

General Rotations in 3D

Quaternions

Róbert Bohdal

`robert.bohdal@fmph.uniba.sk`

<https://flurry.dg.fmph.uniba.sk/webog/bohdal-vyucba>

Department of Algebra and Geometry
Faculty of Mathematics, Physics and Informatics
Comenius University Bratislava

Computer Graphics (1)
Lecture 3



- 1 Cartesian and Spherical Coordinates in 3D
- 2 Rotation in 3D about an Arbitrary Line
- 3 Quaternions



Spherical Coordinates in 3D

- Input: a point $P \in \mathbb{E}^3(\mathbb{R})$
- Output: the Cartesian and spherical coordinates of P
- A point can have different representations:
 - spherical coordinates:

$$r = |OP| \quad \rightsquigarrow r = \sqrt{x^2 + y^2 + z^2}$$

$$\theta = \angle x, \overline{OP^*} \quad \rightsquigarrow \cos \theta = \frac{x}{\sqrt{x^2 + y^2}}$$

$$\rightsquigarrow \sin \theta = \frac{y}{\sqrt{x^2 + y^2}}$$

$$\rightsquigarrow \theta = \text{atan2}(y, x)$$

$$\psi = \angle z, \overline{OP} \quad \rightsquigarrow \psi = \arccos \frac{z}{r}$$

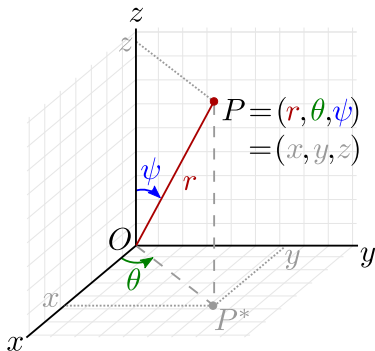
- Cartesian coordinates:

$$x = r \cdot \cos \theta \cdot \sin \psi$$

$$y = r \cdot \sin \theta \cdot \sin \psi$$

$$z = r \cdot \cos \psi$$

- The conversion follows from: $\frac{x}{|OP^*|} = \cos \theta$,
 $\frac{y}{|OP^*|} = \sin \theta$, and $\frac{|OP^*|}{r} = \sin \psi$

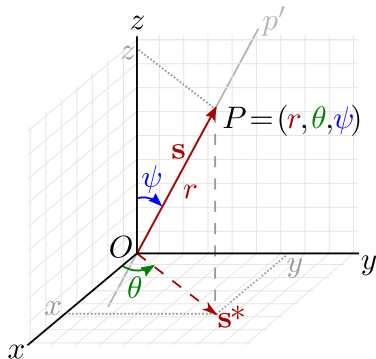


Rotation in 3D about an Arbitrary Line

- Input: a line $p = \langle P, s \rangle$, an angle φ , and an object X (e.g. a point)
- Output: the image of X after rotation about p
- Rotation $\longleftrightarrow$ composition of transformations:
 - I. align p with a coordinate axis (e.g. the $\vec{z}$ -axis)
 - II. apply the rotation $\mathbf{R}_z(\varphi)$
 - III. restore p to its original position

Procedure:

- I. aligning $p \mapsto \vec{z}$:
 - a. translation $P \mapsto P' \rightsquigarrow p \mapsto p' \quad (P' \equiv O)$
 - b. rotation $\mathbf{R}_z(-\theta)$ into the xz -plane
 - c. rotation $\mathbf{R}_y(-\psi)$ within the xz -plane
 - II. rotation $\mathbf{R}_z(\varphi)$
 - III. restoring p to its original position: $\vec{z} \mapsto p$:
 - a. rotation $\mathbf{R}_y(\psi)$ = inverse transformation of I.(c)
 - b. rotation $\mathbf{R}_z(\theta)$ = inverse transformation of I.(b)
 - c. translation $P' \mapsto P \rightsquigarrow p \rightsquigarrow p =$ inverse transformation of I.(a)
- Result: six 4×4 matrix multiplications and one multiplication by a 4×1 vector
- $$\mathbf{X}' = \underbrace{\mathbf{T}_{P-O} \cdot \mathbf{R}_z(\theta) \cdot \mathbf{R}_y(\psi)}_{\text{III} = \mathbf{I}^{-1}} \cdot \underbrace{\mathbf{R}_z(\varphi)}_{\text{II.}} \cdot \underbrace{\mathbf{R}_y(-\psi) \cdot \mathbf{R}_z(-\theta)}_{\text{I.}} \cdot \mathbf{T}_{O-P} \cdot \mathbf{X}$$



Quaternions – Derivation and Definition

- $\mathbb{C} \sim \mathbb{R}^2$: Cartesian product = complex numbers

$$z = (a, b) \in \mathbb{R}^2 \rightsquigarrow a \cdot 1 + b \cdot i \quad \text{where } i^2 = -1$$

$$z = |z| \cdot (\cos \varphi + i \sin \varphi) = |z| \cdot e^{i\varphi}$$

- $\mathbb{H} \sim \mathbb{C}^2$: Cartesian product = hypercomplex numbers (Hamilton)

$$\mathbf{q} = (a + b \cdot i, c + d \cdot i) \in \mathbb{C}^2 \rightsquigarrow (a + b \cdot i) \cdot 1 + (c + d \cdot i) \cdot j =$$

$$a + b \cdot i + c \cdot j + d \cdot i \cdot j =:$$

$$a + b \cdot i + c \cdot j + d \cdot k$$

$$\text{where } i^2 = j^2 = k^2 = i \cdot j \cdot k = -1$$

$$\mathbf{q} = |\mathbf{q}| \cdot (\cos \varphi + \vec{u} \sin \varphi) = |\mathbf{q}| \cdot e^{\vec{u}\varphi}$$

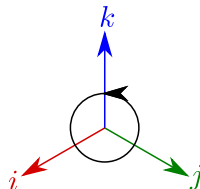
where $\vec{u} = (u_1, u_2, u_3) = u_1 i + u_2 j + u_3 k$ is a purely imaginary **unit** quaternion

- multiplication of the imaginary units is noncommutative:

$$i \cdot j = k = -j \cdot i,$$

$$j \cdot k = i = -k \cdot j,$$

$$k \cdot i = j = -i \cdot k$$



Quaternions – Notation and Operations

- notation:

$$a + b \cdot i + c \cdot j + d \cdot k \rightsquigarrow (a, b, c, d) =: (q_0, q_1, q_2, q_3) =: \mathbf{q}$$

- addition:

$$\mathbf{q} + \mathbf{q}' = (q_0 + q'_0, q_1 + q'_1, q_2 + q'_2, q_3 + q'_3)$$

- multiplication by a scalar $r \in \mathbb{R}$:

$$r \cdot \mathbf{q} = (r \cdot q_0, r \cdot q_1, r \cdot q_2, r \cdot q_3)$$

- quaternion multiplication:

$$\begin{aligned} \mathbf{q} \cdot \mathbf{q}' &= (q_0 + q_1 i + q_2 j + q_3 k) \cdot (q'_0 + q'_1 i + q'_2 j + q'_3 k) = \\ &= (q_0 \cdot q'_0 - q_1 \cdot q'_1 - q_2 \cdot q'_2 - q_3 \cdot q'_3) \cdot 1 + \\ &\quad + (q_0 \cdot q'_1 + q_1 \cdot q'_0 + q_2 \cdot q'_3 - q_3 \cdot q'_2) \cdot i + \\ &\quad + (q_0 \cdot q'_2 + q_2 \cdot q'_0 + q_3 \cdot q'_1 - q_1 \cdot q'_3) \cdot j + \\ &\quad + (q_0 \cdot q'_3 + q_3 \cdot q'_0 + q_1 \cdot q'_2 - q_2 \cdot q'_1) \cdot k \end{aligned}$$



Quaternions – Properties

- conjugate quaternion:

$$\begin{aligned}\mathbf{q} &= q_0 + q_1i + q_2j + q_3k &\rightsquigarrow& \mathbf{q} = q_0 + \vec{q} \\ \mathbf{q}^* &:= q_0 - q_1i - q_2j - q_3k &\rightsquigarrow& \mathbf{q}^* = q_0 - \vec{q}\end{aligned}$$

where $\vec{q} = (q_1, q_2, q_3)$

- properties of conjugate quaternions:

$$\begin{aligned}(\mathbf{q}^*)^* &= \mathbf{q}, \\ \mathbf{q} \cdot \mathbf{q}^* &= \mathbf{q}^* \cdot \mathbf{q}, \\ (\mathbf{p} \cdot \mathbf{q})^* &= \mathbf{q}^* \cdot \mathbf{p}^*,\end{aligned}$$

if $q_0 = 0$, then $\mathbf{q}$ is purely imaginary

- norm (length):

$$\|\mathbf{q}\| = \sqrt{q_0^2 + q_1^2 + q_2^2 + q_3^2} = \sqrt{\mathbf{q} \cdot \mathbf{q}^*} = \sqrt{\mathbf{q}^* \cdot \mathbf{q}}$$

- normalization $\Leftrightarrow$ unit quaternion ($\mathbf{q} \neq \mathbf{0}$):

$$\mathbf{q}_u := \frac{\mathbf{q}}{\|\mathbf{q}\|} = \frac{\mathbf{q}}{\sqrt{\mathbf{q} \cdot \mathbf{q}^*}}$$

- inverse (reciprocal) quaternion ($\mathbf{q} \neq \mathbf{0}$):

$$\mathbf{q}^{-1} = \frac{\mathbf{q}^*}{\|\mathbf{q}\|^2} \Leftrightarrow \frac{1}{\mathbf{q}} = \frac{\mathbf{q}^*}{\mathbf{q} \cdot \mathbf{q}^*} = \frac{\mathbf{q}^*}{(\sqrt{\mathbf{q} \cdot \mathbf{q}^*})^2} = \frac{\mathbf{q}^*}{\|\mathbf{q}\|^2}$$



Quaternions – Definition of the Rotation Operator $L_q()$

- quaternion multiplication (alternative form):

$$\mathbf{pq} = (p_0, \vec{p}) \cdot (q_0, \vec{q}) = p_0 q_0 - \vec{p} \cdot \vec{q} + p_0 \vec{q} + q_0 \vec{p} + \vec{p} \times \vec{q}$$

where $\vec{p} = (p_1, p_2, p_3)$ and $\vec{q} = (q_1, q_2, q_3)$

- the rotation operator $L_q()$ associated with a quaternion $\mathbf{q}$, applied to a vector $\vec{v} \in \mathbb{R}^3$:

$$\begin{aligned} L_q(\vec{v}) &:= \mathbf{q}\vec{v}\mathbf{q}^* = \dots \\ &= (q_0^2 - \|\vec{q}\|^2)\vec{v} + 2(\vec{q} \cdot \vec{v})\vec{q} + 2q_0(\vec{q} \times \vec{v}) \end{aligned}$$

where $\mathbf{q}$ is the unit quaternion

$$\mathbf{q} = q_0 + \vec{q} = \cos \frac{\theta}{2} + \sin \frac{\theta}{2} \vec{u},$$

and $\vec{u}$ has **unit length**, i.e. $\|\vec{u}\| = 1$

Note: we rotate the vector $\vec{v}$ through an angle θ about the axis with direction vector $\vec{u}$.



Quaternions – Procedure for Rotation in 3D

- Input: an object X (e.g. a point), a line $p = \langle P, \vec{u} \rangle$, and a rotation angle θ
- Output: the image X' of X after rotation through θ about the line p
- Preparation:
 1. assume that $p \ni O(0,0,0)$ (otherwise translate both X and P by $\overrightarrow{PO}$ and, after the rotation, translate the result back by $\overrightarrow{OP}$)
 2. represent the coordinates of the object X as a vector $\vec{v}$:

$$X = (x, y, z) \rightsquigarrow \vec{v} := X$$

3. encode the axis of rotation (the line p) as a purely imaginary **unit** quaternion $\mathbf{q}_u$:

- normalize the direction vector $\vec{u}$ of p : $\vec{u} := \frac{\vec{u}}{\|\vec{u}\|}$,
- construct a purely imaginary unit quaternion $\mathbf{q}_u$ from $\vec{u}$:

$$\vec{u} = (u_x, u_y, u_z) \rightsquigarrow \mathbf{q}_u := u_x i + u_y j + u_z k$$

4. construct the quaternion $\mathbf{q}_{u\theta}$ and its conjugate $\mathbf{q}_{u\theta}^*$ so that the rotation operator $L_q()$ defined above can be applied:

$$\mathbf{q}_{u\theta} := \cos \frac{\theta}{2} + \sin \frac{\theta}{2} \mathbf{q}_u = q_0 + \vec{q}, \quad \mathbf{q}_{u\theta}^* := \cos \frac{\theta}{2} - \sin \frac{\theta}{2} \mathbf{q}_u = q_0 - \vec{q}$$

- Procedure (rotation of $\vec{v}$ about the line p):

1. apply the rotation operator to $\vec{v}$ using the quaternions constructed in step 4:

$$X' := \mathbf{q}_{u\theta} \vec{v} \mathbf{q}_{u\theta}^* = (q_0^2 - \|\vec{q}\|^2) \vec{v} + 2(\vec{q} \cdot \vec{v}) \vec{q} + 2q_0(\vec{q} \times \vec{v})$$

2. the image X' of the object X then has coordinates

$$X' = (x', y', z')$$



Quaternions – Matrix Representation of Rotation in 3D

- Every unit quaternion $\mathbf{q}$ corresponds to a rotation matrix $\mathbf{M}_{\mathbf{q}}$
 $\rightsquigarrow$ the same rotation can be performed by matrix multiplication
 $\rightsquigarrow$ well suited for a graphics transformation pipeline
- Input: a point $X = (x, y, z)$ and the quaternion constructed for rotation about the line p (previous slide):

$$\mathbf{q}_{u\theta} := \cos \frac{\theta}{2} + \sin \frac{\theta}{2} \mathbf{q}_u =: (q_0, q_1, q_2, q_3)$$

- Output: the image X' of X after rotation through θ about the line p
- Preparation: construct the general rotation matrix

$$\mathbf{M}_{\mathbf{q}} := \begin{pmatrix} 1 - 2(q_2^2 + q_3^2) & 2(q_1q_2 - q_0q_3) & 2(q_1q_3 + q_0q_2) \\ 2(q_1q_2 + q_0q_3) & 1 - 2(q_1^2 + q_3^2) & 2(q_2q_3 - q_0q_1) \\ 2(q_1q_3 - q_0q_2) & 2(q_2q_3 + q_0q_1) & 1 - 2(q_1^2 + q_2^2) \end{pmatrix}$$

- Procedure:

$$\mathbf{X}' := \mathbf{M}_{\mathbf{q}} \cdot \mathbf{X}$$

$$(x', y', z')^T = \mathbf{M}_{\mathbf{q}} \cdot (x, y, z)^T \rightsquigarrow X' = (x', y', z')$$

Note: If $p \not\equiv O$, translate both the point X and the axis p by $\overrightarrow{PO}$ before applying the matrix. After the rotation, translate the result back by $\overrightarrow{OP}$.



Quaternions – Rodrigues' Rotation Formula

- A vector $\vec{v}$ can be rotated through an angle θ about a unit vector $\vec{u}$ using

$$\vec{v}' = \vec{v} \cos \theta + (\vec{u} \times \vec{v}) \sin \theta + \vec{u}(\vec{u} \cdot \vec{v})(1 - \cos \theta),$$

Equivalently, the matrix form of Rodrigues' rotation formula is

$$(\vec{v}')^T = \mathbf{R} \vec{v}^T,$$

where $\mathbf{R}$ is the rotation matrix

$$\mathbf{R} = \mathbf{I} + \mathbf{U} \sin \theta + \mathbf{U}^2 (1 - \cos \theta)$$

and $\mathbf{U}$ is the skew-symmetric matrix

$$\mathbf{U} = \begin{pmatrix} 0 & -u_z & u_y \\ u_z & 0 & -u_x \\ -u_y & u_x & 0 \end{pmatrix}, \text{ where } \mathbf{U} \mathbf{v}^T = \mathbf{U} \begin{bmatrix} v_x \\ v_y \\ v_z \end{bmatrix} = \begin{bmatrix} u_y v_z - u_z v_y \\ u_z v_x - u_x v_z \\ u_x v_y - u_y v_x \end{bmatrix} = (\mathbf{u} \times \mathbf{v})^T.$$

- This yields the rotation matrix

$$\mathbf{R} = \begin{pmatrix} \cos \theta + u_x^2 (1 - \cos \theta) & u_x u_y (1 - \cos \theta) - u_z \sin \theta & u_x u_z (1 - \cos \theta) + u_y \sin \theta \\ u_y u_x (1 - \cos \theta) + u_z \sin \theta & \cos \theta + u_y^2 (1 - \cos \theta) & u_y u_z (1 - \cos \theta) - u_x \sin \theta \\ u_z u_x (1 - \cos \theta) - u_y \sin \theta & u_z u_y (1 - \cos \theta) + u_x \sin \theta & \cos \theta + u_z^2 (1 - \cos \theta) \end{pmatrix}$$

