CZ2003: Lab 5

Name: YONG WEN SHIUAN	**Last two digits of the matric card:** 37

Q1a

Derive a transformation matrix performing rotation by $\frac{\pi}{2}$ about an axis parallel to axis Y and passing through the point with coordinates (12, 0, 0).

First, translate rotation axis such that it's the same as the y-axis:

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 & -12 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Next, perform rotation by $\frac{\pi}{2}$ about y-axis:

$$\mathbf{R} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Lastly, translate rotation axis back by the same amount as in T, which produces the inverse matrix of T:

$$\mathbf{T}^{-1} = \begin{bmatrix} 1 & 0 & 0 & 12 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Multiplying the matrices above in order, i.e. $T^{-1} \times R \times T$, we obtain the final affine transformation matrix (displayed on the next page):

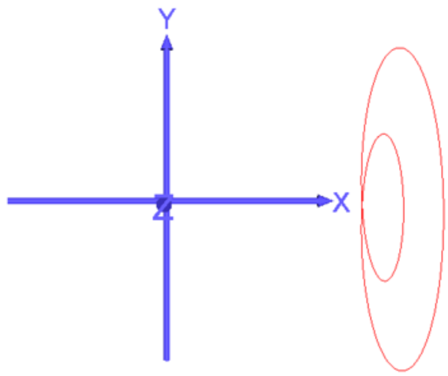
$$\begin{bmatrix} 1 & 0 & 0 & 12 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -12 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

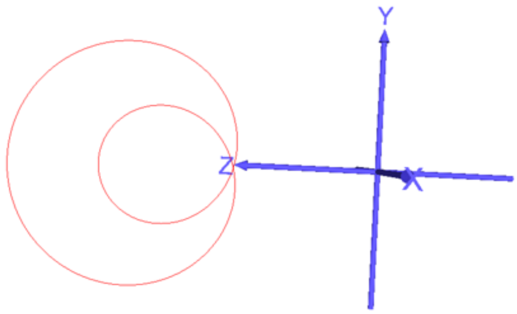
$$= \begin{bmatrix} 0 & 0 & 1 & 12 \\ 0 & 1 & 0 & 0 \\ -1 & 0 & 0 & 12 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

-

-

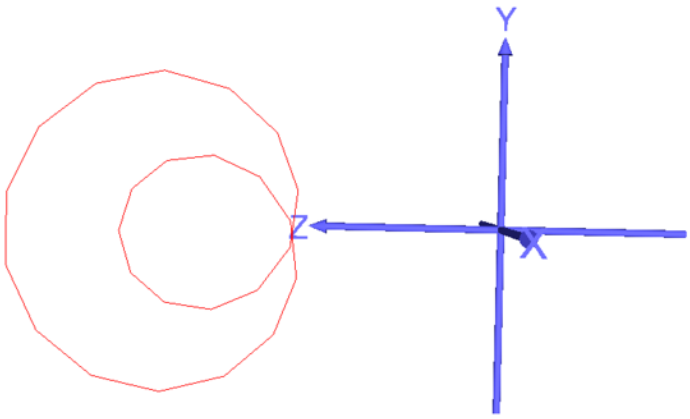
Q1b



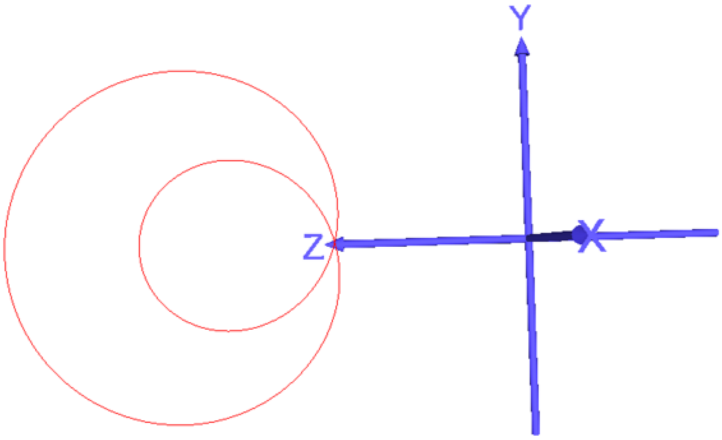


x=12;
*y=(sin(2*pi*u))*(3-12*cos(2*pi*u));*
*z= 12-((cos(2*pi*u))*(3-12*cos(2*pi*u)));*
$u \in [0,1]$
Resolution: [300]
At slightly lower resolutions, the curve may not appear to be as smooth and rounded as it could be, hence this resolution was selected

Name of the file: 1b.wrl



Resolution: [25]
If divided into too few segments, e.g. 25, the smooth curvature is not seen. The straight lines connecting the curve (i.e. linear interpolation) can be observed



Resolution: [500]
If divided into even more segments, e.g. 500, an even smoother curvature is produced. The straight lines connecting the points are not as striking. But further such increases in resolution would result in minor changes (not easily observed by the naked eye) to the curve at the expense of additional computations

Q2

First, we can modify the rotation matrix from Q1 as such:

$$\mathbf{R2} = \begin{bmatrix} \cos\left(\frac{\pi}{2} \cdot t\right) & 0 & \sin\left(\frac{\pi}{2} \cdot t\right) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\left(\frac{\pi}{2} \cdot t\right) & 0 & \cos\left(\frac{\pi}{2} \cdot t\right) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

	Next, we can obtain the new affine transformation matrix by multiplying the matrices in order:

$$\begin{bmatrix} 1 & 0 & 0 & 12 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\left(\frac{\pi}{2}\cdot t\right) & 0 & \sin\left(\frac{\pi}{2}\cdot t\right) & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\left(\frac{\pi}{2}\cdot t\right) & 0 & \cos\left(\frac{\pi}{2}\cdot t\right) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & -12 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\left(\frac{\pi}{2}\cdot t\right) & 0 & \sin\left(\frac{\pi}{2}\cdot t\right) & 12-12\cdot\cos\left(\frac{\pi}{2}\cdot t\right) \\ 0 & 1 & 0 & 0 \\ -\sin\left(\frac{\pi}{2}\cdot t\right) & 0 & \cos\left(\frac{\pi}{2}\cdot t\right) & 12\cdot\sin\left(\frac{\pi}{2}\cdot t\right) \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Multiplying the affine transformation matrix above with the curve from the previous experiment, we get (using intermediate parameter τ instead of t):

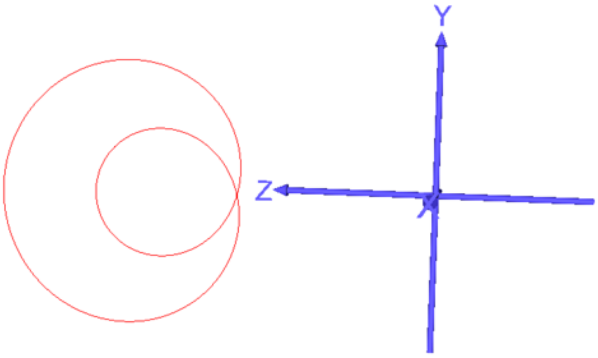
x=(cos(pi/2*τ)*((cos(2*pi*u))*(3-12*cos(2*pi*u))-12))+12;
y=(sin(2*pi*u))*(3-12*cos(2*pi*u));
z=sin(pi/2*τ)*(12-(cos(2*pi*u))*(3-12*cos(2*pi*u)));
$u \in [0,1]$

Lastly, since the curve will be displayed as a 5 seconds rotation motion with some deceleration, we can get an equation for τ=f(t):

τ = sin(pi/2*t);

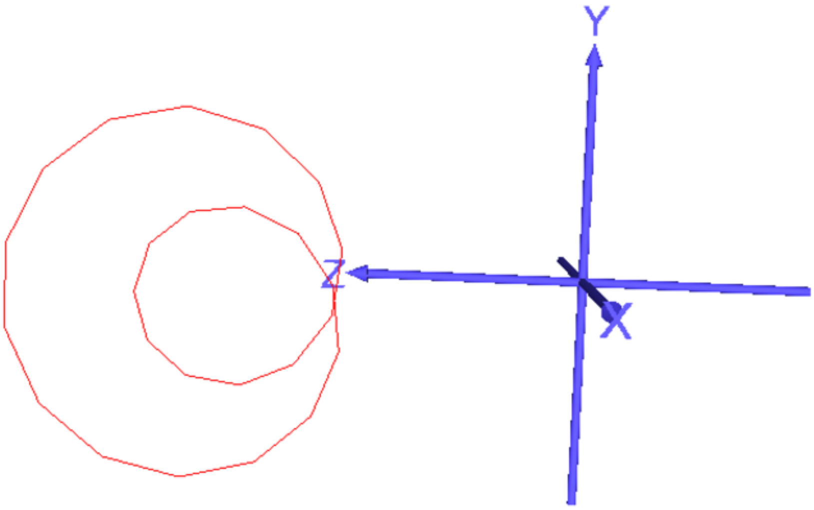
Substituting that in, we end up with the final equation:

*x=(cos(pi/2*sin(pi/2*t))*((cos(2*pi*u))*(3-12*cos(2*pi*u))-12))+12;*
*y=(sin(2*pi*u))*(3-12*cos(2*pi*u));*
*z=sin(pi/2*sin(pi/2*t))*(12-(cos(2*pi*u))*(3-12*cos(2*pi*u)));*
$u \in [0,1]$
cycleInterval = 5
Resolution: [300]

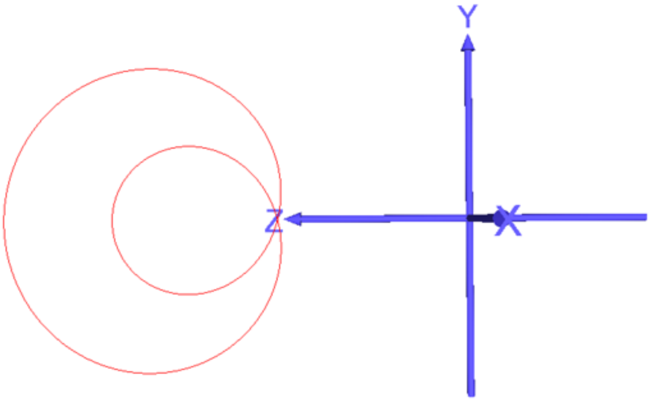


For this question, since the curve displayed is essentially the same as that in q1b (aside from the rotation), the resolution used is the same

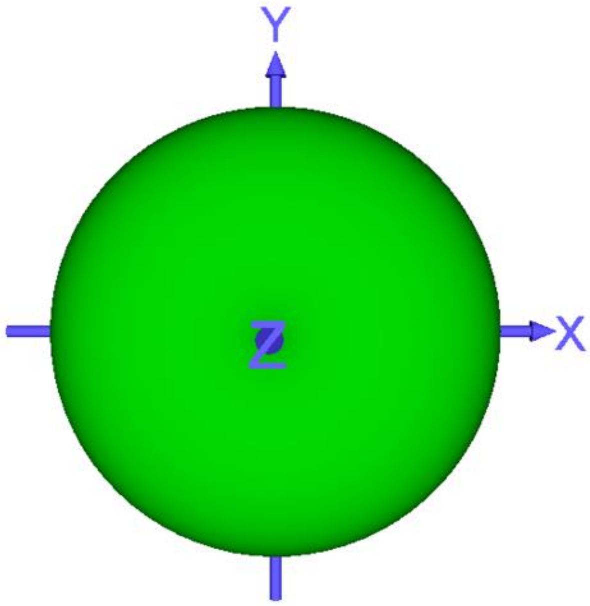
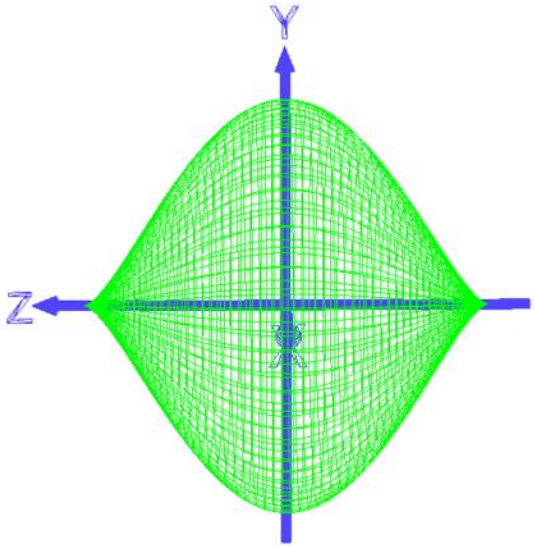
Name of the file: 2.wrl

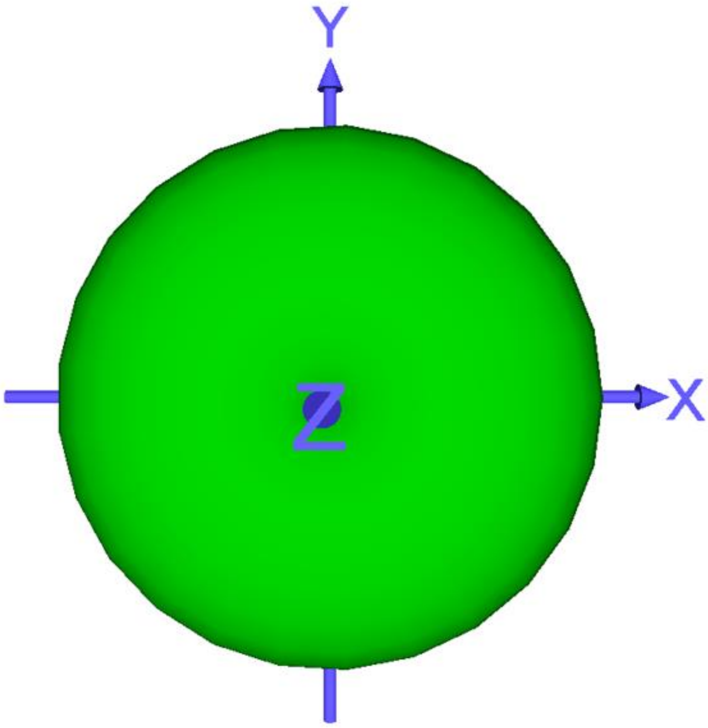
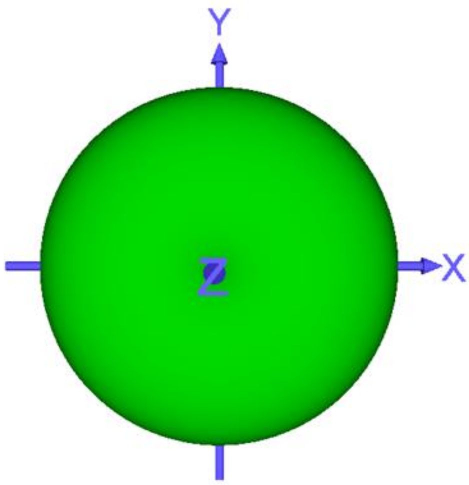
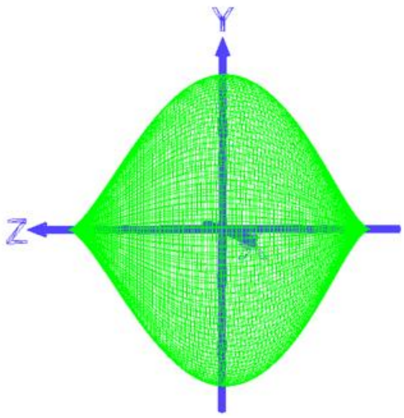


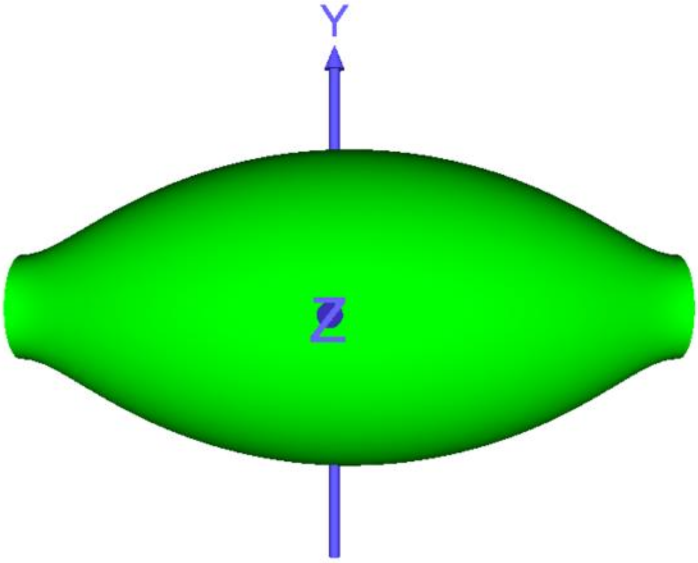
Resolution: [25]
If divided into too few segments, e.g. 25, the smooth curvature is not seen. The straight lines connecting the curve (i.e. linear interpolation) can be observed

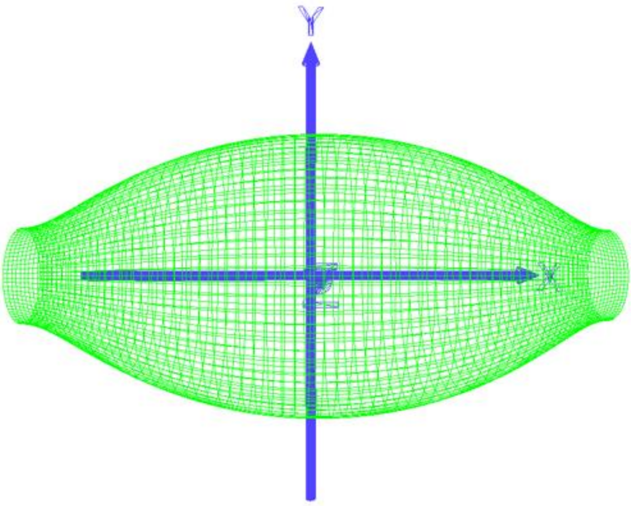


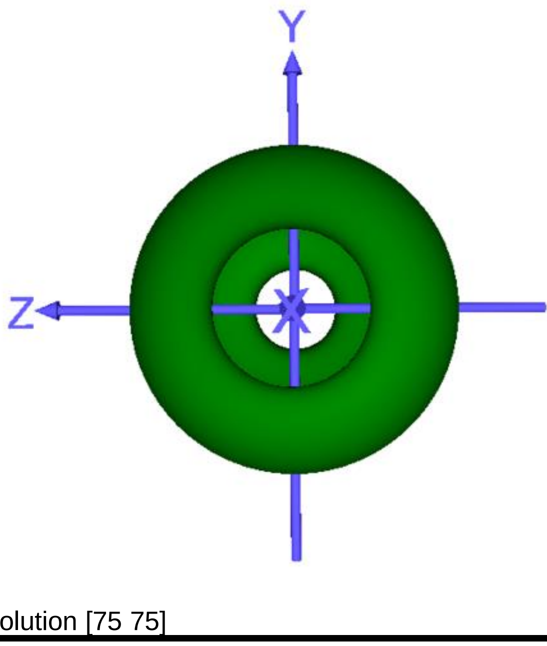
	Resolution: [500] *If divided into even more segments, e.g. 500, an even smoother curvature is produced. The straight lines connecting the points are not as striking. But further such increases in resolution would result in minor changes (not easily observed by the naked eye) to the curve at the expense of additional computations*

Q3a (Surface 7**)**	With reference to Table 3, convert to (u, v), $y(u, v)$, $z(u, v)$, $u, v \in$ [0,1] definitions of surfaces 7 and 10 and display them.
	
	Surface 7:
	
	
	
	
	resolution [75 75]
	x = cos(0.5*4*pi*u)*(sin(pi*v))^3;
	y = sin(0.5*4*pi*u)*(sin(pi*v))^3;
	z = cos(pi*v);
	$u, v \in [0,1]$
	At this resolution, the polygons can be seen to be quite even and not too elongated. Curvature looks quite smooth too

	Name of the file: 3aSurface7.wrl
	Additional screenshots explaining the selection of the sampling resolution, e.g., with a smaller and bigger resolutions. Write the tested resolutions.  resolution [25 25] At this resolution, the polyline interpolation may be observed at the edges and hence the curvature is not so smooth/rounded   resolution [125 125] At this resolution, curvature is smooth, and there are more, smaller polygons. However the polygons are so small the difference is not significantly noticeable from afar or even to the naked eye, at the expense of the extra computations from additional rendering involved. Hence it is not quite the minimal sampling resolution
Q3a (Surface 10**)**	With reference to Table 3, convert to (u, v), $y(u, v)$, $z(u, v)$, $u, v \in [0,1]$ definitions of surfaces 7 and 10 and display them. Surface 10:



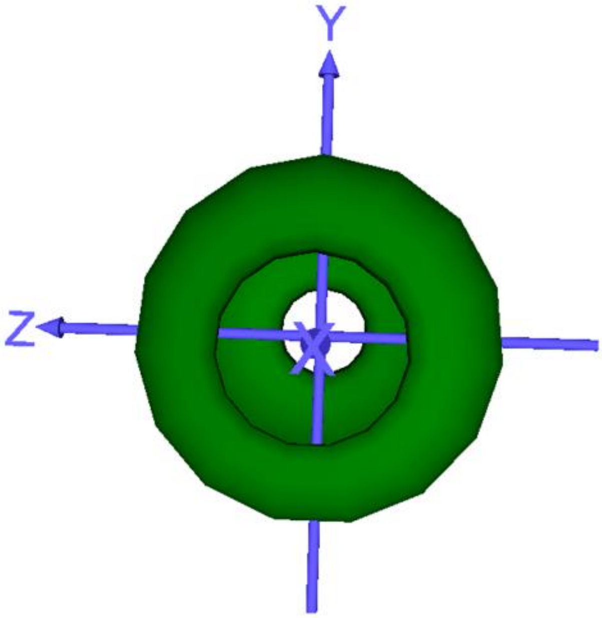




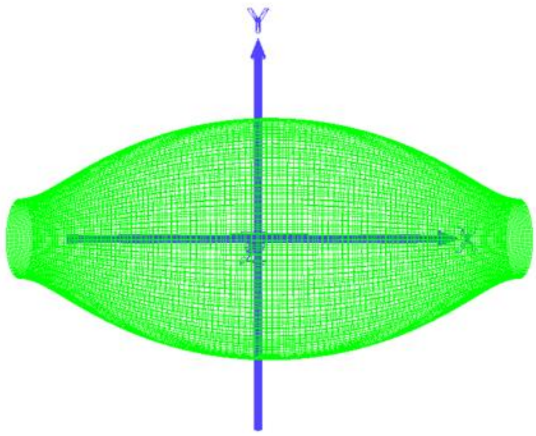
resolution [75 75]

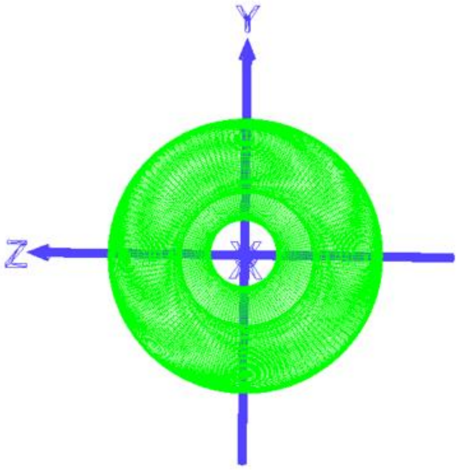
x = 0.5*((2*pi*u - 0.5*sin(2*pi*u)) - 3);
y = 0.5*cos(4*0.5*v*pi)*(1 - 0.5*cos(2*pi*u));
z = 0.5*sin(4*0.5*v*pi)*(1 - 0.5*cos(2*pi*u));
$u, v \in [0,1]$
At this resolution, the polygons can be seen to be quite even and not too elongated. Curvature looks quite smooth and rounded too

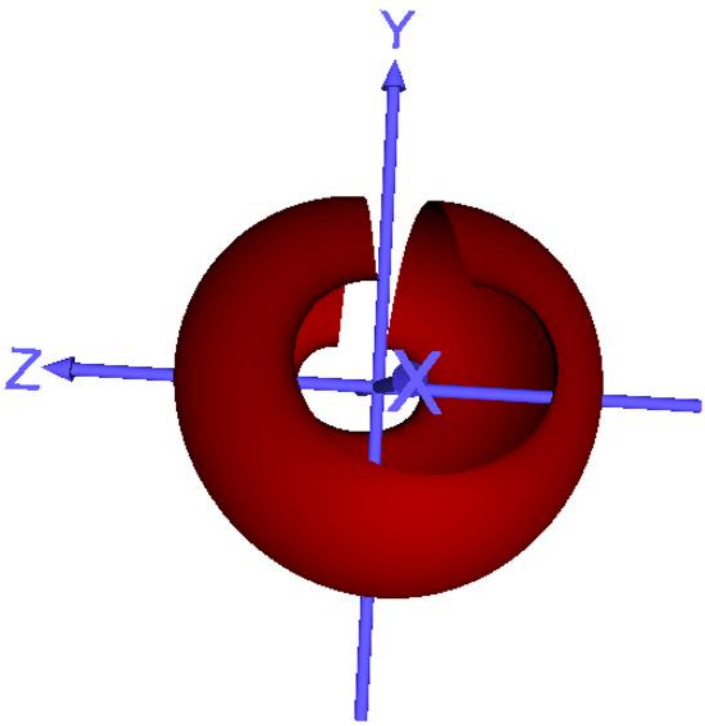
Name of the file: 3aSurface10.wrl

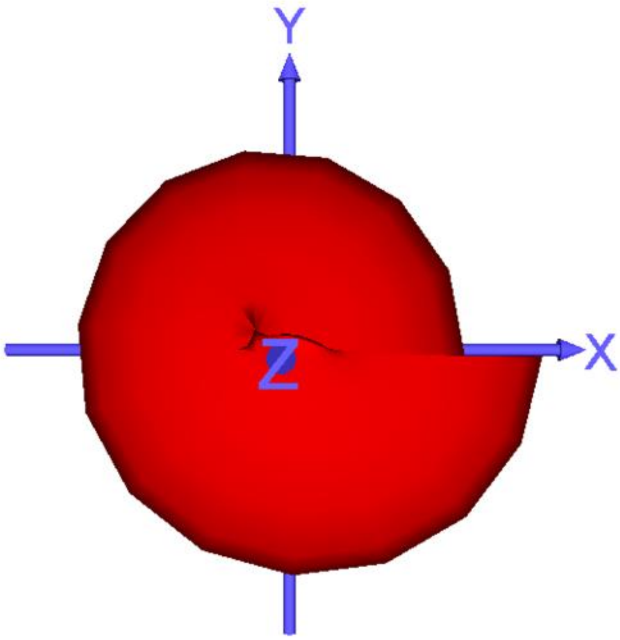
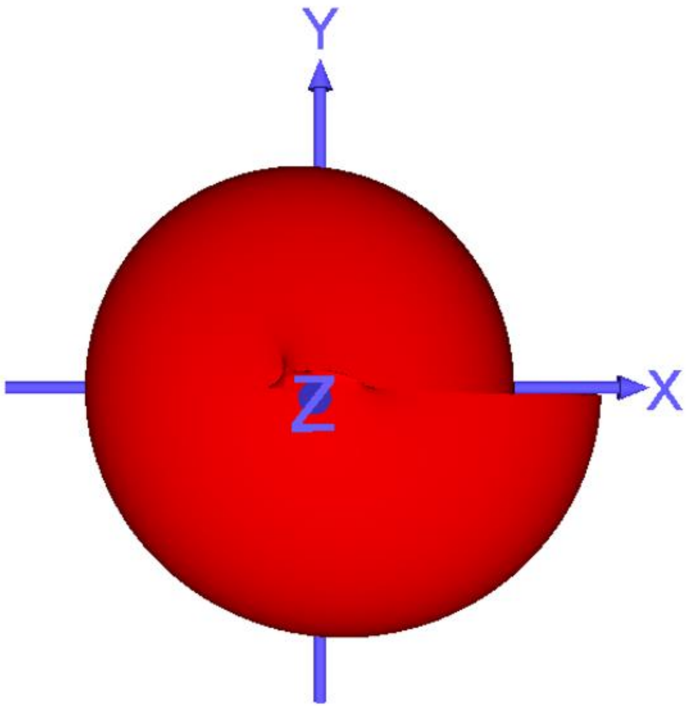


resolution [15 15]
At this resolution, the polyline interpolation may be observed at the edges and hence the curvature is not so smooth/rounded



	 resolution [125 125] At this resolution, curvature is smooth, and there are more, smaller polygons. However the polygons are so small the difference is not significantly noticeable from afar or possibly even to the naked eye, at the expense of the extra computations from additional rendering involved. Hence it is not quite the minimal sampling resolution
Q3b	Define parametrically using (u, v, t), $y(u, v, t)$, $z(u, v, t)$, $u, v, t \in$ [0,1] a swing (back and forth) morphing transformation between surfaces 7 and 10. The morphing animation has to take 5 seconds and it has be done with a uniform speed. function parametric_x(u,v,w,t) { x1=cos(0.5*4*pi*u)*(sin(pi*v))^3; x2=0.5*((2*pi*u - 0.5*sin(2*pi*u)) - 3); return x1*(1-abs(1-2*t))+x2*(abs(1-2*t));} function parametric_y(u,v,w,t) {y1=sin(0.5*4*pi*u)*(sin(pi*v))^3; y2=0.5*cos(4*0.5*v*pi)*(1 - 0.5*cos(2*pi*u)); return y1*(1-abs(1-2*t))+y2*(abs(1-2*t));} function parametric_z(u,v,w,t) {z1 = cos(pi*v); z2 = 0.5*sin(4*0.5*v*pi)*(1 - 0.5*cos(2*pi*u)); return z1*(1-abs(1-2*t))+z2*(abs(1-2*t)); $u, v \in [0,1]$ cycleInterval 5

	resolution [75 75]  Previously, at this resolution, the 2 surfaces above were examined and hence this resolution was selected. From previous examination, at this resolution, the polygons for both surfaces can be seen to be quite even and not too elongated. Curvature for both surfaces looks quite smooth too
	3b.wrl
	Additional screenshots explaining the selection of the sampling resolution, e.g., with a smaller and bigger resolutions. Write the tested resolutions.

	 resolution [15 15] At this resolution, the polyline interpolation may be observed at the edges and the curvature is not so smooth/rounded  resolution [125 125] At this resolution, curvature should be smoother as more rendering is done. However even though the difference is not significantly noticeable from afar, the extra rendering

	computations performed as a result may cause the system to lag if the animation is run at this resolution