

```

// transport déformation d'une dune de fonds par un courant en eau peu profonde
// clin=1 pour transport linéaire et clin=0 pour transport non linéaire
clin=1
ctransp=1
clf()
g=9.81
// caractéristiques du domaine
xmin=0; xmax=12; L=xmax-xmin; ymin=0;
np=121 // nb de points du domaine
lint=L/(np-1)
x=0:L/(np-1):L;x0=x
// position et amplitude de la bosse
xbm=2;xbM=10;Lbosse=xbM-xbm;
nbxbm=xbm/lint+1;nbxbM=xbM/lint+1;nbmed=(nbxbm+nbxbM)/2
k=%pi/Lbosse; hbosse=0.1;
prof=0.3 // profondeur d'eau en amont et en aval
Zs=0.6
Zf=prof*ones(1,np);Zsurf=Zs*ones(1,np)
nbx=x/lint
arg=(nbxbm-nbxbM:nbxbM-nbxbm)*lint;
Zf(nbxbm:nbxbM)=0.3-hbosse*(sin(k.*arg)^2)
f=gcf()
// définit les dimensions de la figure
f.figure_position
f.figure_size=[1000,500]
//couleurs des aires des courbes
id1=color("white")
id2=color(0,191,255)
id3=color(160,64,0)

npas=30;dt=900

// boucle sur les pas de temps
//=====

for j=1:npas
    t=j*dt
    clf
    if clin==1 then ctransp=1.5/3600 // soit 1,5 m/h
    else
        ctransp=2*10^(-9)*(Zf^(-6.5))
    end
    // fixer la couleur du champ de vecteur à "blanc"
    f=gcf()
    f.figure_position; // définit les dimensions de la figure
    d=gca() //définition des axes    d.data_bounds=[xmin,0;xmax,2];
    x=x0+ctransp*t
    d.data_bounds=[xmin,0;xmax,0.6]
    d.labels_font_size=2;
    d.tight_limits="on";
    d.x_label.font_size = 3;
    d.y_label.text="profondeur en m"
    d.x_location="top"
    plot(x,Zsurf-Zf)
    d.data_bounds=[xmin,0;xmax,0.6]
    xfpolys([x';xmax;xmax;xmin;xmin],[((Zsurf-Zf)';prof;Zs;Zs;prof],[id2])
    xfpolys([x';xmax;xmax;xmin;xmin],[((Zsurf-Zf)';prof;ymin;ymin;prof],[id3])
    num=string(j/4)
    if(clin==1) then
        title('Propagation linéaire d'une forme de fond :...
            '+num+ ' heure(s)', 'position',[0 0.65], 'fontsize',5)
    else
        title('Propagation non-linéaire d'une forme de fond :...

```

```

    '+num+ ' heure(s)'),'position',[0 0.65],'fontsize',5)
end
//GIF export

if(clin==1) then
xs2gif(0,'Charriage dune linéaire'+string(j)+'.gif');
else
xs2gif(0,'Charriage dune non-linéaire'+string(j)+'.gif');
end
// arrêt des calculs en cas d'éboulement
arret="false"
for ij=2:np
    if(x(ij-1)>x(ij)) then arret="true" ;disp(t);break; end
end
if(arret=="true") then break; end
end
// dessin des caractéristiques
//g=gcf()

h=scf()
clf
g=gca()
t=3600*10
coef=2*10^(-9)
xc=zeros(1,2);yc=xc;
xp=[6,6.1,6.2,6.3,6.4,6.5,6.6,6.7,6.8,6.9,7,7.1,7.2,7.3,7.4,7.5,7.6,7.7,7.8,7.9,8]
nz=size(xp);
for i=1:nz(2)
    yc(1)=0;yc(2)=t;xc(1)=xp(i)
    np=round(xc(1)/lint+1)
    if(clin==1) then pent=ctransp
    else pent=coef*(Zf(np))^(-6.5)
    end
    xc(2)=xc(1)+yc(2)*pent
    plot(xc,yc)
end
title("Déformation non-linéaire d'une dune - droites caractéristiques",'fontsize',4)
g.x_label.text ="abscisse (m)";g.x_label.font_size = 3;
g.y_label.text="temps (s)";g.y_label.font_size = 3;
if(clin==1) then
    g.data_bounds=[6,0;8.6,10000]
    title('Déformation linéaire d'une dune - droites caractéristiques','fontsize',4)
else
    g.data_bounds=[6,0;8.6,35000]
    title('Déformation non-linéaire d'une dune - droites caractéristiques','fontsize',4)
end
h=gcf()
// définit les dimensions de la figure
h.figure_position
h.figure_size=[800,500]
xs2gif(1,'caractéristiques.gif');

```