



Universidade do Minho
Escola de Engenharia
Licenciatura em Engenharia Informática

Unidade Curricular de Computação Gráfica

Ano Letivo de 2022/2023

Grupo 24

André Castro Alves (a95033) Cláudio Alexandre Freitas Bessa (a97063)
José Fernando Monteiro Martins (a97903)

5 de maio de 2023

Indice

1 Introdução

1.1 *Generator*

1.2 *Engine*

2 Superfícies de *Bezier*

3 Curvas de *Catmull-Rom*

4 Ficheiro XML

5 Controlos do Utilizador

6 Conclusão

1 Introdução

O documento apresentado visa apresentar as metodologias praticadas e processos utilizados durante a terceira fase do projeto que tem vindo a ser realizado no âmbito da UC de Computação Gráfica.

1.1 *Generator*

Nesta fase é requerido voltar à implementação da aplicação do *generator* e adicionar uma nova primitiva baseada em *patches* de *Bezier*. Esta implementação irá ser melhor explicada na secção seguinte.

1.2 *Engine*

Por sua vez, o *engine* sofre novamente alterações, adicionando agora as curvas de *Catmull-Rom*, de modo a animar os diferentes corpos celestes, e.g. planetas, estrelas, etc. Para isso era necessário algumas implementações relativas às transformações, i.e. translações e rotações. Estas alterações descrevem-se da seguinte forma:

- **Translação** -> recebe o tempo que deve demorar a ocorrer e agora pode também ser pai para um conjunto de pontos de modo a definir as curvas
- **Rotação** -> recebe o tempo que uma rotação de 360 deve demorar, eixos a usar e se está alinhado nos eixos ou não

2 Superfícies de *Bezier*

Na aplicação do *generator* era solicitada uma implementação de uma nova *primitiva*. Sendo esta baseada nas superfícies de *Bezier*, para poder ter diferentes níveis de tesselação.

O nível de tesselação define os incrementos a ser feitos a dois vetores, **u** e **v**, sendo o incremento calculado do seguinte modo:

$$vertical_tessellation_inc = horizontal_tessellation_inc = \frac{1}{tessellation_level}$$

Após este cálculo, temos apenas que ler o ficheiro de entrada que contém todos os *patches*. De relembrar que o formato dos ficheiros vai de encontro ao definido pelo corpo docente. De seguida, calcula-se, de forma iterativa, todos os valores de **u** e **v** utilizando a seguinte fórmula:

$$p(u,v) = \begin{bmatrix} u^3 & u^2 & u & 1 \end{bmatrix} \begin{bmatrix} -1 & -3 & 3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_{00} & P_{10} & P_{20} & P_{30} \\ P_{01} & P_{11} & P_{21} & P_{31} \\ P_{02} & P_{12} & P_{22} & P_{32} \\ P_{03} & P_{13} & P_{23} & P_{33} \end{bmatrix} \begin{bmatrix} -1 & -3 & 3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} v^3 \\ v^2 \\ v \\ 1 \end{bmatrix}$$

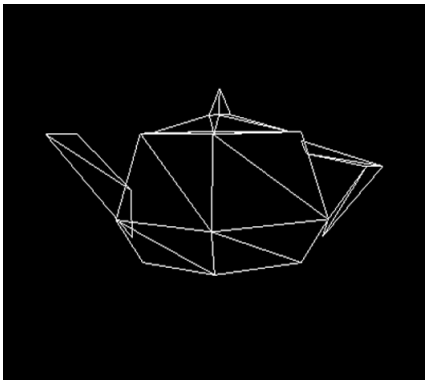


Figura 2.1: Nível de tesselação 1 e 20, respetivamente

3 Curvas de *Catmull-Rom*

De igual forma, era pedido a implementação de animações no, até então, esquema estático do sistema solar. Da seguinte forma, os diferentes planetas podem efetuar uma animação similar às órbitas existentes em torno do sol, ou outros casos, e.g. órbita das luas. Para isso são utilizadas *splines*, mais especificamente, curvas de *Catmull-Rom*, onde tem como obrigação ter no mínimo 4 pontos, sendo a sua equação manipulada da seguinte forma:

$$p(t) = \begin{bmatrix} t^3 & t^2 & t & 1 \end{bmatrix} \begin{bmatrix} -0.5 & 1.5 & -1.5 & 0.5 \\ 1 & -2.5 & 2 & -0.5 \\ -0.5 & 0 & 0.5 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} P_0 \\ P_1 \\ P_2 \\ P_3 \end{bmatrix}$$

Através dos pontos recebidos no ficheiro XML do sistema solar, podemos calcular a órbita de cada corpo celeste. O cálculo das posições de cada componente do sistema resulta num conjunto de *frames* que tem uma dependência com o tempo total que foi definido no atributo *time* da **translação**.

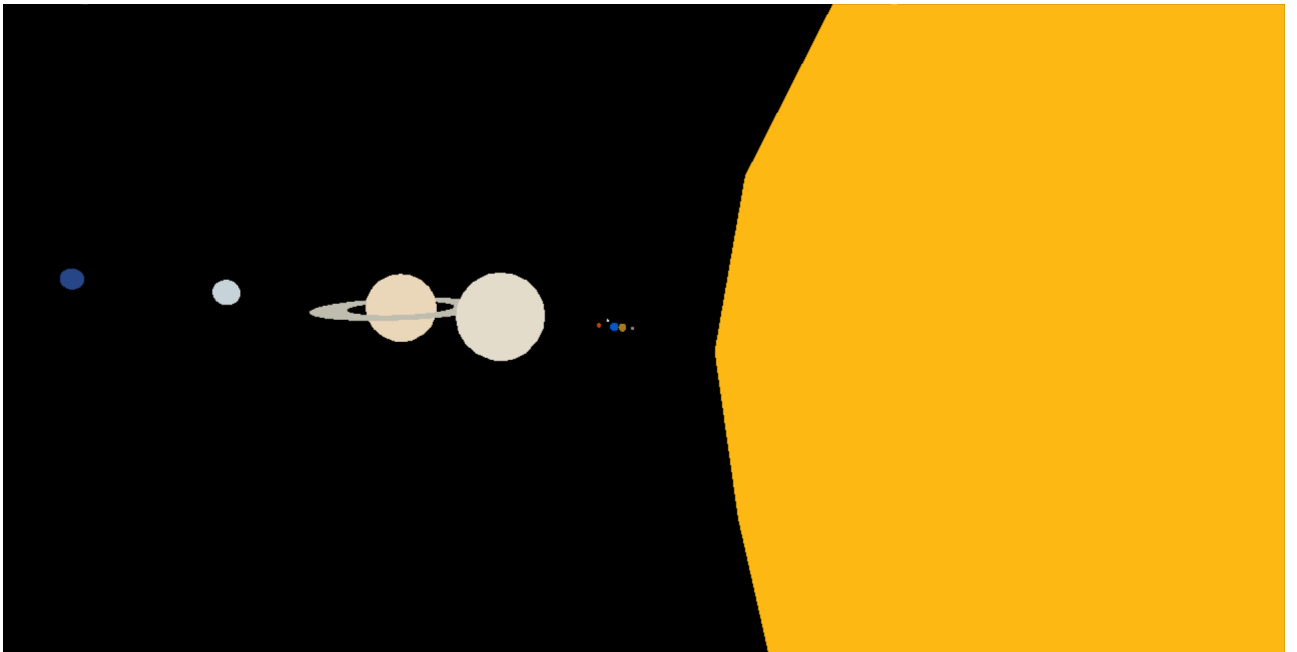


Figura 3.1: Sistema Solar

4 Ficheiro XML

Seguindo o requisito da adição de curvas de *Catmull-Rom* foram implementadas diferentes preparações para a leitura do XML. Isto devido à implementação de novos atributos nas diferentes transformações, exceto a escala, e a implementação dos diferentes pontos das curvas como filhos de uma translação.

Tal como mencionado previamente, o ficheiro XML agora aceita translações onde recebe como atributo o tempo que demora a efetuar a mesma e se está alinhada na curva, curva essa que tem, os respetivos pontos onde define a criação das curvas de *Catmull-Rom*.

Por sua vez, as rotações podem agora receber além dos eixos, o tempo que demora a efetuar uma volta de 360 sobre o(s) eixo(s).

Devido à existência destas novas implementações foi acrescentado um ficheiro titulado de **config.xml** que possui o ambiente pedido para esta fase, contendo um sistema solar animado com um asteróide.

5 Controlos do Utilizador

Visto que já tínhamos um conjunto razoável de controlos de *scene* que podiam ser utilizados pelo utilizador, sendo a maior parte relativos à câmara 3D, foi também acrescentado a apresentação das curvas de *Catmull-Rom*.

- **Key 'C'** -> Ligar/Desligar apresentação de curvas de *Catmull-Rom*

6 Conclusão

Em suma, chegando ao período de entrega desta terceira etapa do trabalho prático, comprovamos que aplicamos os conhecimentos lecionados, porém há partes que podem ser melhoradas na fase final.

Já na fase anterior tínhamos visualizado e efetuado algumas implementações que utilizavam *frustum culling*, sem um propósito. Visto isto, se tudo correr como pretendido, iremos finalizar essa implementação.

Contudo, apesar de tudo ter sido desenvolvido, consideramos que esta foi a fase que encontramos mais dificuldades, quer pelo nível de complexidade desta fase, quer por ter tido menos tempo para efetuar a menos, daí ter havido uma preocupação em adiantar a realização de VBOs, etc, na fase anterior.