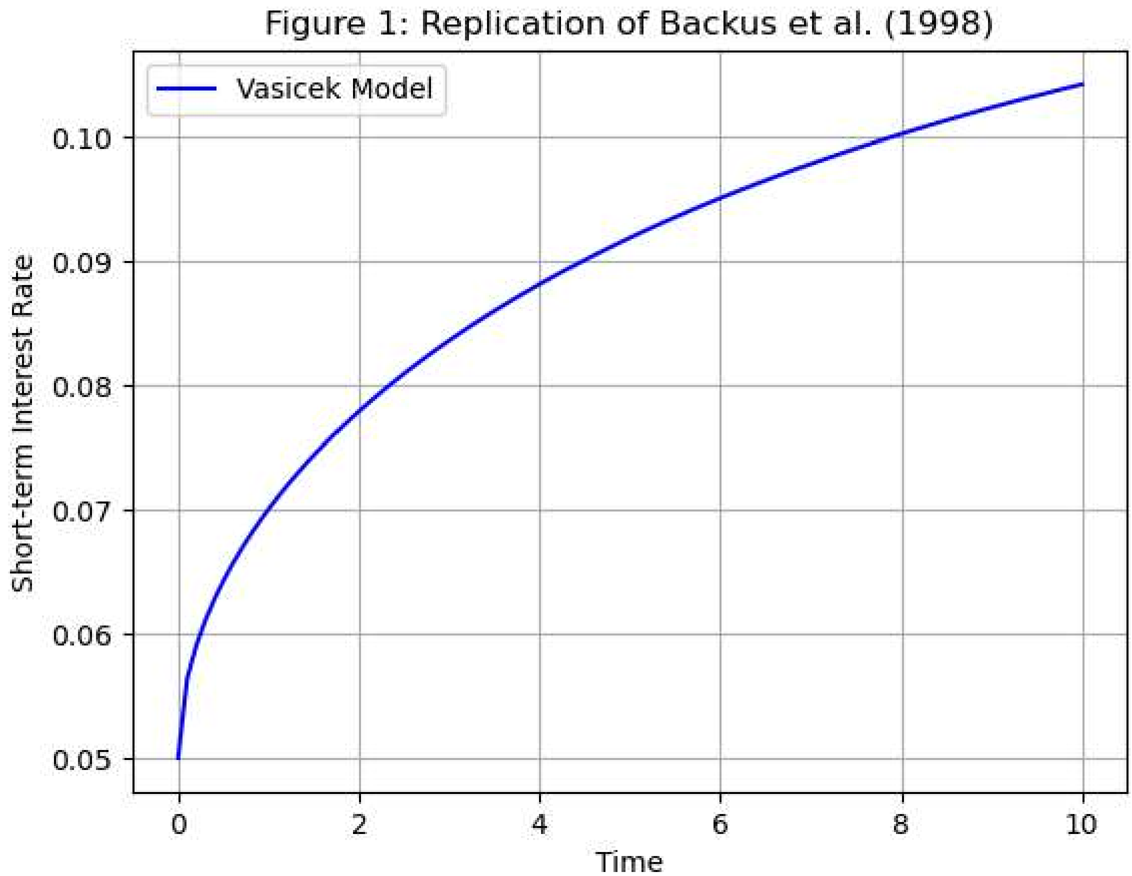
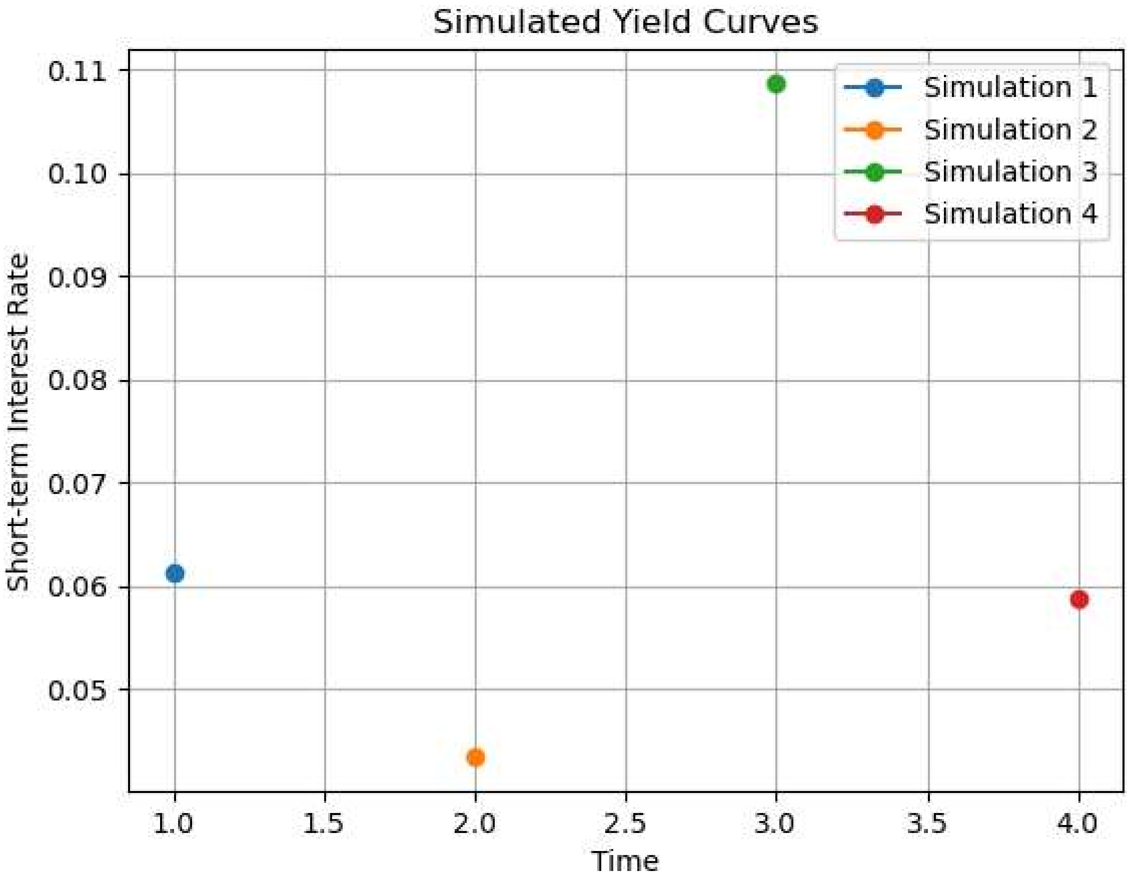
Bonus - Stochastic Interest Rate Models (20 Points)

In [3]:
```
import numpy as np
import matplotlib.pyplot as plt

# Replicate Figure 1 from Backus et al. (1998)

# Define the parameters
mean_reversion_rate = 0.05
long_term_rate = 0.06
volatility = 0.02
initial_short_term_rate = 0.05

# Define the time horizon
t = np.linspace(0, 10, 100)

# Compute the short-term interest rates using the Vasicek model
short_term_rates = long_term_rate + np.exp(-mean_reversion_rate * t) * (initial_sho

# Plot the yield curve
plt.plot(t, short_term_rates, label='Vasicek Model', color='blue')
plt.xlabel('Time')
plt.ylabel('Short-term Interest Rate')
plt.title('Figure 1: Replication of Backus et al. (1998)')
plt.legend()
plt.grid(True)
plt.show()

# Simulate the yield curve over the next four periods

# Define the innovations for εt+s
innovations = [0.55, -0.28, 1.78, 0.19]

# Define the time points for the next four periods
time_points = [1, 2, 3, 4]

# Initialize an empty list to store the simulated short-term interest rates
simulated_short_term_rates = []

# Simulate the short-term interest rates for each period
for i, innovation in enumerate(innovations):
    short_term_rate = long_term_rate + np.exp(-mean_reversion_rate * time_points[i]
    simulated_short_term_rates.append([short_term_rate])

# Plot the simulated yield curves
plt.figure()
for i, rates in enumerate(simulated_short_term_rates):
    plt.plot(time_points[i], rates[0], marker='o', label=f'Simulation {i+1}')

plt.xlabel('Time')
plt.ylabel('Short-term Interest Rate')
```

```
plt.title('Simulated Yield Curves')
plt.legend()
plt.grid(True)
plt.show()
```




Simulated Yield Curves
Short-term Interest Rate
0.11
0.10
0.09
0.08
0.07
0.06
0.05
Time
1.0
1.5
2.0
2.5
3.0
3.5
4.0
Simulation 1
Simulation 2
Simulation 3
Simulation 4

In [ ]: