theta_1)\mathbf{u}_1]$，我们希望找出一些中间变换 q_t，让初始变换 q_0 能够平滑地过渡到最终变换 q_1．t 的取值可以是 $t \in [0, 1]$．当 $t = 0$ 时 q_t 等同于初始变换 q_0，而 $t = 1$ 时 q_t 等同于最终变换 q_1．

由于插值的对象是两个**变换**，想象起来可能非常困难，我们不妨假设 3D 空间中有任意一个向量 v．那么 q_0 会将 v 变换到 $v_0 = q_0 v q_0^*$，而 q_1 会将 v 变换到 $v_1 = q_1 v q_1^*$．我们需要找出中间向量 $v_t = q_t v q_t^*$ 所对应的变换 q_t，使 v 旋转到 v_0 与 v_1 中间的某个位置 v_t：



我们可以看到，这个**旋转的变化量**其实对应的仍是一个旋转，它将由 q_0 变换到 v_0 的向量进一步旋转到 v_t．这个旋转拥有某一个固定的旋转轴 $\mathbf{u}_t$，我们只需要缩放这个变换所对应的角度 ϕ 就能够达到插值的目的了．

那么，现在的问题是，我们该怎么获得这个旋转的变化量呢？我们不妨考虑一下什么变换 Δq 能将已经旋转到 v_0 的向量 v 直接变换到 v_1．这其实就是一个旋转的复合，我们先进行 q_0 变换，再进行 Δq 变换，它们复合的结果需要等于 q_1 变换，也就是说：

$$\Delta q q_0 = q_1$$

那么，

$$\Delta q q_0 q_0^{-1} = q_1 q_0^{-1}$$

$$\Delta q = q_1 q_0^{-1}$$

因为所有的旋转 q 都是单位四元数，$q^{-1} = q^*$，它又可以写成：

$$\Delta q = q_1 q_0^*$$

如果我们对 Δq 取 t 次方，$(\Delta q)^t$ 就能缩放这个旋转所对应的角度了．所以，我们就能得出插值的公式：

$$q_t = Slerp(q_0, q_1; t) = (q_1 q_0^*)^t q_0$$

可以发现,当 $t = 0$ 时,$q_t = (q_1 q_0^*)^0 q_0 = q_0$;而当 $t = 1$ 时,$q_t = (q_1 q_0^*)^1 q_0 = q_1(q_0^* q_0) = q_1$．如果 t 为中间值，比如说 $t = 0.4$ 时，$q_t = (q_1 q_0^*)^{0.4} q_0$，它会先进行 q_0 变换将 v 变换到 v_0，并在此基础上向 v_1 旋转 40%.

这个公式虽然对四元数插值的分析很有帮助，但是它的计算不仅涉及到多个四元数的乘法，而且包含幂运算，在实际应用中的效率很低，我们希望找出一个更高效地插值方法.

我们将这个插值方法叫做「Slerp」，但是我们暂时不会解释它具体是什么意思，为了理解它我们还需要研究一下 3D 空间的旋转与四元数的 4D 向量空间之间的关系.

为了探讨这个关系，我们来实际计算一下 Δq．由于这个关系和角度有关，我们只

需要关心 Δq 的实部就可以了：

$$\begin{aligned}\Delta q &= q_1 q_0^* \\ &= [\cos(\theta_1), \sin(\theta_1)\mathbf{u}_1][\cos(\theta_0), -\sin(\theta_0)\mathbf{u}_0] \\ &= [\cos(\theta_0)\cos(\theta_1) - (\sin(\theta_1)\mathbf{u}_1)\cdot(-\sin(\theta_0)\mathbf{u}_0),\ \dots] \\ &= [\cos(\theta_0)\cos(\theta_1) + (\sin(\theta_1)\mathbf{u}_1)\cdot(\sin(\theta_0)\mathbf{u}_0),\ \dots]\end{aligned}$$

如果将 q_0 和 q_1 看作是两个四维向量，我们能发现，非常巧合的是，Δq 的实部正好是 q_0 与 q_1 点乘的结果 $q_0 \cdot q_1$！

因为 q_0 与 q_1 都是单位四元数，$q_0 \cdot q_1$ 正好是这两个四元数在 4D 空间中夹角的余弦值，我们将这个夹角称为 θ．那么 $q_0 \cdot q_1 = \cos(\theta)$.

我们又知道，Δq 表示的也是一个旋转，而如果它代表的旋转的角度是 2ϕ，那么 Δq 的实数部为 $\Delta q = [\cos(\phi),\ \dots]$.

所以，

$$\Delta q = [\cos(\phi),\ \dots] = [\cos(\theta),\ \dots]$$

$$\cos(\phi) = \cos(\theta)$$

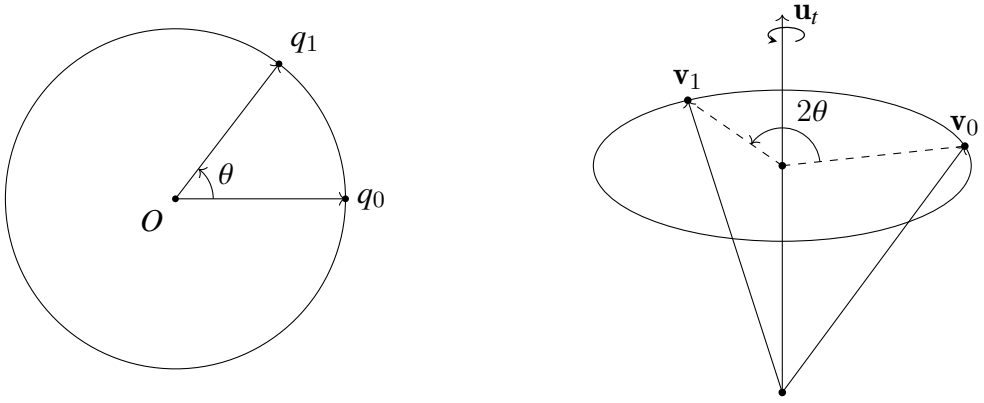
因为 ϕ 和 θ 都是夹角，$\phi, \theta \in [0, \pi]$，所以这个方程有唯一的解：

$$\phi = \theta$$

这也就是说，q_0 与 q_1 作为向量在 4D 四元数空间中的夹角 θ，正好是它们旋转变化量 Δq 的所代表旋转的角度的两倍，即 $2\phi = 2\theta$．所以，我们可以直接用插值向量的方法对旋转进行插值.

为了更直观地理解这层关系，请看下面这两幅图．虽然四元数是处于四维空间之内的，但是因为只有两个四元数，我们可以将它们投影到一个二维的圆上来，也就是左

图．右图则是 3D 空间中发生的旋转改变：



可以看到，当 q_1 与 q_0 之间的夹角为 θ 时，旋转的变化量正好是 2θ．如果我们在圆上有一个单位四元数 q_t，使得它与 q_0 的夹角为 $t\theta$，与 q_1 的夹角为 $(1-t)\theta$，那么我们就能保证在 3D 空间中，它相对于 q_0 的旋转变化量为 $2t\theta$，相对于 q_1 的旋转变化量为 $2(1-t)\theta$．

现在，两个单位四元数的插值就被我们简化为了一个圆上（其实是超球面的一部分）两个向量的插值，我们能直接套用向量的插值公式对两个四元数进行插值．接下来，我们就来讨论如何对两个向量进行插值．

5 Lerp, Nlerp, Slerp

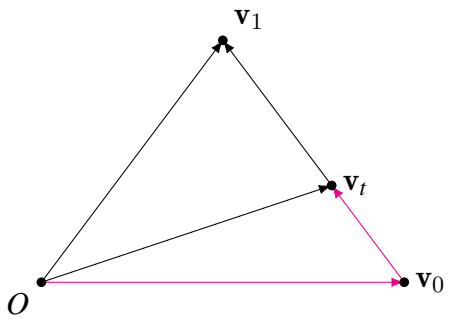
不管是哪种插值方法，我们都希望将中间向量 $\mathbf{v}_t$ 写为初始向量 $\mathbf{v}_0$ 和最终向量 $\mathbf{v}_1$ 的线性组合，也就是说：

$$\mathbf{v}_t = \alpha\mathbf{v}_0 + \beta\mathbf{v}_1$$

其中，系数 α 与 β 都是 t 的函数．不同的插值方法只是拥有不同的系数而已．

5.1 Lerp

我们首先来看一下两个向量插值最简单的一种形式：线性插值（**L**inear Int**erp**olation），也叫做「Lerp」．Lerp 会沿着一条直线进行插值，如果将 $\mathbf{v}_0$ 和 $\mathbf{v}_1$ 看做是三角形的两个边，那么 $\mathbf{v}_t$ 会指向三角形的第三条边：



从图中可以看到，我们能将 $\mathbf{v}_t$ 写为两个向量的和（用红色标出）. 其中一个向量正是 $\mathbf{v}_0$，而另一个向量则是 $\mathbf{v}_1 - \mathbf{v}_0$ 的乘上一个系数，我们直接将 t 作为这个系数，所以:

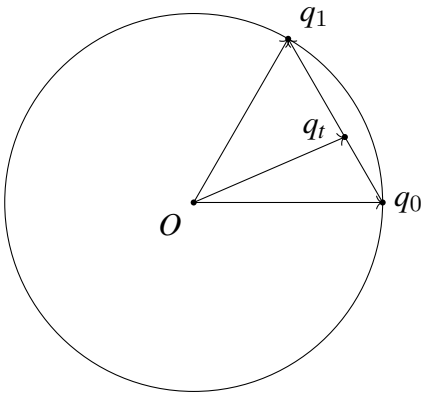
$$\begin{aligned}\mathbf{v}_t = Lerp(\mathbf{v}_0, \mathbf{v}_1, t) &= \mathbf{v}_0 + t(\mathbf{v}_1 - \mathbf{v}_0) \\ &= (1-t)\mathbf{v}_0 + t\mathbf{v}_1\end{aligned}$$

当 $t = 0$ 时，$\mathbf{v}_t = (1-0)\mathbf{v}_0 + 0\mathbf{v}_1 = \mathbf{v}_0$；当 $t = 1$ 时，$\mathbf{v}_t = (1-1)\mathbf{v}_0 + 1 \cdot \mathbf{v}_1 = \mathbf{v}_1$. 这正是我们需要的结果.

如果将 Lerp 的结果应用到单位四元数上，我们就能得到:

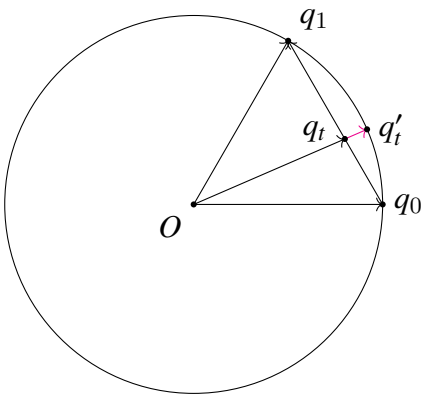
$$q_t = Lerp(q_0, q_1, t) = (1-t)q_0 + tq_1$$

当然，因为我们是沿着一条直线（也就是圆上的一个弦）进行插值的，这样插值出来的四元数并不是单位四元数:



5.2 Nlerp

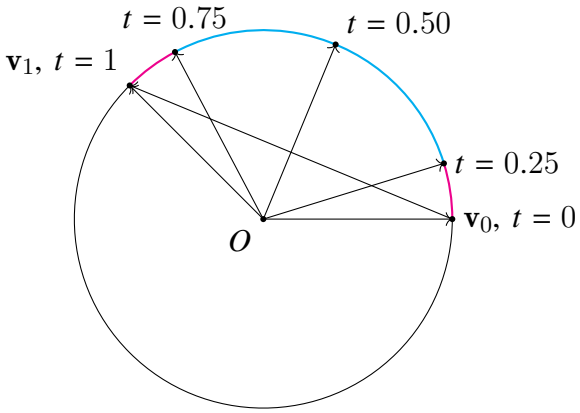
虽然这样插值出来的 q_t 并不是单位四元数，但只要将 q_t 除以它的模长 $\|q_t\|$ 就能够将其转化为一个单位四元数了：



我们将这种先对向量进行插值，再进行正规化 (Normalization) 的插值方法称为正规化线性插值（**N**ormalized **L**inear Int**erp**olation），或者「Nlerp」. 与 Lerp 不同，Nlerp 的两个输入向量必须是单位向量，否则插值出来的结果不会经过初始和最终向量. 下面分别是向量和四元数的 Nlerp 公式：

$$\mathbf{v}_t = Nlerp(\mathbf{v}_0, \mathbf{v}_1, t) = \frac{(1-t)\mathbf{v}_0 + t\mathbf{v}_1}{\|(1-t)\mathbf{v}_0 + t\mathbf{v}_1\|}$$
$$q_t = Nlerp(q_0, q_1, t) = \frac{(1-t)q_0 + tq_1}{\|(1-t)q_0 + tq_1\|}$$

Nlerp 插值仍然存在有一定的问题，当需要插值的弧比较大时，$\mathbf{v}_t$ 的角速度会有显著的变化. 我们可以来看一个例子：



这五个 t 值将整个弧和弦分割成了四个部分．虽然弦上的四段是等长的，但是四个弧是完全不相等的．$t = 0$ 到 $t = 0.25$ 之间的弧（红色）明显比 $t = 0.25$ 到 $t = 0.50$ 的弧（蓝色）要短了不少．

这也就是说，在同等时间内，$\mathbf{v}_t$ 扫过的角度是不同的．$\mathbf{v}_t$ 扫过的速度（或者说角速度）首先会不断地增加，到 $t = 0.50$ 之后会开始减速，所以 Nlerp 插值不能保证均匀的角速度．

5.3 Slerp

为了解决这个问题，我们可以转而对角度进行线性插值．也就是说，如果 $\mathbf{v}_1$ 和 $\mathbf{v}_2$ 之间的夹角为 θ，那么：

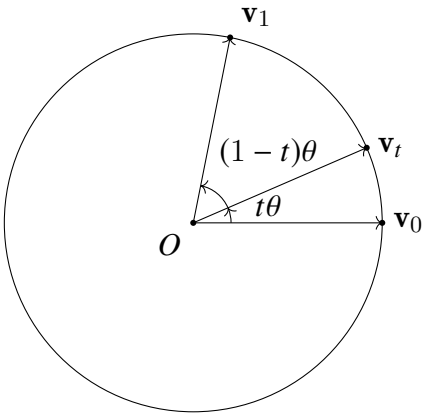
$$\theta_t = (1 - t) \cdot 0 + t\theta = t\theta$$

因为对角度线性插值直接是让向量在球面上的一个弧上旋转，所以又称球面线性插值 (**S**pherical **L**inear Int**erp**olation)，或者「Slerp」．类比于 Lerp 是平面上的线性插值，Slerp 是球面上的线性插值．我们上一章讨论的四元数插值公式正是一个对四元数在四维超球面上的旋转，所以它是 Slerp 的一个等价公式．

上面公式并没有涉及到任何的向量，我们希望将 $\mathbf{v}_t$ 写为 $\mathbf{v}_0$ 和 $\mathbf{v}_1$ 的线性组合：

$$\mathbf{v}_t = \alpha\mathbf{v}_0 + \beta\mathbf{v}_1 \tag{1}$$

注意这里的 $\mathbf{v}_0$ 和 $\mathbf{v}_1$ 仍是单位向量．为了求出这其中的 α 和 β，我们需要借助图像来找出一些关系：



因为图中涉及到很多的角度关系，我们可以先对 (1) 的两边同时点乘 $\mathbf{v}_0$：

$$\mathbf{v}_0 \cdot \mathbf{v}_t = \mathbf{v}_0 \cdot (\alpha \mathbf{v}_0 + \beta \mathbf{v}_1)$$

$$\mathbf{v}_0 \cdot \mathbf{v}_t = \alpha(\mathbf{v}_0 \cdot \mathbf{v}_0) + \beta(\mathbf{v}_0 \cdot \mathbf{v}_1)$$

我们知道，$\mathbf{v}_0$ 和 $\mathbf{v}_t$ 之间的夹角是 $t\theta$，$\mathbf{v}_0$ 与它自身之间的夹角为 0，$\mathbf{v}_0$ 和 $\mathbf{v}_1$ 之间的夹角是 θ，而且所有的向量都是单位向量，所以：

$$\cos(t\theta) = \alpha + \beta \cos(\theta) \tag{2}$$

同理，我们将 (1) 的两边同时点乘 $\mathbf{v}_1$，构造第二个方程：

$$\mathbf{v}_1 \cdot \mathbf{v}_t = \mathbf{v}_1 \cdot (\alpha \mathbf{v}_0 + \beta \mathbf{v}_1)$$

$$\mathbf{v}_1 \cdot \mathbf{v}_t = \alpha(\mathbf{v}_1 \cdot \mathbf{v}_0) + \beta(\mathbf{v}_1 \cdot \mathbf{v}_1)$$

$$\cos((1-t)\theta) = \alpha \cos(\theta) + \beta \tag{3}$$

现在，我们就有了两个方程以及两个未知数，我们只需要解 (2) 、 (3) 这两个方程，求出 α 和 β 就能获得 Slerp 的公式了.

由 (2) 我们能得到：

$$\alpha = \cos(t\theta) - \beta \cos(\theta) \tag{4}$$

将 (4) 代入 (3) ，利用一些三角恒等式，我们能解出 β：

$$\cos((1-t)\theta) = (\cos(t\theta) - \beta \cos(\theta)) \cos(\theta) + \beta$$

$$\cos((1-t)\theta) = \cos(t\theta)\cos(\theta) - \beta \cos^2(\theta) + \beta$$

$$\beta(1 - \cos^2(\theta)) = \cos((1-t)\theta) - \cos(t\theta)\cos(\theta)$$

$$\beta = \frac{\cos(\theta - t\theta) - \cos(t\theta)\cos(\theta)}{\sin^2(t)}$$

$$\beta = \frac{\cos(\theta)\cos(t\theta) + \sin(\theta)\sin(t\theta) - \cos(t\theta)\cos(\theta)}{\sin^2(\theta)}$$

$$\beta = \frac{\sin(\theta)\sin(t\theta)}{\sin^2(\theta)}$$

$$\beta = \frac{\sin(t\theta)}{\sin(\theta)}$$

现在我们就得到了 β，将 β 代入 (4) 就能解出 α：

$$\begin{aligned}\alpha &= \cos(t\theta) - (\frac{\sin(t\theta)}{\sin(\theta)})\cos(\theta) \\ &= \frac{\cos(t\theta)\sin(\theta) - \sin(t\theta)\cos(\theta)}{\sin(\theta)} \\ &= \frac{\sin((1-t)\theta)}{\sin(\theta)}\end{aligned}$$

将 α 和 β 代回 (1)，我们可以得到向量的 Slerp 的公式：

$$\mathbf{v}_t = Slerp(\mathbf{v}_0, \mathbf{v}_1, t) = \frac{\sin((1-t)\theta)}{\sin(\theta)}\mathbf{v}_0 + \frac{\sin(t\theta)}{\sin(\theta)}\mathbf{v}_1$$

类似地，我们有四元数的 Slerp 公式：

$$q_t = Slerp(q_0, q_1, t) = \frac{\sin((1-t)\theta)}{\sin(\theta)}q_0 + \frac{\sin(t\theta)}{\sin(\theta)}q_1$$

q_0 与 q_1 之间的夹角 θ 可以直接使用它们点乘的结果来得出，即

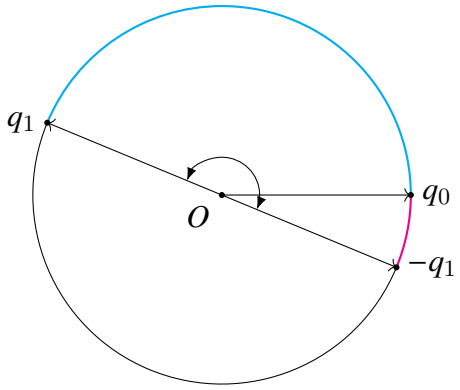
$$\theta = \cos^{-1}(q_0 \cdot q_1)$$

这里导出的公式会比之前利用幂运算的公式要高效很多，但是它仍然涉及到三个三角函数以及一个反三角函数的运算，所以还是会比 Nlerp 要慢一点．如果要插值的角度比较小的话，Nlerp 其实相对于 Slerp 的误差并没有那么大．为了提高效率，我们经常会使用 Nlerp 来代替 Slerp．我们也能用一些数值分析 (Numerical Analysis) 的方法来近似并优化四元数的 Slerp．你可以在一些图形引擎的源代码中找到一些例子．

除了效率问题之外，我们在实现 Slerp 时要注意，如果单位四元数之间的夹角 θ 非常小，那么 $\sin(\theta)$ 可能会由于浮点数的误差被近似为 0.0，从而导致除以 0 的错误．所以，我们在实施 Slerp 之前，需要检查两个四元数的夹角是否过小（或者完全相同）．一旦发现这种问题，我们就必须改用 Nlerp 对两个四元数进行插值，这时候 Nlerp 的误差非常小所以基本不会与真正的 Slerp 有什么区别．

5.4 双倍覆盖带来的问题

如果你还记得，两个不同的单位四元数 q 与 $-q$ 对应的其实是同一个旋转，这个特性显然会对我们的插值造成一些影响．虽然 q 与 $-q$ 对向量变换的最终效果是完全相同的，但是它们作为向量相差了 π 弧度：



可以看到，虽然我们能够将 q_0 向左插值至 q_1（蓝色的弧），但这会将 3D 空间中的向量旋转接近 360°，而实际上这两个旋转相差并没有那么多，它并不是 3D 空间中的弧面最短路径（Geodesic）．而如果我们将 q_0 向右插值至等价的 $-q_1$（红色的弧），它的旋转变化量就会比插值到 q_1 要小很多，所以 q_0 插值到 $-q_1$ 才是插值的最短路径．

这也就告诉我们，在对两个单位四元数进行插值之前，我们需要先检测 q_0 与 q_1 之间是否是钝角，也就是检测它们点积的结果 $q_0 \cdot q_1$ 是否为负数．如果 $q_0 \cdot q_1 < 0$，那么我们就反转其中的一个四元数，比如说将 q_1 改为 $-q_1$，并使用 q_0 与 $-q_1$ 之间新的夹角来进行插值，这样才能保证插值的路径是最短的．

6 Squad

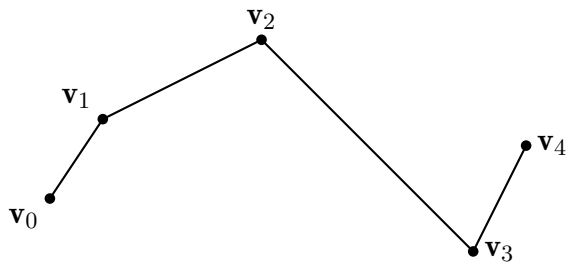
Slerp 其实仍然存在一定的问题．因为我们是对角度进行线性插值，旋转的角速度虽然是固定的，但它的值是与夹角 θ 成正比的（$\frac{d\theta_t}{dt} = \frac{d}{dt}(t\theta) = \theta$，准确说是角速率）．如果我们需要对多个四元数进行插值，对每一对四元数使用 Slerp 插值虽然能够保证每两个四元数之间的角速度是固定的，但是角速度会在切换插值的四元数时出现断点，也就是说在切换点不可导．比如说我们有 q_0, q_1, q_2 三个四元数所组成的序列，如果分别对 q_0q_1、q_1q_2 使用 Slerp 插值，那么当 $q_t = q_1$ 时，角速度会突然改变，这有时候并不是我们所希望的结果．我们希望能以牺牲固定角速度为条件，让插值的曲线不仅是连续的，而且让它的一阶甚至是高阶导数是连续的（曲线连续我们称为 C^0 连续，达到一阶导数连续就叫做 C^1 连续，在此基础上达到二阶导数连续叫做 C^2 连续，以此类推）．

解决这个问题的方法有很多，在网上能找到很多论文，但是每一种方法想要完全理解都不是那么容易的，而且它们一般都比普通的 Slerp 或者 Nlerp 要慢得多．我

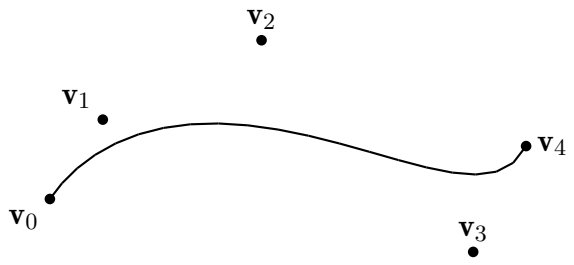
们在这里只会简单介绍最常见到而且也是最直观的球面四边形插值 (**S**pherical and **quad**rangle)，或者「Squad」．它由 Ken Shoemake 在 1987 年提出（这篇 Paper 已经找不到了）．

Squad 仍然是平面上的向量插值衍生到超球面的结果，所以我们关注的重点首先仍然是向量的插值．下面的内容可能会需要一些 Bézier 曲线以及样条的前置知识，但即便你不知道也没有关系，我会在下面简单地概括一下这些知识．

假设我们有一个向量的序列 $\mathbf{v}_0, \mathbf{v}_1, \ldots, \mathbf{v}_n$，如果我们想对这个序列进行插值，那么我们可以分别对每一对向量 $\mathbf{v}_i$ 和 $\mathbf{v}_{i+1}$ 进行插值，然后将插值的曲线连接起来，也就是我们所说的样条（Spline）．如果直接使用 Lerp 的话，我们会得到这样的结果（假设我们只有五个向量需要插值 $\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4$）：



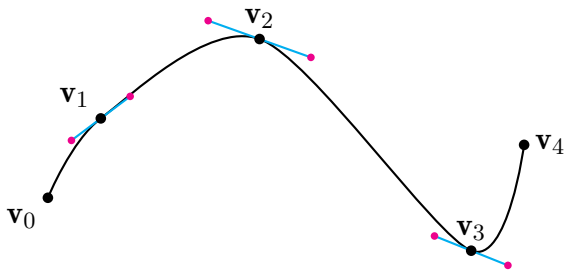
很明显，这个曲线虽然是连续的，但是它的一阶导数（切线）在切换插值向量时都不是连续的．为了解决这个问题，我们最常使用的就是 Bézier 曲线．我们一开始的想法可能会是将中间的 $\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$ 作为控制点，直接使用一个四次 Bézier 曲线（因为有五个点）来生成这个近似曲线．但是 Bézier 曲线只会经过初始点与最终点（插值），一般不会经过中间的控制点（近似），所以这样求出来的曲线虽然是可导的，但是插值曲线不会经过中间的三个向量：



6.1 三次 Bézier 曲线

为了解决这个问题，我们可以分段对每两个向量 $\mathbf{v}_i$ 和 $\mathbf{v}_{i+1}$ 之间使用 Bézier 曲线进行插值，之后将所有的曲线（也叫做样条）连接起来．因为我们需要让曲线的一阶导数，或者说曲线的趋势，连续，我们还需要知道它们的前一个向量 $\mathbf{v}_{i-1}$ 和后一个向量 $\mathbf{v}_{i+2}$，并且用它们生成两个控制点 $\mathbf{s}_i$ 和 $\mathbf{s}_{i+1}$ 来控制曲线的趋势．我们会使用 $\mathbf{v}_i$ 和 $\mathbf{v}_{i+1}$ 作为端点（曲线会经过这两个点），$\mathbf{s}_i$ 和 $\mathbf{s}_{i+1}$ 作为中间的控制点，使用一个三次 Bézier 曲线（Cubic Bézier Curve，四个点）来近似这个两个向量之间的插值．

在我们的例子中，因为我们一共有四对向量（$\mathbf{v}_0\mathbf{v}_1$、$\mathbf{v}_1\mathbf{v}_2$、$\mathbf{v}_2\mathbf{v}_3$、$\mathbf{v}_3\mathbf{v}_4$），我们会使用四个三次 Bézier 曲线对这五个点进行插值．对于三次 Bézier 曲线所产生的样条，如果我们想让最终的插值曲线达到 C^1 连续，那么我们需要让前一个样条在 $\mathbf{v}_i$ 的控制点与当前样条在 $\mathbf{v}_i$ 的控制点分别处于最终曲线在 $\mathbf{v}_i$ 处切线对等的两侧：



在上面的曲线中，蓝色的线就是曲线在点 $\mathbf{v}_i$ 处的切线，红色的点就是三次 Bézier 曲线的控制点，分别处于切线对等的两侧．对于两个端点 $\mathbf{v}_0$ 和 $\mathbf{v}_4$，我们直接将这两个向量的控制点取为它们本身（这不是唯一的做法，但这样是可行的），最终得到一个平滑的曲线．

我们希望将类似的逻辑带到四元数的超球面上，达到四元数序列的插值，但在此之前我们需要了解如何构造一个三次 Bézier 曲线．

6.2 de Casteljau 算法

Bézier 曲线的构造有个著名的递归算法叫做 de Casteljau 算法（de Casteljau's Algorithm），它对任意次方的 Bézier 曲线都是成立的，但是这里我们只关注三次 Bézier 曲线的情况．

这个算法最基本的思想就是线性插值的嵌套．假设我们有四个向量 $\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3$，那么我们可以这样子获得最终的三次 Bézier 曲线：

首先，我们对每一对向量 $\mathbf{v}_0\mathbf{v}_1$、$\mathbf{v}_1\mathbf{v}_2$、$\mathbf{v}_2\mathbf{v}_3$ 进行线性插值，获得 $\mathbf{v}_{01}$、$\mathbf{v}_{12}$、$\mathbf{v}_{23}$：

$$\mathbf{v}_{01} = Lerp(\mathbf{v}_0, \mathbf{v}_1; t)$$

$$\mathbf{v}_{12} = Lerp(\mathbf{v}_1, \mathbf{v}_2; t)$$

$$\mathbf{v}_{23} = Lerp(\mathbf{v}_2, \mathbf{v}_3; t)$$

之后，我们对 $\mathbf{v}_{01}\mathbf{v}_{12}$ 和 $\mathbf{v}_{12}\mathbf{v}_{23}$ 这两对向量进行线性插值，获得 $\mathbf{v}_{012}$ 和 $\mathbf{v}_{123}$：

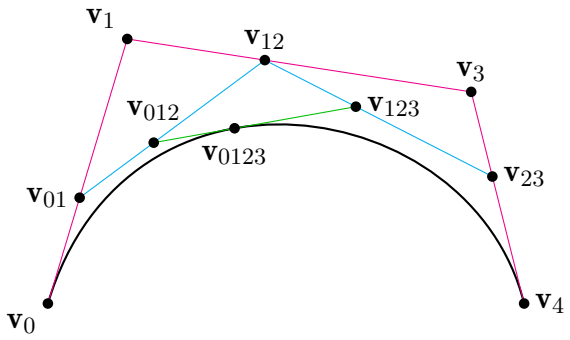
$$\mathbf{v}_{012} = Lerp(\mathbf{v}_{01}, \mathbf{v}_{12}; t)$$

$$\mathbf{v}_{123} = Lerp(\mathbf{v}_{12}, \mathbf{v}_{23}; t)$$

最后，对 $\mathbf{v}_{012}$ 和 $\mathbf{v}_{123}$ 进行线性插值获得 $\mathbf{v}_{0123}$，这个向量就是我们想要的最终结果，它就是三次 Bézier 曲线上的点：

$$\mathbf{v}_{0123} = Lerp(\mathbf{v}_{012}, \mathbf{v}_{123}; t)$$

虽然这个算法看起来很繁琐，但是我们可以通过一张图来理解它（取 $t = 0.4$）：



可以看到，虽然我们一直在使用线性插值，最终获得的却是一条三次 Bézier 曲线.

如果将这些式子合并起来，我们就能得到三次 Bézier 曲线的递归公式. 因为这个式子太长了，我将 $Lerp(\mathbf{v}_i, \mathbf{v}_{i+1}; t)$ 简写为 $L(\mathbf{v}_i, \mathbf{v}_{i+1}; t)$：

$$\text{Bézier}(\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3; t) = L(L(L(\mathbf{v}_0, \mathbf{v}_1; t), L(\mathbf{v}_1, \mathbf{v}_2; t); t), L(L(\mathbf{v}_1, \mathbf{v}_2; t), L(\mathbf{v}_2, \mathbf{v}_3; t); t); t)$$

如果将 Lerp 的定义 $Lerp(\mathbf{v}_i, \mathbf{v}_{i+1}; t) = (1-t)\mathbf{v}_i + t\mathbf{v}_{i+1}$ 不断代入并展开的话，我们能获得这样一个式子：

$$\text{Bézier}(\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3; t) = (1-t)^3\mathbf{v}_0 + 3(1-t)^2t\mathbf{v}_1 + 3(1-t)t^2\mathbf{v}_2 + t^3\mathbf{v}_3$$

因为每项的次数都是 3，所以我们说它是一个三次 Bézier 曲线.

我们可以直接将递归的公式运用到四元数上，得到四元数的球面 Bézier 曲线公式，但因为球面的线性插值不是 Lerp 而是 Slerp，我们需要将公式中所有的 Lerp 全部换成 Slerp（你可以想象一下，将四个向量形成的四边形看作是一个网格（Mesh），之后将这个网格贴在球面上）. 同样因为公式太长，我会将 $Slerp(q_i, q_{i+1}; t)$ 简写为 $S(q_i, q_{i+1}; t)$：

$$\text{SBézier}(q_0, q_1, q_2, q_3; t) = S(S(S(q_0, q_1; t), S(q_1, q_2; t); t), S(S(q_1, q_2; t), S(q_2, q_3; t); t); t)$$

这个其实就是 Ken Shoemake 在 1985 年的那篇 Paper 里提出的插值方法，他也提供了控制点的公式. 然而，很明显这个方法实在是太复杂了. 仅仅是一个 Slerp 就要使用四个三角函数，而我们这里一共有 7 个 Slerp，如果真的要使用它进行插值会对性能产生非常大的影响.

6.3 Squad

于是，Shoemake 在 1987 年提出了一个更高效的近似算法，也就是我们熟悉的 Squad.

我们首先仍然来看平面中向量的情况. 我把向量的 Squad 算法叫做 Quad，代表「Quadrangle」，Quad 有很多歧义，而且我好像没看到别人这么叫过，但是为了简便这里就暂时这样称呼它吧.

与三次 Bézier 曲线嵌套了三层一次插值不同，Quad 使用的是一层二次插值嵌套了一层一次插值：

我们首先是分别对 $\mathbf{v}_0\mathbf{v}_3$ 和 $\mathbf{v}_1\mathbf{v}_2$ 进行插值，获得 $\mathbf{v}_{03}$ 和 $\mathbf{v}_{12}$：

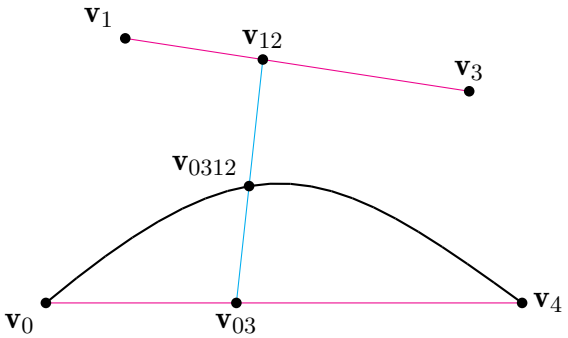
$$\mathbf{v}_{03} = Lerp(\mathbf{v}_0, \mathbf{v}_3; t)$$

$$\mathbf{v}_{12} = Lerp(\mathbf{v}_1, \mathbf{v}_2; t)$$

之后，我们使用 $2t(1-t)$ 为参数，对 $\mathbf{v}_{03}$ 和 $\mathbf{v}_{12}$ 进行二次插值，获得最终的 $\mathbf{v}_{0312}$：

$$\mathbf{v}_{0312} = Lerp(\mathbf{v}_{03}, \mathbf{v}_{12}; 2t(1-t))$$

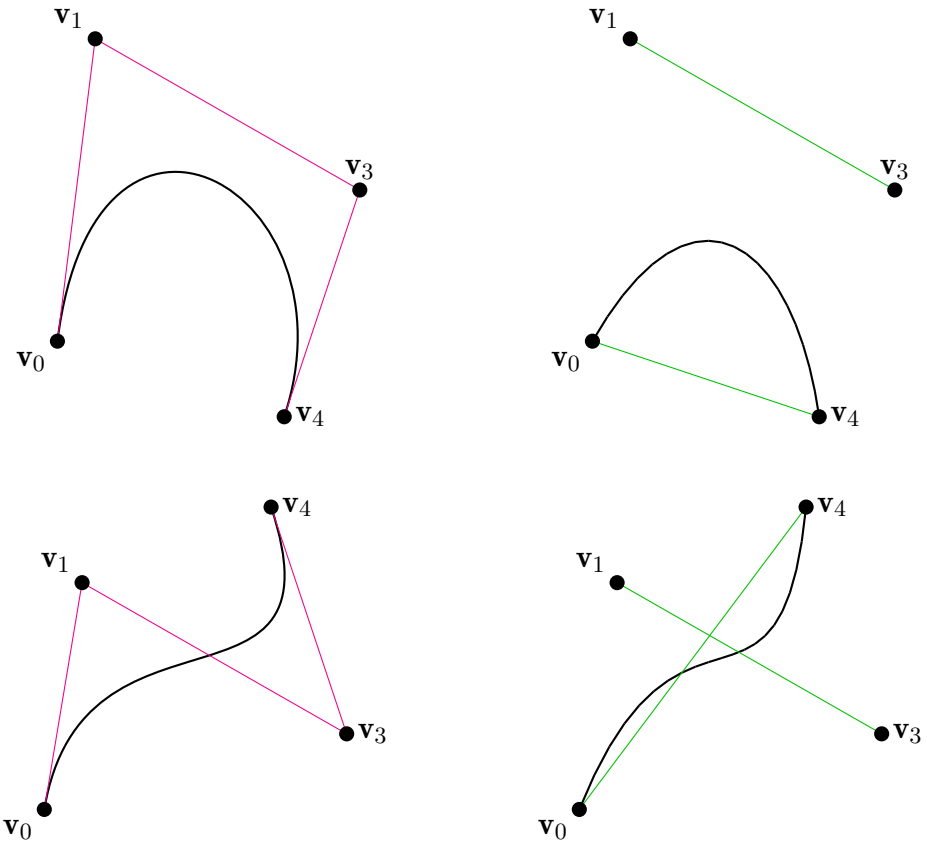
注意最终的 Lerp 使用的参数是二次的 $2t(1-t)$，不是我们一般使用的 t，而且它插值的两个向量也变为了 $\mathbf{v}_{03}$ 和 $\mathbf{v}_{12}$. 我们仍然可以使用图片来理解它（取 $t = 0.4$）：



同样，我们可以将 Quad 写为递归形式：

$$Quad(\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3; t) = Lerp(Lerp(\mathbf{v}_0, \mathbf{v}_3; t), Lerp(\mathbf{v}_1, \mathbf{v}_2; t); 2t(1-t))$$

可以看到，这样的插值要比三次 Bézier 曲线简单很多，将七次 Lerp 减少到了三次. 虽然最终的曲线与三次 Bézier 曲线不完全相同，但是已经很近似了，我们可以看几个对比，左边是三次 Bézier 曲线，右边是 Quad 曲线：



如果我们利用 Lerp 的定义 $Lerp(\mathbf{v}_i, \mathbf{v}_{i+1}; t) = (1-t)\mathbf{v}_i + t\mathbf{v}_{i+1}$ 将递归式展开的话，我

们能得到这样的式子：

$$Quad(\mathbf{v}_0, \mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3; t) = (2t^2 - 2t + 1)(1-t)\mathbf{v}_0 + 2(1-t)^2 t\mathbf{v}_1 + 2(1-t)t^2\mathbf{v}_2 + t(2t^2 - 2t + 1)\mathbf{v}_3$$

它仍是一个三次的曲线，只不过系数有所不同.

如果我们将 Quad 的递归公式用于球面，就能得到四元数的 Squad：

$$Squad(q_0, q_1, q_2, q_3; t) = Slerp(Slerp(q_0, q_3; t), Slerp(q_1, q_2; t); 2t(1-t))$$

我们知道 $Slerp(q_i, q_{i+1}; t) = (q_{i+1}q_i^*)^t q_i$，所以我们可以将 Squad 写成指数形式：

$$Squad(q_0, q_1, q_2, q_3; t) = (Slerp(q_1, q_2; t)(Slerp(q_0, q_3; t))^*)^{2t(1-t)} Slerp(q_0, q_3; t)$$

这个式子会对 Squad 的分析非常有用.

6.4 Squad 应用

接下来，我们回到本章最初的主题，对多个单位四元数进行插值. 如果我们有一个四元数序列 $q_0, q_1, \ldots, q_n$，我们希望对每一对四元数 q_i 和 q_{i+1} 都使用 Squad 进行插值，所以我们有：

$$Squad(q_i, s_i, s_{i+1}, q_{i+1}; t) = Slerp(Slerp(q_i, q_{i+1}; t), Slerp(s_i, s_{i+1}; t); 2t(1-t))$$

现在，留下来的问题就是找出中间的控制点 s_i 和 s_{i+1} 了. 类似于 Bézier 曲线的样条，我们同样需要前一个四元数 q_{i-1} 以及 q_{i+2} 的信息.

s_i 的推导还是比较复杂的，但是它最基本的理念非常简单：让 Squad 在切换点可导，从而达到 C^1 连续. 也就是说，我们希望 $q_{i-1}q_i$ 插值时在 $t = 1$ 处的导数，与 q_iq_{i+1} 插值时在 $t = 0$ 处的导数相等：

$$Squad'(q_{i-1}, s_{i-1}, s_i, q_i; 1) = Squad'(q_i, s_i, s_{i+1}, q_{i+1}; 0)$$

如果我们想要从这里继续推导下去的话，就需要用到单位四元数导数的定义（它和 $q = [\cos(\theta), \sin(\theta)\mathbf{u}] = e^{u\theta}$ 有关），但是因为这一块涉及到很多的证明，我并没有进行介绍. 所以在这里我会省略控制点 s_i 的推导. 如果你感兴趣的话，你可以到最后的参考资料中找到完整的证明.

如果按照这个思路推导下去的话，最终能得到：

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

注意，和 Bézier 曲线的样条不同的是，这里的 s_i 在对 $q_{i-1}q_i$ 插值时和对 $q_i q_{i+1}$ 插值时都是相同的，不像之前是处于切线的两端不同的两个值.

与两个四元数之间的插值一样，Squad 同样会受到双重覆盖的影响. 我们在计算中间控制点和插值之前，需要先选中一个四元数，比如说 q_i，检测它与其它三个四元数之间的夹角，如果是钝角就翻转，将插值的路线减到最小.

7 附加主题、参考资料以及拓展阅读

如果你能一直读到现在，那么你应该对四元数与 3D 旋转的关系以及在图形学的应用有一定基本的了解了. 然而，与四元数和旋转相关的主题还有很多很多. 为了能让你更容易理解这些主题，在这一章中，我会在给出参考资料同时，非常简略地介绍一下它们，并提供一些拓展阅读.

7.1 李群

7.1.1 特殊正交群

特殊正交群（Special Orthogonal Group）其实讨论的是旋转更一般的情况，如果你仔细观察我们得出的 2D 和 3D 旋转矩阵，你会注意到它们具有一些共同的特征.

首先，它们都是正交矩阵 (Orthogonal Matrix)，也就是说矩阵的列向量互相正交，而且是单位向量. 如果我们有正交矩阵 Q，那么 $QQ^T = Q^T Q = I$，而且正交矩阵与正交矩阵的乘积仍然是一个正交矩阵. 正交矩阵的变换会保持向量的长度和内积（也就是夹角），这正是旋转的特性之一.

正交矩阵的行列式可能有两个值，分别是 1 和 −1（我们可以得到 $(\det Q)(\det Q^T) = \det I \Rightarrow (\det Q)^2 = 1$）. 当行列式值为 −1 时，它会翻转向量的相对位置，所以对应的是反射. 而当行列式值为 1 时，它会保持向量的相对位置，所以对应的是旋转. 如果你感兴趣的话，你可以利用三角恒等式求一下我们 2D 和 3D 旋转矩阵的行列式，你会发现它们都属于后者.

我们将行列式值为 1 的 2×2 正交矩阵在矩阵乘法（或者复合）下形成的李群叫做特殊正交群 $SO(2)$（SO 指的是 **S**pecial **O**rthogonal），它包括所有的 2D 旋转．同理，$SO(3)$ 这个群包括所有的 3D 旋转．

7.1.2 特殊酉群

特殊酉群（Special Unitary Group）是同构于单位四元数的一个李群．如果你还记得，我们之前推导出了四元数的两个矩阵形式，这里我们看四元数的右乘矩阵：

$$Q = \left[\begin{array}{cc|cc} a & -b & -c & -d \\ b & a & d & -c \\ \hline c & -d & a & b \\ d & c & -b & a \end{array}\right]$$

将这个矩阵分块之后，我们可以发现，这其中的每一个子矩阵都代表了一个复数（$a + bi = \begin{bmatrix} a & -b \\ b & a \end{bmatrix}$）．如果我们将这个矩阵改写为复值矩阵的话，就能得到：

$$Q = \begin{bmatrix} a + bi & -c + di \\ c + di & a - bi \end{bmatrix}$$

如果我们定义 $\alpha = a + bi$，$\beta = c + di$，那么：

$$Q = \begin{bmatrix} \alpha & -\overline{\beta} \\ \beta & \overline{\alpha} \end{bmatrix}$$

当四元数 $q = a + bi + cj + dk$ 是一个单位四元数时，这个矩阵具有两个性质．首先，它是一个酉矩阵（Unitary Matrix）．如果有一个酉矩阵 U，它满足这样的性质：

$$UU^* = U^*U = I$$

其中 U^* 指的是矩阵的共轭转置，首先对矩阵中每一个元素取共轭，再进行转置．酉矩阵的行列式值可能是 1 或者 -1，但对于单位四元数，$\det Q = \|q\|^2 = 1$．

我们将行列式值为 1 的 2×2 酉矩阵在矩阵乘法下形成的李群叫做特殊酉群 $SU(2)$（SU 指的是 Special Unitary），它同构于所有的单位四元数．

我们之前一直讨论的单位四元数与 3D 旋转之间的关系其实就是在讨论 $SU(2) \to SO(3)$ 这个群同态.

7.1.3 参考资料及拓展阅读

实际上，我们一直在讨论的内容涉及到的远不止这两个群，如果你想从抽象代数的角度深入理解单位四元数和旋转，可以去阅读 John Stillwell 的『Naive Lie Theory』.

除了书之外，也可以去看 Alistair Savage 的讲义『Introduction to Lie Groups』，它可以在网上免费获取到.

Qiaochu Yuan 在他的博客上也写了几篇关于四元数、$SO(3)$、$SU(2)$ 的博文：

- $SO(3)$ and $SU(2)$
- $SU(2)$ and the quaternions
- The quaternions and Lie algebras I
- The quaternions and Lie algebras II

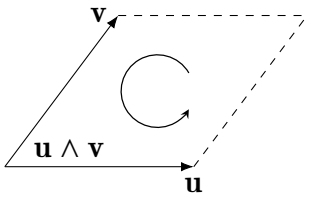
7.2 几何代数（Clifford 代数）

四元数与几何代数（Geometric Algebra）之间也有非常紧密的联系. 由于几何代数是一个庞大的体系，我们在这里无法详细解释每一个定义. 如果你对这部分内容感兴趣，可以到后面的参考资料中找到一些完整的教程. 在这里我们只关心它与四元数之间的关系.

在几何代数中，我们定义了两个向量的几何积（Geometric Product）. 如果我们有两个向量 $\mathbf{u}$ 和 $\mathbf{v}$，那么它们的几何积为：

$$\mathbf{uv} = \mathbf{u} \cdot \mathbf{v} + \mathbf{u} \wedge \mathbf{v}$$

其中，$\mathbf{u} \cdot \mathbf{v}$ 是向量的内积，也就是我们经常说的点积，而 $\mathbf{u} \wedge \mathbf{v}$（读作「u wedge v」）则是向量的外积（Exterior Product）. 外积与三维向量的叉积很类似，但是外积在任意维度都是有定义的. 并且，$\mathbf{u} \wedge \mathbf{v}$ 的结果不是一个普通的向量，它是一个二重向量（Bivector）. 我们不会在此详细讨论它，在这里你只需要知道它和三维向量的叉积一样能定义一个平面以及一个旋转的方向：



我们在四元数的讨论中看过一个非常类似的公式，如果我们有纯四元数 $u = [0, \mathbf{u}]$，$v = [0, \mathbf{v}]$，那么：

$$uv = [-\mathbf{u} \cdot \mathbf{v}, \mathbf{u} \times \mathbf{v}] = -\mathbf{u} \cdot \mathbf{v} + \mathbf{u} \times \mathbf{v}$$

除了点乘的系数差了个符号，其它基本完全相同.

在几何代数中，旋转是利用「两次反射等于一次旋转」这一定理来定义的．这个定理其实可以由我们上面定义的正交矩阵来理解．我们知道，反射变换矩阵其实就是行列式值为 -1 的正交矩阵．如果我们有两个反射变换 A 和 B，那么 $\det A = \det B = -1$，我们有 $\det(AB) = (\det A)(\det B) = (-1)^2 = 1$，并且 AB 同样是正交矩阵，所以 AB 是一个旋转.

如果我们需要将 $\mathbf{v}$ 反射到平面（或者超平面）N 的另外一边，而且我们有一个正交于 N 的单位法向量 $\mathbf{n}$，那么我们可以通过几何积的方式反射 $\mathbf{v}$：

$$\mathbf{v}' = -\mathbf{nvn}$$

如果我们进一步将这个向量反射到平面 M（用单位向量 $\mathbf{m}$ 定义）的另一侧，那么：

$$\begin{aligned}\mathbf{v}' &= -\mathbf{m}(-\mathbf{nvn})\mathbf{m} \\ &= (\mathbf{mn})\mathbf{v}(\mathbf{nm})\end{aligned}$$

在这里，我们定义 $R = \mathbf{mn}$，以及它的共轭 $R^\dagger = \mathbf{nm}$，所以：

$$\mathbf{v}' = R\mathbf{v}R^\dagger$$

这个结果和单位四元数的旋转非常类似．除此之外，我们也可以用指数的形式得出旋转的结果，如果我们有一个单位二重向量：

$$B = \frac{\mathbf{n} \wedge \mathbf{m}}{\sin(\frac{\theta}{2})}$$

其中 $\frac{\theta}{2}$ 是 $\mathbf{n}$ 和 $\mathbf{m}$ 的夹角，那么下面这个变换会将 $\mathbf{v}$ 在以 B 定义的旋转平面中旋转 θ 度：

$$\mathbf{v}' = e^{-B\frac{\theta}{2}}\mathbf{v}e^{B\frac{\theta}{2}}$$

我们知道四元数中也有类似的结论，但是这里是旋转更加通用的情况．我们在这里定义的是一个旋转平面而不是旋转轴．旋转轴仅仅是一个存在于二维和三维的特殊情况（因为正交轴和正交平面的数量相等），如果提升到四维空间，我们有 4 个正交轴以及 $\binom{4}{2} = 6$ 个正交平面，我们就必须将（类比于二维和三维的）旋转定义在一个平面之上了．

7.2.1 参考资料及拓展阅读

我们这里对几何代数的介绍非常粗略，如果你希望了解更多有关几何代数的知识，可以去读 Chris Doran 和 Anthony Lasenby 所著的书『Geometric Algebra for Physicists』．这本书还附有 Lecture 用的讲义．

如果你觉得这本书太长了，可以来阅读 Jaap Suter 写的『Geometric Algebra Primer』．

Mathoma 在 YouTube 上有一个几何代数的视频教程系列，虽然仍然在更新中，内容不是很全，但是他讲得非常通俗易懂．在他的频道上你也可以找到四元数的相关推导．

7.3 二重四元数

除了普通的四元数之外，几何代数中还衍生出来了一个二重四元数（Dual Quaternion）．它不仅能够表示 3D 旋转，还能够表示平移和缩放，所以有时候也被用于图形学中．

二重四元数同样可以表示为 $q = a + bi + cj + dk$，但与普通的四元数不同的是，这里的系数 a, b, c, d 都不再是实数，它们在这里是二重数（Dual Number）．一个二重数 z 可以被定义为：

$$z = a + b\epsilon \quad (\epsilon^2 = 0, \epsilon \neq 0)$$

所以，我们还可以将一个二重四元数写为：

$$q = r + d\epsilon \quad (r, d \in \mathbb{H})$$

其中，r 是这个二重四元数的实部，d 是这个二重四元数的虚部 (Dual Part)．这样的定义会给我们带来更多的性质，让我们能够表示其它的线性变换．

7.3.1 参考资料及拓展阅读

二重四元数相关的资料不是很多，基本都是论文：

首先是 Ladislav Kavan et al. 的论文以及代码：

- Dual Quaternions for Rigid Transformation Blending
- Skinning with Dual Quaternions
- 代码

除此之外，Yan-Bin Jia 写了一份关于二重四元数的讲义．

7.4 四元数

下面是关于四元数本身的参考资料和拓展阅读．

7.4.1 论文及文章

Ken Shoemake 最初提出 Slerp 插值的论文以及一篇介绍四元数的论文：

- Animating Rotation with Quaternion Curves
- Quaternions

Erik B. Dam，Martin Koch，Martin Lillholm 关于四元数插值所写的一篇总结性论文：

- Quaternions, Interpolation and Animation

我们之前所提到的 Squad 的证明以及中间控制点的推导可以在第 52-54 页中找到，对四元数导数的讨论可以在第 23-25 页找到，对 Slerp 以及其四种等价形式的讨论可以在 42-47 页找到．除此之外，这篇论文还讨论了其它的一些插值方法，可以在 56 页之后找到．

Jonathan Blow 对 Slerp 的一些讨论，它提到了 Slerp 的性能问题及一些优化：

- Hacking Quaternions
- Understanding Slerp, Then Not Using It

Euclidean Space 这个网站上提到了 3D 旋转变换在各种形式之间的转换和对比：

- Maths - Rotation conversions

矢野忠在他的期刊『数学・物理通信』中连载的文章对四元数的历史、数学以及应用进行了全面介绍．它从期刊的 1 卷 9 号开始断断续续连载，在 6 卷 10 号结束．期刊中的文章被整理成一本书，叫做『四元数の発見』，于 2014 年出版，但是其中大部分的文章都能够在期刊中找到．

David Eberly 在他的网站 Geometric Tools 上提供的一系列文章和代码：

- Quaternion Algebra and Calculus
- A Linear Algebraic Approach to Quaternions
- Constrained Quaternions Using Euler Angles
- Rotation Representations and Performance Issues
- 代码

7.4.2 书籍

Andrew Hanson 的书『Visualizing Quaternions』对四元数进行了非常详细的介绍．除了四元数在图形学中的应用之外，他还提供了很多的图帮助读者理解更抽象的主题．

John Vince 的书『Quaternions for Computer Graphics』关注点集中在四元数计算机图形学中的应用，但缺少对插值的讨论．

『Graphics Gems』这一系列的书中有很多对于四元数的讨论，但是比较分散．

7.4.3 代码

CesiumJS 库对于四元数的实现：

- Quaternion.js

虽然这个库是用 JavaScript 写的，但是它在四元数类中提供的功能非常全面，而且提供了一个 Slerp 的近似算法．

GLM 库对于四元数和二重四元数的实现，采用 C++ 实现：

- quaternion.inl
- dual_quaternion.inl

Magnum 库对于四元数和二重四元数的实现，采用 C++ 实现：

- Quaternion.h
- DualQuaternion.h

Direct3D 9 的四元数类文档，虽然没有代码，但是里面提到了很多实现的思路以及注意事项：

- Vectors, Vertices, and Quaternions (Direct3D 9)

8 附录：四元数的指数映射

这里我们来证明之前提到的公式：对任意单位向量 $\mathbf{u}$ 所对应的单位四元数 $u = [0, \mathbf{u}]$，有

$$e^{u\theta} = \cos(\theta) + u\sin(\theta)$$

Proof. 和矩阵与复数一样，四元数的指数也是使用级数来定义的.

我们知道，e^x，$\sin(x)$，$\cos(x)$ 分别能用泰勒级数展开为：

$$\begin{aligned}
e^x &= \sum_{n=0}^{\infty} \frac{x^n}{n!} = 1 + x + \frac{x^2}{2!} + \frac{x^3}{3!} + \frac{x^4}{4!} + \dots \\
\sin(x) &= \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!} x^{2n+1} = x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots \\
\cos(x) &= \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!} x^{2n} = 1 - \frac{x^2}{2!} + \frac{x^4}{4!} - \frac{x^6}{6!} + \dots
\end{aligned}$$

我们可以将 $u\theta$ 代入 e^x，得到：

$$\begin{aligned}
LHS = e^{u\theta} &= \sum_{n=0}^{\infty} \frac{(u\theta)^n}{n!} \\
&= 1 + u\theta + \frac{(u\theta)^2}{2!} + \frac{(u\theta)^3}{3!} + \frac{(u\theta)^4}{4!} + \frac{(u\theta)^5}{5!} \dots \\
&= 1 + u\theta + \frac{u^2\theta^2}{2!} + \frac{u^3\theta^3}{3!} + \frac{u^4\theta^4}{4!} + \frac{u^5\theta^5}{5!} + \dots
\end{aligned}$$

因为 $u^2 = [-\mathbf{u}\cdot\mathbf{u}, \mathbf{0}] = -\|\mathbf{u}\|^2 = -1$，我们可以对 $e^{u\theta}$ 进一步化简：

$$\begin{aligned} LHS &= 1 + u\theta + \frac{u^2\theta^2}{2!} + \frac{uu^2\theta^3}{3!} + \frac{(u^2)^2\theta^4}{4!} + \frac{u(u^2)^2\theta^5}{5!} + \dots \\ &= 1 + u\theta - \frac{\theta^2}{2!} - \frac{u\theta^3}{3!} + \frac{\theta^4}{4!} + \frac{u\theta^5}{5!} - \dots \end{aligned}$$

同理，将 θ 代入 $\cos(x)$ 和 $\sin(x)$，

$$\begin{aligned} \sin(\theta) &= \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n+1)!}\theta^{2n+1} \\ &= \theta - \frac{\theta^3}{3!} + \frac{\theta^5}{5!} - \frac{\theta^7}{7!} + \dots \\ \cos(\theta) &= \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!}\theta^{2n} \\ &= 1 - \frac{\theta^2}{2!} + \frac{\theta^4}{4!} - \frac{\theta^6}{6!} + \dots \end{aligned}$$

所以，

$$\begin{aligned} RHS &= \cos(\theta) + u\sin(\theta) \\ &= \sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!}\theta^{2n} + u\sum_{n=0}^{\infty} \frac{(-1)^n}{(2n)!}\theta^{2n} \\ &= 1 + u\theta - \frac{\theta^2}{2!} - \frac{u\theta^3}{3!} + \frac{\theta^4}{4!} + \frac{u\theta^5}{5!} - \dots = LHS \end{aligned}$$

□

我们在证明的其实是李代数 $\mathfrak{su}(2)$ 到李群 $SU(2)$ 的指数映射．我们之所以能够证明它是因为 $u^2 = -1$，与复数的 $i^2 = -1$ 非常类似．与复数不同的是 $x^2 = -1$ 这个方程在四元数中是有无数个解的．比如说我们所用到的 $u = [0, \mathbf{u}]$，只要 $\mathbf{u}$ 是一个单位向量，u 就是这个方程的一个解．