Relatório - miniprojeto 6

Laís Saloum Deghaide, nUSP: 11369767

1. Introdução

Neste miniprojeto, foi nos dada a equação unidimensional de um foguete, com certas propriedades específicas, e a partir dela deveríamos calcular vários aspectos importantes do foguete, como: velocidade máxima, tempo de alcance, aceleração da trajetória, etc.

2. Objetivos

Temos como objetivo solucionar cada um dos aspectos mencionados brevemente na introdução, e melhor discutidos em cada uma das partes separadas do projeto, utilizando quatro métodos diferentes. São eles: **lsode**, **Euler Explícito**, **Runge-Kutta de ordem 2** e **Runge-Kutta de ordem 4**.

3. Desenvolvimento

Parte A:

Devemos resolver numericamente a equação do foguete, usando a função lsode. Essa equação nos retornará uma solução numérica exata para nossa EDO.

Para isso, devemos criar nosso sistema de EDO, como descrito em aula pelo professor Buscaglia:

$$y = \begin{pmatrix} y_1 \\ y_2 \end{pmatrix}, \quad \frac{dy}{dt} = \begin{pmatrix} \frac{dy_1}{dt} \\ \frac{dy_2}{dt} \end{pmatrix}$$

Imagem retirada dos slides

O y_1 representa o espaço percorrido pelo foguete e o y_2, a velocidade.

Sendo assim, obtemos o seguinte sistema de EDO:

$$funcaoFoguete = \begin{pmatrix} z(t) \\ v(t) \end{pmatrix}, \quad \frac{d}{dt}(funcaoFoguete) = \begin{pmatrix} z'(t) \\ v'(t) \end{pmatrix} = \begin{pmatrix} v(t) \\ v'(t) \end{pmatrix},$$

sendo $v' = \dfrac{-c\frac{dm}{dt} - mg - 0.5\rho C_d A|v|v}{m}$

e com PVI: $\begin{cases} z(0) = 0 \\ v(0) = 0 \end{cases}$

Tendo definido nossa EDO, conseguimos resolver a equação do foguete pela lsode da seguinte forma:

```
global H c area Cd g massa_inicial massa_foguete rho_inicial tempo_queima
H = 8000;                       % Altura
c = 3000;                      % Velocidade de expulsão do combustível
area = 1;                       % Area do foguete
Cd = 0.2;                       % Coeficiente de arrasto
g = 9.8;                        % Gravidade
massa_inicial = 1000;            % Massa do combustível
massa_foguete = 100;            % Massa do foguete
rho_inicial = 1.2;              % Densidade inical
tempo_queima = 220 - (20 * 7);  % Tempo de queima do combustível do foguete
```

Primeiramente definimos as constantes utilizadas para a resolução: altura, velocidade de expulsão do combustível, área do foguete, coeficiente de arrasto, gravidade, massa inicial do combustível, massa do foguete, densidade inicial e tempo de queima do combustível dado pela fórmula:

$220 - (20 * nUSP) = 80$, sendo nUSP = 7.

Depois, criamos uma função, funcaoFoguete, que será passada para a função lsode e que calcula a aceleração do nosso foguete, e depois chamamos a lsode de fato.

```
% Função que define a aceleracao do foguete
function A = funcaoFoguete(q, t)
  global H c area Cd g massa_inicial massa_foguete rho_inicial tempo_queima

  z = q(1);
  v = q(2);
  A(1) = v;

  if t <= tempo_queima
    m = massa_foguete + (massa_inicial * (1 - t/tempo_queima));
    A(2) = ((c * massa_inicial/tempo_queima) - (m * g) - (0.5 * (rho_inicial * exp(-z/H)) * Cd * area * abs(v) * v)) / m;
  else
    m = massa_foguete;
    A(2) = (- (m * g) - (0.5 * (rho_inicial * exp(-z/H)) * Cd * area * abs(v) * v)) / m;
 endif
endfunction

T = [1:0.02:2000];
[X, ISTATE, MSG] = lsode("funcaoFoguete",[0 0], T);
```

A função lsode, como visto acima, recebe a função do foguete, a condição inicial e um vetor de tempo T, e nos retorna o vetor X, que possui em sua primeira posição ($X(:,1)$) as alturas que o foguete atingiu ao longo do tempo e na segunda posição ($X(:,2)$), as velocidades que o foguete atingiu ao longo de sua trajetória, até retornar ao chão.

Após encontrarmos o vetor de velocidades, conseguimos derivá-lo, através da função diff, e obter o vetor de acelerações ao longo do tempo, como exibido na imagem do código a seguir:

```
% Encontrando a aceleracao
aceleracao = diff(X(:,2)) / 0.02;
Taceleracao = [1:0.02:1999.98];
```

Análise gráfica - parte A:

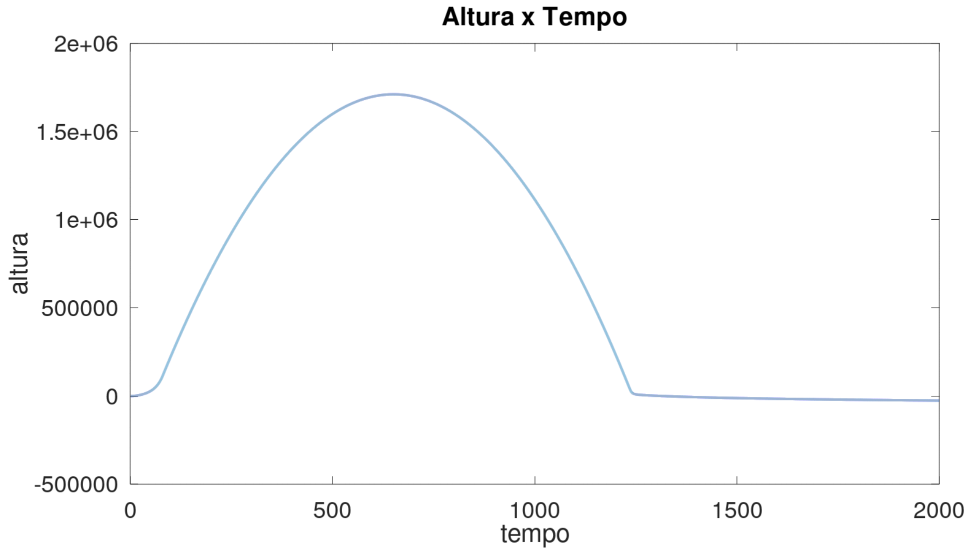
Agora que temos os três vetores pedidos (altura, velocidade e aceleração), conseguimos plotar os gráficos de cada um deles em função do tempo, dados pelos seguintes códigos:

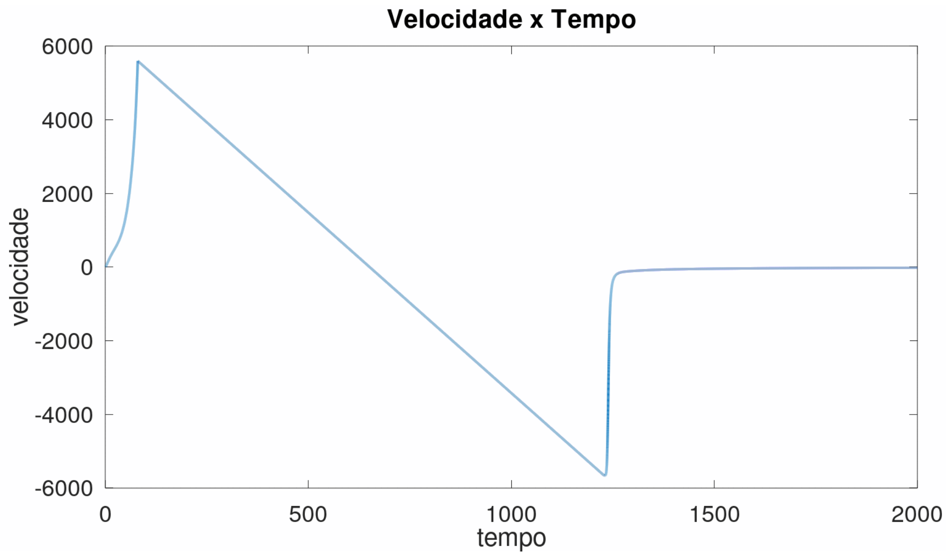
```
% Plot do gráfico de posicao
if(1)
  plot(T, X(:,1), "linewidth", 2)
  set(gca, "fontsize", 18)
  title ("Altura x Tempo");
  xlabel ("tempo");
  ylabel ("altura");
endif
```

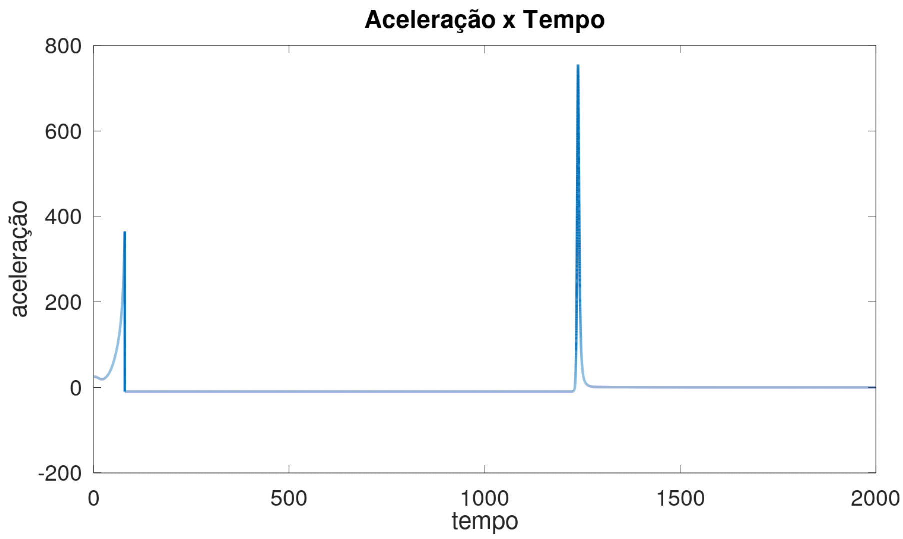
```
% Plot do gráfico de velocidade
if(0)
  plot(T, X(:, 2), "linewidth", 2)
  set(gca, "fontsize", 18)
  title ("Velocidade x Tempo");
  xlabel ("tempo");
  ylabel ("velocidade");
endif

% Plot do gráfico de aceleracao
if(0)
  plot(Taceleracao, aceleracao, "linewidth", 2)
  set(gca, "fontsize", 18)
  title ("Aceleração x Tempo");
  xlabel ("tempo");
  ylabel ("aceleração");
endif
```

Visualização dos gráficos:

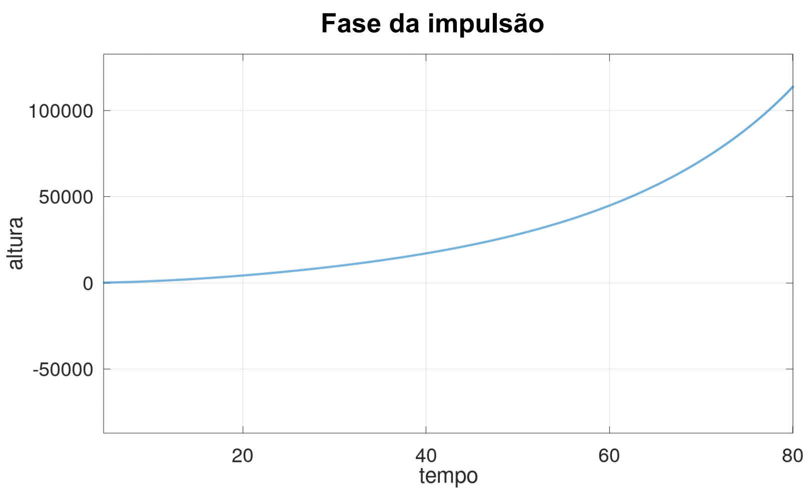




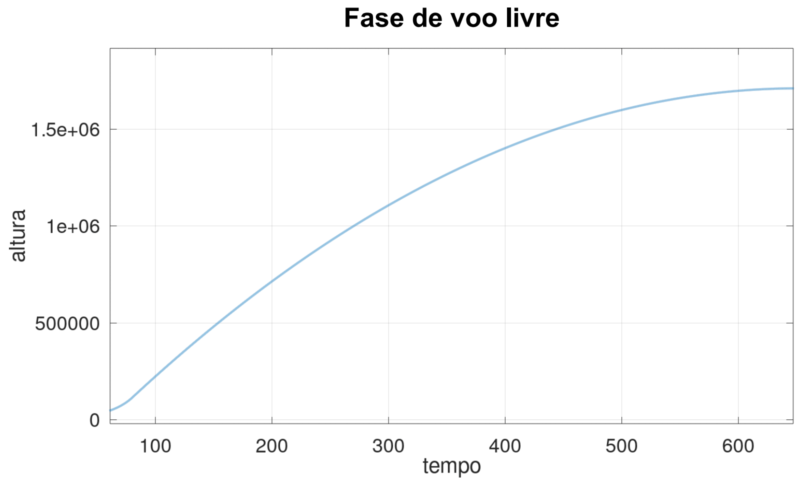


Tendo os gráficos, conseguimos interpretar os resultados identificando as diferentes partes do voo (impulsão, voo livre, reentrada na atmosfera, queda).

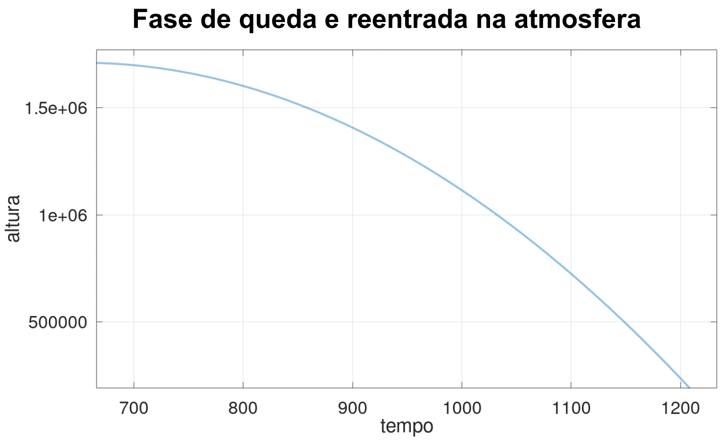
Podemos notar, que a primeira parte, impulsão, começa com $t = 0$ e acaba aproximadamente quando $t = 80$ (t medido em segundos), tempo que leva para queimar o combustível. Em $t = 80$, podemos notar que o foguete está no auge da sua velocidade e com aceleração de, aproximadamente, $400\, m/s^2$.

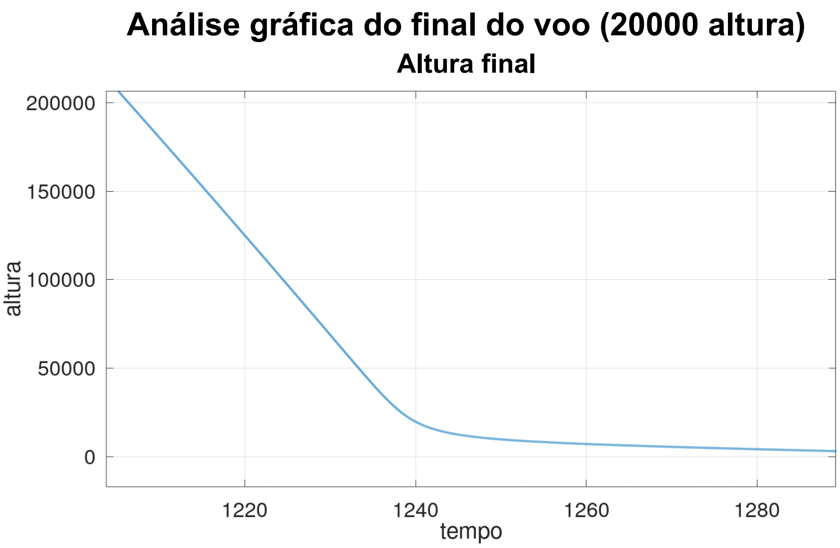


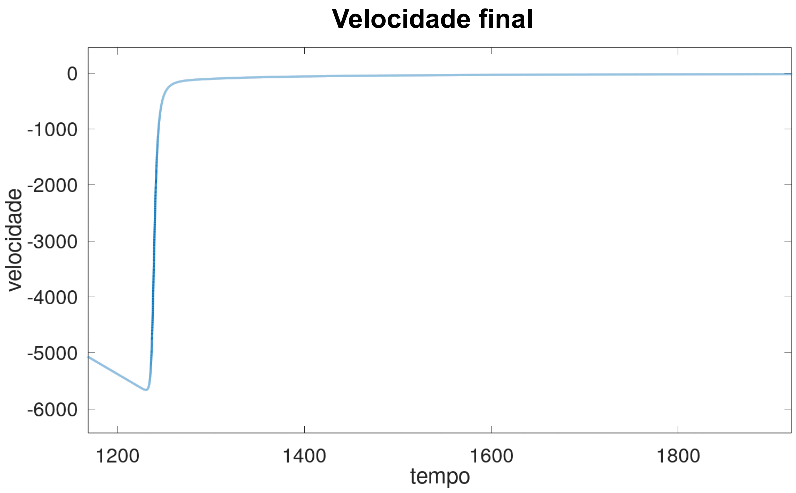
Depois da fase de impulsão, o foguete vai para a fase de voo livre, $t = 80$ até $t \simeq 700$. Nesta fase, as únicas forças atuantes no foguete são a peso e arrasto. Já que essas forças estão em sentido contrário ao deslocamento do foguete, ele começa a desacelerar, até atingir a altura máxima (1.711,3 km). Nessa etapa ocorre a saída do foguete da atmosfera.

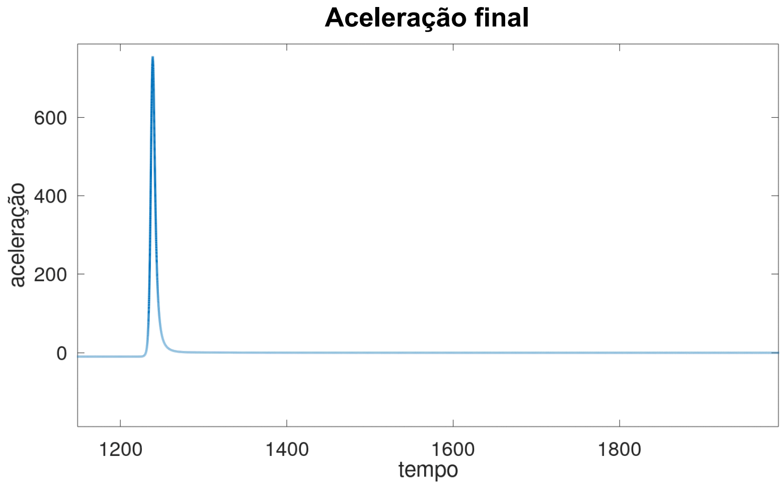


A partir do ponto de altura máxima, onde $t \simeq 700$, o foguete inicia a fase de queda livre, e apenas a força gravitacional atua sobre ele, adquirindo, nessa etapa, uma velocidade acelerada e retrógrada, retornando para a atmosfera, onde a força de arrasto volta a atuar no foguete o desacelerando, até que a força gravitacional se iguala a de arrasto e o foguete passa a andar com uma velocidade terminal até chegar ao chão.









Parte B:

1. Para encontrarmos a altura máxima que o foguete alcança, basta fazermos $max(X(:,1))$, ou seja, pegarmos o valor máximo no nosso vetor de alturas. Fazendo isso, encontramos uma altura máxima de 1711,3 quilômetros.

```
>> max(X(:,1))
ans = 1.7113e+06
```

2. Para encontrarmos a máxima aceleração e altura em que ocorre, basta fazermos o seguinte:

```
[max_aceleracao, posicao_aceleracao] = max(aceleracao)
```

aceleracao é o nosso vetor de vetores, que contém em sua primeira posição o vetor com as acelerações e em sua segunda posição, os tempos em que ocorrem as acelerações.
Então, pegamos a tupla que retorna a maior aceleração e a posição em que ocorre, basta fazermos max(aceleracao).

```
max_aceleracao = 755.57
posicao_aceleracao = 61886
```

3. Para descobrirmos o tempo que leva para o foguete voltar ao chão, basta somarmos todas as alturas positivas que o foguete alcançou. Como o nosso foguete começa em 0 e retorna para zero, a soma dessas alturas nos retorna a posição em que ele estará no chão. Se multiplicarmos a posição que encontramos pelo nosso passo de tempo $dt = 0.02$, encontramos o tempo de voo, ou seja, o tempo em que o foguete alcança o chão.

```
posicao_voo = sum(X(:,1) >= 0)
tempo_voo = posicao_voo * 0.02
```

Com isso, temos que leva 1317.9 segundos para o foguete alcançar o chão.

```
posicao_voo = 65897
tempo_voo = 1317.9
```

4. Para descobrirmos a altura máxima com Cd = 0.1,basta alterarmos nossa variável global Cd, definida no início do nosso programa e repetir o que fizemos no item 1. Com isso, encontramos que a altura máxima seria 1951,5 quilômetros.

```
>> max(X(:,1))
ans = 1.9515e+06
```

Podemos notar que diminuindo o coeficiente de arrasto, conseguimos atingir uma altura maior. Porém, os efeitos da força de arrasto são importantes, pois é essa força que desacelera o foguete e o equilibra até chegar ao chão, impedindo que o foguete colida drasticamente com o solo.

5. Se adicionarmos 1 quilograma a mais na carga do foguete, precisaremos colocar 7.65 kg a mais de combustível para atingir a mesma altura máxima de quando o foguete tinha 100 kg. Ou seja, a massa inicial de combustível deverá ser 1007.65.

```
massa_inicial = 1007.65;
massa_foguete = 101;
```

Parte C:

Nessa parte, o nosso intuito é resolver numericamente a nossa equação unidimensional do foguete, mas com outros métodos, alterando os espaços de tempo em cada uma das resoluções e comparando os resultados com a solução exata que descobrimos pela função lsode. Foi pedido para plotar e comparar as soluções obtidas de altura e velocidade.

Realizei 5 vezes cada um dos métodos Euler Explícito, RK2 e RK4, modificando o $dt = \{2,\ 1,\ 0.5,\ 0.25,\ 0.02\}$.

Análise gráfica - parte C:

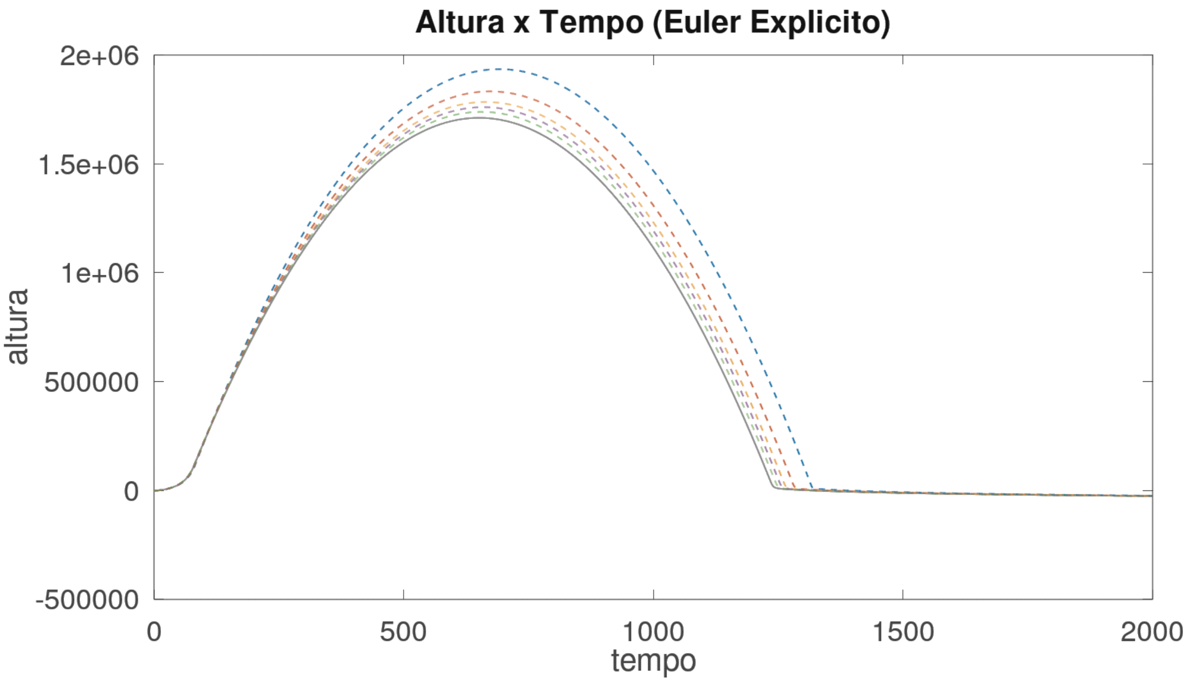
Legenda:

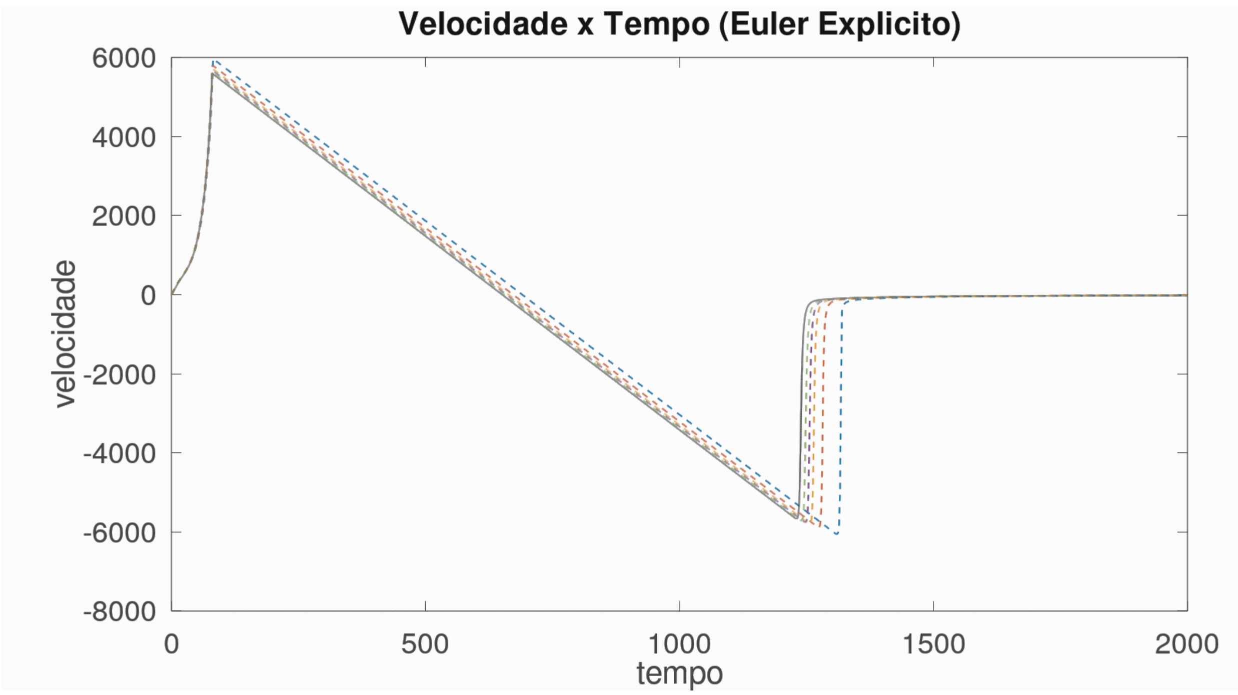
$dt = 2$ - tracejado azul

$dt = 1$ - tracejado laranja

$dt = 0.5$ - tracejado amarelo
$dt = 0.25$ - tracejado magenta
$dt = 0.02$ - tracejado verde
Função lsode - tracejado preto

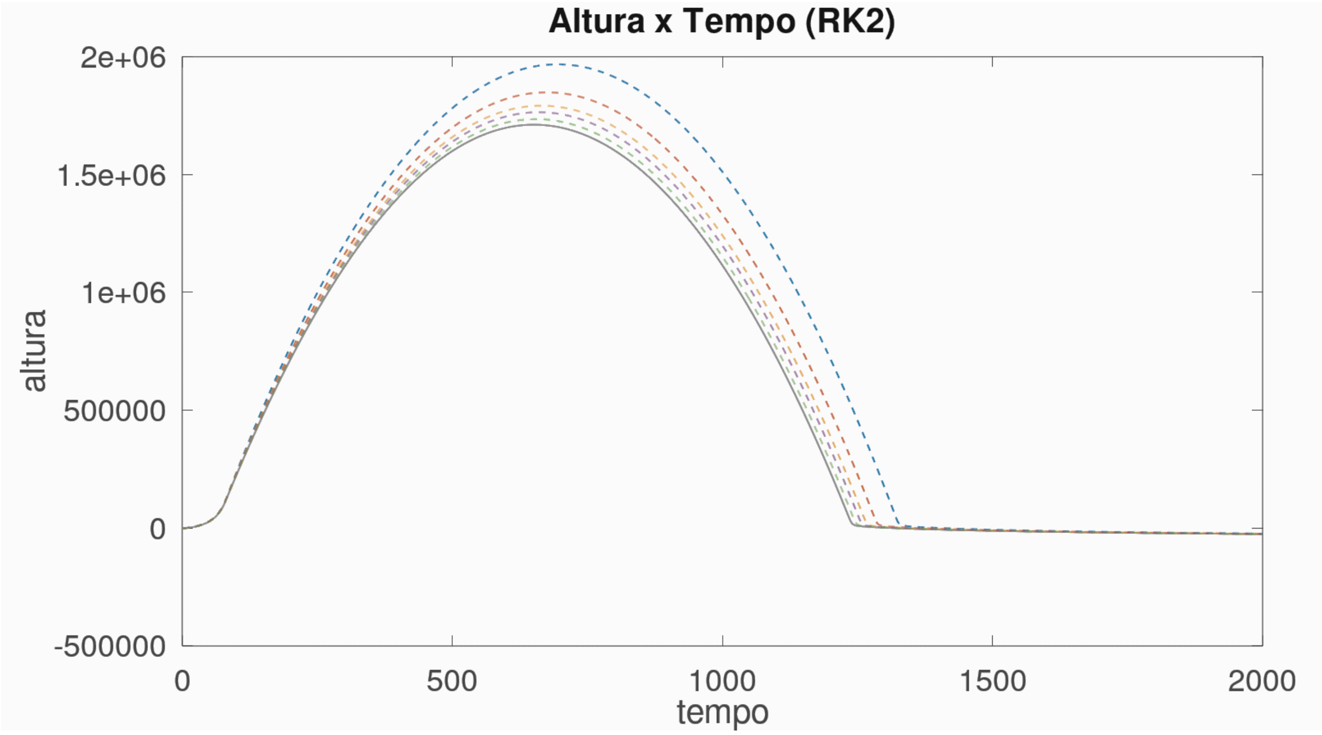
1. Método de Euler Explícito:

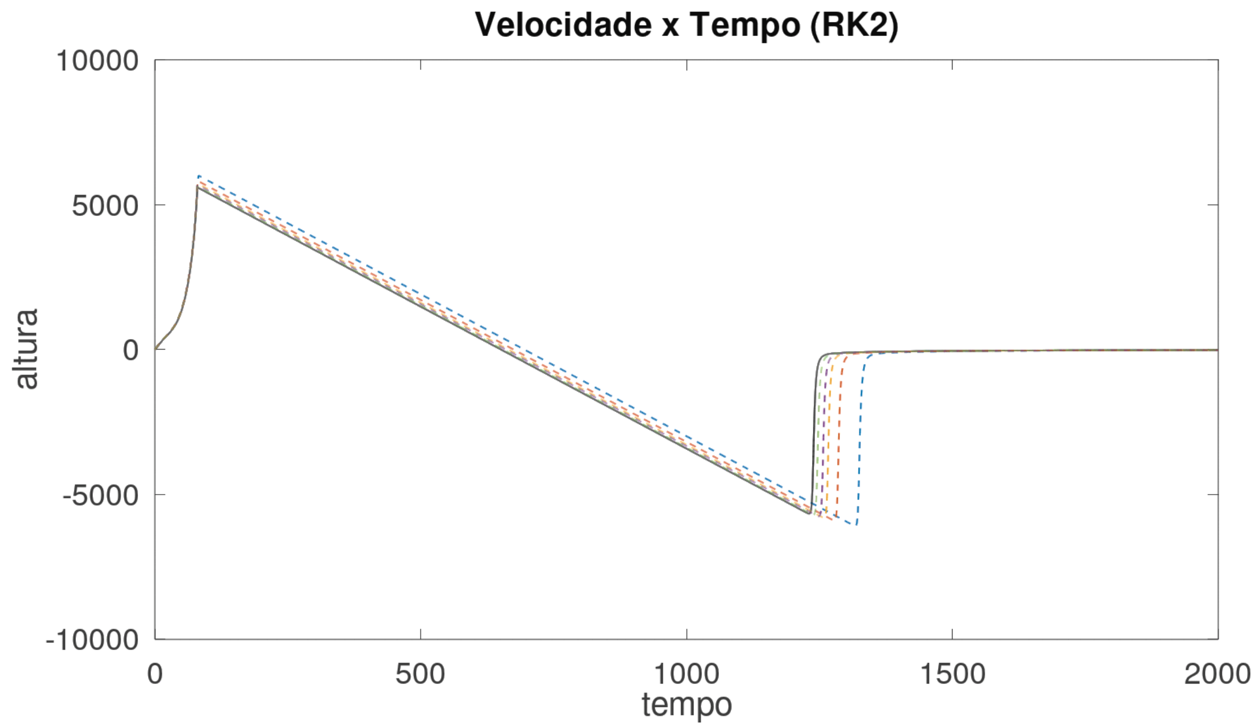




Podemos notar que quanto menor o espaçamento de tempo, mais perto da solução exata (lsode) ficamos.

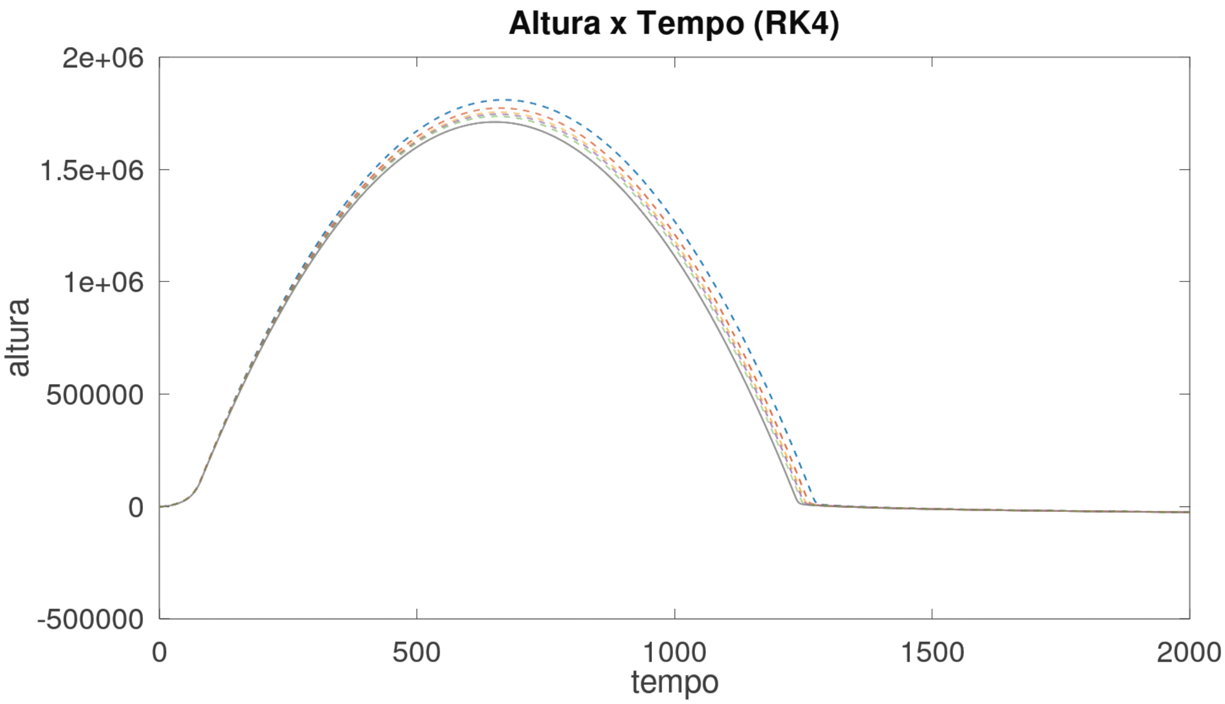
2. Runge-Kutta 2:

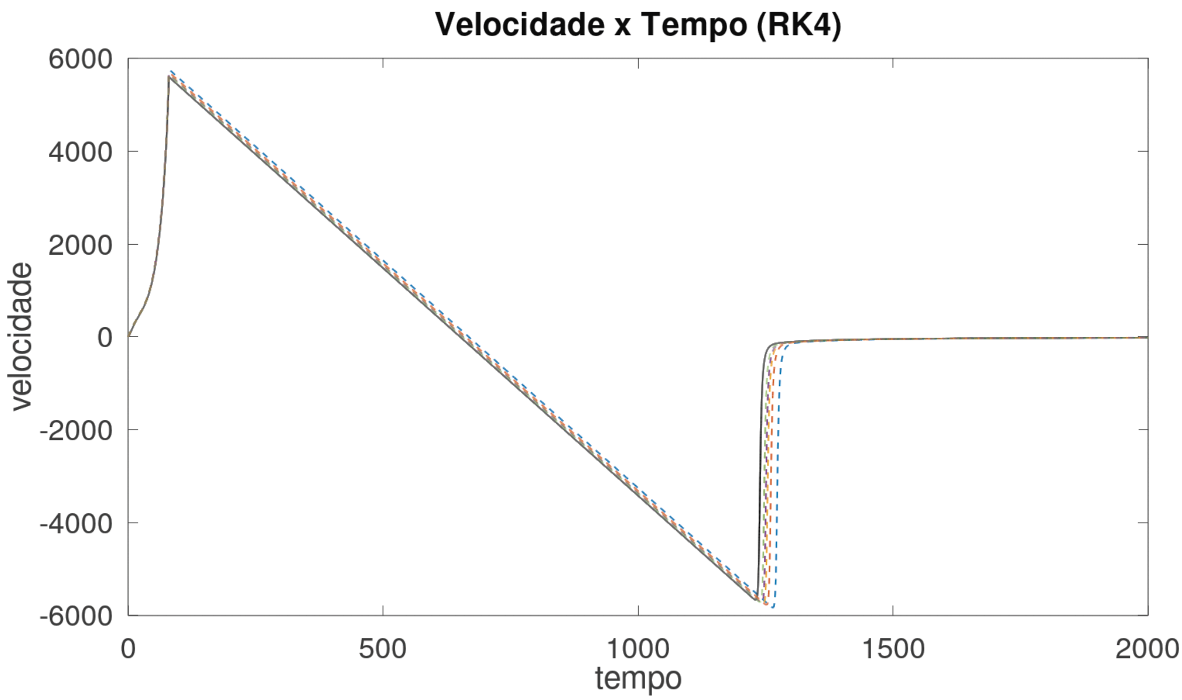




Podemos notar que quanto menor o espaçamento de tempo, mais perto da solução exata (lsode) ficamos.

3. Runge-Kutta 4:





Podemos notar que quanto menor o espaçamento de tempo, mais perto da solução exata (lsode) ficamos.

Parte D:

Nesta etapa, devemos calcular a velocidade com que o foguete chega ao chão através de cada um dos métodos, com os seguintes espaçamentos de tempo: $dt = \{2,\ 1,\ 0.5,\ 0.25\}$.

	$2\ seg$	erro $2\ seg$	$1\ seg$	erro $1\ seg$	$0.5\ seg$	erro $0.5\ seg$	$0.25\ seg$	erro $0.25\ seg$
Solução exata	-94.106 m/s	-	-93.520 m/s	-	-93.230 m/s	-	-93.086 m/s	-
Euler Explícito	-93.214 m/s	-0.892 m/s	-93.326 m/s	-0.194 m/s	-93.222 m/s	-0.008 m/s	-93.074 m/s	-0.012 m/s
RK2	-93.956 m/s	-0.15 m/s	-93.238 m/s	-0.282 m/s	-93.115 m/s	-0.115 m/s	-93.006 m/s	-0.08 m/s
RK4	-93.129 m/s	-0.977 m/s	-93.467 m/s	-0.053 m/s	-93.066 m/s	-0.164 m/s	-93.012 m/s	-0.072 m/s

Tendo a tabela acima, conseguimos perceber que não há uma convergência clara conforme diminuímos os espaçamento de tempo. Podemos ver que o erro variou de acordo com cada dt, mas não seguindo uma regra clara.