

# Lab 8: Simulação Massa-Mola com Verlet

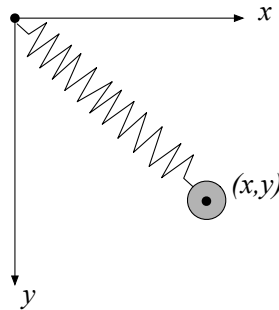
## INF1608 – Análise Numérica

Leonardo Quatrin Campagnolo

lquatrin@tecgraf.puc-rio.br

Departamento de Informática, PUC-Rio

Considere um corpo de massa  $m$ , conectado a um ponto fixo  $A$ , na posição  $(0,0)$ , por uma mola com coeficiente de rigidez  $k$  e comprimento de repouso  $r$ .



Sabe-se que, no instante  $t = 0$ , o corpo está na posição  $(x_0, y_0)$  com velocidade nula, e que as forças atuantes nesse corpo são:

- Força de gravidade (onde  $\mathbf{g}$  representa a aceleração da gravidade):

$$\mathbf{f}_g = m\mathbf{g}$$

- Força de vento, com decaimento exponencial com o tempo  $t$  (onde  $\mathbf{w}$  é a força de vento inicial):

$$\mathbf{f}_w = \mathbf{w} e^{-\frac{t}{20}}$$

- Força de mola (onde  $\mathbf{x} = (x, y)$  representa a posição do corpo):

$$\mathbf{f}_k = -k(\|\mathbf{x}\| - r) \frac{\mathbf{x}}{\|\mathbf{x}\|}$$

Considere o método de integração de Verlet com amortecimento:

$$\mathbf{x}_{i+1} = \mathbf{x}_i + (1 - \delta)(\mathbf{x}_i - \mathbf{x}_{i-1}) + h^2 \frac{\mathbf{f}_i}{m}$$

onde  $\delta$  é o coeficiente de amortecimento,  $m$  é a massa do corpo e  $\mathbf{f}_i$  é a força atuante no instante corrente  $t$ . A posição corrente é dada por  $\mathbf{x}_i$  e a posição anterior por  $\mathbf{x}_{i-1}$ . Como a velocidade

inicial é nula, considera-se  $\mathbf{x}_{i-1} = \mathbf{x}_i$  no início da simulação. O valor de  $\mathbf{x}_{i+1}$  é a posição do corpo no instante  $t + h$ .

Considerando os seguintes parâmetros:

$$\begin{aligned}\text{rigidez da mola: } k &= 0.3 \\ \text{comprimento de repouso da mola: } r &= 200 \\ \text{massa do corpo: } m &= 2.0 \\ \text{aceleração da gravidade: } \mathbf{g} &= [0.0, 9.8]^T \\ \text{força inicial de vento: } \mathbf{w} &= [10.0, 0.0]^T \\ \text{coeficiente de amortecimento: } \delta &= 0.002\end{aligned}$$

Pede-se:

1. Implemente uma função que receba como parâmetros o tempo  $t$  e a posição corrente do corpo  $(x, y)$ . A função deve calcular os valores da força total (gravidade + vento + mola) atuante e preencher os ponteiros também passados como parâmetros:

```
void forca (double t, double x, double y, double* fx, double* fy);
```

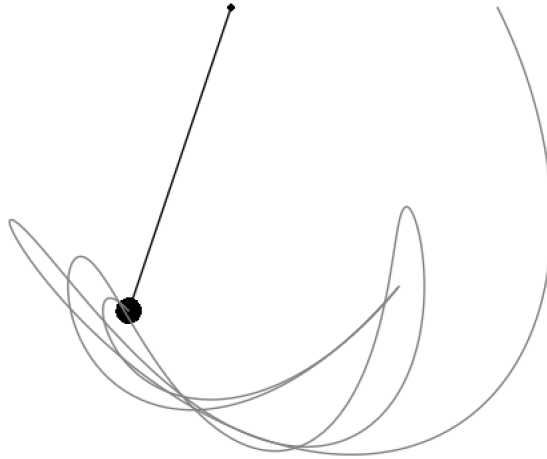
2. Usando o método de Verlet indicado acima, dada uma configuração no instante  $t$ , calcule a nova posição do corpo no instante  $t + h$ . A função deve receber como parâmetros o tempo corrente  $t$ , o passo de integração  $h$ , a posição corrente  $(x, y)$  e a posição anterior  $(x_p, y_p)$ . A função deve preencher a nova posição nos endereços de memória também passados como parâmetros e retornar o tempo atualizado  $(t + h)$ :

```
double evolui (double t, double h, double x, double y, double xp, double yp,
               double* xn, double* yn
);
```

3. Usando a função `evolui`, escreva uma função que preencha um vetor de posições do corpo ao longo do tempo. Sua função recebe como parâmetros a posição inicial  $(x_0, y_0)$ , o tempo total de simulação  $t_{final}$  e o número de iterações  $n$ , fazendo o passo de integração valer  $h = t_{final}/n$ . A função deve preencher os vetores  $x$  e  $y$ , previamente alocados, de dimensão  $n$ , com as posições do corpo nos instantes  $h, 2h, \dots, nh$ .

```
void simula (double x0, double y0, double t_final, int n, double* x, double* y);
```

Para testar sua aplicação, considerando  $h = 0.1$ , saiba que o corpo, inicialmente em repouso, com a posição inicial dada por  $(x, y) = (200, 0)$ , descreve a trajetória ilustrada na figura abaixo até o tempo  $t = 100$ . Note como a força de vento inicial arrasta o corpo para a direita.



Salve as posições  $(x, y)$  computadas e use um programa de gráficos para visualizar a trajetória. Altere os parâmetros da simulação e veja se o comportamento é o esperado.

Agrupe os protótipos das funções pedidas em um módulo “massamola.h” e as implementações em um módulo “massamola.c”. Escreva o teste em outro módulo “main.c”.

**Entrega:** O código fonte deste trabalho (isto é, os arquivos “massamola.c”, “massamola.h” e “main.c”, e *eventuais códigos de laboratórios passados usados na solução*) devem ser enviados via página da disciplina no EAD até 23:59 de hoje (24/10).