

Universidade do Minho
Escola de Engenharia
Licenciatura em Engenharia Informática

Unidade Curricular de Computação Gráfica

Ano Letivo de 2021/2022

Relatório Fase 3 Curvas, Superfícies cúbicas e VBOs

Grupo

Pedro Araújo	a90614
Pedro Fernandes	a84313
João Cardoso	a94595
Rita Gomes	a87960

Índice

1	Introdução	1
2	Generator	2
2.1	Patches de Bezier	2
2.1.1	Desenvolvimento do Teapot	2
3	Engine	4
3.1	VBOs	4
3.2	Curvas Catmull-Rom	4
3.2.1	Translação	5
3.3	XML	5
4	Resultado Final	7
5	Conclusão	8

1 Introdução

O presente relatório serve de suporte ao trabalho realizado no âmbito da unidade curricular de Computação Gráfica. Este trabalho está dividido em quatro fases, sendo que neste relatório será analisada toda a formalização e modelação envolvida na realização da terceira fase. O objetivo desta fase é expandir as funcionalidades tanto no generator como no engine.

No generator é proposto aos alunos a criação de um novo modelo com base nas patches de Bezier, em que, ao receber um ficheiro patch com os índices das patches e os respetivos pontos, irá criar um .3d com as coordenadas dos triângulos gerados para que o engine possa posteriormente ler e desenhar.

Quanto ao engine, é requerido criar funcionalidades para as rotações e translações das primitivas, que irão ser obrigatoriamente desenhadas a partir de VBOs. Com isso, vêm novos parâmetros a acrescentar nos ficheiros xml, tais como o tempo e pontos de controlo. Para além disso, irão ser realizadas as funções necessárias para o desenho das curvas de Catmull-Rom para as trajetórias dos planetas e respetivas luas do sistema solar. O engine irá também ler o ficheiro .3d criado pelo generator para criar um cometa (que neste caso irá ser um teapot) e desenhá-lo junto com as restantes primitivas.

2 Generator

2.1 Patches de Bezier

O generator passou a suportar novas primitivas com recurso a patches de Bezier. Para implementar isto foi fornecido aos alunos um ficheiro `.patch` que fornece o número de índices e pontos a serem guardados.

A tesselação representa o valor de precisão com que a primitiva irá ser desenhada.

Para o cálculo dos pontos de Bezier utilizamos o polinómio de Bernstein como se mostra na seguinte função:

```
static float* genPolyBernstein(float p) {  
    float* b = (float*) malloc(sizeof(float)*4);  
    b[0] = powf(1 - p, 3);  
    b[1] = 3 * p * powf(1 - p, 2);  
    b[2] = 3 * powf(p, 2) * (1 - p);  
    b[3] = powf(p, 3);  
    return b;  
}
```

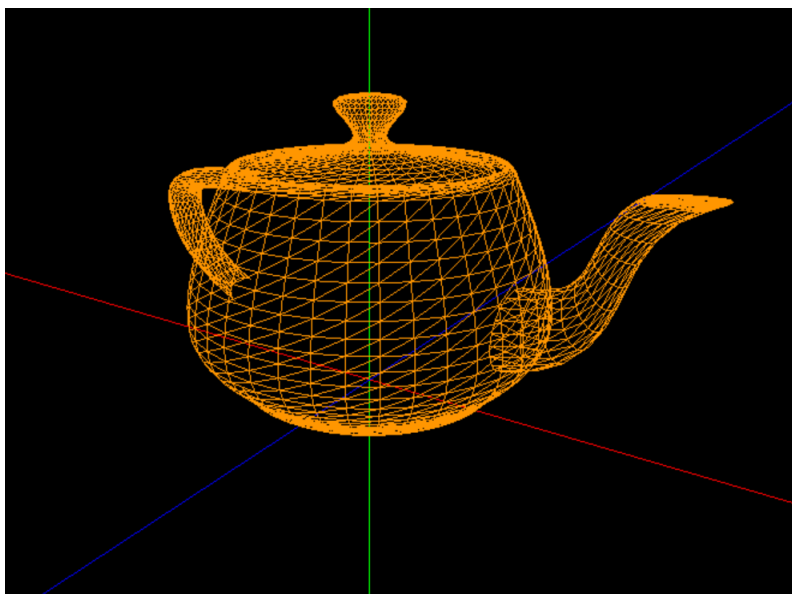
2.1.1 Desenvolvimento do Teapot

Com a adição de um novo modelo para criar, foi implementada uma função *generate-Bezeir* que recebe como argumentos, o nome do ficheiro `.patch`, o nome do ficheiro `.3d`, que irá conter o modelo, e o valor de tesselação.

Numa primeira fase, é feito o parse do ficheiro patch (*teapot.patch*), onde são retirados os o número de patches, os índices de cada patch, que são guardados num vetor de vetores, o numero de pontos de controlo, e por fim, os pontos de controlo, que são guardados num vetor de pontos.

Numa segunda fase, é feito o cálculo dos pontos, segundo o algoritmo de Bezier (Polinómio de Bernstein), onde vão ser definidas as quatro curvas de Bezier. A função implementada para realizar este passo tem como nome *getPointBezier*.

Os pontos gerados irão ser guardados no ficheiro de output, neste caso, denominado por *bezier.3d*.



3 Engine

3.1 VBOs

A utilização de VBOs (Vertex Buffer Objects) apresenta grandes vantagens a nível de desempenho devido ao facto de já não ser necessária a chamada da função glVertex para cada vértice e, para além disso, explora o paralelismo disponibilizado pelo GPU visto que estamos a desenhar cada triângulo imediatamente ao receber cada conjunto de três vértices.

Sendo que na representação do Sistema Solar só serão utilizadas algumas das figuras construídas no generator, o grupo decidiu que apenas as utilizadas seriam renderizadas com o uso de VBOs.

3.2 Curvas Catmull-Rom

Um dos objetivos principais nesta fase era definir as órbitas dos planetas através da definição das curvas de Catmull-Rom. Estas curvas são uma implementação de cálculos de pontos através de matrizes.

Para cálculo das mesmas seja realizado, é necessária a existência de no mínimo 4 pontos de controlo para definir uma curva.

Tendo os pontos de controlo obtido, e o tessellation definido como igual a 100, basta apenas calcular os pontos da curva, assim como a sua derivada para todas as coordenadas seguindo as seguintes fórmulas:

$$P(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} -0.5 & 1.5 & -1.5 & 0.5 \\ 1 & -2.5 & 2 & -0.5 \\ -0.5 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}$$
$$P'(t) = \begin{bmatrix} 3t^2 & 2t & 1 & 0 \end{bmatrix} \begin{bmatrix} -0.5 & 1.5 & -1.5 & 0.5 \\ 1 & -2.5 & 2 & -0.5 \\ -0.5 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

Para posicionar objetos ao longo da curva e percorrer a mesma, é necessário calcular novamente os pontos da curva definida, utilizar a derivada para o cálculo das coordenadas do objeto ao longo da curva. Também foi necessário realizar a leitura do tempo que o objeto necessita para percorrer a curva em 360 graus, para assim definir a "velocidade" com que o objeto percorre a curva.

3.2.1 Translação

Para se poder implementar as novas formas de translação, foi necessário introduzir novas variáveis como o tempo e dois vetores de pontos, um deles guarda os pontos lidos do ficheiro XML e o outro contém os pontos de uma curva Catmull-Rom.

A inserção dos valores no vetor da curva é feita através duma função *geraPontosCurva* que utiliza funções já definidas e utilizadas em aulas práticas, como é o caso de *getCatmullRomPoint* e *getGlobalCatmullRomPoint*, para gerar 100 pontos de uma curva, que representará a órbita dos planetas.

3.3 XML

Como era requerido, tínhamos de criar um XML próprio para o Sistema Solar, foi criado o ficheiro *sistemaSolar.xml* com as características necessárias para que o engine consiga fazer a sua leitura. Deste modo, o ficheiro segue o seguinte padrão para os restantes planetas do sistema solar, bem como a lua e o cometa (que neste caso utiliza o modelo *bezier.3d*).

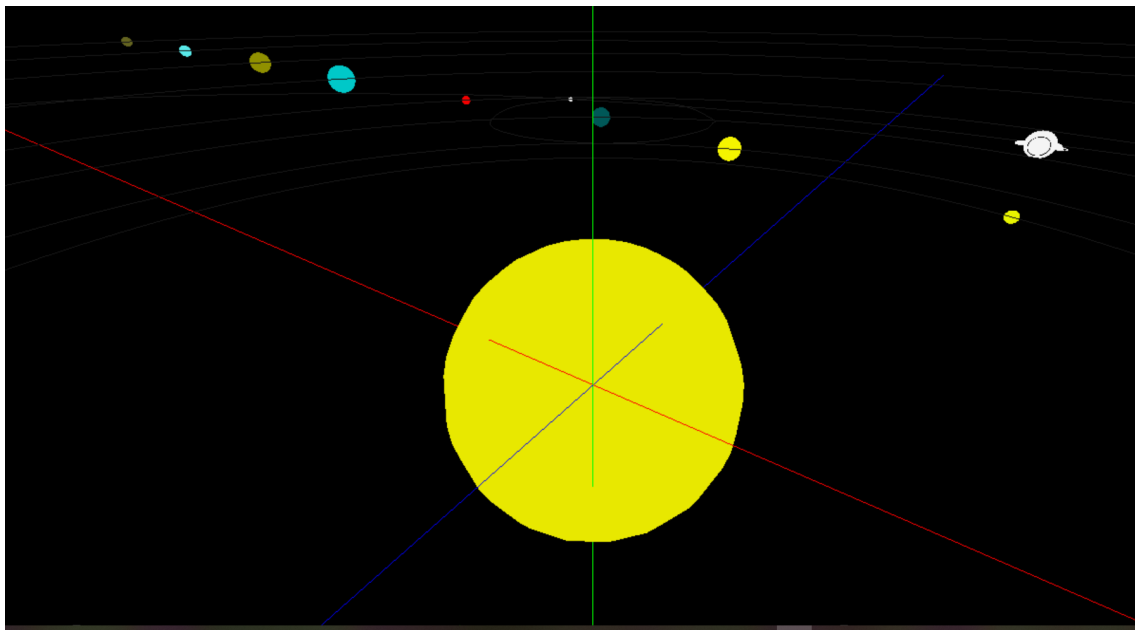
```

<world>
  <camera>
    <position x="10" y="10" z="10" />
    <lookAt x="0" y="0" z="0" />
    <up x="0" y="1" z="0" />
    <projection fov="60" near="1" far="1000" />
  </camera>
  <group>
    <!--SOL-->
    <group>
      <transform>
        <rotate time="45" x="0" y="1" z="0" />
        <scale x="5" y="5" z="5" />
      </transform>
      <models>
        <model file="esferaSS.3d" R="1000" G="1000" B="0" />
      </models>
    </group>
    <!--MERCURIO-->
    <group>
      <transform>
        <translate time="60">
          <point x="30.0" y="0" z="0.0"/>
          <point x="27.406363729278027" y="0" z="12.202099292274005"/>
          <point x="20.073918190765745" y="0" z="22.294344764321824"/>
          <point x="9.270509831248424" y="0" z="28.531695488854606"/>
          <point x="-3.1358538980296" y="0" z="29.835656861048204"/>
          <point x="-14.999999999999993" y="0" z="25.98076211353316"/>
          <point x="-24.27050983124842" y="0" z="17.633557568774197"/>
          <point x="-29.344428022014167" y="0" z="6.237350724532792"/>
          <point x="-29.34442802201417" y="0" z="-6.237350724532772"/>
          <point x="-24.270509831248425" y="0" z="-17.63355756877419"/>
          <point x="-15.000000000000014" y="0" z="-25.980762113533153"/>
          <point x="-3.135853898029627" y="0" z="-29.8356568610482"/>
          <point x="9.270509831248416" y="0" z="-28.53169548885461"/>
          <point x="20.073918190765735" y="0" z="-22.29434476432184"/>
          <point x="27.406363729278016" y="0" z="-12.202099292274028"/>
        </translate>
        <rotate time="15" x="0" y="1" z="0"/>
        <scale x="0.4" y="0.4" z="0.4" />
      </transform>
      <models>
        <model file="esferaSS.3d" R="1000" G="500" B="0" />
      </models>
    </group>
    <!--VENUS-->
    <group>

```


4 Resultado Final

Após todo o trabalho desenvolvido o resultado gráfico obtido pode ser visto a seguir.



5 Conclusão

Com a elaboração desta fase do projeto, o grupo conseguiu aprofundar conhecimento relativo às curvas de Bezier e às de Catmull-Rom, que foram utilizadas para efetuar transformações sobre os objetos através de pontos de controlo e outros dados relativos.

O grupo considera que realizou esta etapa com sucesso na medida em que conseguimos implementar o que era proposto com sucesso e também vemos que o Sistema Solar está cada vez mais próximo da realidade.