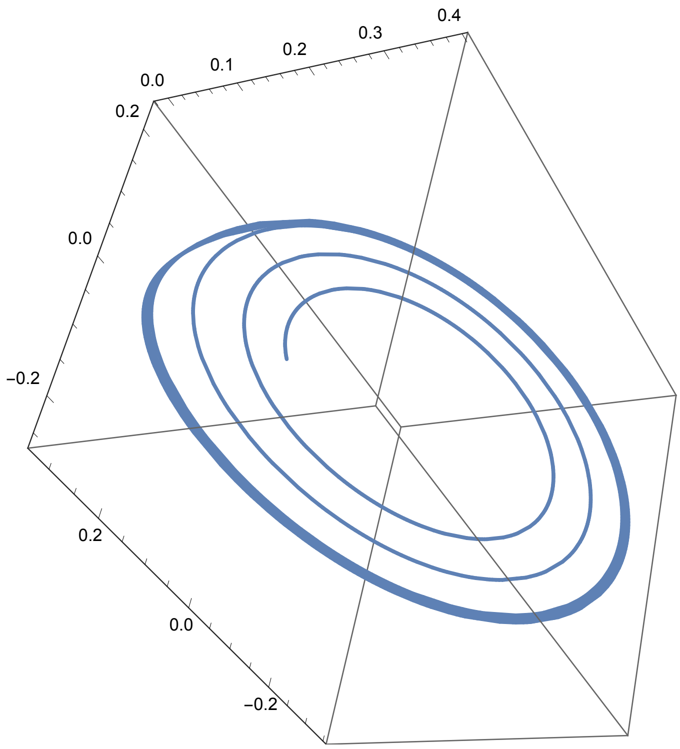
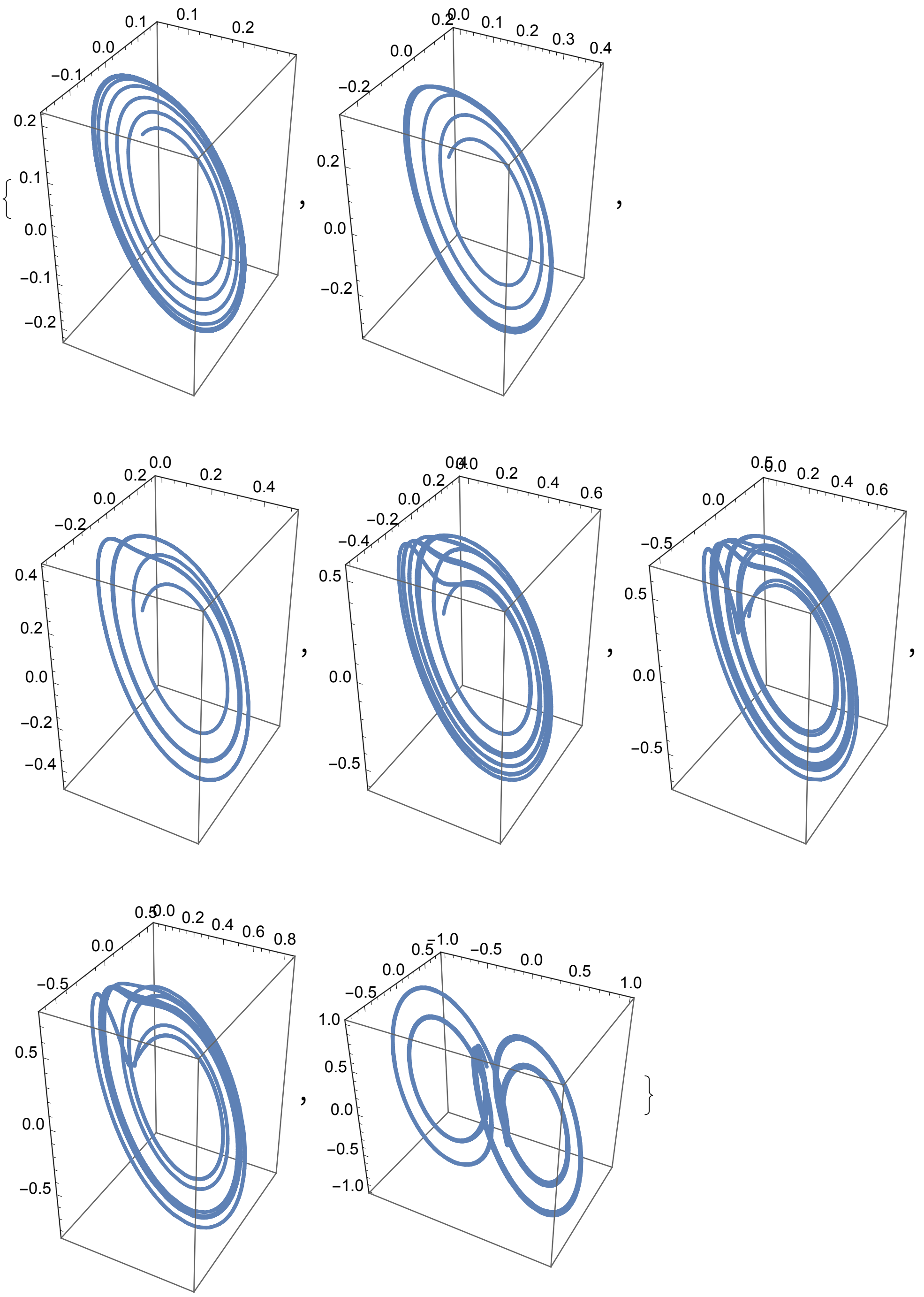
```
In[51]:= (* synthese inverse a partir d'une ode interessante *)

In[52]:= odeChaos = {x1'[t] == x2[t], x2'[t] == x3[t],
           x3'[t] == -2.9 x1[t] - 1.8 x2[t] - 1.1 x3[t] + 0.5 ArcTan[20 x1[t]]}

Out[52]= {x1′[t] == x2[t], x2′[t] == x3[t],
          x3′[t] == 0.5 ArcTan[20 x1[t]] - 2.9 x1[t] - 1.8 x2[t] - 1.1 x3[t]}

In[53]:= xx = {x1[t], x2[t], x3[t]}

Out[53]= {x1[t], x2[t], x3[t]}

In[54]:= odeSol = NDSolve[
           Join[odeChaos, {x1[0] == 0.1, x2[0] == 0.01, x3[0] == 0.1}], xx, {t, 0, 400}]

Out[54]= {{x1[t] → InterpolatingFunction[ Domain: {{0., 400.}} Output: scalar ][t],
           x2[t] → InterpolatingFunction[ Domain: {{0., 400.}} Output: scalar ][t],
           x3[t] → InterpolatingFunction[ Domain: {{0., 400.}} Output: scalar ][t]}}

In[55]:= ParametricPlot3D[xx /. odeSol, {t, 0, 40}]
```

Out[55]=



In[56]:=
```
mesPlotes = Module[{odeChaos, odeSol, NN},
    NN = 50;
    odeChaos = {x1'[t] == x2[t], x2'[t] == x3[t],
      x3'[t] == -2.9 x1[t] - 1.8 x2[t] - 1.1 x3[t] + # ArcTan[20 x1[t]]};
    odeSol = NDSolve[Join[odeChaos, {x1[0] == 0.1, x2[0] == 0.01, x3[0] == 0.1}],
      xx, {t, 0, NN}];
    ParametricPlot3D[xx /. odeSol, {t, 0, NN}]] & /@
  {0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0}
```

Out[56]=



In[57]:=
```
Gs = - (1 / d) 1 / (s^3 + c s^2 + b s + a)
```

Out[57]= $-\frac{1}{d\left(a+b\,s+c\,s^2+s^3\right)}$

In[58]:=
```
ValNum = {a → 2.9, b → 1.8, c → 1.1}
```

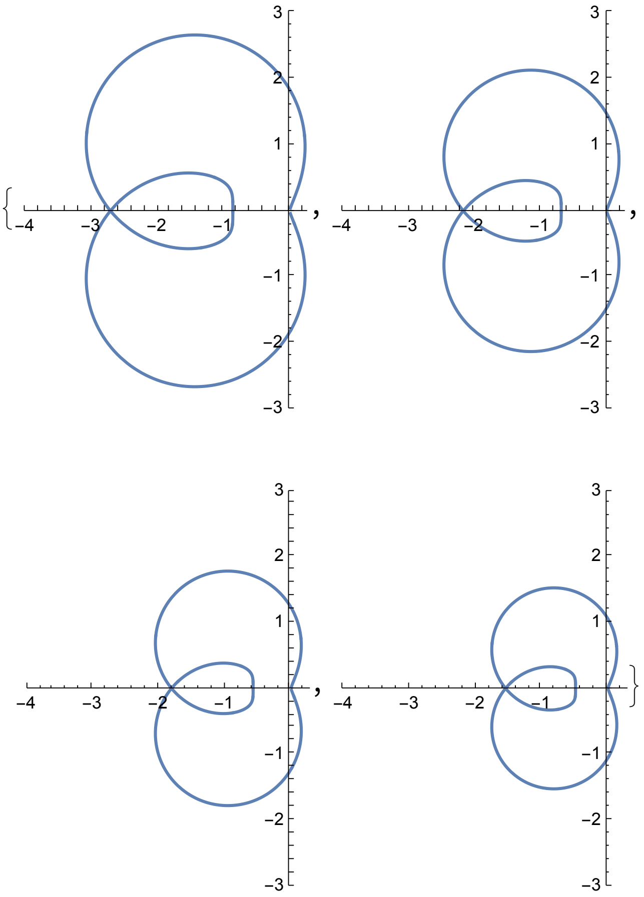
Out[58]= $\{a \to 2.9,\ b \to 1.8,\ c \to 1.1\}$

In[59]:=
```
Gjw = Gs /. {s → I w} /. ValNum
```

Out[59]= $-\frac{1}{d\left(2.9+(0.+1.8\,i)\,w-1.1\,w^2-i\,w^3\right)}$

In[60]:=
```
ParametricPlot[{Re[Gjw], Im[Gjw]} /. {d → #}, {w, -4, 4},
    PlotRange → {{-4, 0.3}, {-3, 3}}] & /@ {0.4, 0.5, 0.6, 0.7}
```

Out[60]=



In[61]:=
```
Gjw
```

Out[61]= $-\frac{1}{d\left(2.9+(0.+1.8\,i)\,w-1.1\,w^2-i\,w^3\right)}$

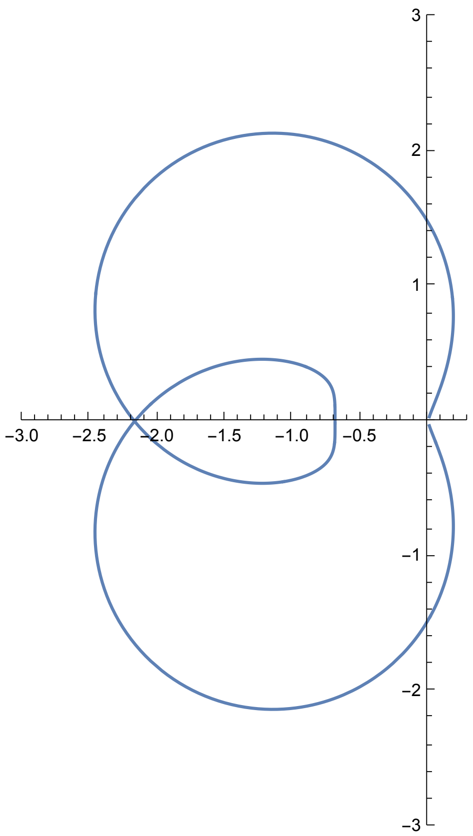
In[62]:=
```
Gs /. ValNum /. {s → I w}
```

Out[62]= $-\frac{1}{d\left(2.9+(0.+1.8\,i)\,w-1.1\,w^2-i\,w^3\right)}$

In[63]:=
```
NyquistPlot = ParametricPlot[{Re[Gjw], Im[Gjw]} /. {d → #}, {w, -4, 4},
    PlotRange → {{-3, 0.3}, {-3, 3}}, PlotStyle -> Thickness[0.007]] & @@ {0.5}
```

··· Set: Symbol NyquistPlot is Protected.

Out[63]=



In[64]:=
```
Gjwa = Gs /. {s → I w}
```

Out[64]= $-\frac{1}{d\left(a + i\, b\, w - c\, w^2 - i\, w^3\right)}$

In[65]:=
```
ff = (a - c w^2) + (w^3 - b w) I
```

Out[65]= $a - c\, w^2 + i\left(-b\, w + w^3\right)$

In[66]:=
```
ddg = Simplify[Expand[Denominator[Gjwa] ff]]
```

Out[66]= $d\left(a^2 + b^2 w^2 - 2\, a\, c\, w^2 - 2\, b\, w^4 + c^2 w^4 + w^6\right)$

In[67]:=
```
nng = Numerator[Gjwa] ff
```

Out[67]= $-a + c\, w^2 - i\left(-b\, w + w^3\right)$

In[68]:=
```
(* la partie imaginaire s'annule pour w^3 - b w = 0, c.-a-d. *)
```

In[69]:=
```
Annule = {w -> Sqrt[b]}
```

Out[69]= $\left\{w \to \sqrt{b}\right\}$

In[70]:=
```
z1  = nng / ddg /. Annule
```

Out[70]= $\frac{-a + b\, c}{\left(a^2 - 2\, a\, b\, c + b^2 c^2\right) d}$

In[71]:= `z1 /. ValNum`

Out[71]= $-\frac{1.08696}{d}$

In[72]:= `Gjw /. {d → 0.5, w → (Sqrt[b] /. ValNum)}`

Out[72]= $-2.17391 + 0.\,i$

In[73]:= `(* la frequence du cycle limite est predite par w qui ne depend que de \sqrt[b] *)`

In[74]:= `(* le gain equivalent d'un relais *)`

In[75]:= `A = - 2 z1`

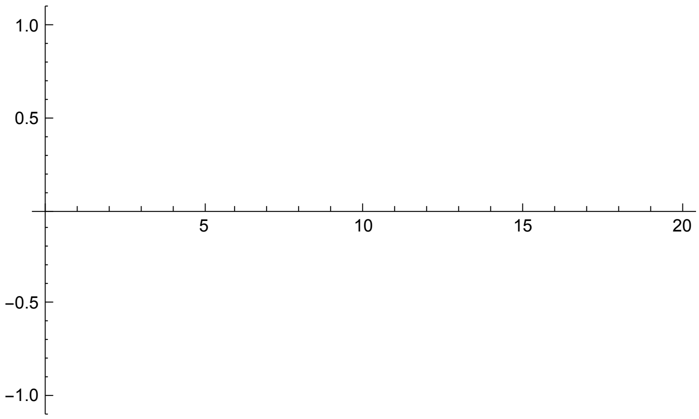
Out[75]= $-\frac{2\,(-a + b\,c)}{\left(a^2 - 2\,a\,b\,c + b^2\,c^2\right)\,d}$

In[76]:= `A /. ValNum /. {d → 0.5}`

Out[76]= 4.34783

In[77]:= `Plot[Evaluate[A Sin[Sqrt[b] t] /. ValNum], {t, 0, 20}]`

Out[77]=

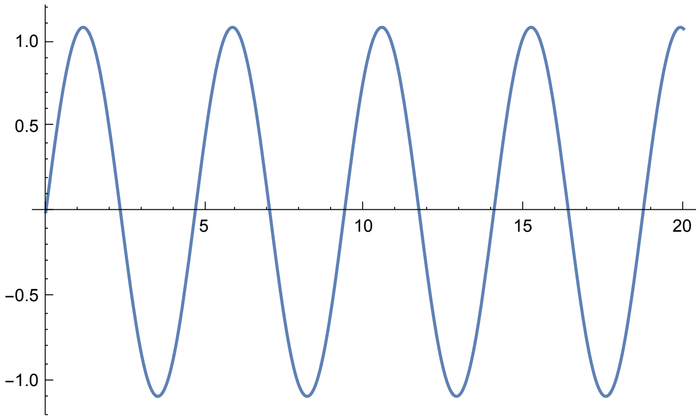


In[78]:= `Sqrt[b] /. ValNum`

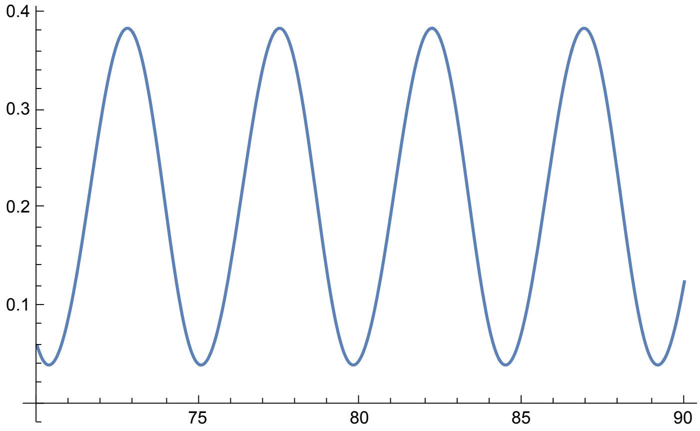
Out[78]= 1.34164

In[79]:= `ondePredite = Plot[1.08696 Sin[1.34164 t], {t, 0, 20}]`

Out[79]=



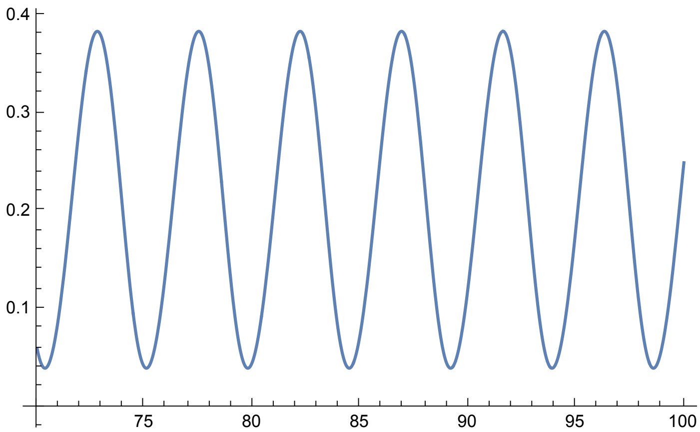
In[80]:=
```
ondeSimul = Plot[x1[t] /. odeSol, {t, 70, 90}]
```

Out[80]=


In[81]:=
```
2 1 / (1 / 0.3) / (a - b c) /. ValNum
```

Out[81]= 0.652174

In[82]:=
```
ondeChaotique = Plot[x1[t] /. odeSol, {t, 70, 100}]
```

Out[82]=


In[83]:=
```
1 / 0.4
```

Out[83]= 2.5