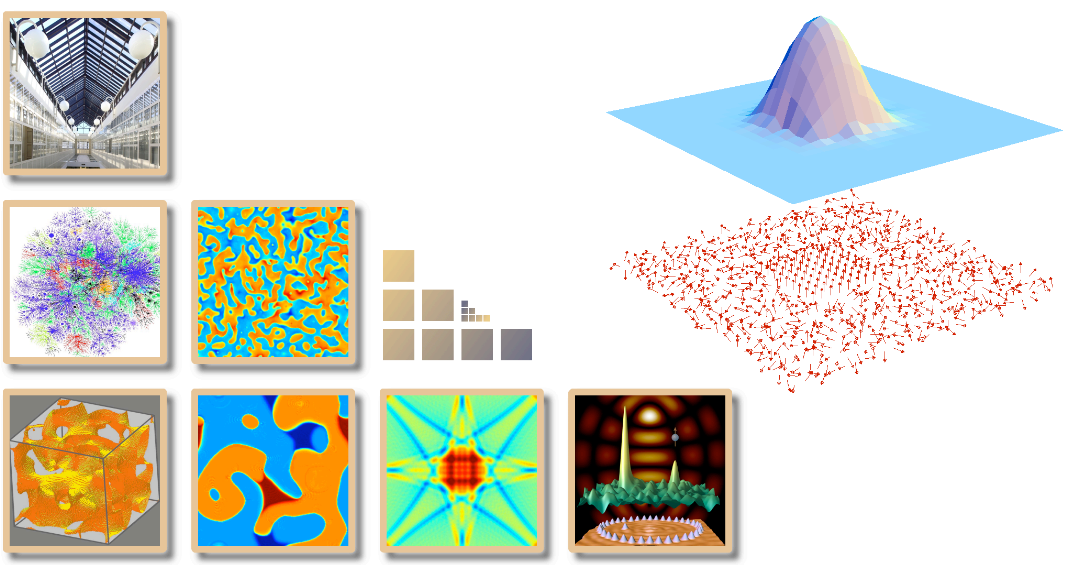
Órbitas de sondas espaciais: *a odisseia da Voyager*



CFP and the Physics Department

Responsável: Prof. Lopes Santos

Orientação: Aires Francisco

A equipa:

António (Cabo Verde)

Daniel (Aveiro)

Diana (Porto)

Gonçalo (Castelo Branco)

Inês (Chaves)

Pedro (Aveiro)

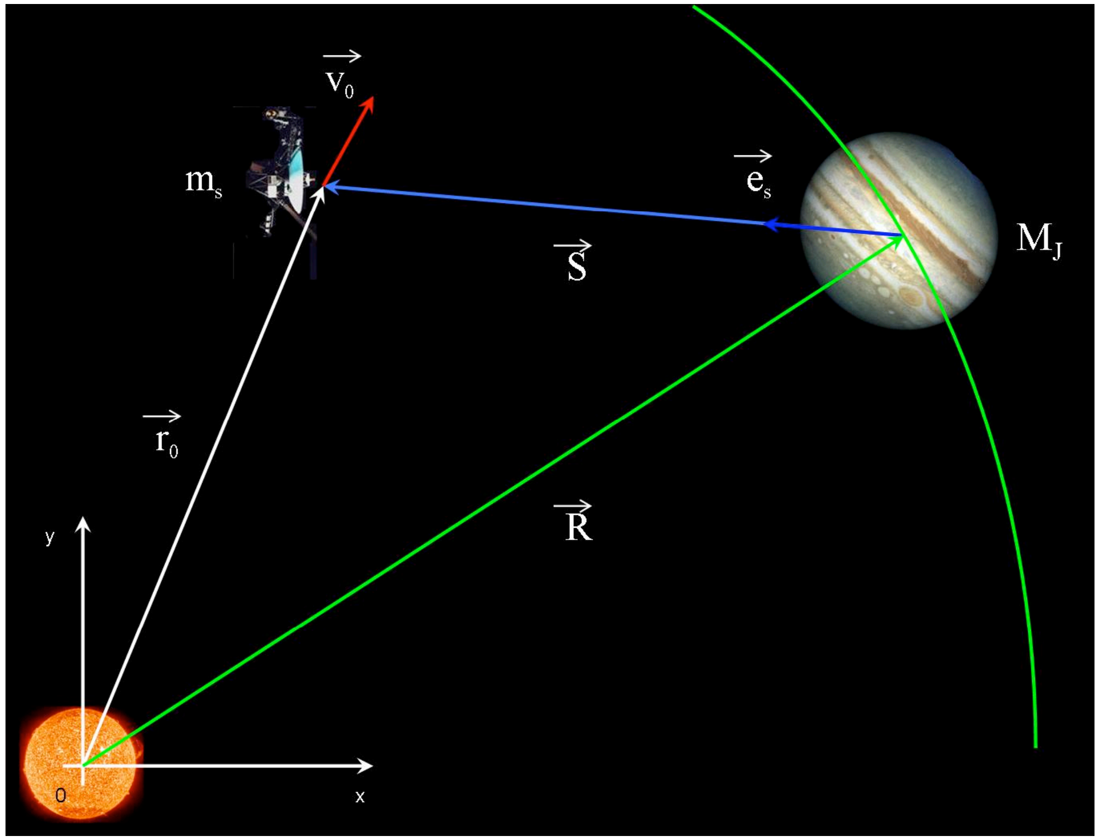
U. PORTO

FC FACULDADE DE CIÊNCIAS UNIVERSIDADE DO PORTO

Outline

- Introdução ao problema
    - Mecânica Newtoniana e método de Euler
        - Resultados
            - Resumo e conclusões

Introdução ao Problema



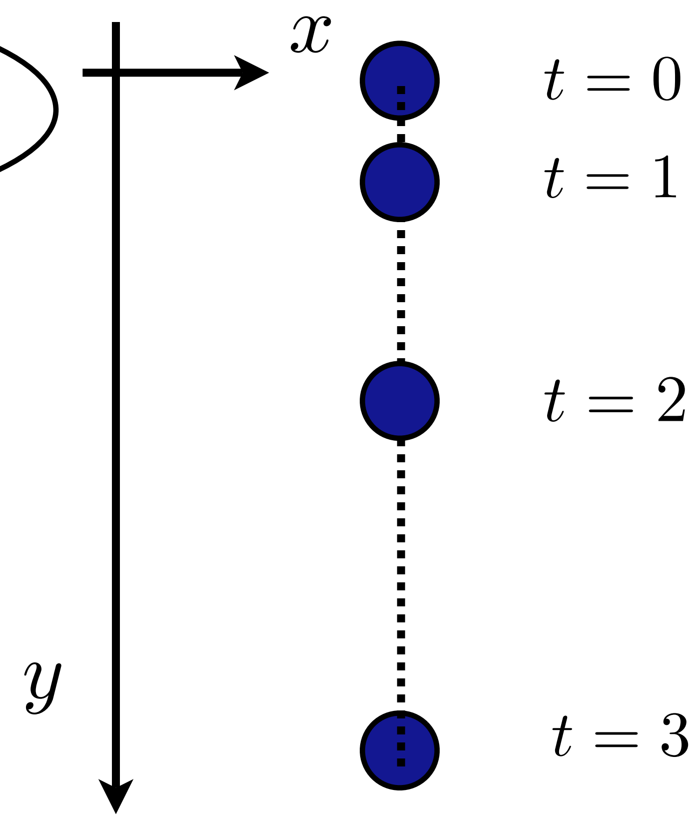
a) Calcular trajectória

b) Calcular velocidade

Questão: Como fazer a sonda voltar da sua aproximação ao planeta com o menor gasto possível de combustível?

Introdução ao Problema

(Galileu, Pisa, XVII)



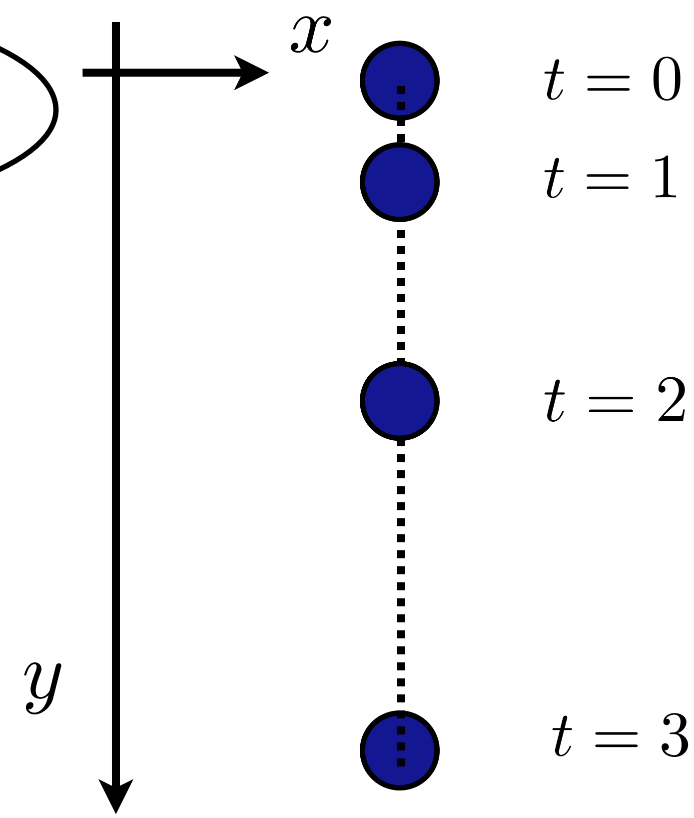
$$
\begin{aligned}
x(t) &= x_0 \\
y(t) &= h - \frac{1}{2} g t^2
\end{aligned}
$$

[lei de movimento]

Problema simples, pois aceleraçao da gravidade é um vector constante nas imediações de Pisa!

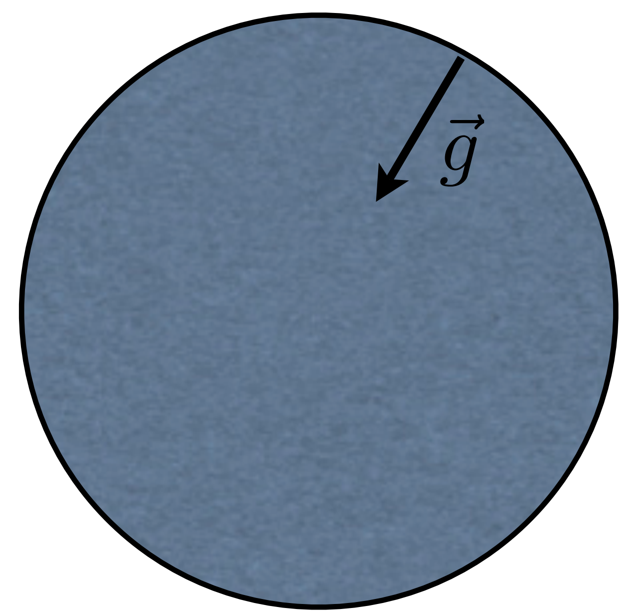
Introdução ao Problema

(Galileu, Pisa, XVII)



$$
\begin{aligned}
x(t) &= x_0 \\
y(t) &= h - \frac{1}{2} g t^2
\end{aligned}
$$

[lei de movimento]

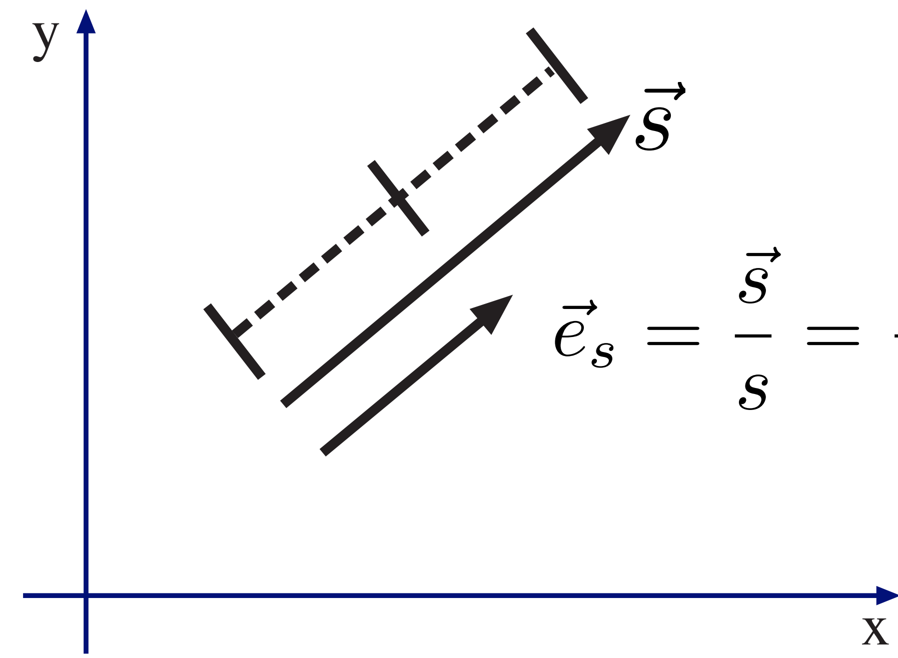


Problema simples, pois aceleraçao da gravidade é um vector constante nas imediações de Pisa!

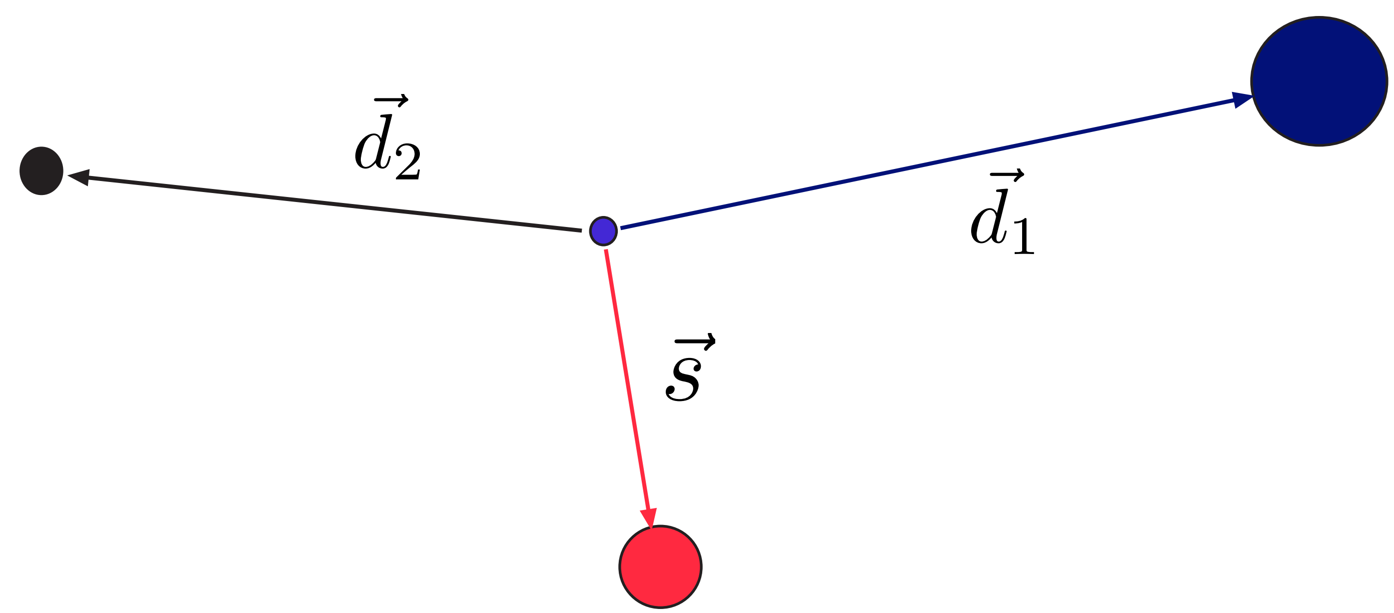
A dinâmica da sonda

$$\vec{F} = m\vec{a}$$

$$\vec{F}_g = -\frac{mMG}{s^2}\vec{e}_s$$



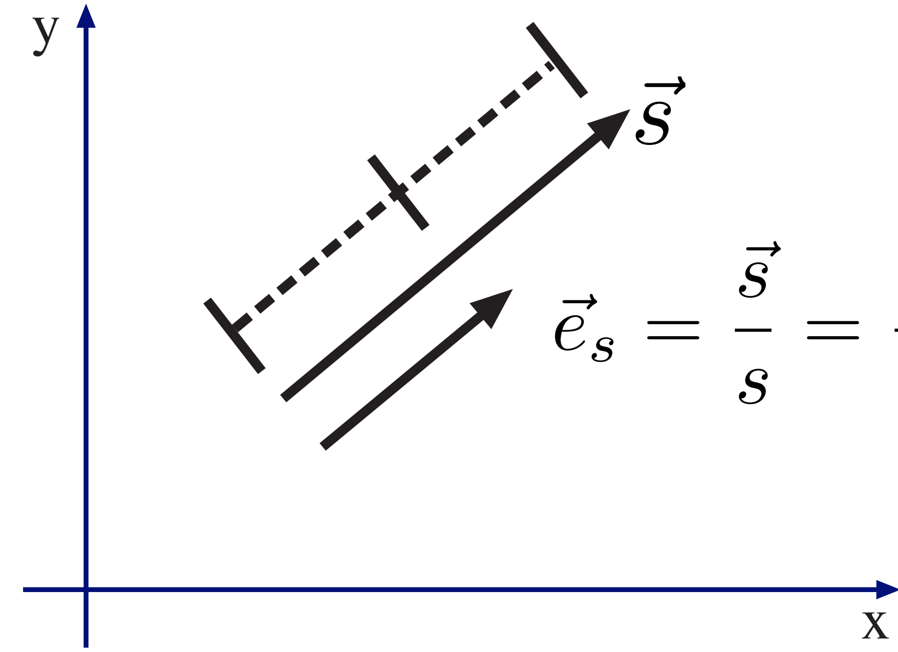
$$\vec{e}_s = \frac{\vec{s}}{s} = \frac{(s_x, s_y)}{\sqrt{s_x^2 + s_y^2}}$$



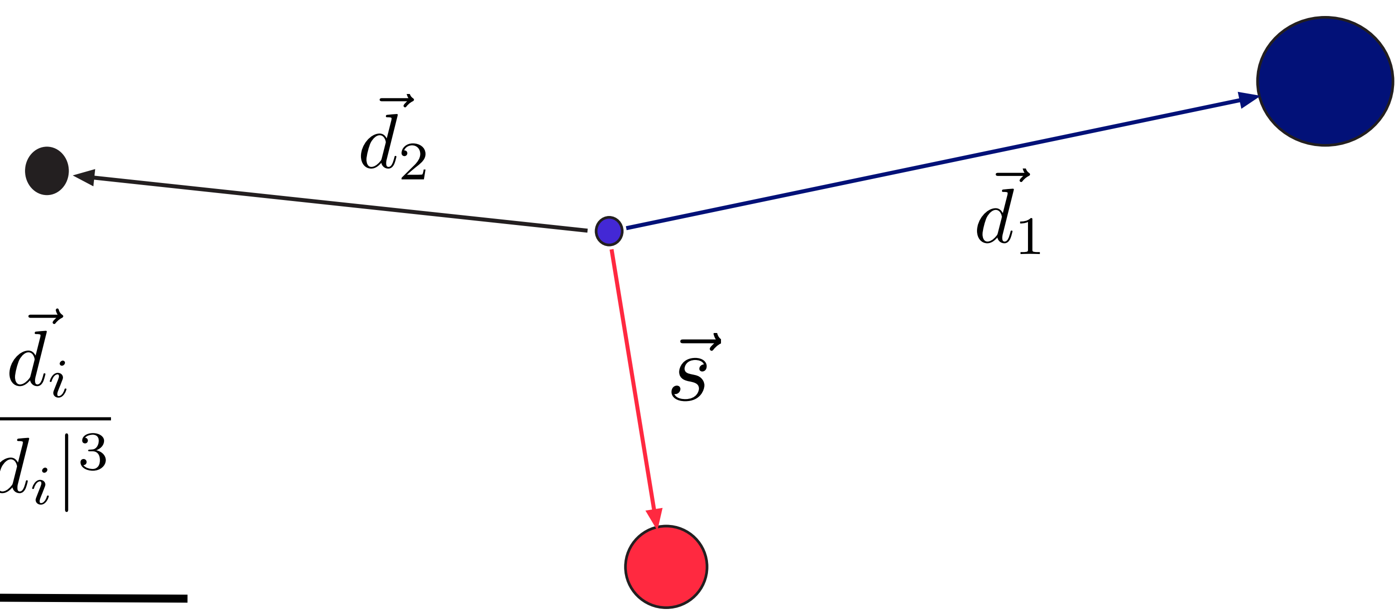
A dinâmica da sonda

$$\boxed{\begin{gathered}\vec{F} = m\vec{a} \\ \vec{F}_g = -\frac{mMG}{s^2}\vec{e}_s\end{gathered}}$$

$$\vec{s} = \vec{r} - \vec{R}_j$$



$$\vec{e}_s = \frac{\vec{s}}{s} = \frac{(s_x, s_y)}{\sqrt{s_x^2 + s_y^2}}$$

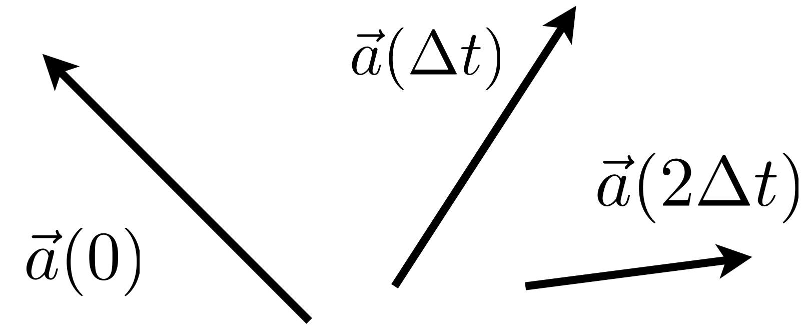


$$\vec{a} = \underbrace{-GM_J \frac{\vec{r} - \vec{R}_J}{|\vec{r} - \vec{R}_J|^3}}_{\text{planeta na vizinhança}} \underbrace{-G \sum_{i=1}^{N} M_i \frac{\vec{d}_i}{|d_i|^3}}_{\text{corpos distantes}}$$

O método de Euler

$$\vec{a} = -GM_J \frac{\vec{r} - \vec{R}_J}{|\vec{r} - \vec{R}_J|^3} - GM_s \frac{\vec{r}}{r^3}$$

As equações simples de movimentos rectilínios não se aplicam. Como resolver isto?

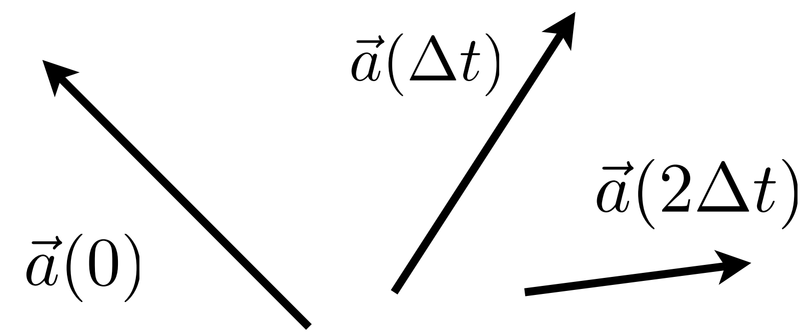


(Movimento variado em duas dimensões)

O método de Euler

$$\vec{a} = -GM_J \frac{\vec{r}-\vec{R}_J}{|\vec{r}-\vec{R}_J|^3} - GM_s \frac{\vec{r}}{r^3}$$

As equações simples de movimentos rectilínios não se aplicam. Como resolver isto?



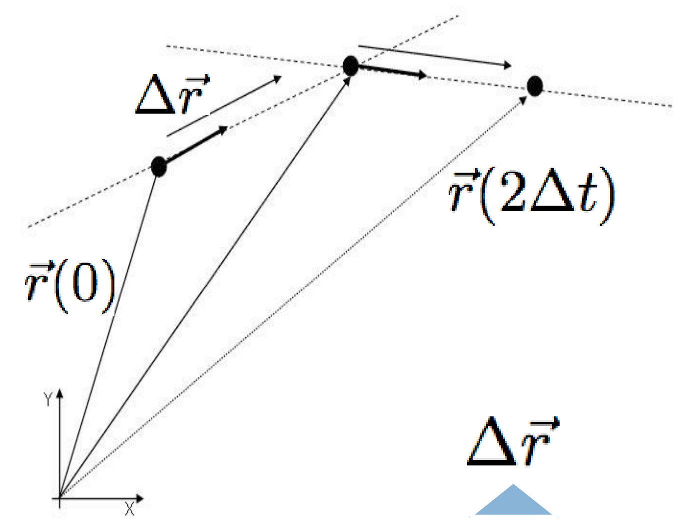
(Movimento variado em duas dimensões)

Método de Euler!

condições-iniciais

$$\begin{aligned}
x(\Delta t) &\cong \boxed{x(0)} + v_x(0)\Delta t\\
y(\Delta t) &\cong \boxed{y(0)} + v_y(0)\Delta t\\
v_x(\Delta t) &\cong \boxed{v_x(0)} + a_x(0)\Delta t\\
v_y(\Delta t) &\cong \boxed{v_y(0)} + a_y(0)\Delta t
\end{aligned}$$

$$\begin{aligned}
x(2\Delta t) &\cong x(\Delta t) + v_x(\Delta t)\Delta t\\
y(2\Delta t) &\cong y(\Delta t) + v_y(\Delta t)\Delta t\\
v_x(2\Delta t) &\cong v_x(\Delta t) + a_x(\Delta t)\Delta t\\
v_y(2\Delta t) &\cong v_y(\Delta t) + a_y(\Delta t)\Delta t
\end{aligned}$$

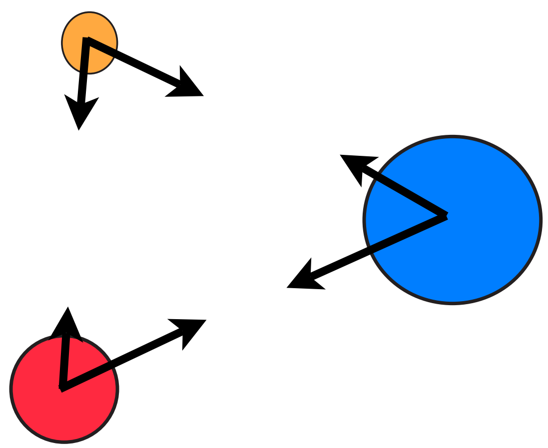


$\Delta\vec{r}$

$$\vec{r}(\Delta t) = \vec{r}(0) + \vec{v}(0)\Delta t$$
$$\vec{v}(\Delta t) = \vec{v}(0) + \vec{a}(0)\Delta t$$

A aceleração é calculada com a fórmula acima assumindo que a trajectória do planeta não se altera.

Revisão (leis de conservação)



$$\sum \vec{F} = \vec{0}$$

$$\sum_i m \frac{\Delta \vec{v}_i}{\Delta t} = \vec{0} \quad \Longrightarrow \quad \sum_i \frac{\Delta \vec{p}_i}{\Delta t} = \vec{0}$$

Resulta da definição de momento: $\vec{p} = m\vec{v}$

(relembrar aula J. Penedones)

$\Longrightarrow$ **O momento total conserva-se!**

Isto justifica a aproximação feita. Vejamos:

$$m_s \vec{v}_i + M_J \vec{V}_i = m_s \vec{v}_f + M_J \vec{V}_f$$ (lei da conservação aplicada à sonda + planeta)

$$\Delta \vec{V} = -\frac{m_s}{M_J} \Delta \vec{v} \simeq 0 \longrightarrow$$ planeta não "sente" a sonda [*]

[*] NASA reclama que a Voyager atrasou Júpiter 30cm por cada trilião de anos...

Odisseia da Voyager (back to 1977)

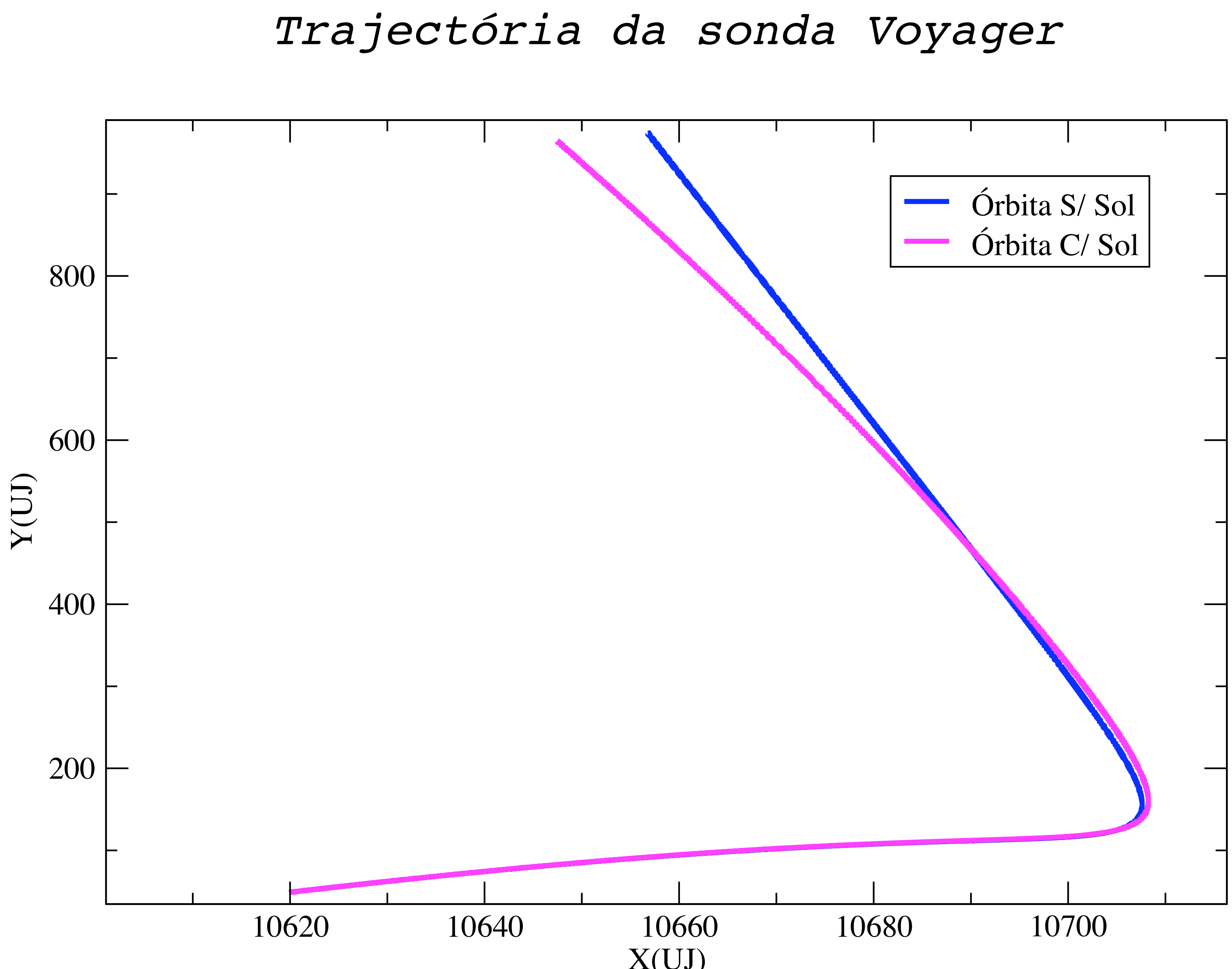
Notação:

<u>distância</u>: UJ [*] - Unidade Joviana (=raio de Júpiter: 71300Km)

<u>tempo</u>: h (horas)

<u>energia por unidade de massa</u>: JJ (Joule Joviano=UJ^2/h^2)

$$x(0) = 10620(\mathrm{UJ})$$
$$y(0) = 49(\mathrm{UJ})$$
$$vx(0) = 0.2939(\mathrm{UJ/h})$$
$$vy(0) = 0.4045(\mathrm{UJ/h})$$



[*] Não confundir com Universidade Júnior.

O efeito 'slingshot'

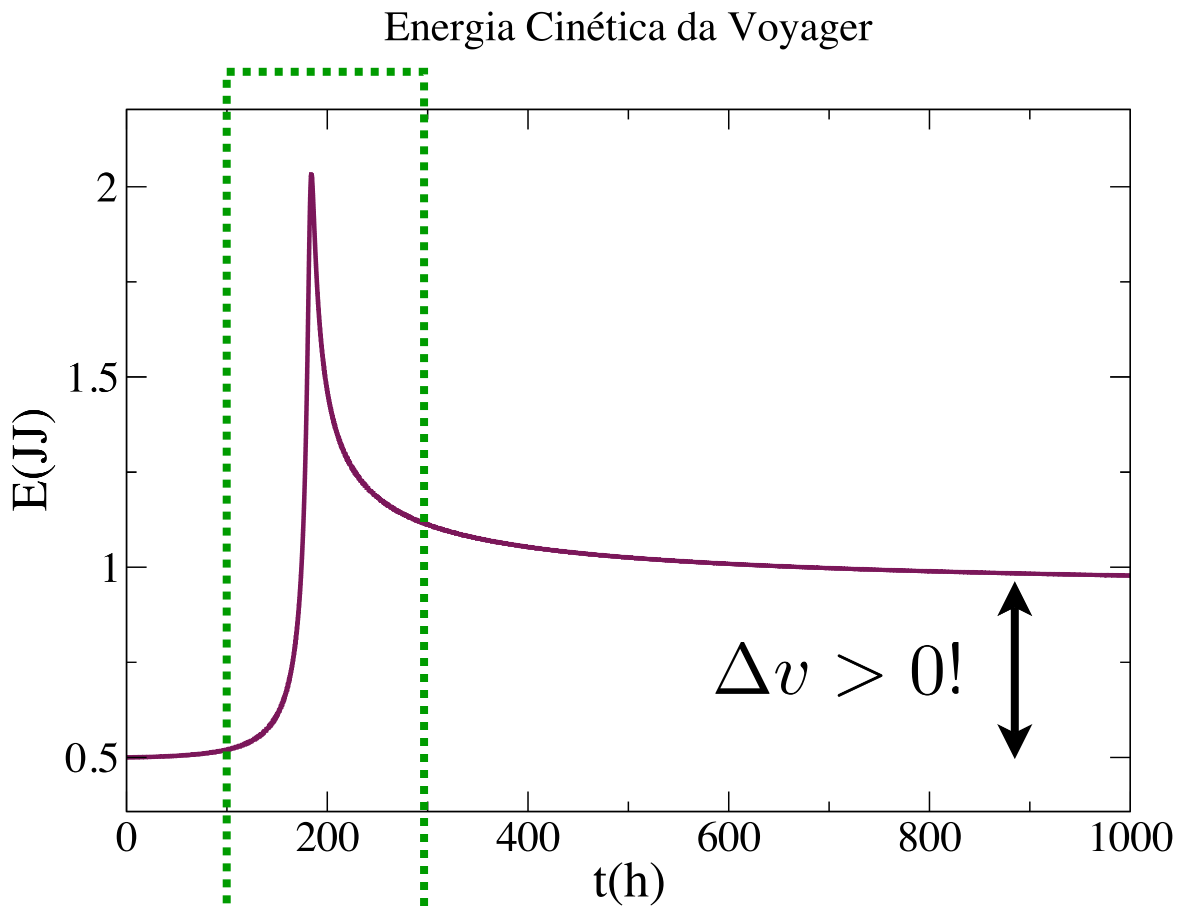
Como explicar a aceleração assistida gravitacionalmente?

No referencial do planeta podemos escrever:

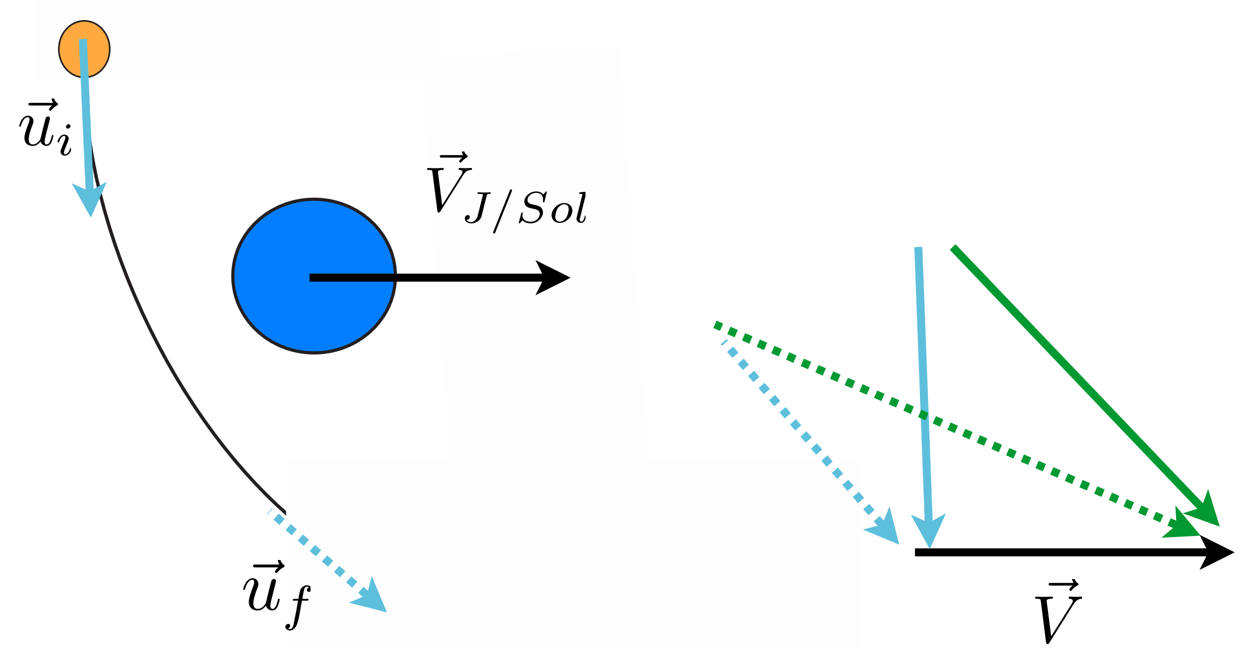
$$E = \frac{1}{2}m\vec{u}^2 - \frac{GM_J}{s}$$

Durante aproximação e afastamento, s>>1:

$$E(\infty) \simeq \frac{1}{2}m\vec{u_\infty}^{\,2} \Rightarrow |\vec{u}_\infty| \text{ conservado}$$



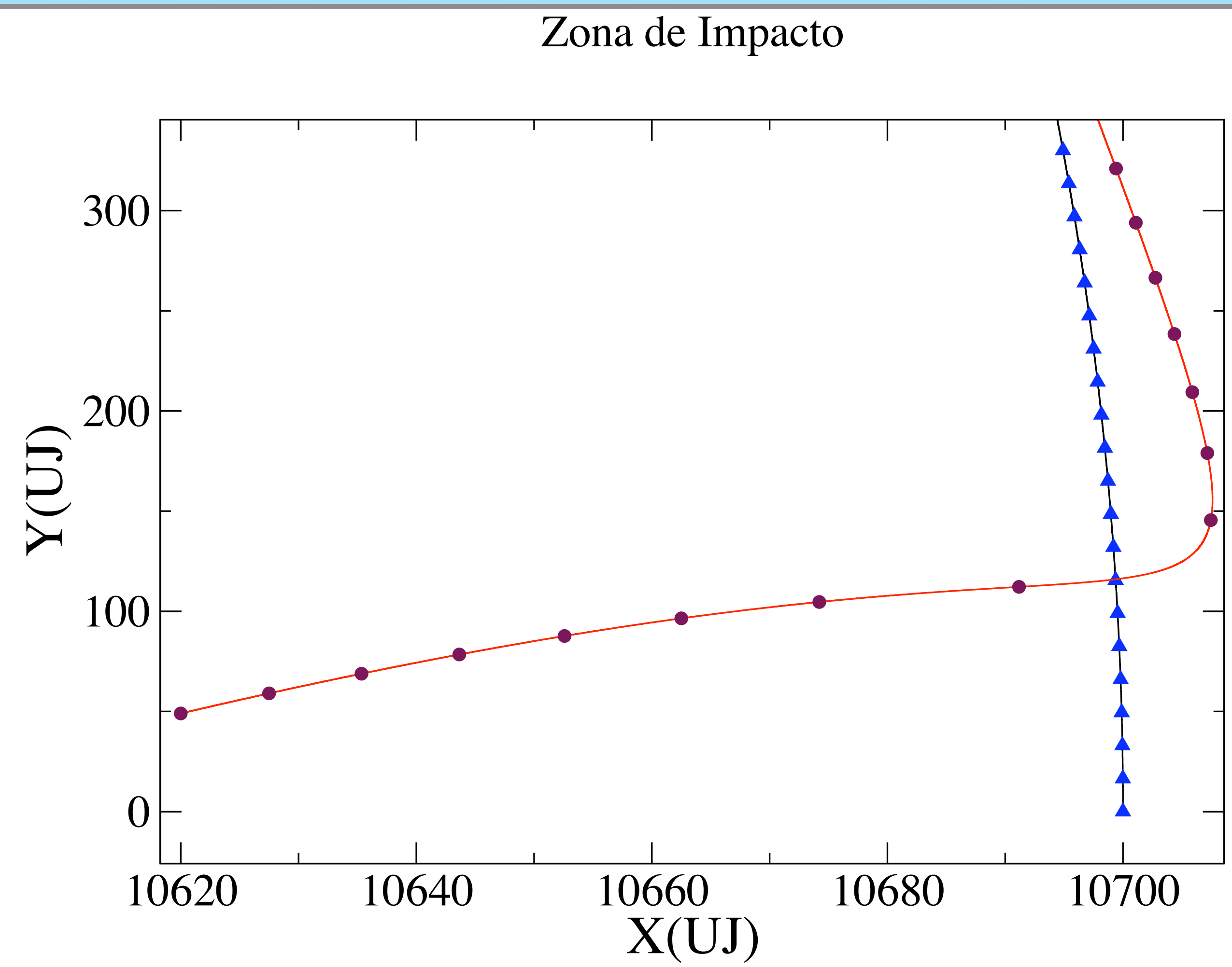
zona de impacto



Adição de velocidades:

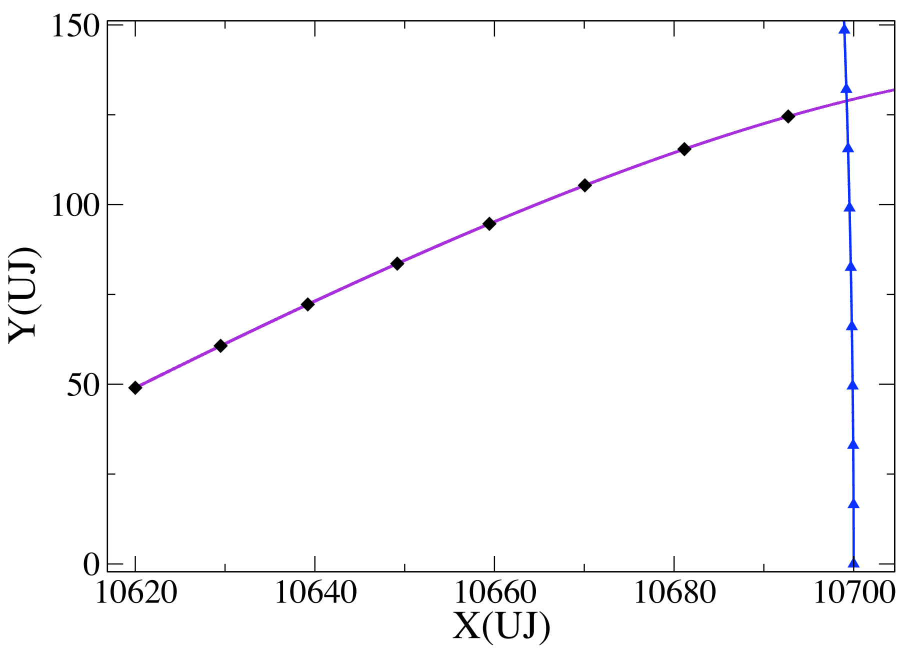
$$\vec{v}_{s/Sol} = \vec{u} + \vec{V}_{J/Sol}$$

O efeito ‘slingshot’ (2)

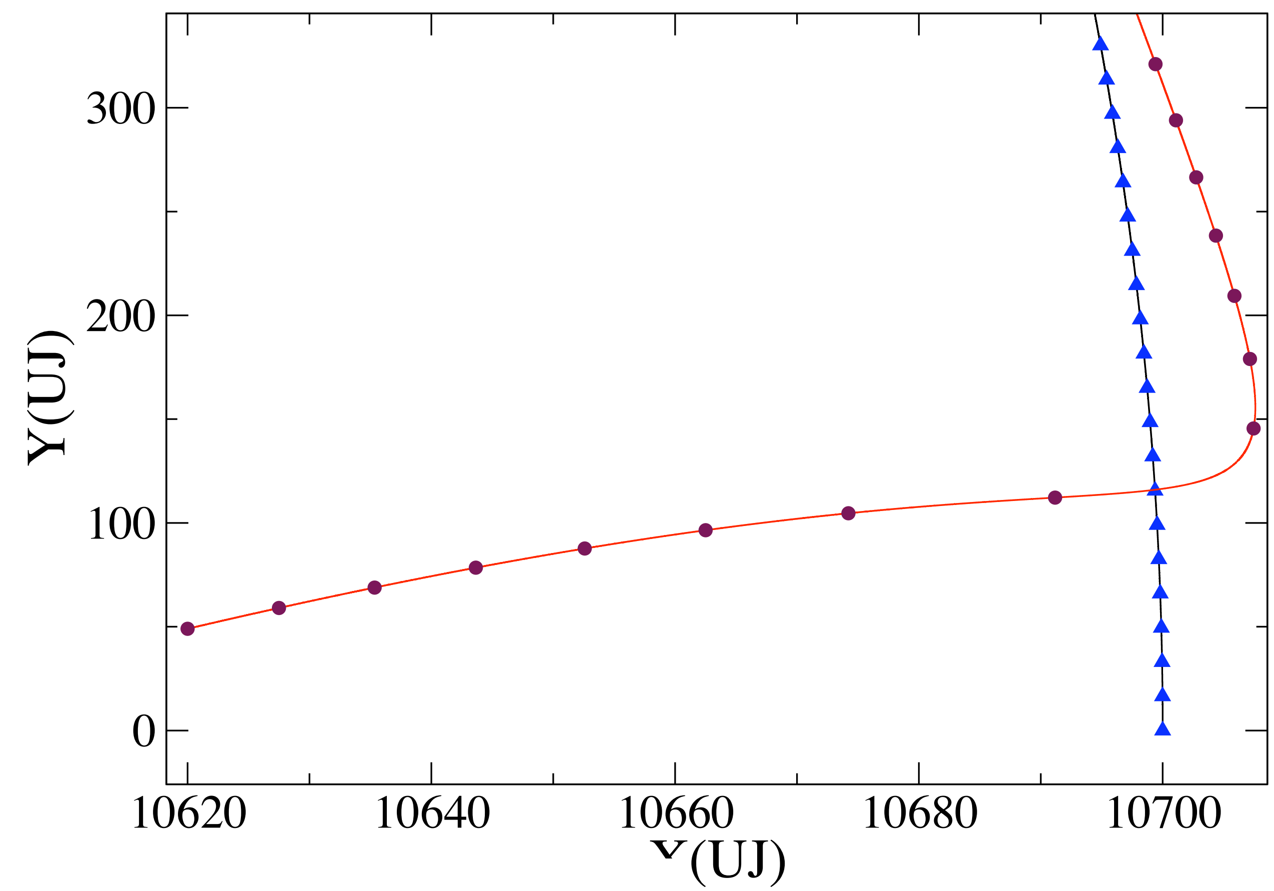


O efeito ‘slingshot’ (2)

O que acontece se a sonda passar pela frente do planeta?



Zona de Impacto

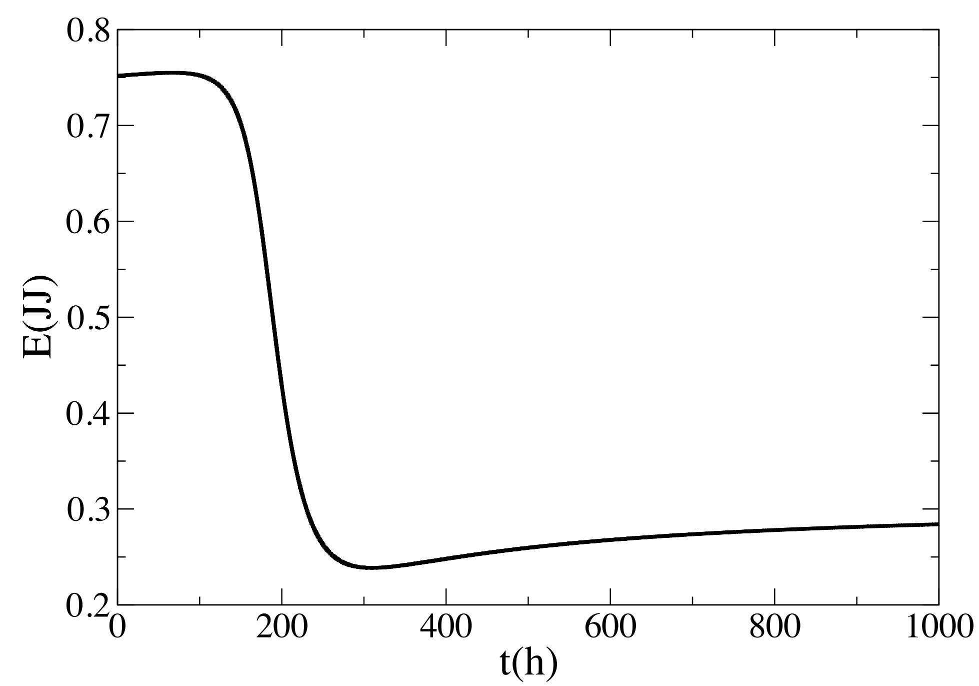


$$\Delta v < 0!$$

‘anti-slingshot’!

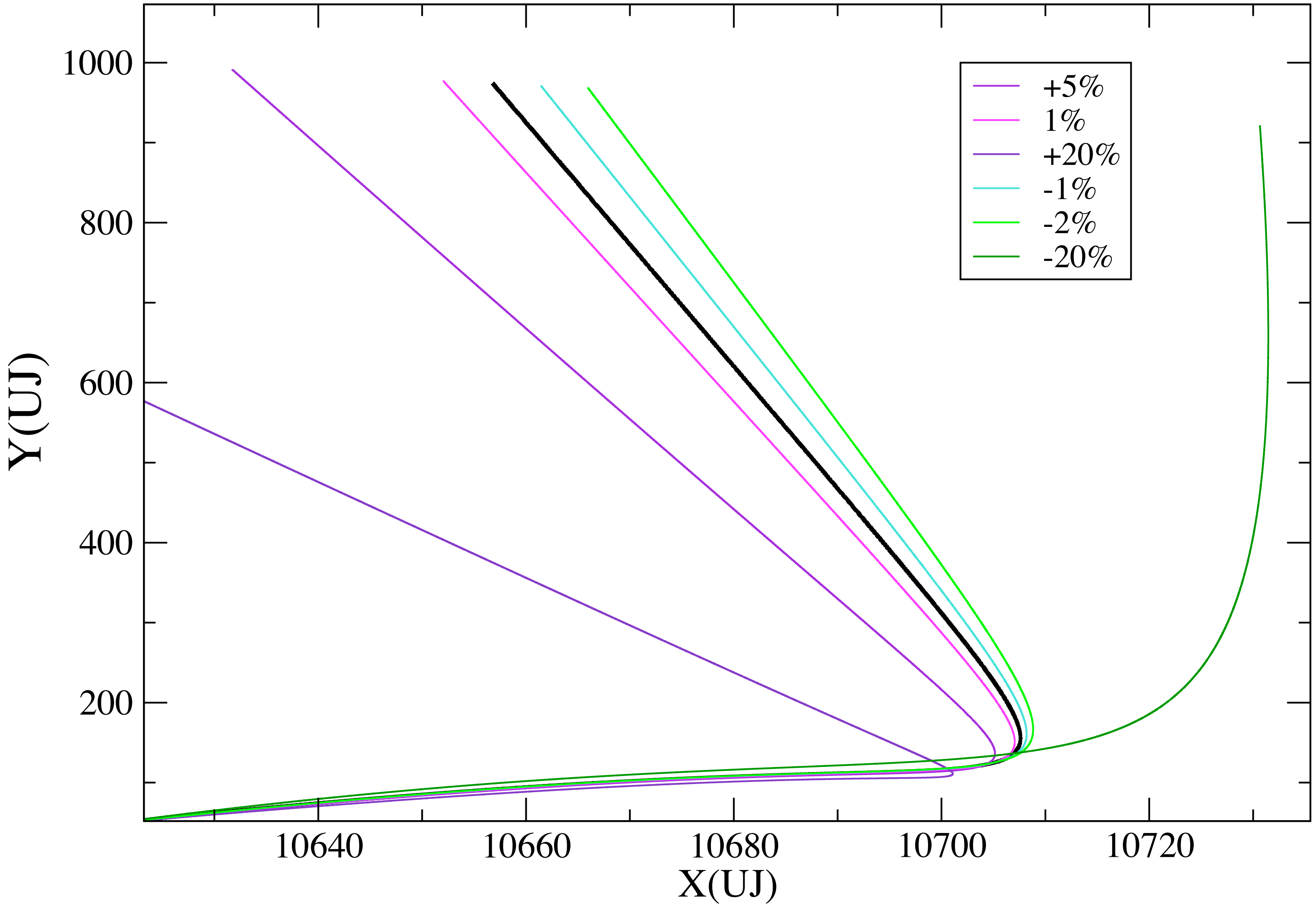
$$\boxed{\begin{array}{c} vx \rightarrow vx * 1.60 \\ vy \rightarrow vy * 1.45 \end{array}}$$

Energia Cinética



Quão difícil é a vida de um eng. da NASA?

Quão sensível é a trajectória às condições iniciais?

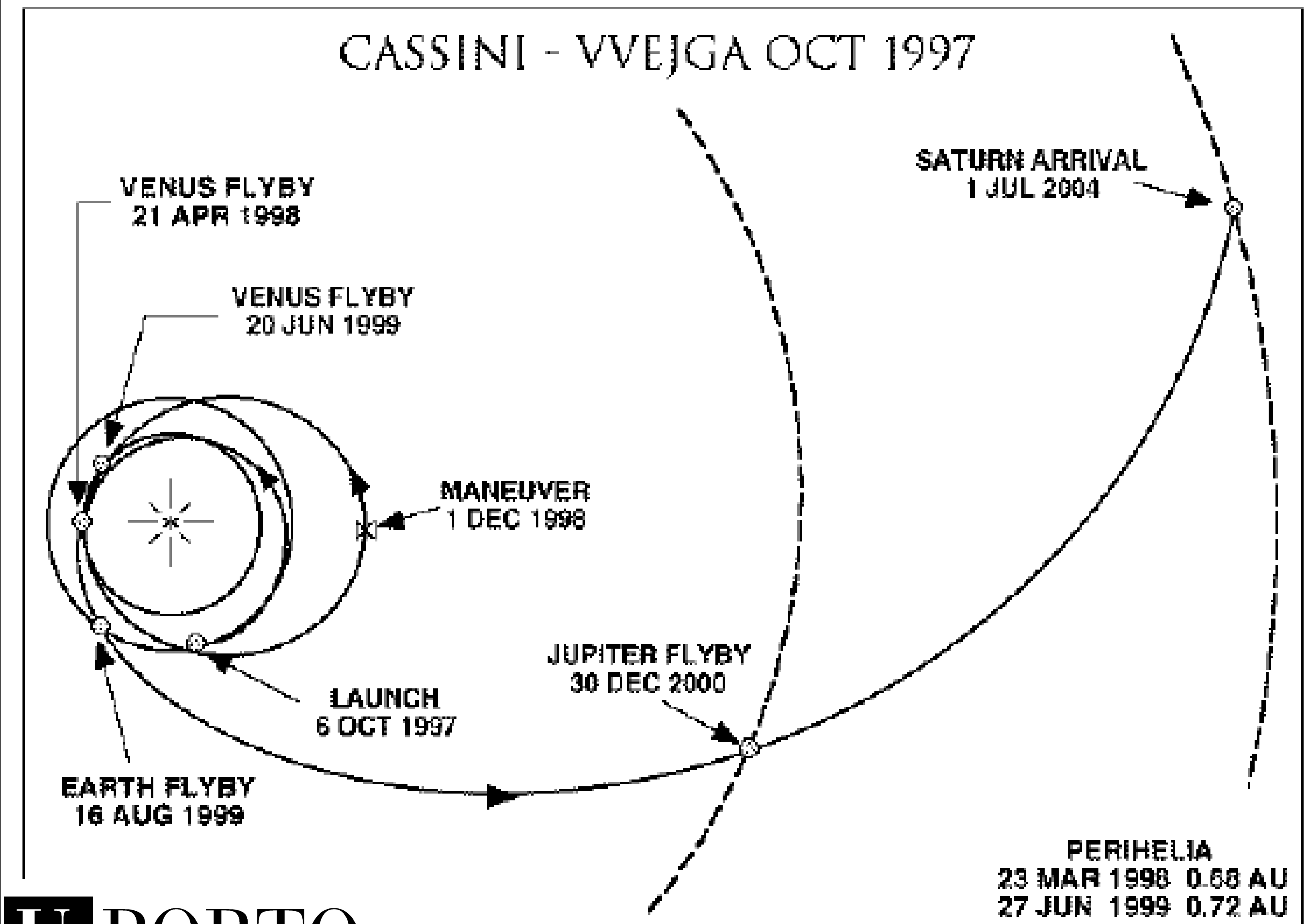


Análise qualitativa e parcial: precisão de 1-5%!

Conclusões

I) é possível calcular trajectória de corpos celestes usando um método simples

II) o envio de sondas deve fazer proveito do efeito slingshot para poupança de combustível




```
for(t=passo;t<=MAX;t+=passo){
        for(i=0;i<4;i++) z[i]=x[i];

        x[0]+=passo*z[2];
        x[1]+=passo*z[3];
        x[2]+=passo*(-G*Mj*(z[0]-xj(t-
passo))/pow(sqrt(pow(z[0]-xj(t-passo),
2)+pow(z[1]-yj(t-passo),2)),3)-
G*Ms*z[0]/pow(sqrt(pow(z[0],
2)+pow(z[1],2)),3));
        x[3]+=passo*(-G*Mj*(z[1]-yj(t-
passo))/pow(sqrt(pow(z[0]-xj(t-passo),
2)+pow(z[1]-yj(t-passo),2)),3)-
G*Ms*z[1]/pow(sqrt(pow(z[0],
2)+pow(z[1],2)),3));
```

```c
#include <stdio.h>
#include <math.h>

#define PI 3.1415926
#define Mt 5.9742E+24       // Massa da Terra
#define Mj 318*Mt           // Massa de Jupiter
#define Ms 332775*Mt        // Massa do Sol
#define k 4*PI*PI           // G*Ms em [distancia]=U.A e [tempo]=ano
#define G 2.37E-27          // Const. Gravitação em unidades Jovianas
#define passo 0.01
#define MAX 1000.0

  int i;
  double t,x[4],z[4];

  FILE *tabela1;
  FILE *tabela2;

double xj(double t);
double yj(double t);

int main()

{

  tabela1 = fopen("ss1.dat","w");
  tabela2 = fopen("ss2.dat","w");

  x[0]=10620.0;
  x[1]=49.0;
  x[2]=0.2939*2;
  x[3]=0.4045*0.02;    //originais

          for(t=passo;t<=MAX;t+=passo){
                for(i=0;i<4;i++) z[i]=x[i];

                x[0]+=passo*z[2];
                x[1]+=passo*z[3];
                x[2]+=passo*(-G*Mj*(z[0]-xj(t-passo))/pow(sqrt(pow(z[0]-xj(t-passo),2)+pow(z[1]-yj(t-passo),2)),
3)-.0*G*Ms*z[0]/pow(sqrt(pow(z[0],2)+pow(z[1],2)),3));
                x[3]+=passo*(-G*Mj*(z[1]-yj(t-passo))/pow(sqrt(pow(z[0]-xj(t-passo),2)+pow(z[1]-yj(t-passo),2)),
3)-.0*G*Ms*z[1]/pow(sqrt(pow(z[0],2)+pow(z[1],2)),3));

                  fprintf(tabela1,"%lf\t%lf\n",x[0],x[1]);
                      fprintf(tabela2,"%lf\t%lf\n",t,sqrt(x[3]*x[3]+x[2]*x[2])-0.0*.5*G*Mj/sqrt(pow(x[0]*x[0]+x[1]*x[1],
2)));
                                                  }

  fclose(tabela1);
  fclose(tabela2);
}

double xj(double t)
{
  return 10700*cos(0.66*t/10700.0);
}
double yj(double t)
{
  return 10700*sin(0.66*t/10700.0);
}
```