Física Computacional

Prática 7 - Questão 1

Alex Enrique Crispim

O método de Euler é o método mais simples para resolução de equações diferenciais ordinarias (EDO's). Uma extensão do método se dá pelo que é chamado *método de Euler-Cromer*, cuja diferença está, basicamente, na ordem de atualização entre a derivada primeira (a qual chamaremos v) e a função em busca ($x(t)$).

Para a EDO
$$\frac{\mathrm{d}x}{\mathrm{d}t} = -x^3 + \sin(t),$$
temos apenas o método de Euler, dos citados anteriormente, visto que é uma EDO de primeira ordem.

Seja $v_i = \frac{\mathrm{d}x}{\mathrm{d}t}(t_i)$ e $x_i = x(t_i)$, podemos utilizar o método das diferenças finitas e escrever

$$x_{i+1} = x_i + \tau v_i, \tag{1}$$
$$v_i = sin(t_i) - x_i^3. \tag{2}$$

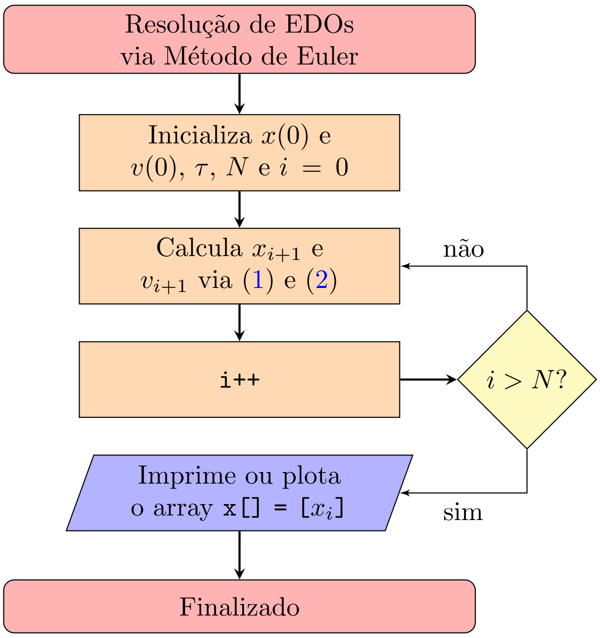
Basta então implementar o algoritmo descrito acima, fornecendo-se as condições iniciais. Com as condições iniciais, o tempo inicial e o número de pontos N que se deseja utilizar para fazer o cálculo, uma possível implementação é mostrada no código em C abaixo, onde `MAX_T` é o valor máximo para t e $t_0 \leftarrow 0.0$ [1].

```c
tau = (double)MAX_T/N;
  for (unsigned int i = 0; i < N; i++) {
  x += tau * v;
    v = -pow(x,3) + sin(i*tau);
    fprintf(fPtr, "%lf\t%lf\n", i*tau, x);
  }
```

O método de Euler (ou Euler-Cromer) tem seu erro da ordem da divisão do intervalo da variável dependente, no nosso caso, o erro é $\mathcal{O}(\tau)$.

Na página seguinte, temos um fluxograma do algoritmo para o método utilizado.

[1] Código completo em https://github.com/AlexEnrique/comp-physics-pratice7


Resolução de EDOs
via Método de Euler
Inicializa $x(0)$ e
$v(0)$, τ, N e $i = 0$
Calcula x_{i+1} e
v_{i+1} via (1) e (2)
i++
$i > N$?
não
sim
Imprime ou plota
o array x[] = [x_i]
Finalizado

The End.