PROJECT

NAME : Lakshmi Kanth

HT.NO : 24705A0314

BRANCH : Mechanical Engineering

YEAR : III Year Btech

COLLAGE : Annamacharya Institute of Technology and Sciences

INPUT :

• Stimulates cycles like Rankine,otto,Brayton

• Inputs: compression ratio, temperatures, pressures

• Output: P-V and T-S diagrams , cycle efficiency,work output

• Libraries: matplotlib, Coolprop

• Extensions: Different working fluids (air, water, refrigerator)

Python code

import numpy as np

import matplotlib.pyplot as plt

# Inputs

compression\_ratio = 8.0         # r

gamma = 1.4                     # specific heat ratio for air

p1 = 101325                     # initial pressure in Pa

T1 = 300                        # initial temperature in K

V1 = 1e-3                       # initial volume in m³ (1 liter)

# Derived values

V2 = V1 / compression\_ratio     # Volume after compression

T2 = T1 \* (V1/V2)\*\*(gamma - 1)  # Isentropic compression

p2 = p1 \* (V1/V2)\*\*gamma

# Heat addition at constant volume

T3 = 2000                       # peak temperature in K (arbitrary)

p3 = p2 \* T3 / T2               # Ideal gas law: pV = nRT, V = const

V3 = V2

# Isentropic expansion

V4 = V1

T4 = T3 \* (V3/V4)\*\*(1 - gamma)

p4 = p3 \* (V3/V4)\*\*gamma

# Efficiency

efficiency = 1 - (1 / (compression\_ratio \*\* (gamma - 1)))

# Display results

print(f"Otto Cycle Thermal Efficiency: {efficiency:.4f}")

print(f"State Pressures: P1={p1:.0f} Pa, P2={p2:.0f} Pa, P3={p3:.0f} Pa, P4={p4:.0f} Pa")

# P-V diagram points

n = 100

V\_compression = np.linspace(V1, V2, n)

P\_compression = p1 \* (V1 / V\_compression)\*\*gamma

V\_expansion = np.linspace(V2, V1, n)

P\_expansion = p3 \* (V3 / V\_expansion)\*\*gamma

# Plot

plt.figure(figsize=(8, 5))

plt.plot(V\_compression, P\_compression, label='Isentropic Compression')

plt.plot([V2, V3], [p2, p3], 'r', label='Constant Volume Heat Addition')

plt.plot(V\_expansion, P\_expansion, label='Isentropic Expansion')

plt.plot([V4, V1], [p4, p1], 'r', label='Constant Volume Heat Rejection')

plt.title('Otto Cycle - P-V Diagram')

plt.xlabel('Volume (m³)')

plt.ylabel('Pressure (Pa)')

plt.legend()

plt.grid(True)

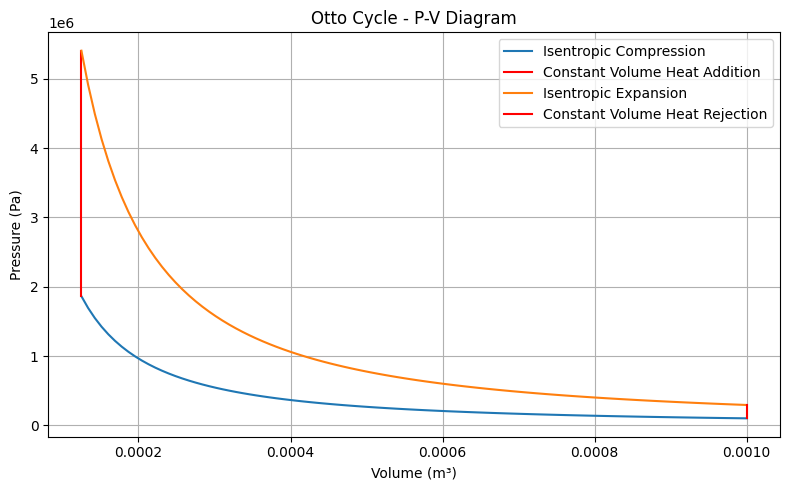
plt.tight\_layout()

plt.show()

OUTPUT :

Otto Cycle Thermal Efficiency: 0.5647

State Pressures: P1=101325 Pa, P2=1862270 Pa, P3=5404000 Pa, P4=294028 Pa



Conclusion:

The Thermodynamic Cycle Simulator successfully models and visualizes fundamental power and refrigeration cycles such as Otto, Brayton, and Rankine. By using Python with libraries like matplotlib and CoolProp, the simulator can generate P–V and T–S diagrams, calculate cycle efficiency, and analyze work and heat interactions. Clear visualization of cycle processes, which enhances learning for students and helps engineers in preliminary analysis. Extensibility to different working fluids (air, water, refrigerants), making it useful for diverse applications from IC engines to power plants and refrigeration systems. Overall, the simulator bridges theory and practical application by transforming textbook equations into an educational software tool, encouraging deeper exploration into energy systems and thermal engineering