Lorenz Model

APPH 4200 Physics of Fluids
Columbia University

Introduction

In 1963, Edward Lorenz modeled the nonlinear thermal convection in 2D. This is the Bénard Instability. Lorenz assumed spatial forms for the convective flow (*i.e.* "rolls"), the temperature difference between rising and falling flows, and the distortion of the average temperature gradient. With this assumed spatial forms, the temporal amplitudes defined a coupled set of nonlinear ordinary differential equations. When the instability drive exceeded a threshold, the temporal solutions became chaotic. This illustrates a route to turbulence.

■ Assummed Spatial Forms

In[1]:=
```
ψ[tX_, x_, z_] := tX Cos[π z] Sin[2 π x];
δT[tY_, tZ_, x_, z_] := tY Cos[π z] Cos[2 π x] + tZ Sin[2 π z]
```

Lorenz Equations

The Lorenz model has three "free" parameters: **pr** (the Prandtl number), **r** (the instability drive), and **b** = 4/5.

In[3]:=
```
eqs = {D[tX[t], t] == pr (tY[t] - tX[t]),
  D[tY[t], t] == - tX[t] tZ[t] + r tX[t] - tY[t],
  D[tZ[t], t] == tX[t] tY[t] - b tZ[t]}
```

Out[3]=
```
{tX′[t] == pr (-tX[t] + tY[t]),
 tY′[t] == r tX[t] - tY[t] - tX[t] tZ[t], tZ′[t] == tX[t] tY[t] - b tZ[t]}
```

Solution (Chaotic)

In[4]:=
```
tmax = 200.0;
```

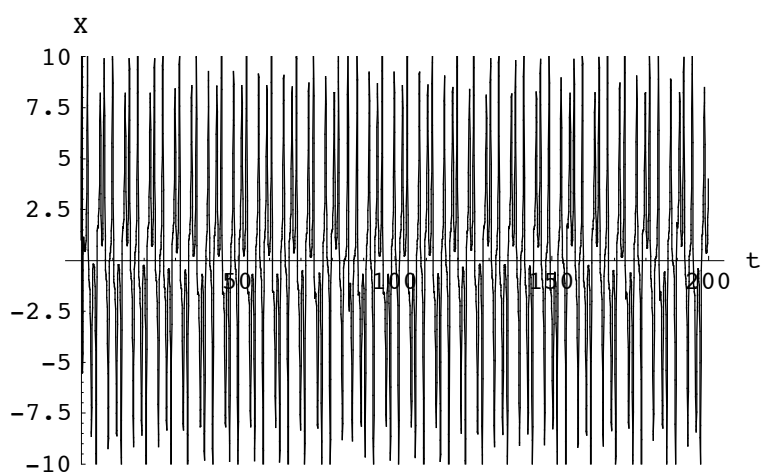
In[5]:=
```
sol = NDSolve[
  eqs ~Join~ {tX[0] == 1, tY[0] == 1, tZ[0] == 1} /. b → 4 / 5 /. pr → 10 /.
   r → 28, {tX[t], tY[t], tZ[t]}, {t, 0, tmax}, MaxSteps → 20000]
```

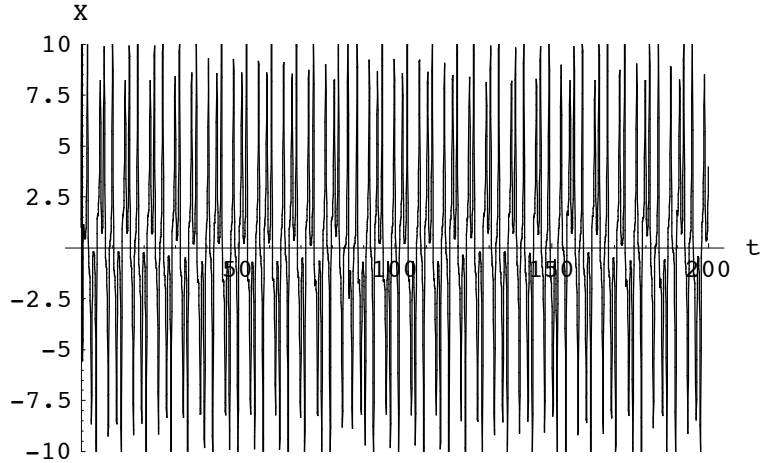
Out[5]=
```
{{tX[t] → InterpolatingFunction[{{0., 200.}}, <>][t],
  tY[t] → InterpolatingFunction[{{0., 200.}}, <>][t],
  tZ[t] → InterpolatingFunction[{{0., 200.}}, <>][t]}}
```

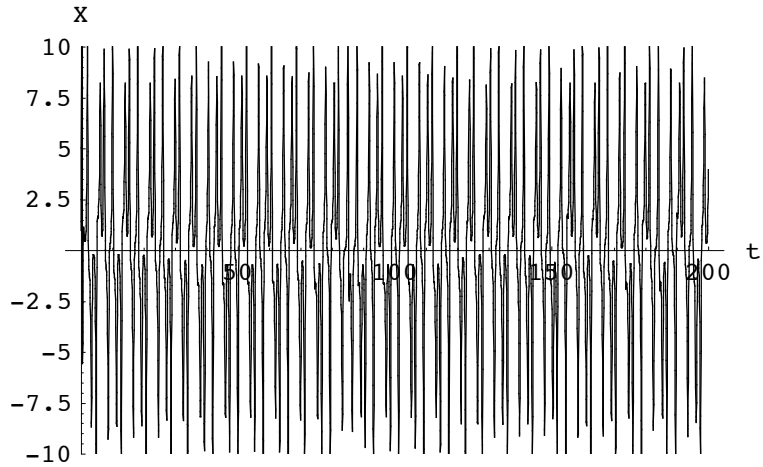
Out[16]=
```
{{tX[t] → InterpolatingFunction[{{0., 200.}}, <>][t],
  tY[t] → InterpolatingFunction[{{0., 200.}}, <>][t],
  tZ[t] → InterpolatingFunction[{{0., 200.}}, <>][t]}}
```

Graphing the Solution

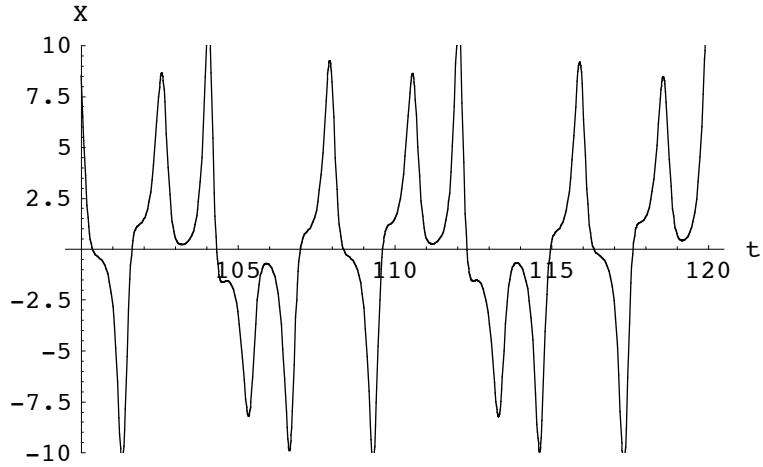
In[6]:=
```
Plot[Evaluate[tX[t] /. First[sol]], {t, 0, tmax},
  PlotPoints → 100, AxesLabel → {"t", "X"}, PlotRange → {-10, 10}];
```

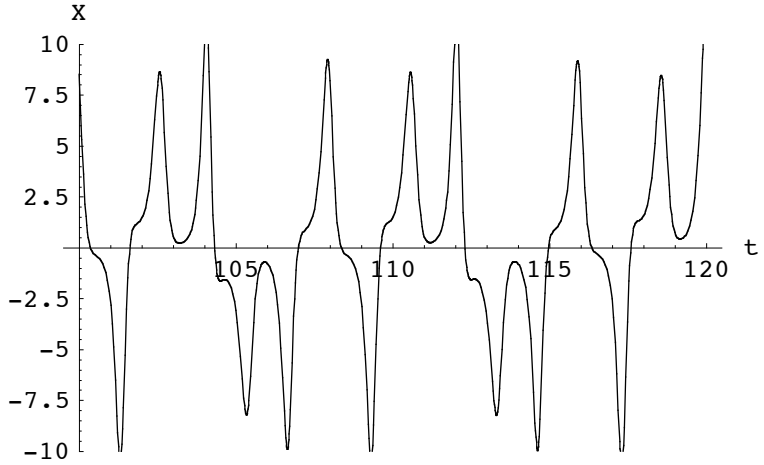


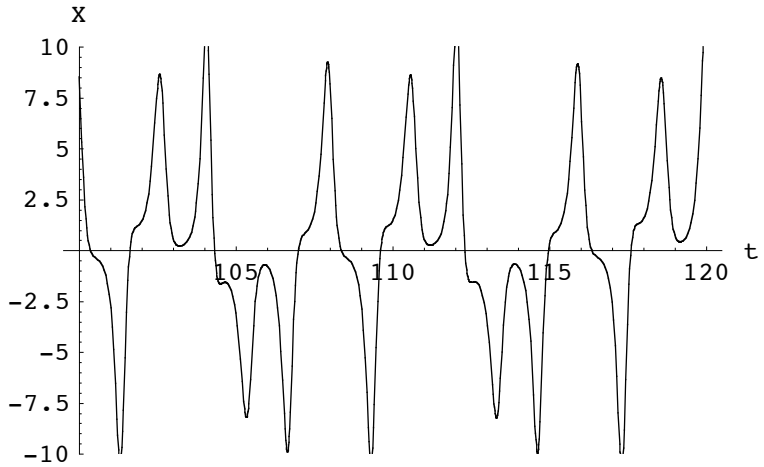




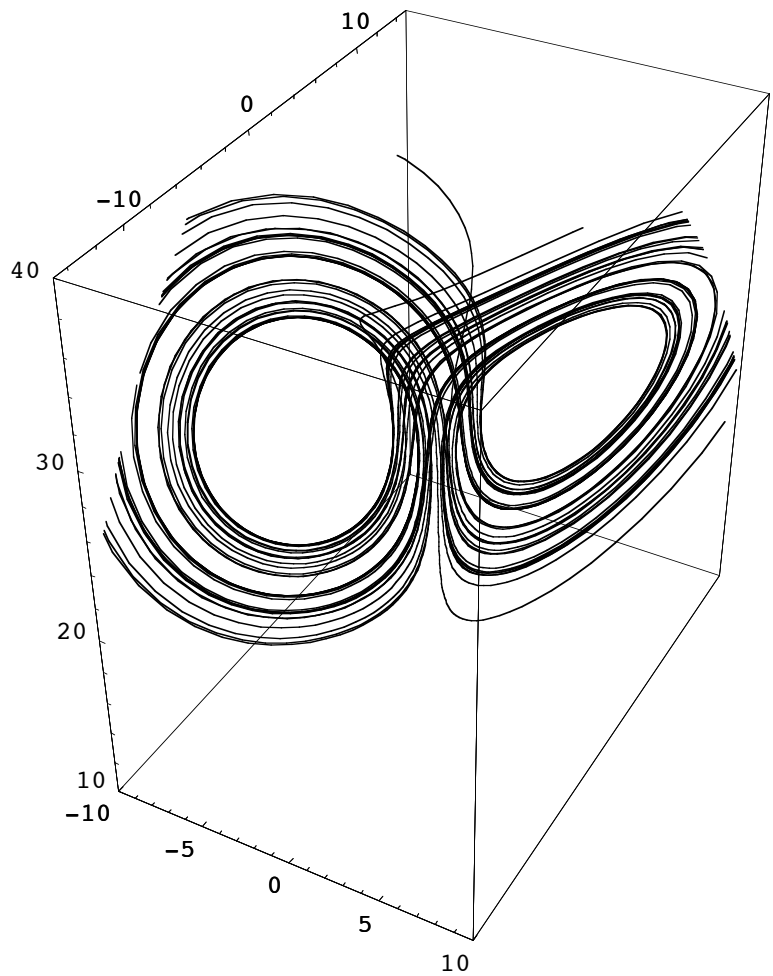
In[7]:=
```
Plot[Evaluate[tX[t] /. First[sol]], {t, tmax / 2, tmax / 2 + 20},
  PlotPoints → 100, AxesLabel → {"t", "X"}, PlotRange → {-10, 10}];
```




x
10
7.5
5
2.5
-2.5
-5
-7.5
-10
105
110
115
120
t


x
10
7.5
5
2.5
-2.5
-5
-7.5
-10
105
110
115
120
t

In[8]:=
```
ParametricPlot3D[Evaluate[{tX[t], tY[t], tZ[t]} /. sol], {t, 0, tmax / 2},
  PlotPoints → 5000, PlotRange → {{-10, 10}, {-15, 15}, {10, 40}}];
```



Above threshold, the convection pattern becomes chaotic (or "turbulent") in time!

Stable Solution (Reduced Drive)

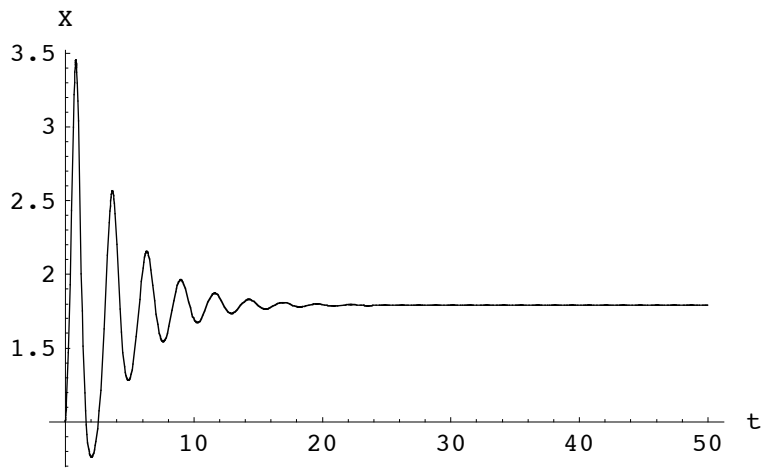
In[9]:=
```
tmax = 50.0;
```

In[10]:=
```
sol = NDSolve[
  eqs ~Join~ {tX[0] == 1, tY[0] == 1, tZ[0] == 1} /. b → 4 / 5 /. pr → 10 /.
   r → 5, {tX[t], tY[t], tZ[t]}, {t, 0, tmax}, MaxSteps → 20000]
```

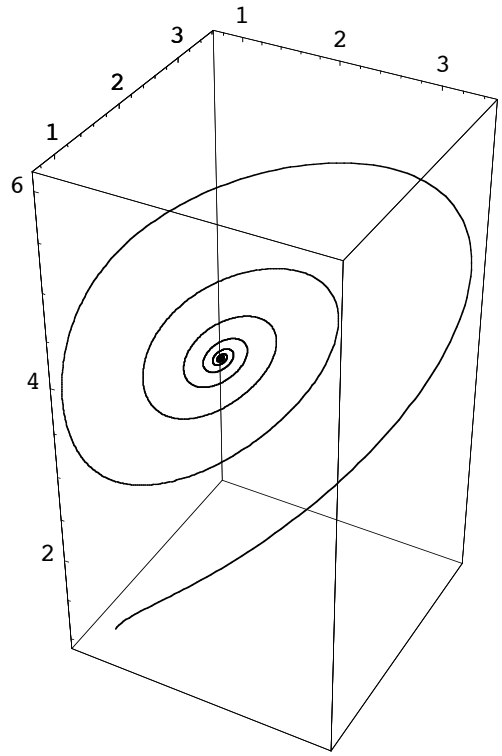
Out[10]=
```
{{tX[t] → InterpolatingFunction[{{0., 50.}}, <>][t],
  tY[t] → InterpolatingFunction[{{0., 50.}}, <>][t],
  tZ[t] → InterpolatingFunction[{{0., 50.}}, <>][t]}}
```

■ Graphing the Solution

In[11]:=
```
Plot[Evaluate[tX[t] /. First[sol]], {t, 0, tmax},
  PlotPoints → 100, AxesLabel → {"t", "X"}, PlotRange → All];
```



In[12]:=
```
ParametricPlot3D[Evaluate[{tX[t], tY[t], tZ[t]} /. sol],
  {t, 0, tmax}, PlotPoints → 5000, PlotRange → All];
```



In[13]:=
```
({tX[t], tY[t], tZ[t]} /. First[sol]) /. t → tmax
```

Out[13]=
```
{1.78885, 1.78885, 4.}
```

Below threshold, the convection pattern becomes steady!

Summary

Lorenz Model illustrates the role of nonlinearity in the devlopment of instability. When the drive for instability is below a threshold, the instability develops into a steady convection cell pattern. When the drive exceeds a threshold, chaotic fluid dynamics develops.