

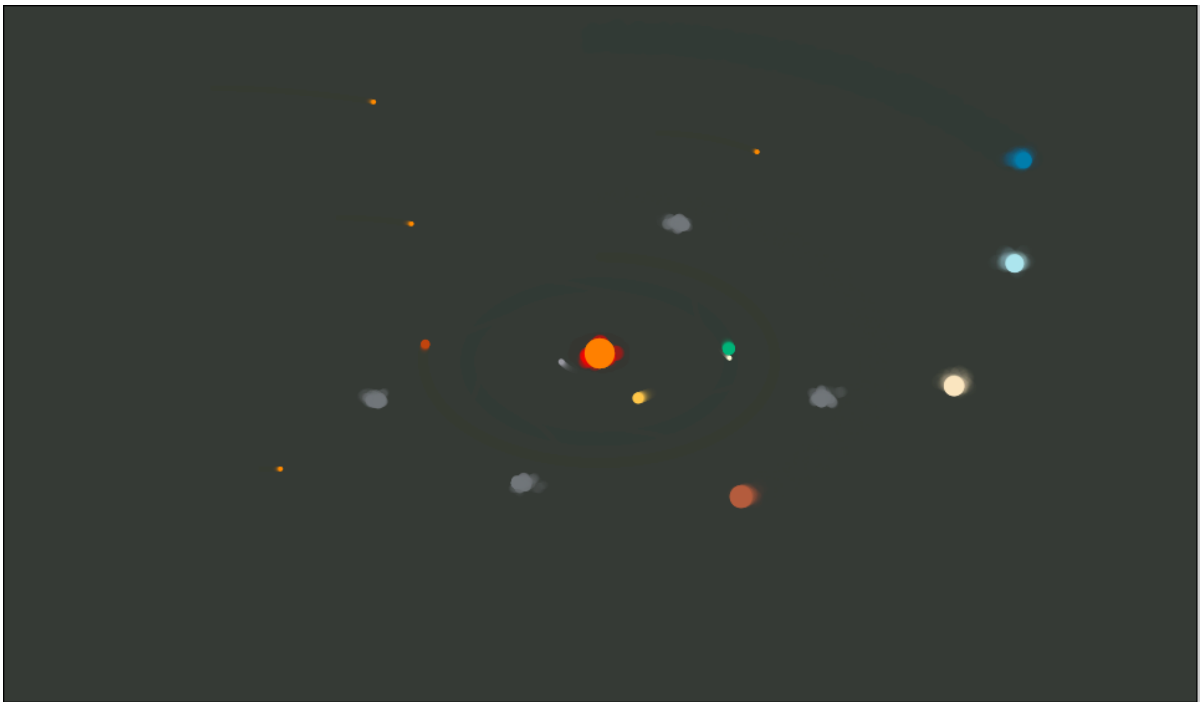


Instituto Superior de Engenharia de Lisboa

Licenciatura em Engenharia Informática e Multimédia

Modelação e Simulação de Sistemas Naturais - MSSN -
2223SI

Projeto 2



Docente Paulo Vieira

Realizado por :
Pedro Silva 48965

20 de novembro de 2022

Conteúdo

1	Introdução	I
2	Queda Livre	I
3	Sistema Solar & Sistemas de Partículas	IV
4	Agentes Autónomos - Comportamentos Individuais	V
5	Agentes Autónomos - Comportamentos de Grupo (Flocking)	VII
6	Conclusões	IX

Lista de Figuras

1	Queda Livre alínea A (Silico)	I
2	Queda Livre alínea B (Silico)	II
3	Queda Livre alínea C (Silico)	III
4	Queda Livre alínea D (Silico)	III
5	Sistema Solar	IV
6	Agentes Autónomos - Comportamentos Individuais	V
7	Função de travagem	VI
8	Agentes Autónomos - Comportamentos de Grupo (Flocking)	VII
9	Verificação de morte do Boid	VIII
10	Verificação de morte do Boid	VIII
11	Período de explosão	VIII

1 Introdução

Este projeto pretende consolidar e avaliar os conhecimentos adquiridos na segunda parte da disciplina Modelação e Simulação de Sistemas Naturais tendo como objetivo o desenvolvimento de uma Queda livre com atrito em Silico, o Sistema Solar & Sistemas de Partículas, comportamentos individuais de Agentes Autónomos e Comportamentos de grupo (Flocking), tendo como base as classes desenvolvidas em aula e no vídeos disponibilizados pelos docentes. Na queda livre com atrito temos várias situações para observar como a abertura de um paraquedas e a queda do paraquedista em água. O objetivo do Sistema Solar era colocar N planetas a girarem em torno do sol e usar sistemas de partículas para representar os vários corpos celestes. Nos comportamentos individuais dos Agentes Autónomos foi pedido a criação de três comportamentos: o Comportamento Arrive, Patrol e Wander e a criação de um Boid que faz "seek" a um alvo mas podemos alterar a sua velocidade assim como travar-lo. Por fim nos comportamentos de grupo dos Agentes Autónomos foi criado um enunciado de um jogo que implementa os tópicos: Predador a seguir uma presa, Boids explosivos, Flock com liderança e Debugging.

2 Queda Livre

1. Pretende-se modelar a queda, na vertical, de um corpo, que para além do seu peso sofre a ação de uma força adicional, de atrito, com sentido contrário à sua velocidade. As variáveis de estado (Stocks) são a posição, $x(100m)$, a velocidade do corpo, $v(0m/s)$, aceleração, a . A força total que o corpo sofre é igual à soma da força do seu peso, $\text{Peso} = \text{Massa}(80kg) * \text{Aceleração gravítica}(9.8)$, com a força de atrito, $0.5 * \text{Área de contacto}(0.5m) \text{ do corpo com o meio} * \text{Densidade do meio}(1.29) \text{ onde o objeto se move} * \text{Módulo da velocidade ao quadrado}$.

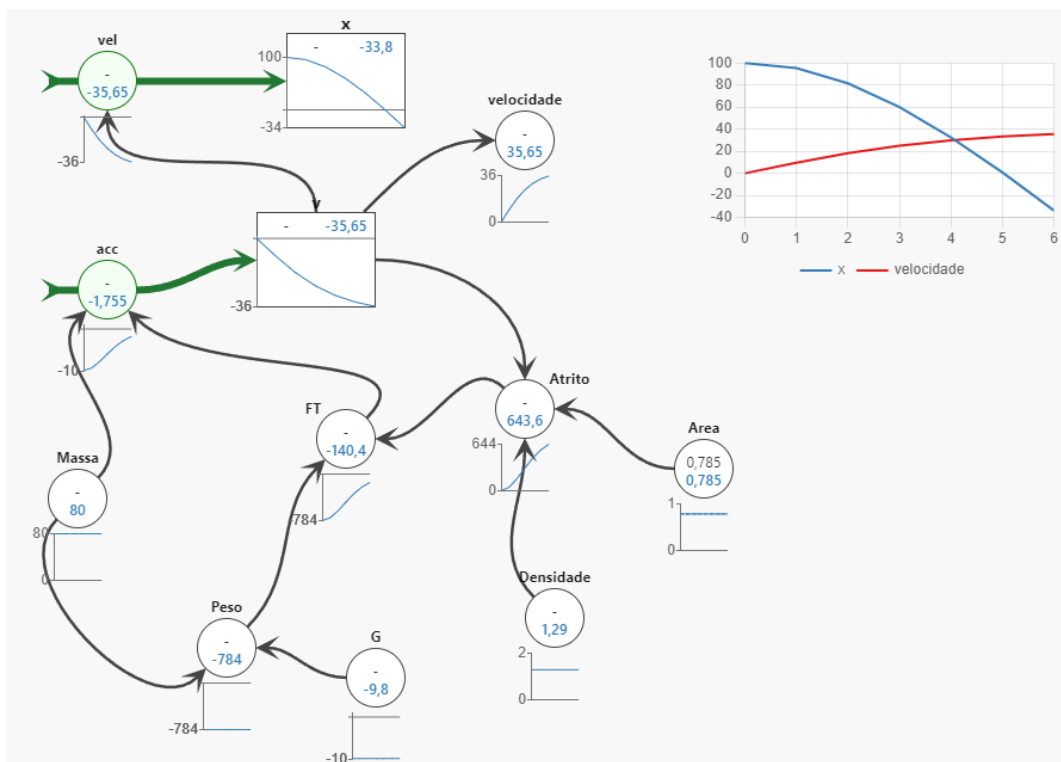


Figura 1: Queda Livre alínea A (Silico)

Ao observar o diagrama de Stocks&Flows na Figura 1 chegamos à conclusão que existe um ciclo de feedback negativo, pois a velocidade afeta a força de atrito, este afeta a força total que afeta a aceleração e por fim afeta a velocidade, o que começa o ciclo novamente. A natureza deste ciclo é negativa pois todas estas variáveis acabam por se estabilizar devido a este ciclo.

2. Temos em baixo na Figura 2 o mesmo sistema mas a massa do corpo é 70kg e a sua área de contacto tem um raio de 0.4m.

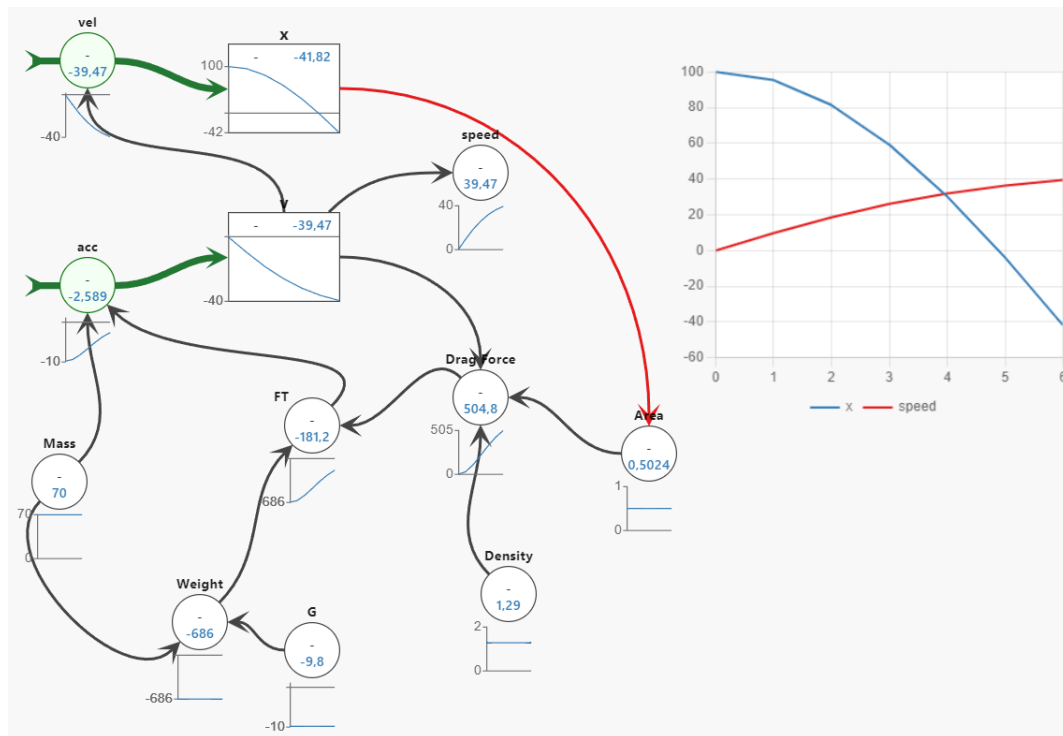


Figura 2: Queda Livre alínea B (Silico)

Podemos observar que o corpo demora 4.9 segundos a aterrar e a velocidade que este embate no solo é 37m/s. Comparando com a figura 1 concluimos que à medida que o peso baixa a velocidade aumenta e diminuindo a área de contacto a força de atrito também diminui.

3. Temos em baixo na Figura 3 o sistema da Alínea A mas desta vez um paraquedas é aberto quando o corpo estiver a 30m do solo e esse paraquedas tem as seguintes características: massa igual a 10kg e diâmetro igual a 6m.

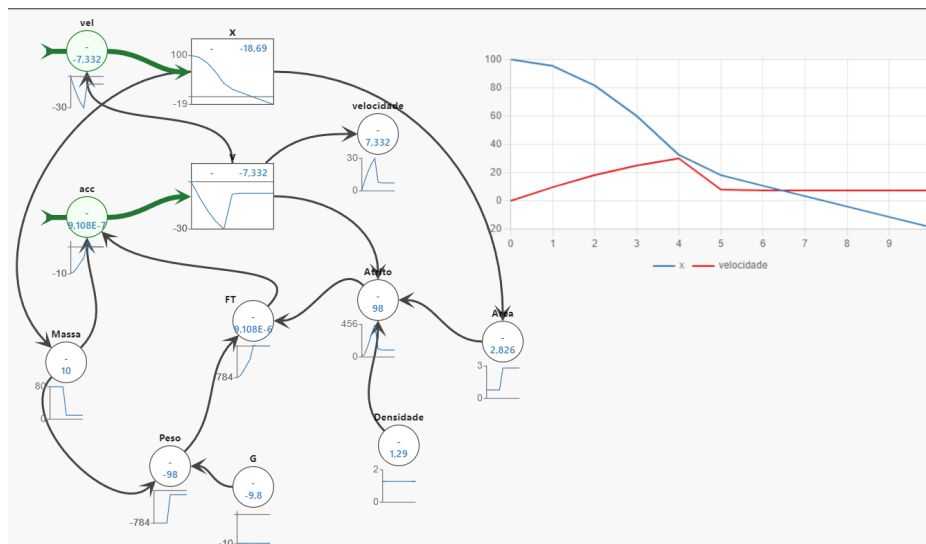


Figura 3: Queda Livre alínea C (Silico)

Podemos observar que o corpo demora 7.5 segundos a aterrar e a velocidade que este embate no solo é 7.33m/s. Conseguimos observar que o momento em que o paraquedas foi aberto foi aos 4 segundos pois nesse momento a velocidade começou a decrescer, o contrário de antes. Depois deste momento a posição do corpo diminui de uma forma mais suave comparando com antes do paraquedas ser aberto. A velocidade vai diminuindo até estabilizar nos 7,33m/s. Antes da abertura do paraquedas o paraquedista aproxima-se do solo a uma velocidade crescente em que a sua distancia percorrida é acentuada. Na abertura do paraquedas o paraquedista sente um efeito de chicote pois a velocidade muda de a aumentar para a diminuir e a partir daí desloca-se a um ritmo mais agradável.

4. Temos em baixo na Figura 4 o sistema da Alínea A mas desta vez o corpo cai no rio Tejo. Consideramos que a densidade da água é de 100Kg/m³.

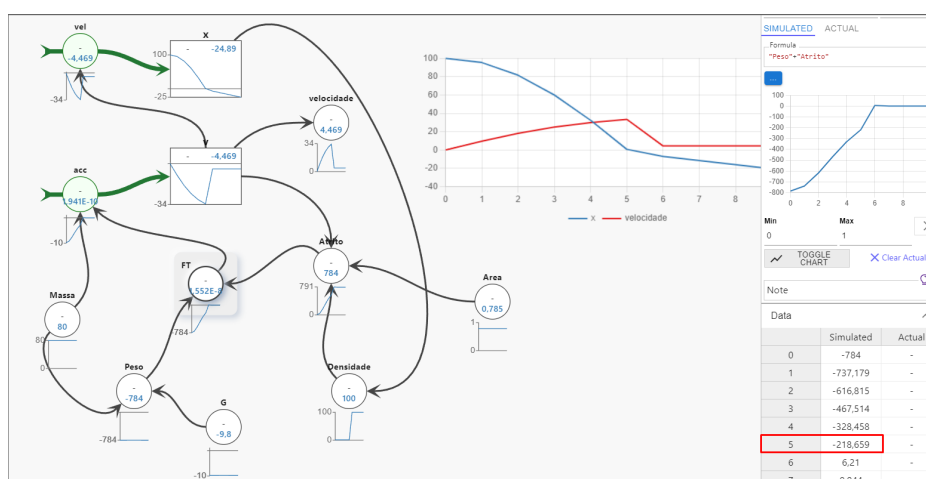


Figura 4: Queda Livre alínea D (Silico)

Podemos observar que a intensidade da força que o paraquedista sofre no momento do embate em água é 218.659N. A velocidade a que ele submerge no rio começa em 33.41m/s e vai diminuindo até estabilizar em 4.47m/s.

3 Sistema Solar & Sistemas de Partículas

Na pergunta C foi pedido para simular o movimento de um planeta em torno de uma estrela, ou seja, a massa da estrela é muitíssimo maior que a do planeta, por isso começamos por introduzir o planeta Terra e a estrela Sol. A segunda alínea consiste em adicionar N planetas a girarem em torno do Sol sendo possível alterar os parâmetros do sistema. Por fim foi pedido o uso de sistemas de partículas para representar os vários corpos celestes.

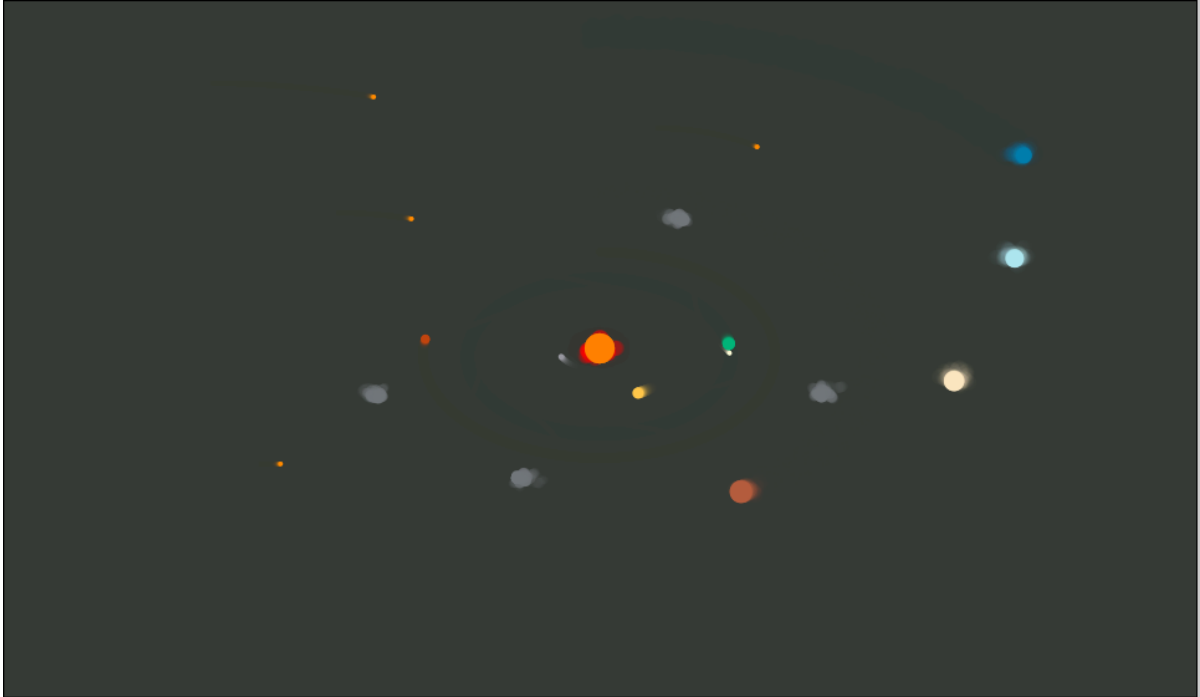


Figura 5: Sistema Solar

De forma a obter-mos um comportamento mais visualmente interessante tivemos que alterar o raio dos planetas assim como a sua distância ao Sol, pois a diferença de dimensões entre planetas era demasiado grande. Utilizámos como referência o raio de Júpiter, que neste caso era 20% menor que o do Sol, e escalámos os restantes planetas a partir dele. Tendo em conta a distância utilizámos a distância de mercúrio ao Sol como referência. Foi adicionado à Terra a sua Lua que orbita à sua volta. Para isto tivemos que aumentar e diminuir, respetivamente o tamanho real da Terra e da Lua pois se usássemos o seus tamanhos reais a Lua não orbitaria em torno da Terra pois a Força gravítica do Sol é muito grande. Foi adicionado a cintura de asteroides constituída por 4 sistemas de partículas que vão ter a mesma posição de 4 "bodys" em que não chamamos o display. Ao Sol e aos planetas gasosos foi adicionado um sistema de partículas para representar a sua natureza. Por fim foi adicionado a possibilidade de criar cometas ao clicar com o rato na posição desejada. Estes são atraídos pelo Sol e por Júpiter pois estes dois são os corpos com maiores dimensões, logo maior força gravítica.

4 Agentes Autónomos - Comportamentos Individuais

Nos comportamentos individuais dos Agentes Autónomos foi pedido a criação de um Boid com acelerador e travão em que este vai fazer seek a um alvo controlado pelo utilizador. O comportamento Arrive é igual ao Seek mas neste caso o Boid ao aproximar-se do alvo reduz gradualmente a sua velocidade. O comportamento Patrol faz com que o Boid percorra a sua lista de coordenadas passadas no construtor pela ordem em que as recebe. O comportamento Wander consiste em o agente vaguear pelo mundo sem objetivo definido.

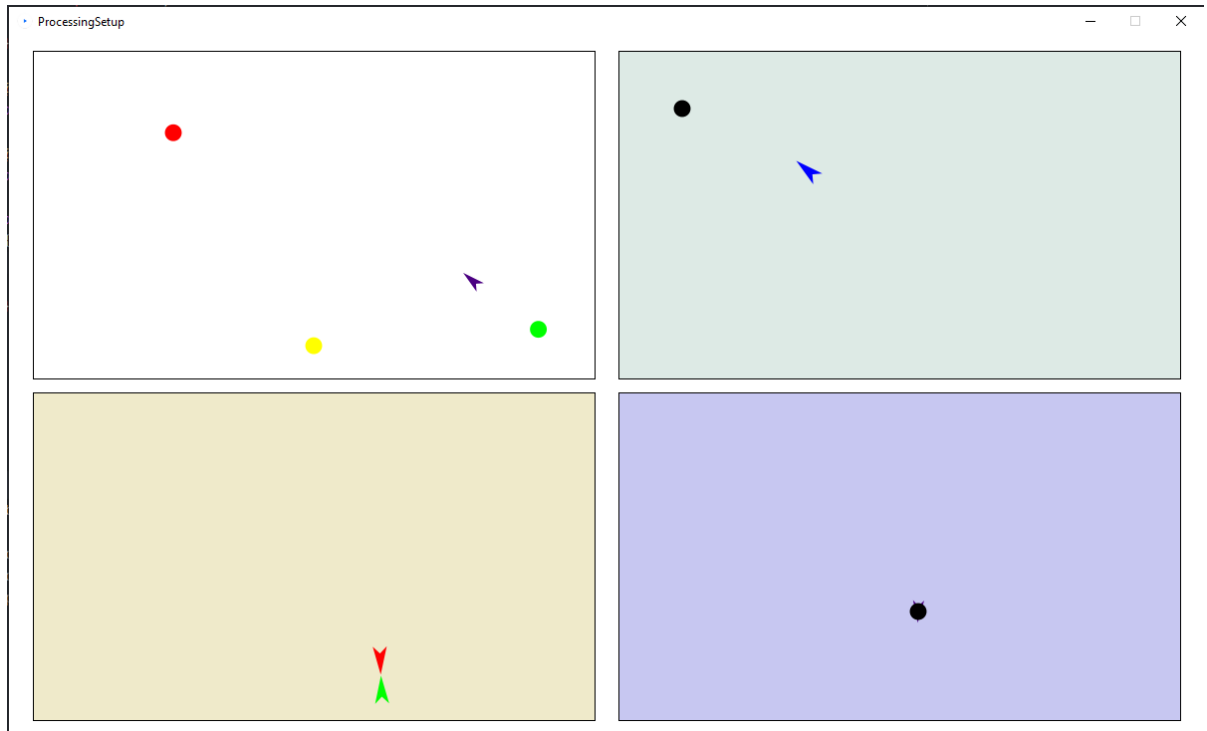


Figura 6: Agentes Autónomos - Comportamentos Individuais

- Na primeira janela temos o Comportamento Patrol em que podemos observar 3 bolas de cores diferentes, em que a vermelha é a posição da primeira coordenada que o Boid patrulha e a verde é a última. Também podemos observar a ciclicidade deste comportamento pois vemos que o agente parte da última coordenada de novo para a primeira.

- Na segunda janela temos o Boid com acelerador e travão onde podemos observar o Boid a perseguir o círculo preto controlado pelo utilizador. Se clicarmos na tecla 'W' e na tecla 'S' aumentamos e diminuímos, respetivamente, a sua velocidade máxima por 0,5. Também é possível travar este agente ao pressionar a barra de espaços. A diferença sente-se mais na redução da velocidade máxima, pois mesmo se aumentarmos significativamente a sua velocidade máxima para o agente atingir essa velocidade ele precisa de se deslocar em retas mas, se tivermos consistentemente a mudar a posição do alvo isso não acontece. Depois de alguns testes foi escolhido o valor 0,5 pois permite um melhor controlo da velocidade do agente.
- Na terceira janela temos o Comportamento Wander, Boid cor verde, e o Comportamento pursuit, Boid cor vermelha, em que o verde passeia livremente pelo mundo enquanto o vermelho o persegue.
- Na última janela temos o Comportamento Arrive em que o agente persegue, abrando à medida que se aproxima, do alvo controlado pelo clique do rato.

$$\begin{aligned} \|\vec{v}_d\| &= v_{max} & \text{se } d \geq R \\ \|\vec{v}_d\| &= v_{max} \left(\frac{d}{R}\right)^k & \text{se } d < R \end{aligned}$$

Figura 7: Função de travagem

Foram testados vários valores para k (ver Figura 7) sendo o valor com melhores resultados 30 pois com menor que este a diminuição da velocidade não é grande o suficiente para parar o agente quando este está em cima do alvo e com valores maiores que este a diminuição da velocidade é demasiado acentuada fazendo com que o Boid demore demasiado a chegar ao alvo.

5 Agentes Autónomos - Comportamentos de Grupo (Flocking)

Na pergunta E foi pedido a implementação dos conceitos predador a perseguir uma presa, Boids explosivos, Flock com liderança e Debugging. Para realizar todos estes conceitos numa só aplicação foi criado um enunciado de jogo. O objetivo do jogo é, controlando um Boid de cor verde utilizando o rato, manter vivo durante o maior tempo possível pelo menos um dos teus seguidores. Tens dois tipos de perigos diferentes as zonas e o predador. Existem 4 zonas de perigo (círculos vermelhos) em que se os teus seguidores tocarem explodem e são eliminados. O outro perigo é o predador, um Boid vermelho que vai perseguir cada um dos teus seguidores até não restar nenhum.

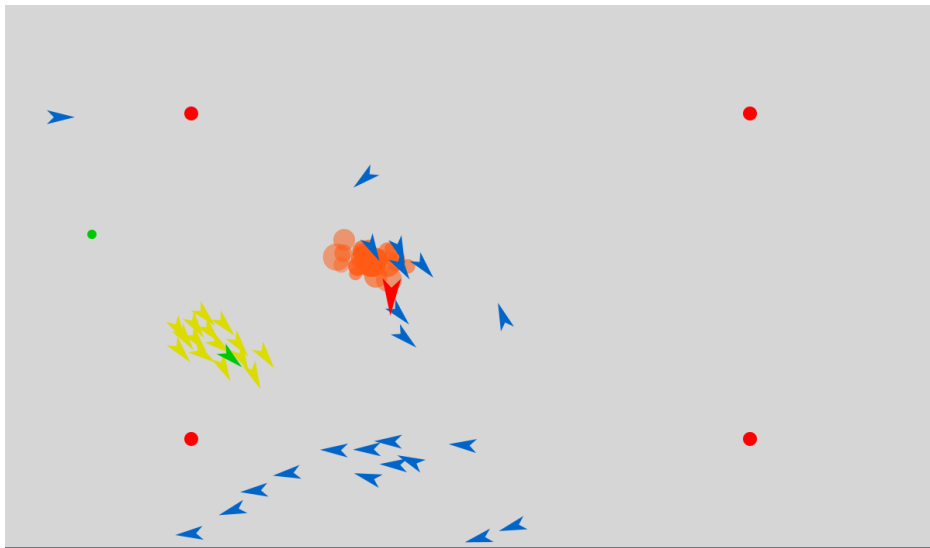


Figura 8: Agentes Autónomos - Comportamentos de Grupo (Flocking)

Na Figura 8 podemos observar um bando de seguidores (boids de cor amarela) a seguir o líder (agente verde) controlado pelo rato que está na posição do círculo verde. Em baixo conseguimos observar um grupo de boids em "flock" a passar por baixo de uma das zonas de perigo. Também podemos ver o predador atrás de uma das suas presas e atrás deles uma das suas vítimas na explosão (partículas laranja). Também podemos acionar o debugger ao clicar na tecla "D" que ativa o display do olho em que é exibido o campo de visão e a distância de segurança do Líder .

```
//Percorre os boids no flock
for (int i = 0; i < flock.getBoids(); i++) {
    flock.getBoid(i).eye.look();
    allTrackingBodies.add(flock.getBoid(i));
    predator.setEye(new Eye(predator, allTrackingBodies));

    //se o boid do flock vir o lider persegue-o e muda de cor
    if (flock.getBoid(i).eye.farSight(leader.getPos()) ) {
        if (flock.getBoid(i).count!=1) {
            flock.getBoid(i).addBehaviour(seek);
            flock.getBoid(i).count++;
        }
        flock.getBoid(i).setShape(p,plt, radius: 0.3f,p.color(220,220,0));
    }
    //se não remove o comportamento de perseguição
    else {
        flock.getBoid(i).removeBehaviour(seek);
        flock.getBoid(i).count=0;
        flock.getBoid(i).setShape(p,plt, radius: 0.3f,p.color(0,100,200));
    }
}
```

Figura 9: Verificação de morte do Boid

Os teus seguidores apenas te seguem se tiveres no seu raio de visão (figura 9) ficando de cor amarela se não continuam azuis e juntam-se ao maior bando de agentes que estiver no seu campo de visão.

```
//se o boid do flock tocar num ponto vermelho ou for apanhado pelo predador explode e removemos do flock
if (flock.getBoid(i).eye.nearSight(new PVector( x: -6, y: 6)) ||
    flock.getBoid(i).eye.nearSight(new PVector( x: -6, y: -6)) ||
    flock.getBoid(i).eye.nearSight(new PVector( x: 6, y: -6)) ||
    flock.getBoid(i).eye.nearSight(new PVector( x: 6, y: 6)) ||
    flock.getBoid(i).eye.nearSight(predator.getPos())) {
    int color = p.color(p.color(255,91,20));
    PSControl psc = new PSControl(velParams, lifetimeParams, radiusParams, flow, color);
    ParticleSystem ps = new ParticleSystem(new PVector(flock.getBoid(i).getPos().x,flock.getBoid(i).getPos().y),
        new PVector(), mass: 1f, radius: 1f, psc);
    explosion.add(ps);
    flock.removeBoid(i);
    System.out.println(flock.getBoids() + " Boids Restantes");
}
```

Figura 10: Verificação de morte do Boid

A eliminação dos Boids é realizada ao efetuarmos a verificação (Figura 10) de que nenhum dos perigos se encontra na sua "safe distance", se este for o caso ele explode e é eliminado, ou seja, criamos um conjunto de partículas de cor alaranjada na posição da morte e removemos-o da lista de Boids no Flock.

```
//explosao
for (int i = 0; i < explosion.size(); i++)
{
    explosion.get(i).move(dt);

    explosion.get(i).display(p, plt);
    if (explosion.get(i).timer>0.8f) {
        explosion.remove(explosion.get(i));
    }
}
```

Figura 11: Período de explosão

Para controlarmos o tempo que a explosão fica no ecrã criamos um cronómetro no "Particle System" e se este for maior que 0.8 segundos a explosão acaba (Figura 11).

6 Conclusões

O objetivo deste projeto era o de consolidar os conhecimentos sobre Física e movimento com e sem atrito, o comportamento e o controlo de movimento em corpos, os comportamentos individuais e os de grupo de agentes autónomos adquiridos ao longo de Modelação e Simulação de Sistemas Naturais. Gostava de ter tido tempo para desenvolver a pergunta G (Guerra Espacial) pois pareceu-me bastante interessante e uma excelente forma de entrelaçar todos os conhecimentos adquiridos. Porém creio que as aplicações desenvolvidas e as suas partes facultativas estão de acordo com o pretendido pelos docentes, demonstrando o domínio que tenho sobre a matéria lecionada.