Lab 4—3D Rotations

Objective: To see that composition of two rotations about different coordinate axes are equivalent to a single rotation about a fixed direction.

MATLAB Commands:

`M=[1 2 3;4 5 6]`	Creates a certain 2x3 matrix and calls it `M`. Use either spaces or commas to separate elements of a row and use semi-colons to separate rows.
`a=[1 2 3]`	Creates a vector and calls it `a`.
`norm(a)`	Computes the length of vector `a`.
`dot(a,b)`	Computes dot product of vectors `a` and `b`.
`M*N`	If `M` and `N` are matrices, the asterix automatically uses matrix multiplication to take the product. Result is stored in variable `ans`, which will be overwritten.
`A=M*N`	As above, but result is stored in variable `A` for later use. Recommended.
`rref(M)`	Computes reduced row echelon form of matrix `M` in a single step (no user input required, unlike `rr(M)`). To store result in variable `A`, type `A=rref(M)` instead.
`acos(x)`	Takes arccos (inverse cosine) of `x`. The answer will be in radians.

Let $R_1(\theta_1)$ be the linear mapping that rotates any vector in $\mathbb{R}^3$ through an angle θ_1 counter-clockwise about the x_1-axis. It will have matrix

$$[R_1(\theta_1)] = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\theta_1 & -\sin\theta_1 \\ 0 & \sin\theta_1 & \cos\theta_1 \end{bmatrix}$$

An important property of this mapping is that it leaves any vector parallel to the x_1-axis unchanged, as you would expect. You can easily check this by confirming that

$$[R_1(\theta_1)]\begin{bmatrix} x_1 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} x_1 \\ 0 \\ 0 \end{bmatrix}$$

By way of contrast, any vector **v** lying in the x_2x_3-plane (the plane whose equation can be written $x_1 = 0$) is rotated through the angle θ_1, while remaining in the

x_2x_3-plane. To see this, define the vector **v** whose components are given by the column matrix:

$$\begin{bmatrix} v_1 \\ v_2 \\ v_3 \end{bmatrix} = [R_1(\theta_1)] \begin{bmatrix} 0 \\ x_2 \\ x_3 \end{bmatrix}$$

If you compute v_1, it will be zero, so **v** lies in the x_2x_3-plane, as claimed. Moreover, if you take the dot product of **v** with the original vector $(0,x_2,x_3)$ and divide by the length of each vector, you will get the cosine of the angle between these two vectors.

Notice that the length of any vector is unchanged by $R_1(\theta_1)$. Indeed, an important property of all rotations is that they leave lengths of vectors unchanged. We can consider this to be the *defining* property of rotations. Thus, a linear mapping R:R^n->R^n is called a rotation if and only if the length of the vector $\mathbf{v}=R(\mathbf{x})$ equals the length of **x** for every **x** in the domain.

---

1. Write the matrices corresponding to the linear mappings $R_2(\theta_2)$, and $R_3(\theta_3)$ that rotate vectors counter-clockwise through angle θ_2 about the x_2-axis and through angle θ_3 about the x_3-axis respectively. What vectors are left unchanged by these mappings?

2. Find the matrix A which corresponds to the linear mapping f_A obtained by first rotating a vector in R^3 through angle $\pi/3$ counter-clockwise about the z-axis and then through angle $3\pi/4$ counter-clockwise about the x-axis. [Hint: Matrix multiplication is much easier using MATLAB than it is by hand. To type π in MATLAB, type `pi` in lower-case letters.]

3. We next determine what axis this rotation is about. To do this, notice that any vector **w** parallel to this axis will be left unchanged by the rotation, so **w** will be a solution of the system of equations $\mathrm{A}[\mathbf{w}]=\mathbf{w}$, which we rewrite as $(\mathrm{A}-\mathrm{I})[\mathbf{w}]=\mathbf{0}$ where I is the identity matrix. Solve this system by performing row reduction of the matrix (A-I). [Hint: Although you can certainly use the `rr(M)` procedure to do this, the MATLAB command `rref(M)` will immediately return the reduced row echelon form of any matrix `M`, saving you plenty of time.] Write the parametric equations of the line through the origin parallel to the **w** direction. This line is the axis of the rotation.

4. Now we compute the angle of the rotation. It will suffice to compute the angle for one specific example. Apply the rotation f_A to the vector $\mathbf{u}=(2,1,2)$ and find the resulting rotated vector $\mathbf{v} = f_A(\mathbf{u})$. Find the vectors $\mathrm{perp}_\mathbf{w}\mathbf{u}$ and $\mathrm{perp}_\mathbf{w}\mathbf{v}$, which are the projections of **u** and **v** into the plane perpendicular to **w**. What is the angle between these two projected vectors?

5. Find the matrix B of the linear mapping obtained by applying our original rotations in reverse order; that is, first applying the rotation by $\pi/4$ counter-clockwise about the x-axis and then rotating the result by angle $\pi/6$ counter-clockwise about the z-axis. This is also equivalent to a single rotation (you do not need to check this). By applying the method of Step 3 above, find the axis of this rotation. Is it different from the axis found before?

Optional: Your TA may assign you one or more of the following:

6. Find the vector whose components are the elements of the column vector A[**i**], where **i** = (1,0,0). Repeat for **j** = (0,1,0) and **k** = (0,0,1). Check that the columns A[**i**], A[**j**], and A[**k**] are mutually orthogonal and represent unit vectors (by computing the dot product of each pair of these vectors using MATLAB). This tells you how the coordinate axes are rotated.

7. Compute the dot product of the vector given in column form by

$$\left[f_{\mathrm{A}}(\mathbf{x})\right]=\left[f_{\mathrm{A}}\left(x_1\mathbf{i}+x_2\mathbf{j}+x_3\mathbf{k}\right)\right]=x_1\mathrm{A}[\mathbf{i}]+x_2\mathrm{A}[\mathbf{j}]+x_3\mathrm{A}[\mathbf{k}]$$

with itself. Since we know that A(**i**), A(**j**), and A(**k**) represent mutually orthogonal unit vectors, this dot product should be equal $x_1^2+x_2^2+x_3^2$, which is just $|\mathbf{x}|^2$, so the length of $[f_{\mathrm{A}}(\mathbf{x})]$ equals the length of **x**. This is an important property of rotations.