

## Proto3d: Um simulador computacional acessível com detecção de colisão.

Róger Matheus Lasch<sup>1</sup>, José Antônio de Figueiredo<sup>1</sup>

<sup>1</sup>Curso Bacharel em Ciência da Computação – Instituto Federal Sul-rio-grandense (IFSUL)  
Passo Fundo – RS – Brasil

rogerlasch@gmail.com, josefigueiredo@ifsul.edu.br

**Abstract.** *Proto3d is a rigid body simulator and collision detection system that presents its results in an Accessible Registry designed for individuals with visual impairments. The application was validated through the simulation of a sphere dropping from a specific height onto a static box, resulting in the sphere bouncing multiple times. The values presented in the Accessible Registry can be verified through calculations based on Newtonian physics. The results demonstrate that the prototype has achieved its simulation goal with accuracy and can be applied in various scenarios, including educational technology.*

**Resumo.** *Proto3d é um simulador de corpos rígidos e detecção de colisão, que apresenta os resultados em um Registro Acessível às pessoas com deficiência visual. A aplicação foi validada com a simulação de uma esfera, que cai de uma determinada altura sobre uma caixa estática. A queda faz a esfera quicar algumas vezes. Os valores apresentados no Registro Acessível podem ser validados por cálculos de física newtoniana. Os resultados demonstram que o protótipo atingiu seu objetivo de simulação com acurácia, podendo ser aplicado em diversos cenários, inclusive com tecnologia educacional.*

### 1. Introdução

O objetivo do trabalho é desenvolver um simulador de corpos rígidos com detecção de colisão, de interface acessível. O estudo justifica-se já que simuladores desenvolvidos com o viés de acessibilidade, ainda são uma área pouco explorada. Ainda segundo [Ericson 2004], a detecção de colisão continua sendo um problema fundamental, de relevância para a ciência moderna.

O artigo apresenta os resultados obtidos com o desenvolvimento do Proto3d, que simula movimento e colisão de corpos rígidos. O projeto apresenta a execução das simulações em uma saída projetada para atender às necessidades de pessoas com deficiência visual. O método proposto registra os resultados no formato textual, em substituição à representação gráfica, tornando a saída acessível à todos os usuários.

O restante do artigo é organizado da seguinte maneira: A seção 2 apresenta um breve referencial teórico que fundamenta a construção do simulador proposto; a seção 3 apresenta o planejamento da aplicação e a modelagem básica das classes; a seção 4 discute os resultados obtidos com a aplicação, através da demonstração de uma simulação; A seção 5 traz algumas considerações finais e registra possibilidades de trabalhos futuros. Por fim, são apresentadas as referências bibliográficas utilizadas.

## 2. Simulação Computacional

Segundo [Durán 2018], a simulação computacional é uma técnica amplamente utilizada em diversas áreas, como por exemplo no estudo de sistemas complexos, que por diversos fatores, não podem ser facilmente reproduzidos.

A simulação permite avaliar diversos cenários de uma pesquisa, apenas alterando suas variáveis de controle, reduzindo, assim, os custos e os riscos de um experimento físico. Conforme [Marschner and Shirley 2018], essa técnica pode ser aplicada em diversas áreas como biologia, engenharia, física, química, entre outras. Simulação é uma representação matemática e/ou computacional do comportamento do sistema a ser simulado.

### 2.1. Formas Geométricas e Movimento

A técnica usada para representar uma forma geométrica computacionalmente dependerá do nível de detalhamento desejado [Pharr et al. 2016]. É possível fazer esta representação de diversas maneiras, por exemplo: forma vetorial, paramétrica ou de forma poligonal. Neste trabalho, adota-se a representação vetorial, com coordenadas cartesianas tridimensionais.

Para fins de simplificação, este trabalho limita-se a modelar objetos na forma de esferas e caixas. Esferas são modeladas com um ponto (centro da esfera) e um raio, enquanto caixas são modeladas com um ponto mínimo e um ponto máximo, que representam o canto inferior esquerdo, e o canto superior direito da caixa.

No contexto desta simulação, os objetos podem ser categorizados em duas classes principais: Objetos estáticos e objetos dinâmicos. Os objetos estáticos não se movem por conta própria e nem sob influência de forças externas. Em contrapartida, os objetos dinâmicos podem se mover por conta própria, bem como em resposta a forças externas.

### 2.2. Colisão e Detecção de Colisão

Uma colisão ocorre quando dois ou mais objetos entram em contato, compartilhando algum espaço do ambiente simulado [Ericson 2004]. Segundo [Bourg and Bywalec 2013], a colisão envolve cálculos para determinar as consequências/efeitos desse contato.

Segundo [Bourg and Bywalec 2013], a detecção de colisão é um problema de geometria computacional, que procura determinar *se* e *onde* dois ou mais objetos colidiram. Pode ser detectada de diversas maneiras, dependendo da natureza dos objetos envolvidos [Foley 1996], por exemplo: 1) se a distância entre os dois objetos está abaixo de uma determinada tolerância; 2) se os objetos estiverem sobrepondo-se em um ou mais pontos.

O processo para detecção de colisão é dividido em fase de previsão e de identificação. A previsão busca objetos que possivelmente estejam colidindo, enquanto a identificação localiza as colisões ocorridas. A separação em duas fases distintas garante um consumo racional do recurso computacional.

Após detectar uma colisão entre dois objetos, torna-se necessário resolver o evento, atualizando a posição da simulação. Em simulações que não requerem realismo, apenas a correção das posições dos objetos é suficiente. Já em simulações mais avançadas, outros métodos podem ser necessários, como a resolução de colisão por impulso, discutida em [Bourg and Bywalec 2013] e [Baraff 1992], onde as características físicas dos objetos são considerados para que a reação seja a mais adequada possível.

### 3. Materiais e Métodos

O objetivo do projeto é desenvolver e implementar um programa para simulação do movimento de corpos rígidos em determinado espaço ou ambiente. A aplicação deverá ter uma interface para configuração do ambiente e dos objetos.

No ambiente, é preciso configurar gravidade e dimensões. Para os corpos rígidos, é preciso configurar restituição, posição, velocidade, orientação, forma e/ou outras propriedades, dependendo do nível de detalhamento que se deseja. A aplicação deve permitir que os objetos sejam adicionados/removidos no ambiente.

A modelagem do simulador é apresentada, de forma simplificada, na listagem de classes a seguir:

- Mundo: Representa o ambiente virtual onde a simulação ocorre, sendo o núcleo central de todo o sistema;
- Corpo Rígido: Responsável por representar objetos no ambiente, podendo ser objetos dinâmicos ou estáticos;
- Pré-filtragem: No início de cada quadro, essa classe realiza uma filtragem em uma lista de objetos, preparando-os para a simulação de movimento;
- Integração Numérica: Responsável por resolver equações de movimento para cada objeto previamente preparado pela classe de pré-filtragem;
- Fase Ampla: Responsável pela verificação em uma lista com pares de objetos cujas caixas delimitadoras (AABBs) estão se sobrepondo, o que pode indicar uma possível colisão entre os objetos;
- Fase Estreita: Aplica testes específicos de interseção entre geometrias, para confirmar ou descartar colisões entre os objetos identificados previamente na fase ampla;
- Resolvedor de Colisões: Responsável por separar objetos que estejam em sobreposição, trabalhando com o conjunto de objetos retornados pela fase estreita;
- Despacho de Eventos: Responsável pelo envio de notificações ao usuário, informando sobre os eventos ocorridos no quadro atual da simulação.

### 4. Resultados Obtidos

O principal resultado é a aplicação Proto3d, que executa simulações de movimento e colisão de corpos rígidos em duas ou três dimensões, dentro do ambiente configurado pelo usuário.

A simulação do movimento é implementada na dinâmica newtoniana, enquanto a detecção de colisão é implementada em uma estrutura de dados *Octree*, com algoritmos *Broad Phase Narrow Phase* para previsão e identificação, respectivamente.

Além disso, o Proto3d é uma ferramenta acessível às pessoas com deficiência visual; desta forma, os resultados das simulações são apresentados no Registro Acessível, um log compatível com os leitores de tela disponíveis no mercado.

#### 4.1. Simulador Proto3d

O Proto3d é um simulador de movimento e colisão de corpos rígidos construído em C++. Para executar uma simulação, a ferramenta permite dimensionar e personalizar um ambiente (mundo), bem como inserir e personalizar corpos rígidos estáticos ou dinâmicos. Os seguintes parâmetros são configuráveis:

- **Mundo:** gravidade, limites físicos do ambiente e passo de tempo;
- **Corpos rígidos estáticos:** forma, posição e orientação;
- **Corpos rígidos dinâmicos:** forma, massa, posição, velocidade e coeficiente de restituição<sup>1</sup>;

Os seguintes padrões foram adotados na implementação do Proto3d: 1) Utiliza o Sistema Cartesiano Tridimensional [Thibaut Tadeo et al. 2004], onde X, Y e Z representam, respectivamente, largura, profundidade e altura. 2) O ponto de origem do mundo (X=0,Y=0,Z=0) é localizado no canto inferior esquerdo. 3) A localização e o tamanho dos corpos rígidos é definida por posição (coordenada onde está o ponto central do objeto no mundo) e espaço que ocupa (definido pelo *Axis Aligned Bounding Box*, ou AABB). 4) A entrada de dados ocorre por meio de um arquivo no formato JSON, preparado previamente. O arquivo contém chaves e valores para setup do ambiente e uma lista de objetos a serem adicionados.

O AABB pode ser compreendido como uma estrutura que encapsula o objeto simulado, sendo especificado por 3 segmentos de reta, um em cada eixo. Cada segmento tem uma posição inicial e uma final.

A execução da simulação é apresentada em formato de texto no Registro Acessível, sendo que esta versão do Proto3d não dispõe de interface gráfica. Para validação das simulações, foram executados alguns testes com grandezas conhecidas.

Para este estudo, optou-se em restringir o formato dos objetos em esferas e caixas, apresentando os resultados da simulação para um destes testes.

## 4.2. Registro Acessível

O Proto3d apresenta os resultados de cada simulação no Registro Acessível, em formato de log de texto e também armazena em um arquivo para análise posterior. O registro é organizado em duas partes principais: *Visão Geral* e *Rastreamento do Movimento*.

*Visão Geral:* Apresenta as configurações gerais do ambiente simulado, como gravidade, especificação dos limites do mundo e a lista de objetos simulados. Por convenção, as grandezas na simulação são lidas como: dimensão em metros ( $m$ ), massa em quilogramas ( $kg$ ), tempo em segundos ( $s$ ), velocidade em metros/segundo ( $m/s$ ) e força de movimento em metros/segundo<sup>2</sup> ( $m/s^2$ ).

*Rastreamento do Movimento:* Apresenta informações das iterações do simulador a cada passo de tempo. Os registros possuem informações conforme a legenda:

- T: Tempo decorrido desde o início da simulação até este registro;
- Dir: Trajetória do objeto dinâmico conforme a legenda: s (direção sul), n (direção norte), o (direção oeste), l (direção leste), c (direção acima/para cima), b (direção abaixo/para baixo). É possível ocorrer combinações entre as trajetórias básicas.
- Pos: Posição do objeto em um determinado instante (em metros) nos eixos X (horizontal), Y (profundidade) e Z (vertical);
- Vel: Velocidade do objeto neste instante (m/s).

Nesta versão do protótipo é possível realizar o rastreamento de apenas um objeto dinâmico.

<sup>1</sup>Valor que varia de 0 a 1, especificando o quanto de energia cinética o objeto conserva após uma colisão.

### 4.3. Testando uma simulação

O funcionamento do Proto3d é demonstrado com a simulação de um objeto dinâmico que, atraído pela gravidade, cai e colide com um objeto estático. Depois desta colisão, sob efeito de uma nova força horizontal, o objeto dinâmico é movido novamente. As etapas da simulação para o teste são detalhadas na sequência.

O Quadro 1 mostra o Registro Acessível<sup>2</sup> de inicialização do ambiente, para o teste proposto.

O registro inicia mostrando Tempo atual=0,00s e Último passo de tempo=0,00s, indicando que a simulação ainda não começou. Na linha 2 são mostrados os parâmetros Gravidade=True e Gravidade: 0,00 : 0,00 : -9,81, indicando que a gravidade está ativada, com força  $9,81m/s^2$  apenas no eixo Z (o valor é negativo para que a força "atraia" os objetos para baixo). Por fim, na linha 3 são mostrados os limites do ambiente, sendo os 3 eixos iniciados em 0 (origem) vão até 1000000m.

O chão do mundo (piso) é definido a seguir. A linha 5 informa que este elemento é um objeto em forma de caixa e seu ponto central encontra-se na posição  $X = 500000, Y = 500000, Z = -0,5$ . Pode-se observar que o ponto em Z está negativo, ou seja, está um pouco abaixo da origem.

A linha 6 mostra o espaço ocupado pelo chão com  $X = 0,00 : 1000000$  e  $Y = 0,00 : 1000000$ , indicando que esta forma geométrica ocupa toda a largura e comprimento do mundo. Já a altura do chão é definida por  $Z = -1,00 : 0,00$ , indicando que a altura do mesmo é de 1m, mas ocupa o espaço abaixo do mundo.

**Quadro 1. Registro Acessível: Inicialização do ambiente**

```
1 Tempo atual: 0.00s; Último passo de tempo: 0.00;
2 Gravidade: True; Gravidade: 0.00:0.00:-9.81
3 Limites: 0.00:1000000.00,0.00:1000000.00,0.00:1000000.00
4
5 Nome: Chão; Forma Geom: Caixa; Pos: 500000.00:500000.00:-0.50
6 AABB -> X: 0.00:1000000.00; Y: 0.00:1000000.00; Z: -1.00:0.00
7
8 Nome: Caixa estática; Forma Geom: Caixa; Pos: 35.00:35.00:5.00
9 AABB -> X: 10.00:60.00; Y: 10.00:60.00; Z: 0.00:10.00;
10
11 Nome: Esfera dinâmica; Forma Geo: esfera; Pos: 10.00:10.00:50.00
12 Raio: 2.50; Masss: 5.00; Restituição: 0.25
13 AABB -> X: 7.50:12.50; Y: 7.50:12.50; Z: 47.50:52.50
```

Nas linhas 8 e 9 encontram-se as especificações de um objeto, denominado caixa estática, de forma cúbica, com lado de 10m, que está repousada sobre o chão.

A linha 8 informa que a posição do ponto central deste objeto está na coordenada  $X = 35, Y = 35, Z = 5$ . Na linha 9, estão os valores do AABB, que demarcam os limites da caixa, com as seguintes dimensões:  $X = 10,00 : 60,00$  (segmento de 10 a 60), indicando largura de 50m;  $Z = 10,00 : 60,00$  (segmento de 10 a 60), indicando profundidade de 50m;  $Y = 0,00 : 10,00$  (segmento de 10 a 60), indicando altura de 10m.

<sup>2</sup>Para fins de adequação ao formato de artigo, algumas linhas do Registro Acessível original são omitidas nos quadros apresentados

Nas linhas 11 à 14 estão as especificações de um objeto dinâmico, denominado esfera dinâmica, de forma esférica, situado a  $50m$  de altura a partir do chão.

A linha 11 informa que o ponto central da esfera está na posição  $10,00 : 10,00 : 50,00$ , indicando que o ponto central da esfera está em  $10m$  da origem nos eixos X e Y e  $50m$  da origem no eixo Z. A linha 10 informa que a esfera possui raio= $2,5m$ , massa= $5kg$  e coeficiente de restituição= $0,25$ . Na linha 13 estão os valores do AABB, indicando os limites deste objeto, com as seguintes dimensões:  $X = 7.50 : 12.50$ ;  $Y = 7.50 : 12.50$ ;  $Z = 47.50 : 52.50$ , indicando que o diâmetro da esfera é de  $5m$  em todos os eixos.

### 4.3.1. Simulação de queda da esfera

A simulação consiste em deixar a esfera cair, atraída pela força da gravidade, até colidir com o objeto caixa estática. Entre a borda da esfera e a borda da caixa, a queda simulada é de  $37,5m$ , e, segundo a física clássica, com uma força  $G=9,81m/s^2$  levaria cerca de  $2,76s$  para ocorrer a colisão. No momento em que os objetos colidem, a velocidade estimada da esfera é de  $27,12m/s$ .

Com a execução da simulação planejada no Quadro 1, a esfera é atraída para baixo pela gravidade, até que eventualmente colida com a caixa que está sobre o chão. Quando a colisão ocorre, a esfera recebe um impulso com sentido contrário à queda, que faz o objeto quicar algumas vezes.

Este trecho da simulação pode ser verificado no Quadro 2, onde as coordenadas apresentadas representam o centro da esfera. Destaca-se que apenas alguns instantes da simulação são mostrados no quadro. O detalhamento dos eventos apresentados no Registro Acessível encontra-se a seguir:

**Quadro 2. Registro Acessível: Queda da esfera.**

1	T: 0.20s, Dir: b, Pos: 10.00:10.00:49.61, Vel: 1.96m/s
2	T: 2.40s, Dir: b, Pos: 10.00:10.00:19.39, Vel: 23.54m/s
3	T: 2.60s, Dir: b, Pos: 10.00:10.00:14.29, Vel: 25.51m/s
4	T: 2.80s, Dir: c, Pos: 10.00:10.00:12.50, Vel: 6.87m/s
5	T: 3.60S, Dir: c, Pos: 10.00:10.00:15.64, Vel: 0.98m/s
6	T: 3.80S, Dir: b, Pos: 10.00:10.00:15.25, Vel: 1.96m/s
7	T: 4.20S, Dir: b, Pos: 10.00:10.00:13.28, Vel: 5.89m/s
8	T: 4.40S, Dir: c, Pos: 10.00:10.00:12.50, Vel: 1.96m/s

- $T = 0,20s$ : Gravidade começa atrair a esfera para baixo;
- $T = 2,40$  a  $2,60$ : Gravidade continua atraindo a esfera, aumentando sua velocidade;
- $T = 2,60s$ : Esfera caiu  $35,32m$  em relação a posição inicial e agora tem velocidade de  $25,51m/s$ . A colisão acontece em  $T = 2,76s$  (sem registro no log);
- $T = 2,80s$ : Esfera já colidiu com a caixa, recebendo um impulso de força contrária à queda, fazendo-a mudar de direção. Agora a esfera está subindo (na altura  $12,50m$ ) a uma velocidade de  $6,87m/s$ ;
- $T = 3,60s$ : Esfera atinge o pico do primeiro salto com sua velocidade praticamente anulada pela gravidade;
- $T = 3,80s$ : Esfera começa a cair novamente após a gravidade conseguir anular a velocidade;

- $T = 4,20s$ : Esfera está prestes a colidir com a caixa pela segunda vez;
- $T = 4,40s$ : Esfera colide com a caixa e salta novamente, mas agora seu pico será em uma altura menor que a anterior;

No instante  $4,40s$  a esfera está parando de quicar, quando uma força lateral de  $35m/s^2$  é aplicada, causando um deslocamento para leste (direita da origem). A esfera será empurrada até chegar ao limite da caixa. Este trecho da simulação pode ser verificado no Quadro 3.

**Quadro 3. Registro Acessível: Esfera sobre a caixa.**

1	T: 4.40s, Dir: c, Pos: 10.00:10.00:12.50, Vel: 1.96m/s
2	T: 4.60s, Dir: lc, Pos: 11.40:10.00:12.89, Vel: 7.27m/s
3	T: 4.80s, Dir: l, Pos: 14.20:10.00:12.89, Vel: 14.00m/s
4	T: 5.60s, Dir: l, Pos: 39.40:10.00:12.50, Vel: 42.00m/s

- $T = 4,40s$ : Instante que a força lateral é aplicada;
- $T = 4,60s$ : Esfera começa a mover-se também para leste. A esfera ainda está "quicando", mas em uma distância muito pequena da caixa (apenas  $39cm$ ). As duas forças causam um movimento na direção lc, para leste e acima, descrevendo uma pequena parábola;
- $T = 4,80s$ : Esfera estabilizou verticalmente sobre a caixa e se movimenta apenas para leste, com velocidade  $14m/s$ , devido a força aplicada desde o instante  $4,40s$ ;
- $T = 5,60s$ : A esfera continua acelerando na direção leste em cima do cubo, agora na posição  $39,40m$  com  $42m/s$ .

A esfera continua sendo "empurrada" para leste, até o momento em que "sai de cima da caixa", iniciando um movimento diagonal, para leste e abaixo. A força lateral ainda está agindo sobre o objeto. Este trecho da simulação pode ser verificado no Quadro 4.

**Quadro 4. Registro Acessível: Esfera se movendo diagonalmente**

1	T: 6.00s, Dir: l Pos: 60.40:10.00:12.50, Vel: 56.00m/s
2	T: 6.20s, Dir: lb Pos: 73.00:10.00:12.11, Vel: 63.03m/s
3	T: 7.40s, Dir: c Pos: 178.00:10.00:2.50, Vel: 105.06m/s
4	T: 8.00s, Dir: lb Pos: 249.40:10.00:3.38, Vel: 126.00m/s
5	T: 8.60s, Dir: l, Pos: 333.40:10.00:2.50, Vel: 147.00m/s
6	T:243.20s, Dir: l, Pos:998791.00:10.00:2.50, Vel: 8358.00m/s
7	T:244.20s, Dir: lb, Pos:1007170.00:10.00:-1.42, Vel: 8393.00m/s

- $T = 6,00s$ : Esfera continua sua trajetória na direção leste e agora seu centro ultrapassou a borda da caixa;
- $T = 6,20s$ : Gravidade começa a agir novamente juntamente com a força lateral. Esfera descreve um movimento diagonal, na direção lb (leste e abaixo), com velocidade de  $63,03m/s$ ;
- $T = 7,40s$ : Esfera atingiu o chão pela primeira vez quicando e subindo;
- $T = 8,00s$ : Depois de atingir sua altura máxima, em função de ter quicado, a esfera volta a cair descrevendo um movimento de parábola. A direção agora é leste e baixo, com velocidade de  $126m/s$ ;
- $T = 8,60s$ : Força restituída da queda termina e a esfera para de quicar, movimentando-se para leste, com velocidade crescente até a "borda do mundo".

- $T = 243, 20s$ : Esfera está com velocidade de  $8358m/s$  na direção leste, próxima ao limite do mundo simulado;
- $T = 244, 20s$ : Esfera ultrapassa os limites do mundo onde começa a cair novamente, descrevendo uma trajetória diagonal.;

Fora do mundo simulado não existem mais objetos que possam causar colisão, então a esfera continuará caindo, no sentido leste, abaixo (lb) até que a simulação finalize ou novas forças sejam aplicadas a esfera.

## 5. Considerações finais

O Proto3d é uma aplicação desenvolvida com a finalidade de simular o movimento de um ou mais corpos sólidos, detectar e resolver as colisões entre estes corpos e, ainda, apresentar a simulação em uma saída acessível a pessoas com deficiência visual. O protótipo implementado permite configurar vários parâmetros da simulação.

A simulação executada pelo Proto3d atingiu os objetivos estabelecidos. Os resultados de movimento mostrados no Registro Acessível podem ser validados com cálculos de física newtoniana. As colisões são detectadas e o movimento resultante também têm resultados coerentes com os esperados.

O Proto3d poderia ser utilizado em diversas aplicações, com destaque para tecnologia educacional acessível, dedicada ao estudo de física, simulação computacional, representação gráfica de movimentos, entre outras.

Como trabalhos futuros para o desenvolvimento do Proto3d, planeja-se implementar o rastreamento de múltiplos objetos, criar uma interface gráfica para tornar o simulador mais acessível, aprimorar o método de integração usado para calcular os movimentos dos objetos e também considerar outras propriedades físicas, como velocidade angular. Além disso, pretende-se introduzir no ambiente simulado um parâmetro para resistência do ar ou atrito.

## Referências

- Baraff, D. (1992). *Dynamic simulation of nonpenetrating rigid bodies*. Cornell University.
- Bourg, D. M. and Bywalec, B. (2013). *Physics for Game Developers: Science, math, and code for realistic effects*. "O'Reilly Media, Inc."
- Durán, J. M. (2018). *Computer simulations in science and engineering: Concepts-Practices-Perspectives*. Springer.
- Ericson, C. (2004). *Real-time collision detection*. Crc Press.
- Foley, J. D. (1996). *Computer graphics: principles and practice*, volume 12110. Addison-Wesley Professional.
- Marschner, S. and Shirley, P. (2018). *Fundamentals of computer graphics*. CRC Press.
- Pharr, M., Jakob, W., and Humphreys, G. (2016). *Physically based rendering: From theory to implementation*. Morgan Kaufmann.
- Thibaut Tadeo, E. et al. (2004). Proyecto cube: una introducción a la geometría tridimensional. *Suma*.