Computação Gráfica (3ºano)

Trabalho Prático – Fase 3

Relatório de Desenvolvimento

Alexis Correia	João Fonseca	Moisés Ferreira	Rita Machado
a102495	a102512	a97020	a102508

27 de abril 2025

Conteúdo

Capítulo 1

Introdução

Nesta terceira fase, temos como principal objetivo a criação de um novo tipo de modelo baseado em patches e Superfícies de Bezier. Para que tal seja possível, foi necessário atualizar o Generator, para que este recebesse novos parâmetros, que consiste num ficheiro em que nele é definido os pontos de controlo e também o nível de tesselação da superfície.

Além disso, realizamos alterações no Engine, uma vez que este precisou de ser estendido para poder receber novos elementos de rotação e translação. Considerando a translação, um conjunto de pontos será fornecido para definir uma curva cubica Catmull-Rom, bem como o tempo, em segundos, para percorrer toda a curva. Já a rotação, poderá fornecer um valor de tempo – ao invés de um ângulo – e este será o tempo necessário para realizar uma volta completa.

No fim, de forma a melhorar o desempenho de renderização, foi necessário desenhar os modelos com VBO's, que consiste em armazenar dados num buffer de memória, o que permite que estes sejam consultados de forma mais rápida.

Capítulo 2

Desenvolvimento

2.1 Generator

Nesta etapa geram-se os pontos, guardando-os num ficheiro, os quais mais tarde serão usados para desenhar as figuras no Engine. A funcionalidade adicionada permite construir modelos com base em curvas de Bezier. Para criar uma curva de Bezier, são necessários 4 pontos de controlo, como só temos um conjunto de pontos, precisamos de calcular os restantes usando a fórmula de Bezier. A alteração do Generator consiste em que este possa ler ficheiros patches de Bezier, realizar os cálculos matriciais dos pontos de acordo com o número da tesselação e retornar a lista de triângulos necessários para desenhar a superfície para o ficheiro de saída.

Para a obtenção dos pontos necessários à construção de cada superfície foi utilizada a seguinte fórmula:

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

Em que:

$$M = \begin{bmatrix} -1 & 3 & -3 & 1 \\ 3 & -6 & 3 & 0 \\ -3 & 3 & 0 & 0 \\ 1 & 0 & 0 & 0 \end{bmatrix}$$

- M^T é a transposta de M (neste caso: $M=M^T$, pois M é uma matriz simétrica);
- $C_0 = \{P_{00}, P_{10}, P_{20}, P_{30}\}$;
- $C_1 = \{P_{01}, P_{11}, P_{21}, P_{31}\}$;
- $C_2 = \{P_{02}, P_{12}, P_{22}, P_{32}\}$;
- $C_3 = \{P_{03}, P_{13}, P_{23}, P_{33}\}$;
- C_0, C_1, C_2, C_3 curvas de Bezier.

Os vários pontos da superfície são obtidos através da função *p* - variando u e v entre [0, 1], de acordo com a especificação do nível de tesselação – e como resultado obtemos uma superfície mais suave e realista.

2.2 Engine

Foram feitas alterações quanto às translações e rotações, de forma que estas fossem capazes de, no caso da translação, definir uma curva cubica *Catmull-Rom*, bem como o número de segundos para percorrer toda a curva e no caso da rotação fazer com que esta funcionasse em função do tempo.

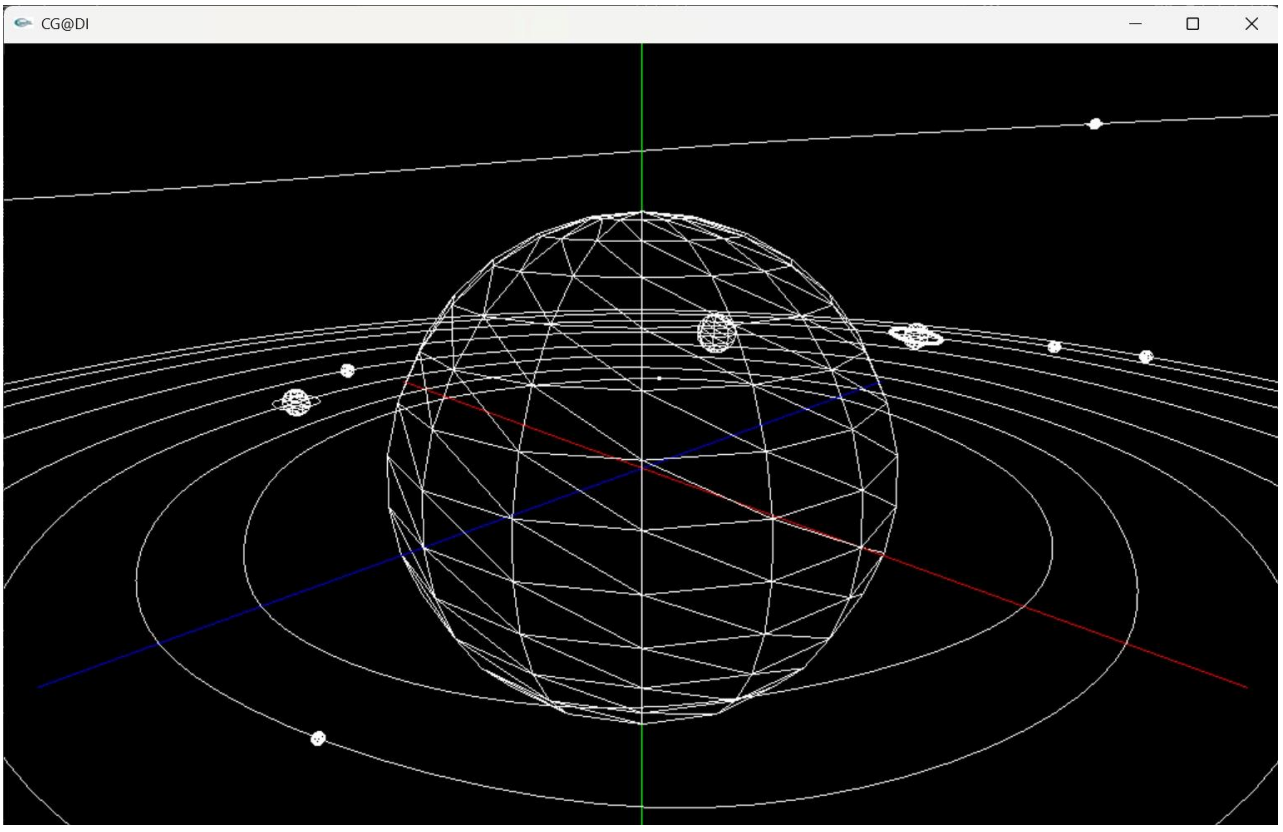
Para alcançar este resultado, começamos por alterar a função que lê o ficheiro XML com o parser e adicionar os novos casos. Depois disso, alteramos o **renderScene** para atender às novas possibilidades de transformações e, por fim, criamos a função **renderCatmullRomCurve**, **getGlobalCatmullRomPoint** e **getCatmullRomPoint.** Estas funções, basicamente, desenham a curva descrita e garantem que o objeto em questão segue o caminho no tempo determinado e com (o sem, se for o caso) o alinhamento desejado.

Capítulo 3

Sistema Solar

De acordo com o enunciado, alteramos o ficheiro XML formulado na fase anterior. Agora, o nosso modelo do Sistema solar é dinâmico, ou seja, os planetas orbitam ao redor do Sol - em orbitas e tempos variados – e a Lua orbita a Terra. Além disso, os planetas agora possuem uma rotação própria e giram continuamente em torno do seu eixo. Além disso, adicionamos um "cometa" que passa no fundo da imagem.

Vale notar que, as órbitas, o tempo de translação e rotação dos planetas, bem como suas escalas e distâncias foram feitos com valores arbitrários, escolhidos pelo grupo.



Capítulo 4

Conclusão

O grupo sentiu que até agora esta foi a fase mais complexa do trabalho o que consequentemente levou a uma atenção especial e pesquisa exaustiva. Contudo, o grupo considera que conseguiu concluir com sucesso mais uma fase do trabalho, mas sempre com a certeza de que há aspetos que futuramente podem ser melhorados.

Capítulo 5

Extras

- **Movimentação da câmara:** através das teclas *a*, *w*, *s*, *d* ou *arrowkey*. Isso é feito através do calcula da coordenada esférica da câmara, e posteriormente calculamos novamente as coordenadas cartesianas após cada input do teclado.
    - Alteramos parte do cálculo da coordenada esférica do ponto inicial da câmara.

- Tanto no Generator quanto no Engine, os cálculos matriciais necessários para calcular a superfície de Bezier e a curva de Catmull-Rom, respetivamente, são feitos com auxílio das funções auxiliares **multMtrixVector**, **multVectorMatrix**, **multVectorVector**, **normalize**, **length**, **cross** e **buildRotMatrix**
    - Estas funções foram escritas com base em funções vistas durante as aulas práticas, por isso utilizamos arrays de floats e nos certificamos de, no final, salvar os resultados em variáveis do tipo *vertice* (Struct criada pelo grupo na Fase 1 para o trabalho).