# 4.1a

In[768]:=
```
ClearAll["Global`*"]


sigma = 10;
b = 8 / 3;
r = 28;

bC = N[(sigma (sigma + 4) / (sigma - 2))];
rHopf = N[sigma * ((sigma + b + 3) / (sigma - b - 1))];

(*Fixed points for the system*)
FixedPoint1 = {0, 0, 0};
FixedPoint2 = {Sqrt[b (r - 1)], Sqrt[b (r - 1)], r - 1};
FixedPoint3 = {-Sqrt[b (r - 1)], -Sqrt[b (r - 1)], r - 1};

f1 = sigma (y - x);
f2 = r * x - y - x * z;
f3 = x * y - b * z;

Jacobi = D[{f1, f2, f3}, {{x, y, z}}];

(*Fix point 1*)
x = 0;
y = 0;
z = 0;

Eig = N[Eigenvalues[Jacobi]]

(*Fix point 2*)
x = Sqrt[b (r - 1)];
y = Sqrt[b (r - 1)];
z = r - 1;

Eig = N[Eigenvalues[Jacobi]]

(*Fix point 3*)
x = -1 * Sqrt[b (r - 1)];
y = -1 * Sqrt[b (r - 1)];
z = (r - 1);

Eig = N[Eigenvalues[Jacobi]]
```

Out[784]=
```
{-22.8277, 11.8277, -2.66667}
```

Out[788]=
```
{-13.8546, 0.0939556 + 10.1945 i, 0.0939556 - 10.1945 i}
```

Out[792]=

$\{-13.8546, 0.0939556 + 10.1945\, i, 0.0939556 - 10.1945\, i\}$

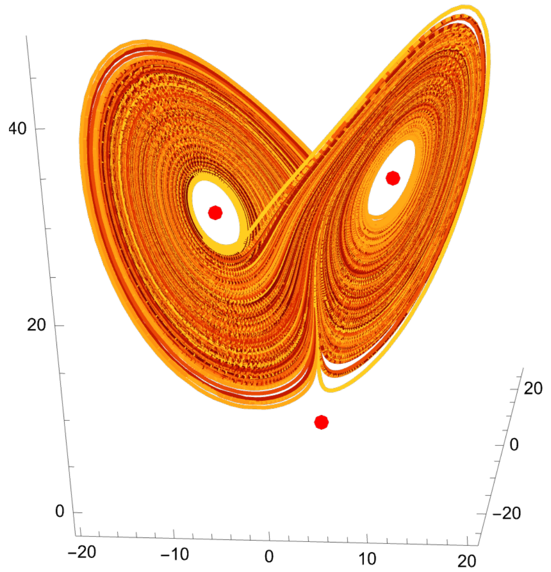
To be a stabel system in all directions all eigenvalues real part needs to be negative. Therfore none of the above is stabel in all directions

# 4.1 b

In[652]:=

```
ClearAll["Global`*"]

(*Setting up the Lorenz variables*)
sigma = 10;
r = 28;
b = 8 / 3;

(*Establishing the differential equation system*)
eq1 = x'[t] == sigma * (y[t] - x[t]);
eq2 = y'[t] == r * x[t] - y[t] - x[t] * z[t];
eq3 = z'[t] == x[t] * y[t] - b * z[t];
system = {eq1, eq2, eq3};

(*Determining fixed points for the system of equations*)
fixedPoint1 = {0, 0, 0};
fixedPoint2 = {Sqrt[b (r - 1)], Sqrt[b (r - 1)], r - 1};
fixedPoint3 = {-Sqrt[b (r - 1)], -Sqrt[b (r - 1)], r - 1};

(*Parameters for the plot*)
t0 = 0;
tMax = 1000;
tZeroPlot = 10;

(*Defining the starting point for the trajectory*)
startingPoint = {x[0] == 0, y[0] == 0.01, z[0] == 0.01};

(*Solving the differential equations*)
solution =
  NDSolve[{system, startingPoint}, {x, y, z}, {t, t0, tMax}, MaxSteps → ∞];

(*Plotting the parametric representation*)
plot = ParametricPlot3D[Evaluate[{x[t], y[t], z[t]} /. solution],
   {t, tZeroPlot, 500}, PlotPoints → 1000, ColorFunction →
    (ColorData["SolarColors", #4] &), PlotRange → All, Boxed → False];

(*Displaying the fixed points on the plot*)
fixedPts = Graphics3D[
   {Red, PointSize[0.03], Point[{fixedPoint1, fixedPoint2, fixedPoint3}]}];
Show[plot, fixedPts]
```

Out[670]=




# 4.1 c)

```
ClearAll["Global`*"]

eq1 = sigma * (y - x);
eq2 = r * x - y - x * z;
eq3 = x * y - b * z;
system = {eq1, eq2, eq3};

jacobiMatrix = D[system, {{x, y, z}}];
MatrixForm[jacobiMatrix]
```

Out[690]//MatrixForm=

$$\begin{pmatrix} -\text{sigma} & \text{sigma} & 0 \\ r - z & -1 & -x \\ y & x & -b \end{pmatrix}$$

# 4.1 d)

In[691]:=

```
eigenvalues = Eigenvalues[jacobiMatrix];
totalSum = eigenvalues[[1]] + eigenvalues[[2]] + eigenvalues[[3]] // Simplify
```

Out[692]=

```
-1 - b - sigma
```