



Travaux Pratiques de Mécanique Analytique - MatLab

Simulation en temps réel du mouvement d'un pendule double

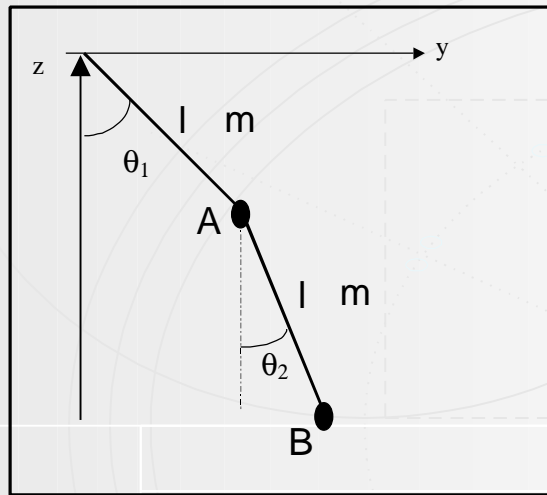
- ➔ Problème : pendule double
- ➔ Visualisation du mouvement



- ➔ Problème : pendule double
- ➔ Visualisation du mouvement



Mise en situation...



Positions:

Point A: $(l \sin \theta_1, -l \cos \theta_1)$

Point B: $(l.(\sin \theta_1 + \sin \theta_2), -l.(\cos \theta_1 + \cos \theta_2))$

Vitesses:

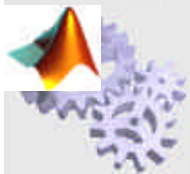
$(\theta_1)' \Rightarrow \dot{\theta}_1$

$(\theta_2)' \Rightarrow \dot{\theta}_2$

$$angle = \begin{bmatrix} \mathbf{q}_{1t1} & \mathbf{q}_{2t1} \\ \mathbf{q}_{1t2} & \mathbf{q}_{2t2} \\ \dots & \dots \\ \mathbf{q}_{1tn} & \mathbf{q}_{2tn} \end{bmatrix}$$

$\mathbf{P} \ [A_y \ A_z \ B_y \ B_z]$

plot (y , z) $\mathbf{P} \ (0, 0) \longrightarrow (A_y, A_z) \longrightarrow (B_y, B_z)$



- ➔ Problème : pendule double
- ➔ Visualisation du mouvement



Stratégie de de visualisation

- ⇒ dessin d'une position
- ⇒ boucle d'affichage
- ⇒ temps réel
- ⇒ **méthode d'affichage**
 - méthode clf-drawnnow
 - méthode effacement-dessin
 - méthode orienté objet



Premier affichage



```
%Allocation mémoire
```

```
y=zeros(3,1);
```

```
z=zeros(3,1);
```



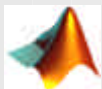
```
%Calcul de coordonnées
```

```
y(2)=1*sin(angle(1,1));
```

```
z(2)=-1*cos(angle(1,1));
```

```
y(3)=y(2)+1*sin(angle(1,2));
```

```
z(3)=z(2)-1*cos(angle(1,2));
```



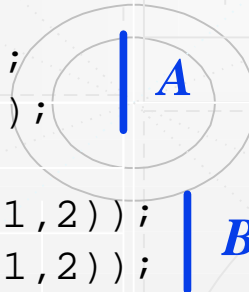
```
%Préparation session graphique et premier affichage
```

```
figure;
```

```
axis([-30 30 -30 30])
```

```
hold on;
```

```
line(y,z,'LineWidth',2);
```



Boucle d'affichage



```
%Boucle d'affichage  
for i=1:1000
```

```
    y(2)=1*sin(angle(i,1));  
    z(2)=-1*cos(angle(i,1));
```

A

```
    y(3)=y(2)+1*sin(angle(i,2));  
    z(3)=z(2)-1*cos(angle(i,2));
```

B

```
    line(y,z,'LineWidth',2);
```

```
end
```

```
--> dem3.m
```



Gestion dynamique



```
%Boucle d'affichage
```

```
for i=1:100 ←
```

```
    y(2)=1*sin(angle(i,1));
```

```
    z(2)=-1*cos(angle(i,1));
```

```
    y(3)=y(2)+1*sin(angle(i,2));
```

```
    z(3)=z(2)-1*cos(angle(i,2));
```

```
→ clf;
```

```
→ axis([-30 30 -30 30]);
```

```
    line(y,z, 'LineWidth',2);
```

```
→ drawnow;
```

```
--> dem4.m
```



Temps réel



```
%Boucle d'affichage  
for i=1:100
```

```
→ tic;
```

```
y(2)=1*sin(angle(i,1));  
z(2)=-1*cos(angle(i,1));
```

```
y(3)=y(2)+1*sin(angle(i,2));  
z(3)=z(2)-1*cos(angle(i,2));
```

```
→ while toc<0.1;
```

```
→ end;
```

```
clf;  
axis([-30 30 -30 30]);  
line(y,z, 'LineWidth',2);  
drawnow;
```

```
end
```

```
--> dem5.m
```



Gestion anciennes positions



`%Allocation mémoire`

`yold=zeros(3,1);` ←

`zold=zeros(3,1);` ←

`%Boucle d'affichage`

`for i=1:100`

`tic;`

→ `yold(2)=y(2);`

→ `zold(2)=z(2);`

→ `yold(3)=y(3);`

→ `zold(3)=z(3);`

`y(2)=1*sin(angle(i,1));`

`z(2)=-1*cos(angle(i,1));`

`y(3)=y(2)+1*sin(angle(i,2));`

`z(3)=z(2)-1*cos(angle(i,2));`

`while toc<0.1;`
`end;`

→ `plot(yold,zold,'w','LineWidth',2);`
`plot(y,z,'LineWidth',2);`
`drawnow;`

`end`

--> `dem6.m`



Approche "orienté-objet"



Objet graphique

- *Adresse*
- *Propriété 1*
- *Propriété 2*
- *Propriété ...*
- *Propriété n*

- *Instruction 1*
- *Instruction 2*
- *Instruction ...*
- *Instruction n*



Approche "orienté-objet"



Objet graphique

- Adresse
- XData
- YData
- ZData
- Color
- Style
- plot
- axis



Approche "orienté-objet"

```
graph1=plot(y,z,'w*', 'EraseMode', 'none');
```



Objet graphique

Adresse = 1.05684523 = graph1

XData = y

YData = z

ZData

Color = w

*Style = **

EraseMode = none

Instruction = plot

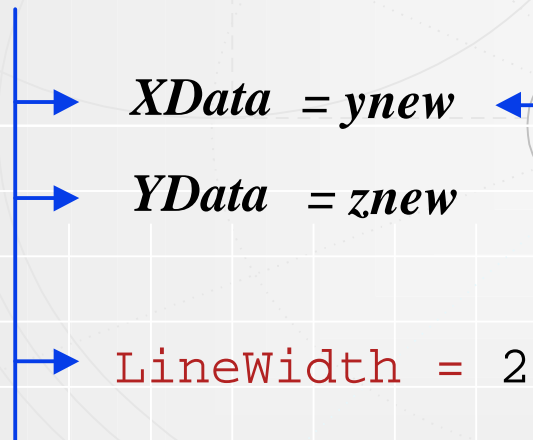


Approche "orienté-objet"




```
set(graph1, 'XData', ynew, 'YData', znew, 'LineWidth', 2);
```

graph1



Approche “orienté-objet”



```
%Préparation session graphique et premier affichage  
p=plot(y,z,'EraseMode','none');  
q=plot(yold,zold,'w','EraseMode','none');
```



```
%Boucle d'affichage  
set(q,'XData',yold,'YData',zold,'LineWidth',2);  
set(p,'XData',y,'YData',z,'LineWidth',2);
```

P dem7.m

