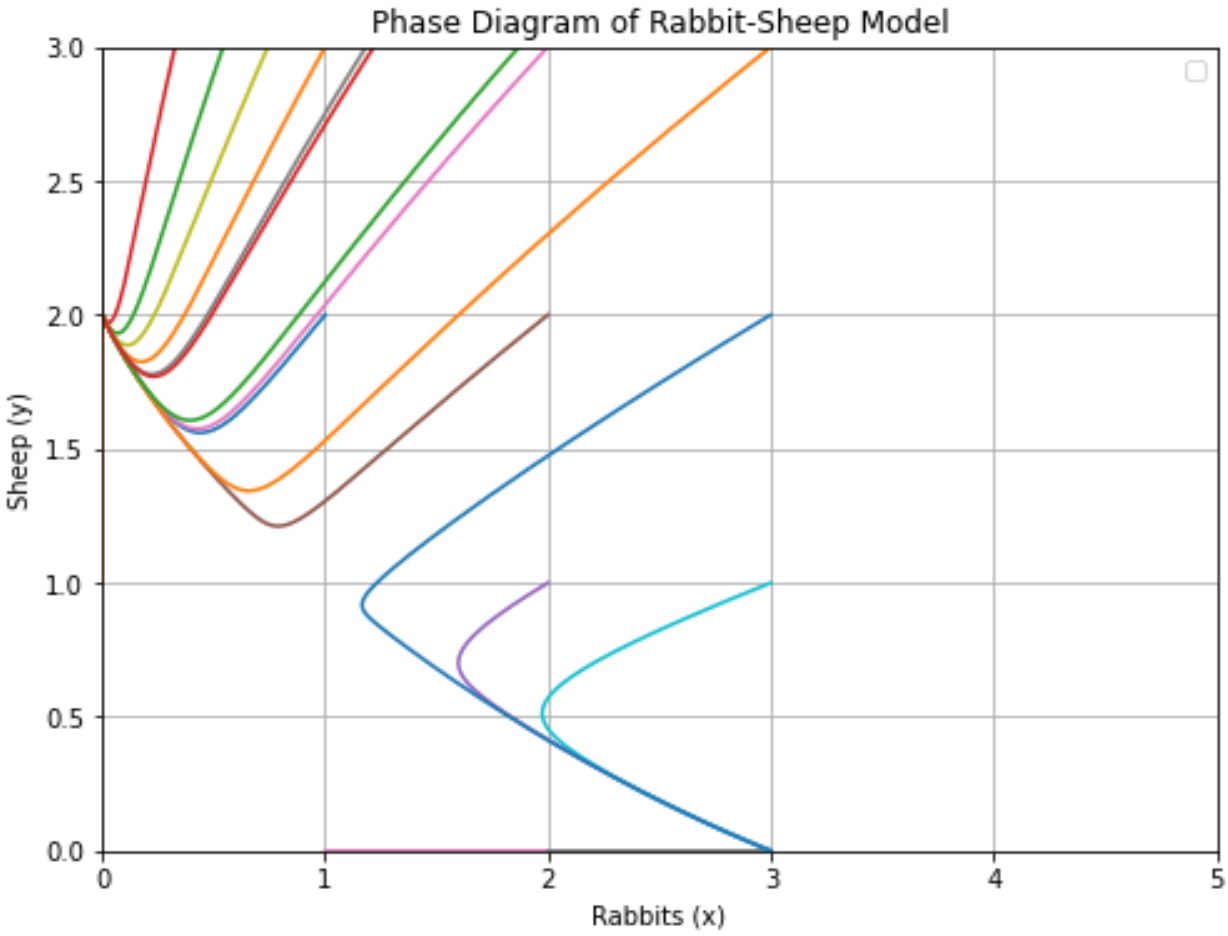


```
import numpy as np
import matplotlib.pyplot as plt

def rabbit_sheep_equations(x, y):

    x_prime = x * (3 - x - 2 * y)
    y_prime = y * (2 - x - y)
    return x_prime, y_prime

def rk4_step(x, y, h):

    k1x, k1y = rabbit_sheep_equations(x, y)
    k2x, k2y = rabbit_sheep_equations(x + h/2 * k1x, y + h/2 * k1y)
    k3x, k3y = rabbit_sheep_equations(x + h/2 * k2x, y + h/2 * k2y)
    k4x, k4y = rabbit_sheep_equations(x + h * k3x, y + h * k3y)

    x_next = x + h/6 * (k1x + 2*k2x + 2*k3x + k4x)
    y_next = y + h/6 * (k1y + 2*k2y + 2*k3y + k4y)
    return x_next, y_next
```

```
def solve_ode(x0, y0, h, t_end):

    x_values = [x0]
    y_values = [y0]
    t = 0
    while t < t_end:
      x_next, y_next = rk4_step(x_values[-1], y_values[-1], h)
      x_values.append(x_next)
      y_values.append(y_next)
      t += h
    return x_values, y_values

initial_conditions = [
    (0, 0),
    (0, 1),
    (0, 5),
    (0, 4),
    (0, 2),
    (0, 3),
    (1, 0),
    (2, 0),
    (3, 0),
    (1, 1),
    (1, 2),
    (1, 3),
    (1, 4),
    (1, 5),
    (2, 1),
    (2, 2),
    (2, 3),
    (2, 4),
    (2, 5),
    (3, 1),
    (3, 2),
    (3, 3),
    (3, 4),
    (3, 5),
    ]

step_size =0.01
t_end_simulation = 20

plt.figure(figsize=(8, 6))

for x0, y0 in initial_conditions:
  x_traj, y_traj = solve_ode(x0, y0, step_size, t_end_simulation)
  plt.plot(x_traj, y_traj)

plt.xlabel("Rabbits (x)")
plt.ylabel("Sheep (y)")
plt.title("Phase Diagram of Rabbit-Sheep Model")
```

```
plt.xlim(0, 5)
plt.ylim(0, 3)
plt.grid(True)
plt.legend()
plt.show()
```