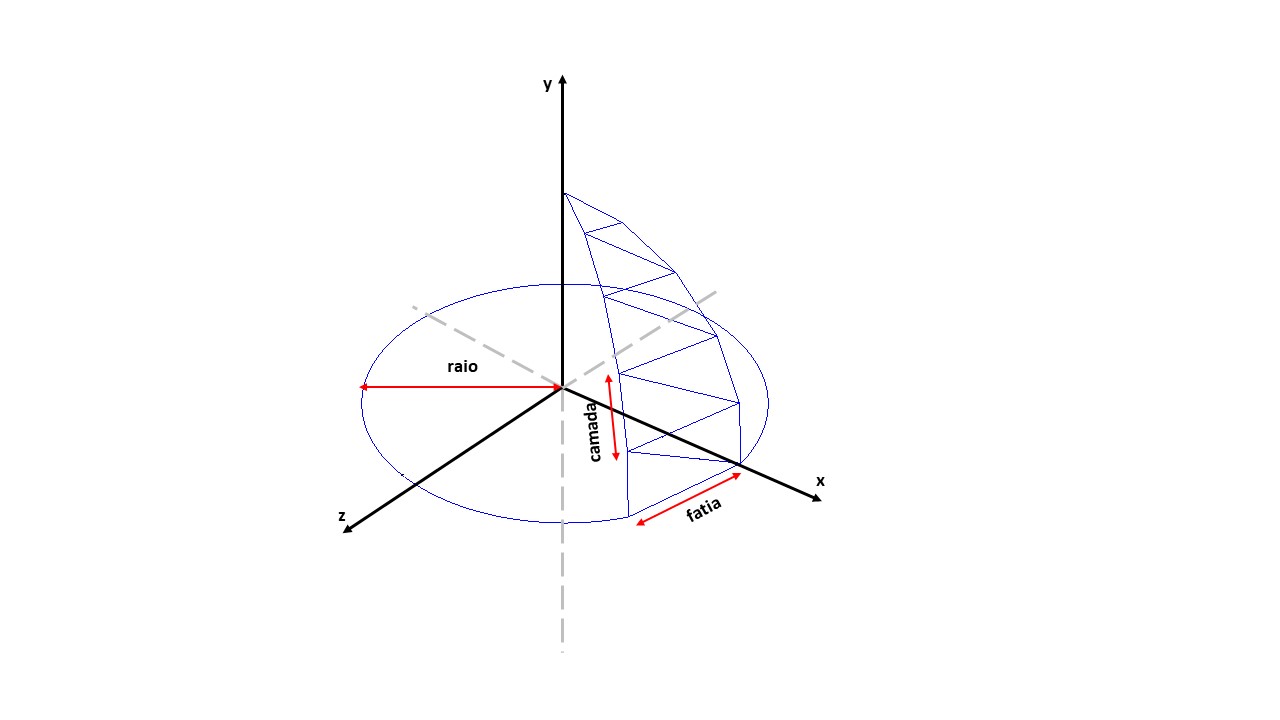
Para desenhar a figura geométrica "esfera" precisamos de um raio, que poderia tomar qualquer valor real positivo, o número de fatias e o número de camadas. Para além disto, desenhar a esfera implicava que trabalhássemos com coordenadas esféricas, de modo que, foi necessário preceder à transformação destas coordenas para cartesianas. Aplicando o conhecimento adquirido de trigonometria convertemos através das seguintes formas:

y = raio \* cos(beta)

x = raio \* sin(beta) \* sin(alpha)

z = raio \* sin(beta) \* cos(alpha)

Sendo que beta está no intervalo 0 < beta < PI e alpha no intervalo 0 < alpha < 2\*PI. Isto implica que por cada fatia que é iterada é desenhada uma parte da esfera (em relação à fatia a esfera é totalmente desenhada no eixo dos yy), ao fim de iterarmos por todas as fatias temos então desenhada a esfera completa. Ou seja, ângulo alpha é referente ao deslocamento na vertical, enquanto que o ângulo beta é o deslocamento na horizontal.

A estratégia adotada pressupunha que para cada fatia eram desenhadas as N camadas, recebidas como parâmetro. Por cada camada eram desenhados 2 triângulos seguindo a regra da mão direita, para que ficasse orientado para o exterior. A iteração foi realizada atrvés de 2 ciclos for, um para a deslocação horizontal e outro para a vertical. O incremento dos ângulos (o que verdadeiramente nos permitia mover) era feito através da fórmula i\*alpha e j\*beta, sendo que tanto i como j estavam delimitados, respetivamente, ao número de fatias e de camadas. Ao aumentar o valor de alpha e beta seguindo uma operação de multiplicação, permitiu diminuir a percentagem de erro que poderia aparecer nas operações de virgula flutuante quando se usam somas.

Sendo assim os pontos seriam:

yCima = raio \* cos(j\*beta)

xAtualCima = raio \* sin(j\*beta)\* cos(i\*alpha)

zAtualCima = raio \* sin(j\*beta)\* sin(i\*alpha)

xProxCima = raio \* sin(j\*beta)\* cos((i+1)\* alpha)

zProxCima = raio \* sin(j\*beta)\* sin((i+1) \* alpha)

yBaixo = raio \* cos((j+1) \* beta)

xAtualCima = raio \* sin((j+1) \* beta)\* cos(i\*alpha)

zAtualCima = raio \* sin((j+1) \* beta)\* sin(i\*alpha)

xProxCima = raio \* sin((j+1) \* beta)\* cos((i+1)\* alpha)

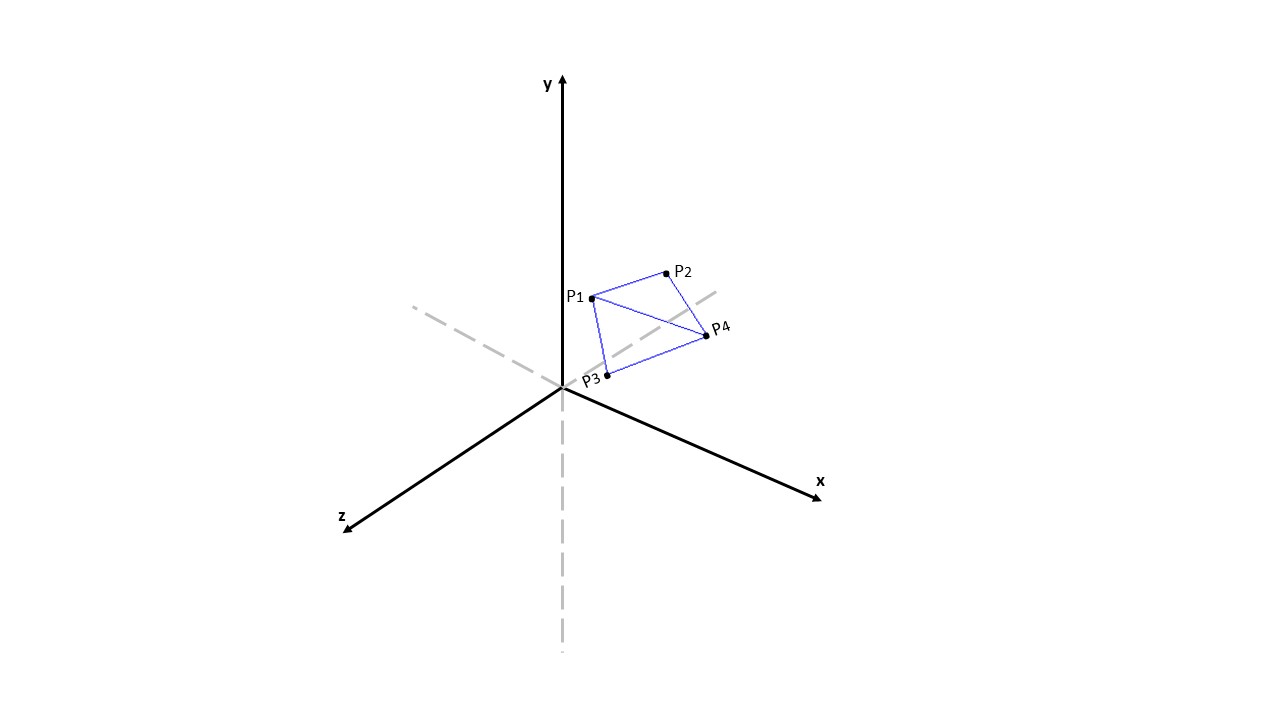
zProxCima = raio \* sin((j+1) \* beta)\* sin((i+1) \* alpha)

P1 = (xAtualCima, yCima, zAtualCima)

P2 = (xProxCima, yCima, zProxCima)

P3 = (xAtualBaixo, yBaixo, zAtualBaixo)

P4 = (xProxBaixo, yBaixo, zProxBaixo)



A orientação dos pontos, para formar o triângulo, seguiu então a seguinte ordem: P4 -> P2 -> P1 e P4 -> P1 -> P3.

De notar que tivemos de ter em atenção que tanto o alpha como o beta iriam começar sempre com o valor de zero. Isto implica que o "alpha começasse sobre o eixo dos zz" e rodasse no sentido anti-horário ao passo que o "beta iniciava no eixo dos yy" tendo uma orientação de cima para baixo. Este facto é explicado pois cos(0) = 1 e sin(0) = 0.