Модуль 2 «Математические методы, модели и алгоритмы компьютерной геометрии»
Лекция 8 «Геометрические преобразования в трёхмерном пространстве»

к.ф.-м.н., доц. каф. ФН-11, Захаров Андрей Алексеевич,
ауд.:930а(УЛК)
моб.: 8-910-461-70-04,
email: azaharov@bmstu.ru

МГТУ им. Н.Э. Баумана

4 ноября 2025 г.

Методы геометрических преобразований в трёх измерениях являются развитием двухмерных методов, в которые добавляется ещё одна координата z. Перемещение объекта выполняется с помощью заданного трёхмерного вектора перемещения, который определяет, насколько нужно переместить объект по каждой из трёх координат. Аналогично, чтобы масштабировать объект, выбирают масштабный коэффициент для каждой из трёх декартовых координат. Базовые трёхмерные повороты получаются расширением двухмерных.

Трёхмерное положение, выраженное в однородных координатах, представляется четырехэлементным вектором-столбцом. Следовательно, каждый оператор геометрического преобразования — это матрица 4×4, которая умножается слева на вектор-столбец. Кроме того (как и в двух измерениях), любая последовательность преобразований представляется одной матрицей, сформированной последовательной сверткой матриц отдельных преобразований. Каждая следующая матрица последовательности преобразований действует слева на предыдущую матрицу преобразования.

Трёхмерное перемещение

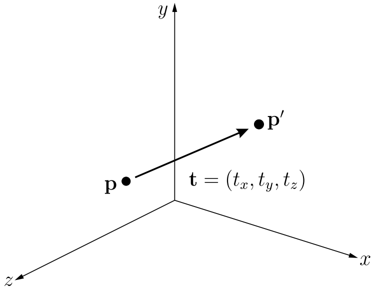


Рис.: Перемещение точки на вектор $\mathbf{t} = (t_x, t_y, t_z)$

Точка $\mathbf{p} = (x, y, z)$ трёхмерного пространства перемещается в точку $\mathbf{p}' = (x', y', z')$ путём прибавления компонент вектора перемещения t_x, t_y и t_z к декартовым координатам точки $\mathbf{p}$:

$$x' = x + t_x, \qquad y' = y + t_y, \qquad z' = z + t_z. \tag{1}$$

Трёхмерное перемещение

Трёхмерное перемещение можно выразить в матричной форме. Точки $\mathbf{p}$ и $\mathbf{p}'$ представляются в однородных координатах четырехэлементными векторами-столбцами, а матрица перемещения $\mathbf{T}$ имеет вид:

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & 1 & 0 & t_y \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \tag{2}$$

или

$$\mathbf{p}' = \mathbf{T} \cdot \mathbf{p}. \tag{3}$$

Трёхмерное перемещение

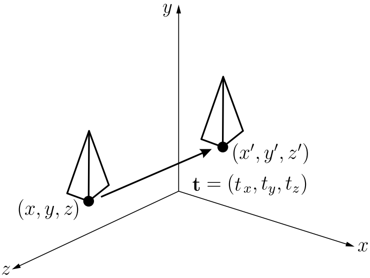


Рис.: Перемещение трёхмерного объекта с использованием вектора перемещения $\mathbf{t}$

Чтобы переместить объект в трёх измерениях, нужно преобразовать все координаты объекта, затем визуализировать объект в новом положении. Если объект представлен как набор многоугольных поверхностей, то перемещается каждая вершина каждой поверхности, и в конечных положениях строятся многоугольные грани.

Трёхмерное перемещение

Матрица, обратная к матрице трёхмерного перемещения, получается с использованием тех же процедур, что применялись для двухмерного перемещения. Следовательно, меняется знак у компонент вектора перемещения t_x, t_y и t_z. В результате получается перемещение в противоположном направлении, причём произведение матрицы перемещения и обратной к ней равно единичной матрице.

Трёхмерный поворот

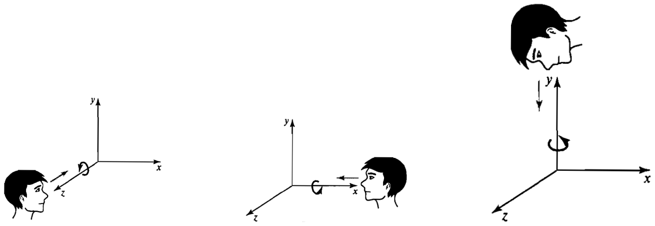


Рис.: Положительные повороты вокруг координатной оси выполняются против часовой стрелки, если смотреть вдоль положительной половины оси к началу координат

Двухмерные повороты на плоскости xy требовали рассмотрения только поворотов вокруг осей, перпендикулярных плоскости xy. В трёхмерном пространстве можно выбирать любую пространственную ориентацию осей поворота. Легче всего рассчитывать повороты вокруг осей, параллельных декартовым осям, поэтому, вначале рассмотрим операции, фигурирующие при поворотах вокруг координатных осей.

Будем считать, что положительные углы поворота дают повороты против часовой стрелки вокруг координатной оси при условии, что мы смотрим в отрицательном направлении координатной оси.

Трёхмерные повороты вокруг координатной оси

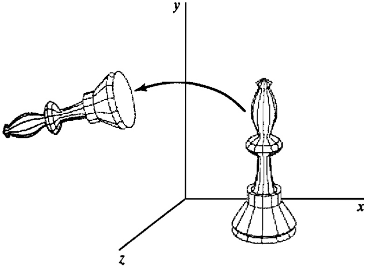


Рис.: Поворот объекта вокруг оси z

Уравнения двухмерного *поворота вокруг оси* z легко распространяются на три измерения:

$$\begin{aligned} x' &= x\cos\theta - y\sin\theta, \\ y' &= x\sin\theta + y\cos\theta, \\ z' &= z. \end{aligned} \tag{4}$$

Параметр θ задаёт угол поворота вокруг оси z, а значения координаты z при этом преобразовании не меняются.

Уравнения трёхмерного поворота вокруг оси z записываются через однородные координаты следующим образом:

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}, \tag{5}$$

что можно записать более компактно:

$$\mathbf{p}' = \mathbf{R}_z(\theta) \cdot \mathbf{p}. \tag{6}$$

Трёхмерные повороты вокруг координатной оси

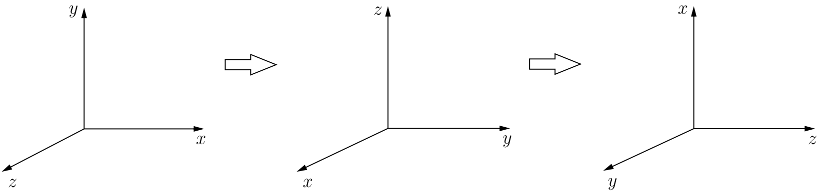


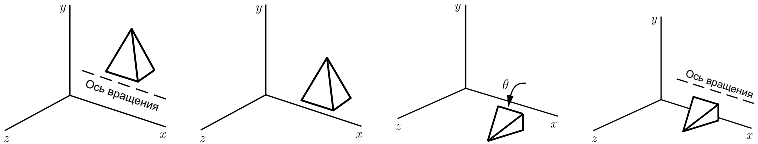
Рис.: Циклическая перестановка декартовых осей, дающая три набора уравнений поворота вокруг координатных осей

Уравнения преобразования для поворотов вокруг двух других координатных осей можно получить с помощью циклической перестановки параметров x, y и z в уравнениях (4):

$$x \to y \to z \to x. \tag{7}$$

Матрица, обратная к матрице трёхмерного поворота, получается аналогично матрице обратного поворота в двух измерениях. Нужно всего лишь заменить угол θ углом $-\theta$. Отрицательные значения углов поворота вращают тело по часовой стрелке, и при умножении матрицы поворота на обратную к ней получается единичная матрица. Матрицу, обратную к любой матрице поворота $\mathbf{R}$, можно найти, вычислив транспонированную матрицу $(\mathbf{R}^{-1} = \mathbf{R}^T)$.

Поворот вокруг оси, параллельной одной из координатных осей



а) исходное положение объекта

б) перемещение оси вращения до совмещения с осью x

в) поворот объекта на угол θ

г) перемещение оси вращения в исходное положение

Рис.: Последовательность преобразований при повороте объекта вокруг оси, параллельной оси x

Если нужно повернуть объект вокруг оси, параллельной одной из координатных осей, требуется выполнить следующие действия:

1. так переместить объект, чтобы ось вращения совпала с параллельной координатной осью;
2. выполнить заданный поворот вокруг этой оси;
3. так переместить объект, чтобы ось вращения вернулась в исходное положение.

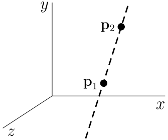
С помощью этой последовательности точка $\mathbf{p}$ преобразуется следующим образом: $\mathbf{p}' = \mathbf{R}(\theta) \cdot \mathbf{p} = \mathbf{T}^{-1} \cdot \mathbf{R}_x(\theta) \cdot \mathbf{T} \cdot \mathbf{p}$.

Произвольные трёхмерные повороты

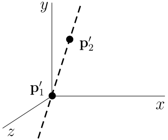
Когда нужно повернуть объект вокруг оси, не параллельной ни одной из координатных осей, требуются дополнительные преобразования. Они включают в себя повороты, выравнивающие ось вращения с выбранной координатной осью, а также возврат оси поворота к исходной ориентации. Для общего случая положения оси вращения, требуемый поворот можно выполнить за пять шагов:

1. переместить объект так, чтобы ось вращения прошла через начало координат;
2. так повернуть объект, чтобы ось вращения совместилась с одной из координатных осей;
3. выполнить заданный поворот вокруг выбранной координатной оси;
4. применить обратные повороты для возврата оси вращения к исходной ориентации;
5. применить обратное перемещение, чтобы вернуть ось вращения к исходному пространственному положению.

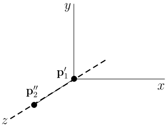
Ось вращения можно совместить с любой из трёх координатных осей. Часто удобно использовать ось z. Рассмотрим последовательность преобразований с использованием матрицы поворота вокруг оси z.



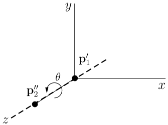
а) Исходное положение



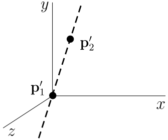
б) Этап 1. Переместить $\mathbf{p_1}$ в начало координат



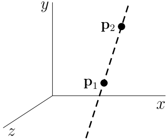
в) Этап 2. Повернуть $\mathbf{p}_2'$ до попадания на ось z



г) Этап 3. Повернуть объект вокруг оси z

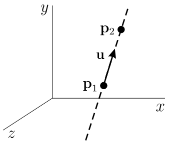


д) Этап 4. Повернуть ось в начальное положение



е) Этап 5. Переместить ось в исходное положение

Рис.: Пять этапов преобразования, при выполнении которых находится сложная матрица поворота вокруг произвольной оси. Ось вращения совмещается с осью z



Ось вращения можно определить двумя точками, или одной точкой и направляющими углами (или направляющими косинусами), которые ось вращения образует с координатными осями. Предположим, что ось вращения определяется двумя точками, и что направление поворота — против часовой стрелки, если смотреть по оси в направлении от $\mathbf{p}_2$ к $\mathbf{p}_1$. Компоненты вектора оси вращения вычисляются следующим образом:

$$\mathbf{v} = \mathbf{p}_2 - \mathbf{p}_1 = (x_2 - x_1, y_2 - y_1, z_2 - z_1). \tag{8}$$

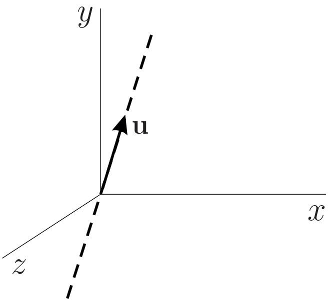
Единичный вектор оси вращения $\mathbf{u}$ записывается так:

$$\mathbf{u} = \frac{\mathbf{v}}{|\mathbf{v}|} = (a, b, c), \tag{9}$$

где компоненты a, b и c — направляющие косинусы оси вращения:

$$a = \frac{x_2 - x_1}{|\mathbf{v}|}, \qquad b = \frac{y_2 - y_1}{|\mathbf{v}|}, \qquad c = \frac{z_2 - z_1}{|\mathbf{v}|}. \tag{10}$$

Произвольные трёхмерные повороты



Первый шаг при выполнении поворота — задать матрицу перемещения, так переносящую ось вращения, чтобы она проходила через начало координат. Поскольку требуется, чтобы вращение выполнялось против часовой стрелки, если смотреть вдоль оси от $\mathbf{p}_2$ к $\mathbf{p}_1$, в начало координат перемещается точка $\mathbf{p}_1$. Данная матрица перемещения записывается так:

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & -x_1 \\ 0 & 1 & 0 & -y_1 \\ 0 & 0 & 1 & -z_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{11}$$

Произвольные трёхмерные повороты

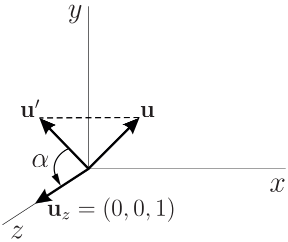
а) поворот вокруг оси x до плоскости xz

б) поворот вокруг оси y до совмещения с осью z

Рис.: Преобразования единичного вектора $\mathbf{u}$

Далее нужно сформулировать преобразования, которые совместят ось вращения с осью z. Чтобы выполнить это совмещение в два этапа, можно использовать повороты вокруг координатных осей. В данном примере мы вначале выполним поворот вокруг оси x, а затем вокруг оси y. Поворот вокруг оси x переводит вектор $\mathbf{u}$ на плоскость xz, а поворот вокруг оси y совмещает $\mathbf{u}$ с осью z.

В расчёте поворота фигурируют функции синус и косинус. Чтобы найти косинус, можно использовать скалярное произведение векторов, а для расчёта синуса применяется векторное произведение.



Чтобы записать матрицу преобразования, характеризующую поворот вокруг оси x, нужно определить синус и косинус угла поворота, необходимого для перевода вектора $\mathbf{u}$ на плоскость xz. Данный угол поворота образуется проекцией $\mathbf{u}$ на плоскость yz и положительным направлением оси z. Если представить проекцию $\mathbf{u}$ на плоскость yz как вектор $\mathbf{u}' = (0, b, c)$, тогда косинус угла поворота α можно определить по скалярному произведению $\mathbf{u}'$ и единичного вектора $\mathbf{u}_z$, идущего вдоль оси z:

$$\cos\alpha = \frac{\mathbf{u}' \cdot \mathbf{u}_z}{|\mathbf{u}'||\mathbf{u}_z|} = \frac{c}{d}, \tag{12}$$

где d — модуль вектора $\mathbf{u}'$.

Подобным образом синус α можно определить по векторному произведению $\mathbf{u}'$ и $\mathbf{u}_z$:

$$\mathbf{u}' \times \mathbf{u}_z = \mathbf{u}_x |\mathbf{u}'||\mathbf{u}_z| \sin\alpha, \tag{13}$$

которое в компонентном виде выглядит следующим образом:

$$\mathbf{u}' \times \mathbf{u}_z = \mathbf{u}_x \cdot b. \tag{14}$$

Приравнивая правые части уравнений (13) и (14) и, учитывая, что $|\mathbf{u}_z| = 1$ и $|\mathbf{u}'| = d$, получаем:

$$d \sin\alpha = b$$

или

$$\sin\alpha = \frac{b}{d}. \tag{15}$$

Теперь значения $\cos\alpha$ и $\sin\alpha$ определены через компоненты вектора $\mathbf{u}$, и можно задавать элементы матрицы поворота этого вектора вокруг оси x:

$$\mathbf{R}_x(\alpha) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \frac{c}{d} & -\frac{b}{d} & 0 \\ 0 & \frac{b}{d} & \frac{c}{d} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{16}$$

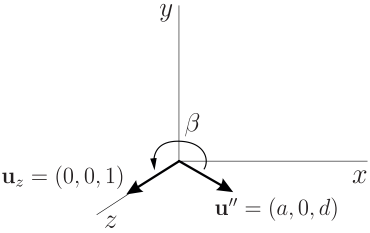


Рис.: Поворот единичного вектора $\mathbf{u}''$ (вектор $\mathbf{u}$ после поворота на плоскость xz) вокруг оси y на угол β совмещает $\mathbf{u}''$ с вектором $\mathbf{u}_z$

Следующий этап формирования последовательности преобразований состоит в определении матрицы, под действием которой единичный вектор, находящийся в плоскости xz, поворотом против часовой стрелки вокруг y совмещается с положительным направлением оси z. Обозначим его $\mathbf{u}''$, его x-я компонента равна a, поскольку поворот вокруг оси x не меняет значение компоненты x. Компонент z вектора равен d (модуль $\mathbf{u}'$). Наконец, компонент y вектора $\mathbf{u}''$ — 0, поскольку вектор сейчас лежит в плоскости xy. Косинус угла поворота β можно определить по скалярному произведению единичных векторов $\mathbf{u}''$ и $\mathbf{u}_z$.

Произвольные трёхмерные повороты

Следовательно,

$$\cos\beta = \frac{\mathbf{u}'' \cdot \mathbf{u}_z}{|\mathbf{u}''||\mathbf{u}_z|} = d, \tag{17}$$

поскольку $|\mathbf{u}_z| = |\mathbf{u}''| = 1$. Сравнивая векторное произведение

$$\mathbf{u}'' \times \mathbf{u}_z = \mathbf{u}_y|\mathbf{u}''||\mathbf{u}_z|\sin\beta \tag{18}$$

с представлением в компонентном виде

$$\mathbf{u}'' \times \mathbf{u}_z = \mathbf{u}_y(-a), \tag{19}$$

получаем, что

$$\sin\beta = -a. \tag{20}$$

Следовательно, матрица преобразования, характеризующая поворот вектора $\mathbf{u}''$ вокруг оси y, равна

$$\mathbf{R}_y(\beta) = \begin{bmatrix} d & 0 & -a & 0 \\ 0 & 1 & 0 & 0 \\ a & 0 & d & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{21}$$

Произвольные трёхмерные повороты

С помощью матриц (11), (16) и (21) мы выровняли ось вращения по положительному направлению оси z. Теперь поворот на заданный угол θ можно выполнить вокруг оси z:

$$\mathbf{R}_z(\theta) = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \tag{22}$$

Чтобы завершить требуемый поворот вокруг данной оси, нужно вернуть ось вращения в исходное положение. Для этого применяются преобразования, обратные к (11), (16) и (21). Матрицу преобразования, описывающую поворот вокруг произвольной оси, можно выразить через совокупность семи отдельных преобразований:

$$\mathbf{R}(\theta) = \mathbf{T}^{-1} \cdot \mathbf{R}_x^{-1}(\alpha) \cdot \mathbf{R}_y^{-1}(\beta) \cdot \mathbf{R}_z(\theta) \cdot \mathbf{R}_y(\beta) \cdot \mathbf{R}_x(\alpha) \cdot \mathbf{T}. \tag{23}$$

Трёхмерное масштабирование

Матричное выражение трёхмерного масштабирования точки $\mathbf{P} = (x, y, z)$ относительно начала координат является простым расширением двухмерного масштабирования. В матрицу преобразования просто включается параметр, отвечающий за масштабирование по оси z:

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \begin{bmatrix} s_x & 0 & 0 & 0 \\ 0 & s_y & 0 & 0 \\ 0 & 0 & s_z & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}. \tag{24}$$

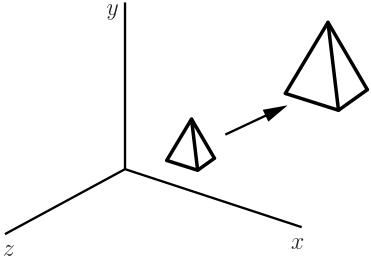
Преобразование трёхмерного масштабирования точки можно представить как

$$\mathbf{P}' = \mathbf{S} \cdot \mathbf{P}, \tag{25}$$

где масштабным коэффициентам s_x, s_y и s_z присваиваются любые положительные значения. Явные выражения для преобразования масштабирования относительно начала координат записываются так:

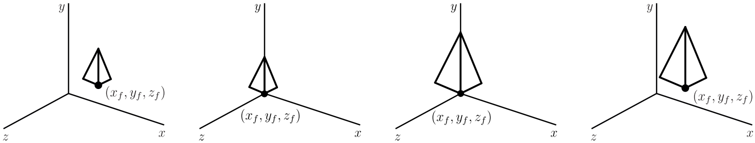
$$x' = x \cdot s_x, \qquad y' = y \cdot s_y \qquad z' = z \cdot s_z. \tag{26}$$

Трёхмерное масштабирование



Масштабирование объекта с помощью преобразования (24) меняет положение объекта относительно начала координат. При коэффициенте масштабирования, превышающем 1, точка удаляется от начала координат в соответствующем направлении. Если же коэффициент меньше 1, точка приближается к началу координат по этой координате. Кроме того, если не все коэффициенты масштабирования равны, меняются относительные размеры объекта. Чтобы сохранить исходную форму объекта, используется пропорциональное масштабирование: $s_x = s_y = s_z$.

Трёхмерное масштабирование



а) исходное положение б) перемещение в) масштабирование г) обратное перемещение

Рис.: Последовательность преобразований при масштабировании объекта относительно выбранной фиксированной точки с использованием уравнения (24)

Хотя некоторые графические пакеты предлагают только процедуры, масштабирующие объект относительно начала координат, всегда можно построить преобразование масштабирования относительно любой выбранной неподвижной точки (x_f, y_f, z_f), применив для этого следующую последовательность операций:

1. Переместить неподвижную точку в начало координат.
2. Применить преобразование масштабирования относительно начала координат, используя уравнение (24).
3. Переместить неподвижную точку обратно.

Матричное представление масштабирования относительно произвольной неподвижной точки можно выразить как свертку указанных преобразований перемещения–масштабирования–перемещения:

$$\mathbf{T}(x_f, y_f, z_f) \cdot \mathbf{S}(s_x, s_y, s_z) \cdot \mathbf{T}(-x_f, -y_f, -z_f) = \begin{bmatrix} s_x & 0 & 0 & (1-s_x)x_f \\ 0 & s_y & 0 & (1-s_y)y_f \\ 0 & 0 & s_z & (1-s_z)z_f \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{27}$$

Обратная матрица трёхмерного масштабирования для уравнения (24) или уравнения (27) задаётся заменой каждого масштабного коэффициента s_x, s_y и s_z обратной величиной. Однако данное обратное преобразование не определено, если любому из коэффициентов присвоено значение 0. Обратная матрица генерирует противоположное преобразование масштабирования, и свертка матрицы трёхмерного масштабирования с обратной даёт единичную матрицу.

Сложные трёхмерные преобразования

Как и при двухмерных преобразованиях, матрица сложного трёхмерного преобразования формируется умножением матричных представлений отдельных операций последовательности преобразований. Последовательность преобразований можно реализовать путём свертки отдельных матриц справа налево или слева направо в зависимости от порядка, в котором заданы матричные представления. Крайний справа член произведения матриц всегда является первым преобразованием, применяемым к объекту, а крайний слева — последним.

Другие трёхмерные преобразования

Помимо перемещения, поворота и масштабирования во многих трёхмерных задачах используются другие преобразования, такие как отражение и сдвиг.

Трёхмерные отражения

Отражение в трёхмерном пространстве можно выполнить относительно выбранной оси отражения или относительно плоскости отражения. Отражение относительно данной оси эквивалентно повороту на 180° вокруг этой оси.
Матричное представление отражения относительно плоскости xy, которое меняет знак координаты z, записывается так:

$$\mathbf{M}_{z\mathsf{reflect}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Матрицы отражений относительно плоскостей yz или xz, выполняющие инверсию координат x или y, определяются аналогично. Отражения в других плоскостях можно получить как комбинации поворотов и отражений в координатных плоскостях.

Трёхмерные сдвиги

Трёхмерные сдвиги используются для изменения форм объектов, аналогично двухмерному случаю. Они также применяются в трёхмерных преобразованиях наблюдения.

Произвольное преобразование сдвига вдоль оси z относительно опорной точки z_{ref} выражается следующей матрицей преобразования:

$$\mathbf{M}_{z\text{shear}} = \begin{bmatrix} 1 & 0 & \lambda_x & -\lambda_x z_{\text{ref}} \\ 0 & 1 & \lambda_y & -\lambda_y z_{\text{ref}} \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Параметры сдвига λ_x и λ_y могут принимать любые действительные значения. Под действием этой матрицы преобразования значения координат x и y меняются на величину, пропорциональную расстоянию от z_{ref}, координата z при этом не меняется.

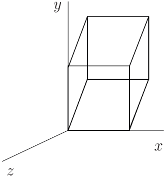
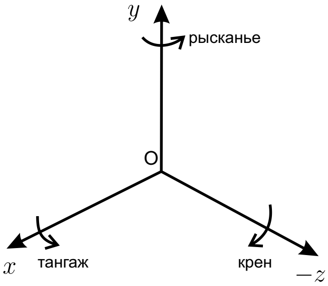
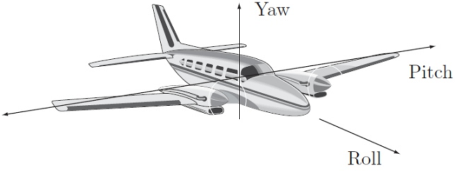


Рис.: Пример действия матрицы сдвига, где $\lambda_x = \lambda_y = 1$ и $z_{\text{ref}} = 0$

Углы Эйлера





Как было показано, поворот вокруг любой оси можно свести к поворотам вокруг трёх координатных осей. Следовательно для указания ориентации в пространстве можно использовать три угла поворота, которые названы в честь великого математика Леонарда Эйлера. Обычно направлением просмотра (в котором смотрит камера) является отрицательное направление оси z, поэтому вращение происходит вокруг осей $-z$, x и y. Угол поворота вокруг оси $-z$ называется углом крена (roll или bank, $-\pi < r \leqslant \pi$), поворот угол вокруг оси x называется углом тангажа (pitch, $-\frac{\pi}{2} \leqslant p \leqslant \frac{\pi}{2}$), а угол поворота вокруг оси y называется углом рысканья (yaw или heading, $-\pi < h \leqslant \pi$).

Углы Эйлера

$$\mathbf{E}(h,p,r) = \mathbf{R}_{-z}(r)\cdot\mathbf{R}_x(p)\cdot\mathbf{R}_y(h) = \tag{28}$$

$$= \begin{pmatrix} \cos r\cos h - \sin r\sin p\sin h & -\sin r\cos p & -\sin h\cos r - \sin r\sin p\cos h \\ \sin r\cos h + \cos r\sin p\sin h & \cos r\cos p & -\sin r\sin h + \cos r\sin p\cos h \\ \cos p\sin h & -\sin p & \cos p\cos h \end{pmatrix}. \tag{29}$$

Матрица $\mathbf{E}$, определяющая ориентацию, определяется произведением матриц вращения, и является также матрицей вращения, для которой $\mathbf{E}^{-1} = \mathbf{E}^T$.

Представление ориентации с помощью углов Эйлера имеет один серьезный недостаток — при определённых комбинациях углов одна степень свободы может потеряться. Эта ситуация называется шарнирным замком (*gimbal lock*).

При $p = \pi/2$ матрица $\mathbf{E}(h,\pi/2,r)$ зависит только от $r+h$:

$$\mathbf{E}(h,\pi/2,r) = \begin{pmatrix} \cos(r+h) & 0 & -\sin(r+h) \\ \sin(r+h) & 0 & -\cos(r+h) \\ 0 & -1 & 0 \end{pmatrix}. \tag{30}$$

Если мы знаем элементы матрицы $\mathbf{E}(h, p, r)$, можно извлечь из неё углы h, p и r. Запишем $\mathbf{E}$ в общем виде

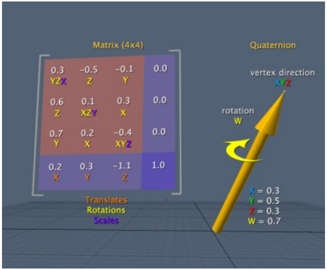
$$\mathbf{E} = \begin{pmatrix} e_{11} & e_{12} & e_{13} \\ e_{21} & e_{22} & e_{23} \\ e_{31} & e_{32} & e_{33} \end{pmatrix}. \tag{31}$$

Если мы сравним матрицы в (31) и (29), мы сразу увидим, что

$$\begin{aligned} e_{32} &= -\sin p, \\ \frac{e_{12}}{e_{22}} &= \frac{-\sin r \cos p}{\cos r \cos p} = -\tan r, \\ \frac{e_{31}}{e_{33}} &= \frac{\cos p \sin h}{\cos p \cos h} = \tan h \end{aligned}$$

Заметим, что существует специальный случай $\cos p = 0$, который соответствует приведённому выше случаю шарнирного замка. В этом случае, можно извлечь только значение $r + h$ из матрицы $\mathbf{E}(h, \pi/2, r)$. Чтобы разрешить эту неоднозначность, возникающую при $p = \pm\pi/2$, примем, что в этом случае $h = 0$, и тогда r просто извлекается из (30).

Кватернионы



Более эффективным методом генерации поворота вокруг выбранной произвольной оси является использование кватернионного представления в преобразовании поворота, аналогично использованию комплексных чисел для поворота векторов в плоскости. Кватернионы полезны во множестве процедур компьютерной графики, включая генерацию фрактальных объектов. Они требуют меньше памяти для хранения, чем матрицы 4×4, и для последовательностей преобразований процедуры с использованием кватернионов пишутся легче. Это особенно важно в анимации, где часто нужны изощренные последовательности движения и интерполяция с помощью движения двух данных положений объекта.

Кватернионы

Ирландский математик Уильям Гамильтон искал способ расширить комплексные числа от 2D к 3D в течение многих лет. Этот новый тип комплексного числа, подумал он, будет иметь одну действительную часть и две мнимые части. Тем не менее, Гамильтон не смог создать полезный тип комплексного числа с двумя мнимыми частями. Тогда, как гласит история, на своем пути к выступлению в Ирландской королевской академии, его вдруг осенило, что необходимы три мнимые части, а не две. На перилах Брукгамского моста через Королевский канал он высек уравнения, определяющие свойства этого нового типа комплексного числа. Таким образом, были изобретены кватернионы. Сейчас формула на мосту сделана каменной.

Кватернионы — это величины с одним действительным и тремя мнимыми компонентами:

$$\mathbf{q} = s + ia + jb + kc, \tag{32}$$

где коэффициенты a, b и c — действительные числа, а параметр s — действительное число, называемое скалярной частью. Т.о. при работе с кватернионами приходится работать с четырьмя числами, отсюда происходит и их название (от лат. quaterni, по четыре). Три мнимые части i, j, и k связаны следующим образом:

$$i^2 = j^2 = k^2 = -1, \qquad ij = -ji = k. \tag{33}$$

Из этих свойств следует, что

$$jk = -kj = i, \qquad ki = -ik = j. \tag{34}$$

Можно также использовать для записи кватернионов следующую форму упорядоченной записи, состоящей из *скалярной части* и *векторной части*, подобную упорядоченному представлению комплексного числа:

$$\mathbf{q} = (s, \mathbf{v}). \tag{35}$$

Параметр $\mathbf{v}$ в этом представлении — вектор (a, b, c).

Многие свойства обычных комплексных чисел также применимы к кватернионам. Умножение кватерниона на скаляр определяется по аналогии с соответствующей операцией для векторов и комплексных чисел. Для заданного скаляра k и кватерниона $\mathbf{q}$:

$$k\mathbf{q} = k\,(s, \mathbf{v}) = (ks, k\mathbf{v})\,.$$

Следовательно, все четыре компонента кватерниона умножаются на скалярное значение.
Сложение кватернионов определяется как сложение соответствующих элементов:

$$\mathbf{q}_1 + \mathbf{q}_2 = (s_1 + s_2) + i\,(a_1 + a_2) + j\,(b_1 + b_2) + k\,(c_1 + c_2)\,. \tag{36}$$

Используя упорядоченную форму записи, сложение кватернионов можно выразить следующим образом:

$$\mathbf{q}_1 + \mathbf{q}_2 = (s_1 + s_2, \mathbf{v}_1 + \mathbf{v}_2)\,. \tag{37}$$

Операция сложения коммутативна и ассоциативна.

Произведение кватернионов

Кватернионы могут быть перемножены в соответствии с их комплексной интерпретацией:

$$
\begin{gathered}
(s_1 + a_1 i + b_1 j + c_1 k)(s_2 + a_2 i + b_2 j + c_2 k) = \\
= s_1 s_2 + s_1 a_2 i + s_1 b_2 j + s_1 c_2 k + a_1 s_2 i + a_1 a_2 i^2 + a_1 b_2 ij + a_1 c_2 ik + \\
+ b_1 s_2 j + b_1 a_2 ji + b_1 b_2 j^2 + b_1 c_2 jk + c_1 s_2 k + c_1 a_2 ki + c_1 b_2 kj + c_1 c_2 k^2 = \\
= s_1 s_2 + s_1 a_2 i + s_1 b_2 j + s_1 c_2 k + a_1 s_2 i + a_1 a_2 (-1) + a_1 b_2 k + a_1 c_2 (-j) + \\
+ b_1 s_2 j + b_1 a_2 (-k) + b_1 b_2 (-1) + b_1 c_2 i + c_1 s_2 k + c_1 a_2 j + c_1 b_2 (-i) + c_1 c_2 (-1) = \\
= s_1 s_2 - a_1 a_2 - b_1 b_2 - c_1 c_2 + (s_1 a_2 + a_1 s_2 + b_1 c_2 - c_1 b_2)\, i + \\
+ (s_1 b_2 + b_1 s_2 + c_1 a_2 - a_1 c_2)\, j + (s_1 c_2 + c_1 s_2 + a_1 b_2 - b_1 a_2)\, k.
\end{gathered}
$$

Это приводит нас к стандартному определению умножения кватернионов:

$$
\mathbf{q}_1 \mathbf{q}_2 = (s_1 s_2 - \mathbf{v}_1 \cdot \mathbf{v}_2, s_1 \mathbf{v}_2 + s_2 \mathbf{v}_1 + \mathbf{v}_1 \times \mathbf{v}_2)\,. \tag{38}
$$

Операция умножения кватернионов ассоциативна, но не коммутативна:

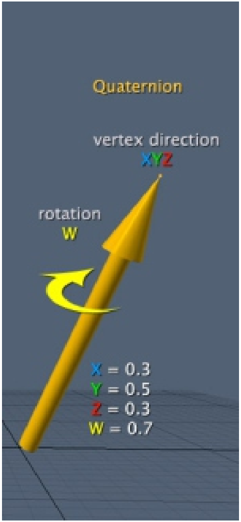
$$
\begin{gathered}
(\mathbf{q}_1 \mathbf{q}_2)\, \mathbf{q}_3 = \mathbf{q}_1 (\mathbf{q}_2 \mathbf{q}_3)\,, \\
\mathbf{q}_1 \mathbf{q}_2 \neq \mathbf{q}_2 \mathbf{q}_1.
\end{gathered} \tag{39}
$$

Кватернион, представляющий поворот

Кватернионы могут использоваться для представления поворотов в трёхмерном пространстве.
Поворот вокруг любой оси, проходящей через начало координат, выполняется с помощью кватерниона с такими скалярной и векторной частями:

$$s = \cos\frac{\theta}{2}, \qquad \mathbf{v} = \sin\frac{\theta}{2}\mathbf{u}, \tag{40}$$

где $\mathbf{u}$ — единичный вектор, направленный по заданной оси вращения, а θ — заданный угол поворота вокруг этой оси.
Если $\mathbf{q_1} = \left(\cos\frac{\theta}{2}, \sin\frac{\theta}{2}\mathbf{u}\right)$ и $\mathbf{q_2} = \left(\cos\frac{\varphi}{2}, \sin\frac{\varphi}{2}\mathbf{u}\right)$, то $\mathbf{q_1}\mathbf{q_2} = \left(\cos\frac{\theta+\varphi}{2}, \sin\frac{\theta+\varphi}{2}\mathbf{u}\right)$



Единичный кватернион(ы)

С геометрической точки зрения, есть два «идентичных» кватерниона, которые дают нулевой поворот: $(1, \mathbf{0})$ и $(-1, \mathbf{0})$, где $\mathbf{0}$ — нулевой вектор. Когда θ кратно 360° и частное — чётное число, то $\cos(\theta/2) = 1$, и мы получаем первую форму. Если θ кратно 360° и частное — нечётное число, то $\cos(\theta/2) = -1$, и мы получаем вторую форму. В обоих случаях, $\sin(\theta/2) = 0$, так что величина $\mathbf{u}$ не имеет значения.
С алгебраической точки зрения, существует только один единичный кватернион: $(1, \mathbf{0})$. Когда мы умножаем любой кватернион $\mathbf{q}$ на единичный кватернион, получаем $\mathbf{q}$. Когда мы умножаем кватернион $\mathbf{q}$ на другой «геометрически эквивалентный» кватернион $(-1, \mathbf{0})$, мы получаем $-\mathbf{q}$. Геометрически это один и тот же кватернион, так как $\mathbf{q}$ и $-\mathbf{q}$ представляют один и тот же поворот:

$$\left(-\cos\frac{\theta}{2}, -\sin\frac{\theta}{2}\mathbf{u}\right) = \left(\cos\left(\pi + \frac{\theta}{2}\right), \sin\left(\pi + \frac{\theta}{2}\right)\mathbf{u}\right) =$$
$$= \left(\cos\left(\frac{2\pi + \theta}{2}\right), \sin\left(\frac{2\pi + \theta}{2}\right)\mathbf{u}\right).$$

Математически, $\mathbf{q}$ и $-\mathbf{q}$ не равны, поэтому $(-1, \mathbf{0})$ не является «истинным» единичным кватернионом.

Модуль кватерниона

Модуль кватерниона вычисляется так же, как для векторов и комплексных чисел:

$$\begin{aligned} \|\mathbf{q}\| &= \|s + ia + jb + kc\| = \sqrt{s^2 + a^2 + b^2 + c^2} && (41) \\ &= \|(s, \mathbf{v})\| = \sqrt{s^2 + \|\mathbf{v}\|^2} && (42) \end{aligned}$$

Вычислим чему равен модуль кватерниона, представляющего поворот. Подставляя θ и $\mathbf{u}$:

$$\|\mathbf{q}\| = \sqrt{\cos^2\left(\frac{\theta}{2}\right) + \left(\sin\left(\frac{\theta}{2}\right)\|\mathbf{u}\|\right)^2} = \sqrt{\cos^2\left(\frac{\theta}{2}\right) + \sin^2\left(\frac{\theta}{2}\right)\|\mathbf{u}\|^2}$$

Учитывая, что $\mathbf{u}$ имеет единичную длину, получаем:

$$\|\mathbf{q}\| = \sqrt{\cos^2\left(\frac{\theta}{2}\right) + \sin^2\left(\frac{\theta}{2}\right)}$$

Используя тождество $\sin^2 x + \cos^2 x = 1$:

$$\|\mathbf{q}\| = \sqrt{\cos^2\left(\frac{\theta}{2}\right) + \sin^2\left(\frac{\theta}{2}\right)} = \sqrt{1} = 1$$

Таким образом, кватернионы для представления поворота всегда имеют единичную длину и образуют сферу S^3 в $\mathbb{R}^4$.

Модуль кватерниона

Вычислим значение модуля произведения двух кватернионов.

$$\|\mathbf{q}_1\mathbf{q}_2\| = \sqrt{\begin{matrix} (s_1s_2 - a_1a_2 - b_1b_2 - c_1c_2)^2 + (s_1a_2 + a_1s_2 + b_1c_2 - c_1b_2)^2 + \\ + (s_1b_2 + b_1s_2 + c_1a_2 - a_1c_2)^2 + (s_1c_2 + c_1s_2 + a_1b_2 - b_1a_2)^2 \end{matrix}}$$

Упрощая данное выражение, получаем:

$$\|\mathbf{q}_1\mathbf{q}_2\| = \sqrt{\begin{matrix} s_1^2s_2^2 + a_1^2a_2^2 + b_1^2b_2^2 + c_1^2c_2^2 \\ +s_1^2a_2^2 + a_1^2s_2^2 + b_1^2c_2^2 + c_1^2b_2^2 \\ +s_1^2b_2^2 + b_1^2s_2^2 + c_1^2a_2^2 + a_1^2c_2^2 \\ +s_1^2c_2^2 + c_1^2s_2^2 + a_1^2b_2^2 + b_1^2a_2^2 \end{matrix}} = \sqrt{\begin{matrix} s_1^2\left(s_2^2 + a_2^2 + b_2^2 + c_2^2\right) \\ +a_1^2\left(s_2^2 + a_2^2 + b_2^2 + c_2^2\right) \\ +b_1^2\left(s_2^2 + a_2^2 + b_2^2 + c_2^2\right) \\ +c_1^2\left(s_2^2 + a_2^2 + b_2^2 + c_2^2\right) \end{matrix}}$$

$$= \sqrt{\left(s_1^2 + a_1^2 + b_1^2 + c_1^2\right)\left(s_2^2 + a_2^2 + b_2^2 + c_2^2\right)}$$

Исходя из формулы определения модуля кватерниона, получаем:

$$\|\mathbf{q}_1\mathbf{q}_2\| = \sqrt{\left(s_1^2 + a_1^2 + b_1^2 + c_1^2\right)\left(s_2^2 + a_2^2 + b_2^2 + c_2^2\right)} = \sqrt{\|\mathbf{q}_1\|^2\|\mathbf{q}_2\|^2} = \|\mathbf{q}_1\|\,\|\mathbf{q}_2\| \tag{43}$$

Таким образом, модуль произведения кватернионов равен произведению модулей. Отсюда следует, что длина кватерниона при умножении на кватернион единичной длины не изменяется. А также при умножении двух кватернионов единичной длины, снова получается кватернион единичной длины.

Сопряжённый кватернион

Сопряжённый кватернион, обозначается $\mathbf{q}^*$, получается инвертированием векторной части кватерниона:

$$\mathbf{q}^* = (s \quad \mathbf{v})^* = (s \quad -\mathbf{v}) = s - ia - jb - kc. \tag{44}$$

Обратный кватернион

Обратный кватернион, обозначаемый $\mathbf{q}^{-1}$, определяется как сопряжённый кватернион, разделённый на модуль кватерниона в квадрате. Для вычисления обратного кватерниона используется формула

$$\mathbf{q}^{-1} = \frac{\mathbf{q}^*}{\|\mathbf{q}\|^2}, \tag{45}$$

так что

$$\mathbf{q}\mathbf{q}^{-1} = \mathbf{q}^{-1}\mathbf{q} = (1, \mathbf{0}).$$

Так как кватернионы для представления поворота всегда имеют единичную длину, сопряжённый и обратный кватернионы будут одинаковы. Сопряжённый и обратный кватернионы интересны тем, что $\mathbf{q}$ и $\mathbf{q}^*$ представляют собой повороты в противоположные стороны. Легко понять, почему это так. Инвертируя знак $\mathbf{v}$, мы переворачиваем ось вращения $\mathbf{u}$, то есть изменяем направление вращения. Таким образом, $\mathbf{q}$ осуществляет поворот вокруг оси на угол θ, и $\mathbf{q}^*$ осуществляет поворот в противоположном направлении на тот же угол.
Обращение произведения кватернионов равно произведению обратных кватернионов, записанных в обратном направлении:

$$(\mathbf{a}\mathbf{b})^{-1} = \mathbf{b}^{-1}\mathbf{a}^{-1} \tag{46}$$

$$(\mathbf{q}_1\mathbf{q}_2 \cdots \mathbf{q}_{n-1}\mathbf{q}_n)^{-1} = \mathbf{q}_n^{-1}\mathbf{q}_{n-1}^{-1} \cdots \mathbf{q}_2^{-1}\mathbf{q}_1^{-1} \tag{47}$$

Трёхмерные повороты с помощью кватернионов

Любую точку $\mathbf{p}$, которую нужно повернуть с помощью кватерниона, можно представить в кватернионной форме записи:

$$\mathbf{P} = (0, \mathbf{p}),$$

где координаты точки образуют векторную часть кватерниона $\mathbf{p} = (x, y, z)$. Пусть $\mathbf{q}$ — кватернион, представляющий поворот в виде $(s, \mathbf{v}) = (\cos\theta/2, \mathbf{u}\sin\theta/2)$. Тогда мы можем повернуть точку $\mathbf{P}$ вокруг $\mathbf{u}$ на угол θ, выполнив следующее умножение:

$$\mathbf{P}' = \mathbf{q}\mathbf{P}\mathbf{q}^{-1}. \tag{48}$$

В результате данного преобразования получается новый кватернион:

$$\mathbf{P}' = \left(0, \mathbf{p}'\right). \tag{49}$$

Второй член данной упорядоченной пары — это повернутая точка $\mathbf{p}'$, которая вычисляется с помощью формулы (38):

$$\mathbf{p}' = s^2\mathbf{p} + \mathbf{v}(\mathbf{p}\cdot\mathbf{v}) + 2s\left(\mathbf{v}\times\mathbf{p}\right) + \mathbf{v}\times\left(\mathbf{v}\times\mathbf{p}\right). \tag{50}$$

Во множестве систем компьютерной графики используются эффективные аппаратные реализации данных векторных расчётов, с помощью которых быстро находятся трёхмерные повороты объектов.

Покажем, что преобразование (48) эквивалентно повороту вокруг оси, проходящей через начало координат. Т.е., что это то же, что и последовательность преобразований поворота в уравнении (23). Запишем каждое слагаемое из (48) в матричной форме:

$$s^2\mathbf{p} = \begin{bmatrix} s^2 & 0 & 0 \\ 0 & s^2 & 0 \\ 0 & 0 & s^2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix},$$

$$\mathbf{v}(\mathbf{p}\cdot\mathbf{v}) = (xa+yb+zc)(a\mathbf{i}+b\mathbf{j}+c\mathbf{k}) = \begin{bmatrix} a^2 & ab & ac \\ ab & b^2 & bc \\ ac & bc & c^2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix},$$

$$2s(\mathbf{v}\times\mathbf{p}) = 2s(bz-cy)\mathbf{i} + 2s(cx-az)\mathbf{j} + 2s(ay-bx)\mathbf{k} =$$

$$= \begin{bmatrix} 0 & -2sc & 2sb \\ 2sc & 0 & -2sa \\ -2sb & 2sa & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix},$$

$$\mathbf{v}\times(\mathbf{v}\times\mathbf{p}) = [b(ay-bx)-c(cx-az)]\mathbf{i} + [c(bz-cy)-a(ay-bx)]\mathbf{j} +$$

$$+[a(cx-az)-b(bz-cy)]\mathbf{k} = \begin{bmatrix} -b^2-c^2 & ab & ac \\ ab & -a^2-c^2 & bc \\ ac & bc & -a^2-b^2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Просуммировав эти матрицы и учитывая, что кватернион для представления поворота имеет единичную длину, получим:

$$\mathbf{p}' = \begin{bmatrix} 1-2b^2-2c^2 & 2ab-2sc & 2ac+2sb \\ 2ab+2sc & 1-2a^2-2c^2 & 2bc-2sa \\ 2ac-2sb & 2bc+2sa & 1-2a^2-2b^2 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix} = \mathbf{M}_R \cdot \mathbf{p}. \qquad (51)$$

Подставим явные значения параметров a, b, c и s для кватерниона, представляющего поворот, а затем используем следующие тригонометрические тождества для упрощения членов:

$$\cos^2\frac{\theta}{2} - \sin^2\frac{\theta}{2} = 1 - 2\sin^2\frac{\theta}{2} = \cos\theta, \qquad 2\cos\frac{\theta}{2}\sin\frac{\theta}{2} = \sin\theta.$$

Тогда матрицу $\mathbf{M}_R$ из (51) можно переписать как

$$\begin{bmatrix} u_x^2(1-\cos\theta)+\cos\theta & u_xu_y(1-\cos\theta)-u_z\sin\theta & u_xu_z(1-\cos\theta)+u_y\sin\theta \\ u_yu_x(1-\cos\theta)+u_z\sin\theta & u_y^2(1-\cos\theta)+\cos\theta & u_yu_z(1-\cos\theta)-u_x\sin\theta \\ u_zu_x(1-\cos\theta)-u_y\sin\theta & u_zu_y(1-\cos\theta)+u_x\sin\theta & u_z^2(1-\cos\theta)+\cos\theta \end{bmatrix} \qquad (52)$$

Если в (23) выполнить перемножение матриц $\mathbf{R}_x^{-1}(\alpha) \cdot \mathbf{R}_y^{-1}(\beta) \cdot \mathbf{R}_z(\theta) \cdot \mathbf{R}_y(\beta) \cdot \mathbf{R}_x(\alpha)$, то придем к аналогичному результату.

Трёхмерные повороты с помощью кватернионов

Посмотрим теперь, что будет происходить, когда мы применяем несколько поворотов. Повернём точку $\mathbf{p}$ кватернионом $\mathbf{q}_1$ и затем повернём этот результат другим кватернионом $\mathbf{q}_2$:

$$\begin{aligned}
\mathbf{p}' &= \mathbf{q}_2 \left(\mathbf{q}_1 \mathbf{p} \mathbf{q}_1^{-1} \right) \mathbf{q}_2^{-1} \\
&= (\mathbf{q}_2 \mathbf{q}_1)\, \mathbf{p} \left(\mathbf{q}_1^{-1} \mathbf{q}_2^{-1} \right) \\
&= (\mathbf{q}_2 \mathbf{q}_1)\, \mathbf{p}\, (\mathbf{q}_2 \mathbf{q}_1)^{-1}
\end{aligned}$$

Заметим, что поворот с помощью $\mathbf{q}_1$, а затем $\mathbf{q}_2$ эквивалентен выполнению одного поворота с помощью кватерниона $\mathbf{q}_2\mathbf{q}_1$. Таким образом, перемножение нескольких кватернионов может быть использовано для комбинации нескольких поворотов подобно умножению матриц.

Чтобы завершить последовательность преобразований для поворота вокруг произвольно расположенной оси вращения, нужно добавить перемещения, переносящие ось вращения в начало координат и возвращающие её в исходное положение. Таким образом, полное выражение для поворота с помощью кватернионов, соответствующее уравнению (23), выглядит так:

$$\mathbf{R}(\theta) = \mathbf{T}^{-1} \cdot \mathbf{M}_R \cdot \mathbf{T}. \tag{53}$$

Кватернион поворота между векторами

Найдем компоненты кватерниона поворота $\mathbf{d} = (s, \mathbf{v}) = \left(\cos\frac{\theta}{2}, \sin\frac{\theta}{2}\mathbf{u}\right)$, который выполняет поворот исходного положения, определяемого вектором $\mathbf{a}$ до конечного положения, определяемого вектором $\mathbf{b}$.
Очевидно, что вектор $\mathbf{u}$ является нормалью к плоскости, в которой лежат векторы $\mathbf{a}$ и $\mathbf{b}$, а угол θ определяется углом между векторами $\mathbf{a}$ и $\mathbf{b}$. Тогда:

$$\mathbf{a}\times\mathbf{b} = |\mathbf{a}||\mathbf{b}|\sin(\theta)\,\mathbf{u} = 2|\mathbf{a}||\mathbf{b}|\cos\left(\frac{\theta}{2}\right)\sin\left(\frac{\theta}{2}\right)\mathbf{u} = 2|\mathbf{a}||\mathbf{b}|\cos\left(\frac{\theta}{2}\right)\mathbf{v},$$

$$\mathbf{a}\cdot\mathbf{b} = |\mathbf{a}||\mathbf{b}|\cos(\theta) = |\mathbf{a}||\mathbf{b}|\left(2\cos^2\left(\frac{\theta}{2}\right) - 1\right),$$

$$\mathbf{a}\cdot\mathbf{b} + |\mathbf{a}||\mathbf{b}| = 2|\mathbf{a}||\mathbf{b}|\cos^2\left(\frac{\theta}{2}\right) = 2|\mathbf{a}||\mathbf{b}|\cos\left(\frac{\theta}{2}\right)\cos\left(\frac{\theta}{2}\right) = 2|\mathbf{a}||\mathbf{b}|\cos\left(\frac{\theta}{2}\right)s.$$

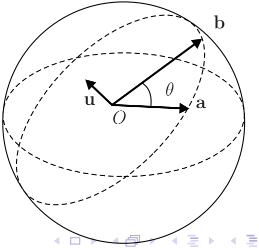
Откуда получаем:

$$\tilde{\mathbf{v}} = \mathbf{a}\times\mathbf{b},$$

$$\tilde{s} = \mathbf{a}\cdot\mathbf{b} + |\mathbf{a}||\mathbf{b}|,$$

$$\mathbf{d} = (s, \mathbf{v}) = \frac{\mathbf{q}}{\|\mathbf{q}\|},$$

где $\mathbf{q} = (\tilde{\mathbf{s}}, \tilde{\mathbf{v}})$.



Кватернион «углового смещения»

Пусть теперь $\mathbf{a}$ и $\mathbf{b}$ являются не векторами, а кватернионами, задающими ориентацию объекта в пространстве. Определим кватернион «углового смещения» $\mathbf{d}$, который выполняет преобразование объекта из ориентации, определяемой кватернионом $\mathbf{a}$ в ориентацию, определяемую кватернионом $\mathbf{b}$, следующим образом:

$$\mathbf{da} = \mathbf{b}.$$

Разрешим это уравнение относительно $\mathbf{d}$. Умножим обе части на $\mathbf{a}^{-1}$ справа:

$$(\mathbf{da})\,\mathbf{a}^{-1} = \mathbf{ba}^{-1}.$$

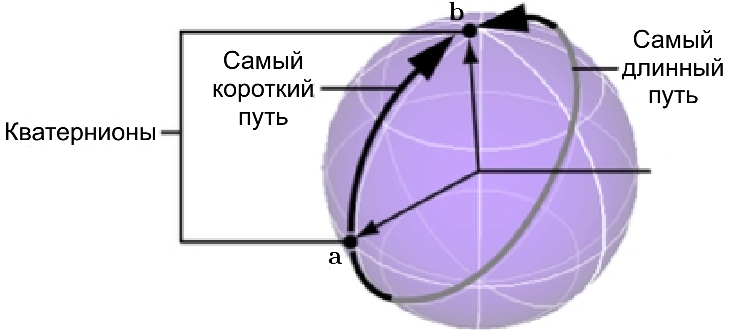
Применяя свойство ассоциативности умножения кватернионов и упрощая, получаем:

$$\begin{aligned} \mathbf{d}\left(\mathbf{aa}^{-1}\right) &= \mathbf{ba}^{-1}, \\ \mathbf{d}\,(1, \mathbf{0}) &= \mathbf{ba}^{-1}, \\ \mathbf{d} &= \mathbf{ba}^{-1}. \end{aligned}$$

Теперь у нас есть способ получения кватерниона для представления углового смещения от одного положения к другому.
Математически, угловая «разница» между двумя кватернионами больше похожа на деление чем на вычитание.

Кватернион «углового смещения»

Важно отметить, что угловое смещение всегда рассчитывается по кратчайшему расстоянию. Например, между поворотом на 270° против часовой стрелки вокруг оси x и поворотом на 90° по часовой стрелке вокруг той же оси, будет получено именно последнее. С точки зрения конечного результата это одно и то же, однако в некоторых ситуациях, например, когда мы определяем по этому смещению угловую скорость, результат может быть не тот, который ожидался.
Угловое смещение также не может состоять из нескольких поворотов.



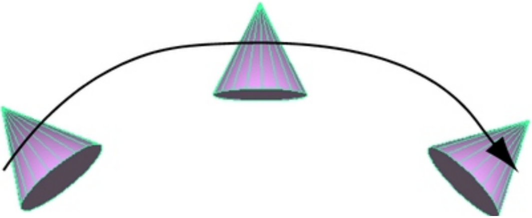
Скалярное произведение кватернионов

Будем интерпретировать кватернионы как элементы четырёхмерного евклидова пространства. Тогда для кватернионов можно определить операцию скалярного произведения:

$$\mathbf{q}_1 \cdot \mathbf{q}_2 = s_1 s_2 + \mathbf{v}_1 \cdot \mathbf{v}_2 = s_1 s_2 + a_1 a_2 + b_1 b_2 + c_1 c_2.$$

Скалярное произведение кватернионов имеет интерпретацию, аналогичную скалярному произведению векторов. Его результатом является скаляр. Для единичных кватернионов скалярное произведение равно косинусу угла между ними, который измеряется в четырёхмерном пространстве $\mathbf{a} \cdot \mathbf{b} = \cos\theta$. Чем больше абсолютная величина скалярного произведения единичных кватернионов ($|\mathbf{q}_1 \cdot \mathbf{q}_2| \approx 1$), тем более «близкими» являются вращения, представленные $\mathbf{q}_1$ и $\mathbf{q}_2$.
Соответственно, ортогонально нормированные кватернионы ($\mathbf{q}_1 \cdot \mathbf{q}_2 = 0$) дают чрезвычайно разные вращения.
Если рассмотреть кватернион «углового смещения» $\mathbf{d} = \mathbf{b}\mathbf{a}^*$, где $\mathbf{a}$ и $\mathbf{b}$ — единичные кватернионы, то его s-компонента равна скалярному произведению $\mathbf{a} \cdot \mathbf{b}$.

Интерполяция кватернионов



Поскольку кватернионы представлены векторами, они хорошо подходят для интерполяции. Когда объект анимируется, интерполяция полезна для генерации промежуточных ориентаций, которые лежат между предварительно рассчитанными ключевыми кадрами. Это является и одной из основных причин использования кватернионов в компьютерной графике.
Функции интерполяции как правило принимают три аргумента. Первые два аргумента являются кватернионами, которые задают начальную $\mathbf{q}_0$ и конечную $\mathbf{q}_1$ ориентации, между которыми нужно выполнить интерполяцию. Третий параметр t является параметром интерполяции, который изменяется от 0 до 1. Функция возвращает ориентацию, которая является интерполяцией значений $\mathbf{q}_0$ и $\mathbf{q}_1$ в точке t.

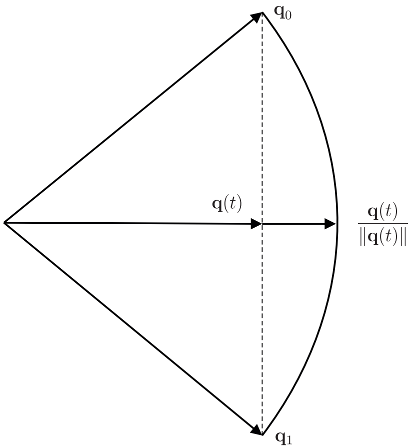
Линейная интерполяция кватернионов (lerp)

Простейшим типом интерполяции является *линейная интерполяция*. Для двух единичных кватернионов $\mathbf{q}_0$ и $\mathbf{q}_1$, линейно интерполированный кватернион $\mathbf{q}(t)$ задаётся формулой

$$\mathbf{q}(t) = (1 - t)\mathbf{q}_0 + t\mathbf{q}_1.$$

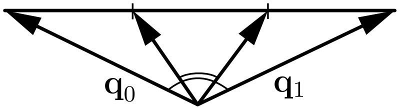
Функция $\mathbf{q}(t)$ плавно меняется вдоль отрезка, соединяющего $\mathbf{q}_0$ и $\mathbf{q}_1$ при изменении t от 0 до 1. Кватернион $\mathbf{q}(t)$ не является единичным, но его можно нормировать в каждой точке:

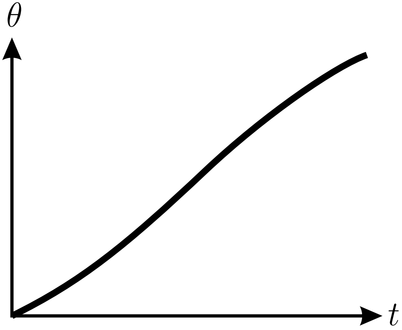
$$\mathbf{q}(t) = \frac{(1 - t)\mathbf{q}_0 + t\mathbf{q}_1}{\|(1 - t)\mathbf{q}_0 + t\mathbf{q}_1\|}.$$



Линейная интерполяция кватернионов (lerp)

Хотя линейная интерполяция эффективна, у неё есть недостаток: функция $\mathbf{q}(t)$ не обеспечивает движение с постоянной скоростью. График $\theta = \cos^{-1}(\mathbf{q}(t) \cdot \mathbf{q}_0)$ демонстрирует что скорость изменения угла между $\mathbf{q}(t)$ и $\mathbf{q}_0$ изменяется относительно медленно в конечных точках $t = 0$ и $t = 1$ и меняется быстрее всего при $t = 1/2$.

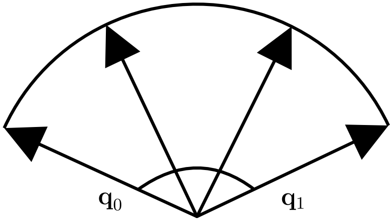


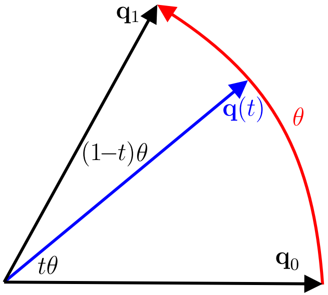


Сферическая интерполяция кватернионов (slerp)

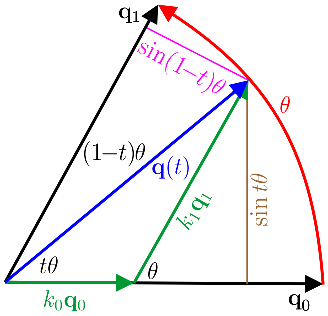
Построим теперь интерполяцию, которая обеспечивает движение с постоянной скоростью.

Для этого найдем функцию $\mathbf{q}(t)$, которая интерполирует два кватерниона $\mathbf{q}_0$ и $\mathbf{q}_1$ на поверхности единичной гиперсферы S^3 в $\mathbb{R}^4$ по дуге, которая их соединяет. (Отсюда и название *slerp*, которое расшифровывается как (**s**pherical **l**inear int**erp**olation — сферическая линейная интерполяция). Пусть θ — угол, охватываемой дугой от $\mathbf{q}_0$ до $\mathbf{q}_1$, тогда $\mathbf{q}(t)$ будет являться результатом вращения $\mathbf{q}_0$ по этой дуге на угол $t\theta$.





Сферическая интерполяция кватернионов (slerp)



Мы можем выразить $\mathbf{q}(t)$ в виде линейной комбинации: $\mathbf{q}(t) = k_0\mathbf{q}_0 + k_1\mathbf{q}_1$.
Рассмотрим прямоугольный треугольник с гипотенузой $k_1\mathbf{q}_1$ (и помня, что $\mathbf{q}(t)$ и $\mathbf{q}_1$ являются единичными векторами) получим что:

$$\sin\theta = \frac{\sin t\theta}{k_1} \qquad \Rightarrow \qquad k_1 = \frac{\sin t\theta}{\sin\theta}.$$

Аналогичным образом получим k_0:

$$k_0 = \frac{\sin(1-t)\theta}{\sin\theta}.$$

Тогда $\mathbf{q}(t)$ может быть выражен следующим образом:

$$\mathbf{q}(t) = k_0\mathbf{q}_0 + k_1\mathbf{q}_1 = \frac{\sin(1-t)\theta}{\sin\theta}\mathbf{q}_0 + \frac{\sin t\theta}{\sin\theta}\mathbf{q}_1.$$

Для вычисления угла θ можно использовать скалярное произведение:

$$\theta = \cos^{-1}(\mathbf{q}_0 \cdot \mathbf{q}_1),$$

и, следовательно, $\sin\theta$ можно вычислить по формуле:

$$\sin\theta = \sqrt{1 - (\mathbf{q}_0 \cdot \mathbf{q}_1)^2}.$$

Сферическая интерполяция кватернионов (slerp)

Есть две небольшие сложности:

1. Два кватерниона $\mathbf{q}$ и $-\mathbf{q}$ представляют одну и ту же ориентацию, но они могут давать различные результаты при использовании в качестве аргументов функции slerp.
   Чтобы избежать этой проблемы следует выбирать знаки $\mathbf{q}_0$ и $\mathbf{q}_1$ таким образом, чтобы скалярное произведение $\mathbf{q}_0 \cdot \mathbf{q}_1$ было неотрицательно. В результате всегда будет выбираться кратчайший поворот от $\mathbf{q}_0$ до $\mathbf{q}_1$.
2. Если $\mathbf{q}_0$ и $\mathbf{q}_1$ очень близки, то угол θ будет очень малым и $\sin\theta$ тоже будет очень малым, что вызовет проблемы с делением.
   Чтобы избежать этого, в случае, если $\sin\theta$ очень мал, следует использовать линейную интерполяцию lerp.

Получение кватерниона по углам Эйлера

В случае, если нужно перейти от ориентации, заданной с помощью углов Эйлера, к ориентации с использованием кватернионов потребуется выполнить следующее умножение:

$$\mathbf{q} = \mathbf{q}_z \mathbf{q}_y \mathbf{q}_x,$$

где

$$\mathbf{q}_x = \left(\cos\frac{p}{2}, \sin\frac{p}{2}\mathbf{i}\right), \qquad \mathbf{q}_y = \left(\cos\frac{h}{2}, \sin\frac{h}{2}\mathbf{j}\right), \qquad \mathbf{q}_z = \left(\cos\frac{r}{2}, \sin\frac{r}{2}\mathbf{k}\right).$$

Выполнив умножение, получим:

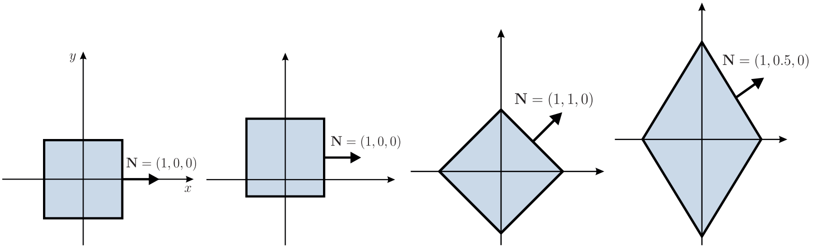
$$\mathbf{q} = (s, x_q\mathbf{i} + y_q\mathbf{j} + z_q\mathbf{k}),$$

где

$$\begin{aligned}
s &= \cos\frac{r}{2}\cos\frac{h}{2}\cos\frac{p}{2} + \sin\frac{r}{2}\sin\frac{h}{2}\sin\frac{p}{2},\\
x_q &= \cos\frac{r}{2}\cos\frac{h}{2}\sin\frac{p}{2} - \sin\frac{r}{2}\sin\frac{h}{2}\cos\frac{p}{2},\\
y_q &= \cos\frac{r}{2}\sin\frac{h}{2}\cos\frac{p}{2} + \sin\frac{r}{2}\cos\frac{h}{2}\sin\frac{p}{2},\\
z_q &= \sin\frac{r}{2}\cos\frac{h}{2}\cos\frac{p}{2} - \cos\frac{r}{2}\sin\frac{h}{2}\sin\frac{p}{2}.
\end{aligned}$$

Преобразование векторов

Вершины полигональной модели, кроме данных о своих координатах, обычно содержат дополнительную информацию. Например, координаты касательного и нормального векторов. Когда мы выполняем преобразование модели, то нужно пересчитывать не только положения вершин, но и координаты этих векторов.



а) исходное положение нормали
б) перемещение вдоль оси y
в) поворот на 45°
г) масштабирование вдоль оси y

Рис.: Изменение направления нормали при различных преобразованиях координат

Преобразование векторов

Поскольку касательные и нормальные векторы не изменяются при преобразованиях перемещения, то для их пересчёта достаточно будет из исходной матрицы $\tilde{\mathbf{M}}$ взять подматрицу $\mathbf{M}$ размером 3×3, вычеркнув последнюю строку и столбец.

Рассмотрим исходное соотношение для преобразования координат вершин геометрической модели, заданной в параметрическом виде:

$$\mathbf{p}' = \tilde{\mathbf{M}} \cdot \mathbf{p},$$

и продиффиренцируем его по параметру. Получим аналогичное соотношение для касательных векторов:

$$\mathbf{t}' = \mathbf{M} \cdot \mathbf{t}.$$

Преобразование векторов нормалей

Так как касательные и нормали перпендикулярны, то касательный вектор $\mathbf{t}$ и нормальный вектор $\mathbf{n}$, связанные с вершиной, должны удовлетворять уравнению: $\mathbf{n} \cdot \mathbf{t} = 0$. Это соотношение должно быть справедливо и для преобразованных касательного $\mathbf{t}'$ и нормального $\mathbf{n}'$ векторов. Обозначив матрицу преобразования нормального вектора как $\mathbf{G}$, запишем:

$$\mathbf{n}' \cdot \mathbf{t}' = (\mathbf{G}\mathbf{n}) \cdot (\mathbf{M}\mathbf{t}) = 0.$$

Записав скалярное произведение в компонентном виде, получим:

$$(\mathbf{G}\mathbf{n}) \cdot (\mathbf{M}\mathbf{t}) = (\mathbf{G}\mathbf{n})^T (\mathbf{M}\mathbf{t}) = \mathbf{n}^T \mathbf{G}^T \mathbf{M}\mathbf{t}.$$

Так как $\mathbf{n} \cdot \mathbf{t} = 0$, то уравнение $\mathbf{n}^T \mathbf{G}^T \mathbf{M}\mathbf{t} = 0$ выполняется, если $\mathbf{G}^T \mathbf{M} = \mathbf{E}$, где $\mathbf{E}$ — единичная матрица. Отсюда следует, что $\mathbf{G} = \left(\mathbf{M}^{-1}\right)^T$. То есть, нормальный вектор преобразуется с помощью *транспонированной обратной матрицы*. Векторы, которые должны быть преобразованы таким образом, называются *ковариантными* векторами, а векторы, которые преобразуются обычным образом с использованием матрицы $\mathbf{M}$ (такие как радиус-векторы точек и касательные векторы), называются *контравариантными* векторами.

Если матрица $\mathbf{M}$ — ортогональная, например, является матрицей поворота, то $\mathbf{M}^{-1} = \mathbf{M}^T$ и, следовательно, $\left(\mathbf{M}^{-1}\right)^T = \mathbf{M}$.

Функции для работы с матрицами преобразования векторов в glMatrix

В библиотеке glMatrix для получения левой верхней подматрицы 3×3 из исходной матрицы 4×4 вычеркиванием последней строки и столбца, можно использовать метод `mat3.fromMat4(out, m)`.
Для получения обратной транспонированной матрицы, которая используется для преобразования нормали, можно использовать метод `mat3.normalFromMat4(out, m)`.

Функции для работы с кватернионами в glMatrix

Для работы с кватернионами в библиотеке glMatrix существует объект `quat`. Для создания единичного кватерниона требуется вызвать функцию:

```
quat.create();
```

Чтобы создать кватернион и сразу инициализировать его компоненты, используется следующая команда:

```
const q = quat.fromValues(a, b, c, s);
```

где s — скалярная часть , а компоненты a, b и c формируют векторную часть.
Функция `quat.str()` преобразует кватернион в строку.
В объекте `mat4` существует метод `fromQuat(out, q)` который формирует матрицу вращения по заданному кватерниону.

Функции для работы с кватернионами в glMatrix

`add(out, q1, q2)`	Складывает два кватерниона
`clone(q)`	Возвращает копию кватерниона
`conjugate(out, q)`	Вычисляет сопряжённый кватернион
`dot(q1, q2)`	Рассчитывает скалярное произведение
`fromEuler(out,pitch,yaw,roll)`	Задаёт кватернион по углам Эйлера
`getAxisAngle(out_axis, q)`	Возвращает ось поворота и угол
`identity(out)`	Возвращает единичный кватернион
`invert(out, q)`	Рассчитывает обратный кватернион
`length()`	Возвращает модуль кватерниона
`lerp(out, q1, q2, t)`	Выполняет линейную интерполяцию
`multiply(out, q1, q2)`	Перемножает два кватерниона
`normalize(out, q)`	Нормирует кватернион
`rotateX(out, q, rad)`	Поворачивает кватернион вокруг x
`rotateY(out, q, rad)`	Поворачивает кватернион вокруг y
`rotateZ(out, q, rad)`	Поворачивает кватернион вокруг z
`rotationTo(out, a, b)`	Находит кватернион поворота от $\mathbf{a}$ к $\mathbf{b}$
`scale(out, q, k)`	Умножает кватернион на число
`set(out, a, b, c, s)`	Задаёт кватернион по компонентам
`setAxisAngle(out, axis, rad)`	Задаёт кватернион через ось и угол
`slerp(out, q1, q2, t)`	Сферическая линейная интерполяция

Методы класса THREE.Quaternion в библиотеке three.js

`Quaternion(a, b, c, s)`	Конструктор
`clone()`	Возвращает копию кватерниона
`conjugate()`	Возвращает сопряжённый кватернион
`dot(q)`	Рассчитывает скалярное произведение с кватернионом $\mathbf{q}$
`setFromEuler(euler)`	Задаёт кватернион по углам Эйлера
`identity()`	Делает кватернион единичным
`invert()`	Обращает кватернион
`length()`	Возвращает модуль кватерниона
`multiply(q)`	Умножает кватернион на кватернион $\mathbf{q}$
`normalize()`	Нормирует кватернион
`setFromUnitVectors( a, b)`	Находит кватернион поворота от $\mathbf{a}$ к $\mathbf{b}$
`set(a, b, c, s)`	Задаёт кватернион по компонентам
`setFromAxisAngle(axis, rad)`	Задаёт кватернион через ось и угол
`slerpQuaternions(q1, q2, t)`	Задаёт кватернион сферической линейной интерполяции