

```

0001 // propagation d'onde en eaux peu profondes
0002 cas=0
0003 // Canal ouvert à droite (cas=0) ou stationnaire (cas=1)
0004 clf()
0005 clearglobal()
0006 g=9.81
0007 xmin=20000
0008 xmax=100000
0009 fact=1 // facteur d'échelle pour représentation des vecteurs vitesse
0010 L=xmax-xmin
0011 yminaffect=-100 //affiche la profondeur jusqu'à une profondeur de 50 m
0012 S0=0.001
0013 a0=7
0014 f=gcf()
0015 // définit les dimensions de la figure
0016 f.figure_position
0017 f.figure_size=[1500,500]
0018 //couleurs des aires des courbes
0019 id1=color("white")
0020 id2=color(0,191,255)
0021 TT=3600/4
0022 tmin=0
0023 tmax=TT
0024
0025 dx=L/360
0026 dt=TT/15
0027 n1=tmax/dt
0028 n=L/dx
0029 sigma=2*pi/TT
0030 k=sqrt(sigma^2/(g*S0)) //k des fonctions de Bessel
0031
0032 celerite0=sqrt(g*S0*xmin)
0033
0034 nbonde=sigma/celerite0
0035
0036 t=tmin
0037 xx=xmin:dx:xmax
0038 tt=tmin:dt:tmax
0039 // arguments
0040 arg0=2*k*sqrt(xmin)
0041 arg1=2*k*sqrt(xmax)
0042 if cas==0 then
0043     Calpha=-%i*sigma/celerite0
0044     Cbeta=k/(sqrt(xmin))
0045     // PROPAGATION DE L'AVANT VERS L'AMONT
0046     D=bessely(0,arg1)*(Cbeta*besselj(1,arg0)+Calpha*besselj(0,arg0))-...
0047     bessely(0,arg1)*(Cbeta*bessely(1,arg0)+Calpha*bessely(0,arg0))
0048     c1=-a0/D*(Cbeta*bessely(1,arg0)+Calpha*bessely(0,arg0))
0049     c2= a0/D*(Cbeta*besselj(1,arg0)+Calpha*besselj(0,arg0))
0050     arg=2*k*sqrt(xx)
0051     A2x=c1*besselj(0,arg)+c2*bessely(0,arg)
0052     //B2x=c1*besselj(1,arg)+c2*bessely(1,arg)
0053     Cmoy=(1/L)*(2/3)*sqrt(g*S0)*(xmax^(3/2)-xmin^(3/2)) //célérité moyenne de l'onde
0054     // boucle sur les pas de temps
0055     //=====
0056
0057     for j=1:n1
0058         clf
0059         d=gca() //définition des axes
0060         d.data_bounds=[xmin,yminaffect;L,20];
0061         d.labels_font_size=2;
0062         d.x_label.font_size = 3;d.y_label.font_size = 3;
0063         d.x_label.text="abscisse en m"
0064         d.y_label.text="profondeur en m"
0065         // fixer la couleur du champ de vecteur à "blanc"
0066         f=gcf()
0067         // définit les dimensions de la figure
0068         f.figure_position
0069         f.figure_size=[1500,500]
0070         t=tt(j)
0071         h=real(A2x.*exp(-%i*sigma*t))
0072         plot(xx/fact,h,id2)
0073         xfpolys([xx'/fact;xmax/fact;xmin],[h';-xmax*S0;-xmin*S0],[id2])

```

```

0074         num=string(j)
0075         title('Propagation d'une onde en estuaire pente constante, section constante: '+num+ '
minute(s)', 'position', [25000 25], 'fontsize', 5)
0076         //GIF export
0077         xs2gif(0, 'Estuaire propagation'+string(j)+'.gif');
0078     end
0079     // REFLEXION TOTALE EN AMONT
0080     //=====
0081 else
0082     D=bessely(1,arg0)*besselj(0,arg1)-bessely(0,arg1)*besselj(1,arg0)
0083     c1=a0/D*bessely(1,arg0)
0084     c2=-a0/D*besselj(1,arg0)
0085     arg=2*k*sqrt(xx)
0086     // boucle sur les pas de temps
0087     //=====
0088     for j=1:n1
0089         clf
0090         d=gca() //définition des axes
0091         d.data_bounds=[xmin,yminaffich;L,20];
0092         d.labels_font_size=2;
0093         d.x_label.font_size = 3;d.y_label.font_size = 3;
0094         d.x_label.text="abscisse en m"
0095         d.y_label.text="profondeur en m"
0096         // fixer la couleur du champ de vecteur à "blanc"
0097         f=gcf()
0098         // définit les dimensions de la figure
0099         f.figure_position
0100         f.figure_size=[1500,500]
0101         t=tt(j)
0102         h=real((c1*besselj(0,arg)+c2*bessely(0,arg))*exp(-%i*sigma*t))
0103         plot(xx/fact,h,id2)
0104         xfpolys([xx'/fact;xmax/fact;xmin],[h';-xmax*S0;-xmin*S0],[id2])
0105         num=string(j)
0106         title('Réflexion d'une onde en estuaire pente constante, section constante: '+num+ '
minute(s)', 'position', [25000 25], 'fontsize', 5)
0107         //GIF export
0108         xs2gif(0, 'Estuaire reflexion'+string(j)+'.gif');
0109     end
0110 end

```