

```

0001 // Diffraction autour d'une île
0002 clf()
0003 scf(1)
0004 g=9.81
0005 a=25 // rayon de l'île
0006 b=200 // rayon frontière extérieure
0007 T=10 // période de l'onde incidente
0008 H=200 // profondeur d'eau
0009 h=1 // hauteur de houle incidente
0010 sigma=2*pi/T // fréquence
0011 // calcul du nombre d'onde par itération
0012 function delta=dispersion(k, sigma, g, H)
0013     delta=sigma^2-(g*k*tanh(k*H))
0014 endfunction
0015 // résolution de l'équation de diffusion
0016 function k=resdisp(g, T, H)
0017     sigma=2*pi/T;
0018     k1=0; // minimum
0019     k2=1000; // maximum
0020     delta=dispersion(k1,sigma,g,H)
0021     if(delta<0) then kmin=k1;kmax=k2;
0022     else kmin=k2;kmax=k1;
0023     end
0024     while abs(delta)>0.001
0025         k=(kmin+kmax)/2;
0026         delta=dispersion(k,sigma,g,H)
0027         if(delta<0) then kmin=k;
0028         else kmax=k;
0029         end
0030     end
0031 endfunction
0032 // dérivée de fonction de Bessel
0033 function jp=jprim(n, x)
0034     jp=0.5*((besselj(n-1,x)-(besselj(n+1,x))))
0035 endfunction
0036 // dérivée de fonction de Hankel
0037 function hp=hprim(n, x)
0038     hp=0.5*((besselh(n-1,x)-(besselh(n+1,x))))
0039 endfunction
0040 function ile(Radius, Height, SideCount)
0041     // Vertices
0042     n_side = SideCount;
0043     for i_ver=1:n_side
0044         VertexData(i_ver,:) = [Radius*cos(2*pi/n_side*i_ver),Radius*sin(2*pi/n_side*i_ver),0];
0045     end
0046     VertexData(n_side+i_ver,:) = [Radius*cos(2*pi/n_side*i_ver),Radius*sin(2*pi/n_side*i_ver),Height];
0047     // Side Patches
0048     for i_pat=1:n_side-1
0049         Index_Patch1(i_pat,:) = [i_pat,i_pat+1,i_pat+1+n_side,i_pat+n_side];
0050     end
0051     Index_Patch1(n_side,:) = [n_side,1,1+n_side,2*n_side];
0052     for i_pat=1:n_side
0053         // Side patches data
0054         PatchData1_X(:,i_pat) = VertexData(Index_Patch1(i_pat,:),1);
0055         PatchData1_Y(:,i_pat) = VertexData(Index_Patch1(i_pat,:),2);
0056         PatchData1_Z(:,i_pat) = VertexData(Index_Patch1(i_pat,:),3);
0057     end
0058     // descente du cylindre
0059     PatchData1_Z= PatchData1_Z-1
0060     // Draw side patches
0061
0062     // définition de la couleur et plot3d (couleur sur le dessus de l'île)
0063     id=color('gray82')
0064     plot3d(PatchData1_X,PatchData1_Y,PatchData1_Z,20,20," ",[-id 0 0]);
0065     hs=gce()
0066     hs.color_flag=1
0067
0068     // Bottom Patches
0069     Index_Patch2(1,:) = [1:n_side];
0070     Index_Patch2(2,:) = [n_side+1:2*n_side];
0071     hl_fac3d = gce();
0072     hl_fac3d.color_mode =2;

```

```

0073     hl_fac3d.foreground = 1;
0074     hl_fac3d.hiddencolor = id;
0075
0076     for i_pat=1:2
0077
0078         // Bottom patch data
0079         PatchData2_X(:,i_pat) = VertexData(Index_Patch2(i_pat,:),1);
0080         PatchData2_Y(:,i_pat) = VertexData(Index_Patch2(i_pat,:),2);
0081         PatchData2_Z(:,i_pat) = VertexData(Index_Patch2(i_pat,:),3);
0082     end
0083 endfunction
0084 // dimension du vecteur SIGPHI
0085 nr=51; // nombre de points sur chaque rayon
0086 nt=51; // nombre d'angles (répartition sur [0:2*pi])
0087 nrmin=a;nrmax=b;dr=(nrmax-nrmin)/(nr-1)
0088 ntmin=0;ntmax=2*pi;dt=(ntmax-ntmin)/(nt-1)
0089 // initialisation des vecteurs des coordonnées
0090 SIGPHI=zeros(nr,nt);xpts=zeros(nr,nt);ypts=zeros(nr,nt)
0091 k=resdisp(g,T,H)
0092 arga=k*a
0093 PHI=0
0094 it=0
0095 //
0096 // CALCUL DE LA HAUTEUR DE HOULE
0097 //
0098 itheta=0
0099 // boucle sur les angles
0100 for theta=ntmin:dt:ntmax
0101     itheta=itheta+1
0102     ir=0
0103     // Boucle sur les rayons
0104     for r=nrmin:dr:nrmax
0105         ir=ir+1
0106         PHI=0
0107         argr=k*r
0108         // boucle sur les fréquences
0109         PHI=besselj(0,argr)*hprim(0,arga)-jprim(0,arga)*besselh(0,argr)
0110         PHI=PHI/hprim(0,arga)
0111         SIGPHI(ir,itheta)=PHI
0112
0113         for n=1:100
0114             PHI=besselj(n,argr)*hprim(n,arga)-jprim(n,arga)*besselh(n,argr)
0115             PHI=PHI*2*(%i^n)*cos(n*theta)/hprim(n,arga)
0116             SIGPHI(ir,itheta)=SIGPHI(ir,itheta)+PHI
0117             xpts(ir,itheta)=r*cos(theta)
0118             ypts(ir,itheta)=r*sin(theta)
0119         end
0120     end
0121 end
0122 // BOUCLE SUR LE TEMPS
0123 //
0124 it=0
0125 for t=0:0.5:10
0126     it=it+1
0127     vfunct=real((SIGPHI)*exp(-%i*sigma*t))
0128     // choix de la palette de couleurs
0129     xset("colormap", wintercolormap(128) )
0130     // tracé de la surface libre
0131     surf(xpts,ypts,vfunct)
0132     hh=gce()
0133     hh.color_flag=1; //change the figure colormap
0134     hh.color_mode=-2; //color according to z
0135     hh.color_flag=3; //remove the facets boundary
0136     // tracé de l'île
0137     ile(25,5,20)
0138     // Ecriture du titre
0139     format('v',8)
0140     titre='DIFFRACTION autour d'une île : t = '+string(t)+' s, prof.'+string(H)+'m'
0141     a=gca()
0142     a.box='off'
0143     a.rotation_angles=[50 160]; // angle solide de vue
0144     a.parent.figure_size= [800,800]; // dimensions de la figure
0145     a.title.text = (string(titre)); // titre de la figure

```

```

0146     a.title.font_size = 5;           // taille de la police du titre
0147     a.data_bounds=[-200, -200, -4; 200, 200,4] // limites des axes
0148     a.margins=[0.1, 0.1, 0.1, 0.1]; // limites des marges du graphique
0149     // impression des fichiers images
0150     format('v',4)
0151     iprint='ILE_'+string(H)+'m_'+string(it)'+'.png'
0152     xs2png(1,iprint)
0153     clf()
0154     // fin de la boucle en temps
0155 end
0156 // cartographie des hauteurs de houle
0157 scf(2)
0158 vhoule=abs(SIGPHI)
0159 xset("colormap", jetcolormap(128) )
0160 surf(xpts,ypts,vhoule)
0161 hh=gce()
0162 hh.color_flag=1; //change the figure colormap
0163 hh.color_mode=-2; //color according to z
0164 hh.color_flag=3; //remove the facets boundary
0165 iprint='Hauteur de houle'+string(H)+'m.png'
0166 ile(25,5,20)
0167
0168 // Ecriture du titre
0169 titre='Hauteur de houle - profondeur '+string(H)+'m'
0170 a=gca()
0171 a.box='off'
0172 a.rotation_angles=[0 0]; // angle solide de vue
0173 a.parent.figure_size= [850,800]; // dimensions de la figure
0174 a.title.text = (string(titre)); // titre de la figure
0175 a.title.font_size = 5; // taille de la police du titre
0176 a.data_bounds=[-200, -200, -4; 200, 200,4] // limites des axes
0177 a.margins=[0.1, 0.1, 0.1, 0.1]; // limites des marges du graphique
0178 a.axes_visible = ["off","off","off"]
0179 // représentation de l'île - construction d'un cylindre qui représente l'île
0180 colorbar(min(vhoule),max(vhoule))
0181 isoview="on"
0182 xs2png(2,iprint)

```