Project 1 - Simple Circuit Simulator
Milestone 5: Documentation Creation

1 Project Overview

1.1 Introduction

The Simple Circuit Simulator (SCS) is a Python-based tool designed to model and solve a direct current (DC) circuit that includes a voltage source, a series resistor, and a load resistor connected between buses A and B, as shown in **Figure 1**. The purpose of this simulator is to facilitate the analysis of the DC circuit.

The key features and functionalities of this simulator are:

- **Model DC Circuit:** Users can construct simple DC circuits by adding multiple circuit elements such as voltage source, buses, resistors, and load and specifying component values.
- **Solve DC Circuit:** The simulator calculates and displays the voltage at each bus and the current flowing through the circuit, providing instant output on the circuit's behavior.

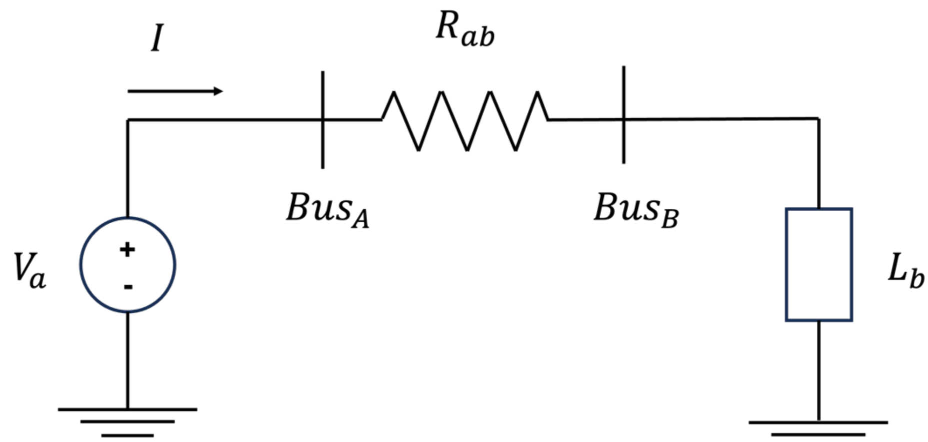

Figure 1. Schematic of the DC Circuit

1.2 Problem Solving and Real-World Applications

The SCS provides a virtual environment where users can experiment with circuit configurations and instantly observe the outcomes. Its real-world applications are:

- **Educational Tools**: Acts as a teaching aid in classrooms to demonstrate the fundamental principles of DC circuit design and analysis.
- **Prototype Testing**: Allows engineers to test DC circuit configurations before building physical prototypes, saving time and resources.
- **Troubleshooting**: Helps in diagnosing issues with DC circuit designs by allowing for hypothetical modifications and observing potential impacts on circuit functionality.

2 Class Diagrams

Figure 2 shows the class diagrams of the SCS. The diagrams are generated using the diagram.puml file, which is processed by the PlantUML plugin in PyCharm.

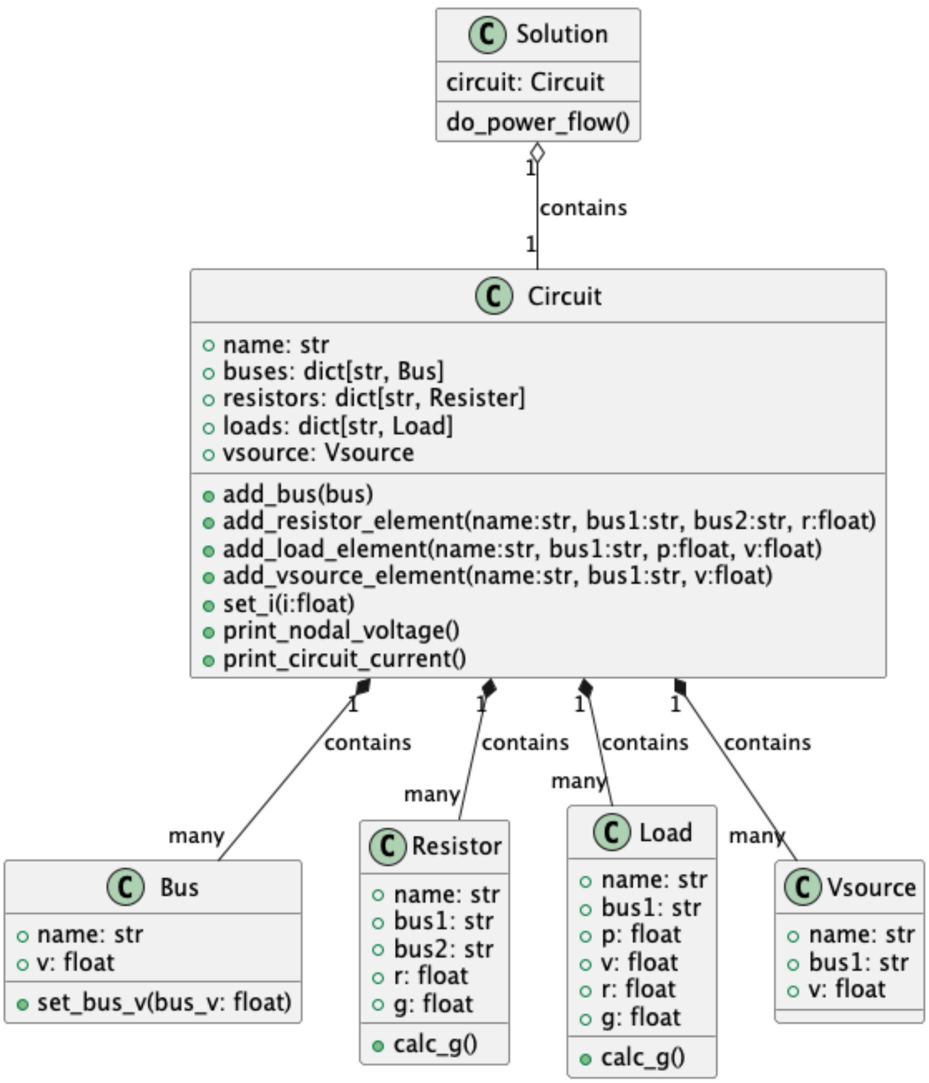

Figure 2. Diagram of the DC Circuit

Class Explanations:

1. Bus: A bus is an electrical junction where multiple circuit components like resistors and voltage sources connect, facilitating the distribution and routing of electrical power.
    a. Attributes:
        i. name (str): Provided by the user as an argument when defining a bus.
        ii. v (float): Represents the voltage at the bus.
    b. Methods:
        i. set_bus_v(bus_v: float): Sets the voltage at the bus.
2. Resistor: A resistor is a passive electrical component that opposes and reduces current flow, thereby regulating the voltage within a circuit based on Ohm's law.
    a. Attributes:
        i. name (str): Provided by the user as an argument when defining a resistor.
        ii. bus1 (str): Provided by the user as one bus connected to the resistor.
        iii. bus2 (str): Provided by the user as the other bus connected to the resistor.
        iv. r (float): Provided by the user as the resistance of the resistor.
        v. g (float): Calculated internally using the calc_g method.
    b. Methods:
        i. calc_g(): Calculates the conductance value (g).
3. Load: In an electrical circuit, a load is any component or device that consumes power, such as lights or motors, and typically represents the functional purpose of the circuit.
    a. Attributes:
        i. name (str): Provided by the user as an argument when defining a load.
        ii. bus1 (str): Provided by the user as the bus connected to the load.
        iii. p (float) : Provided by the user as the power at the load.
        iv. v (float): Provided by the user as the voltage at the load.
        v. r (float): Provided by the user as the resistance of the load.
        vi. g (float): Calculated internally using the calc_g method.
    b. Methods:
        i. calc_g(): Calculates the conductance value (g).
4. Vsource: A voltage source is a component that provides a fixed or variable potential difference between its terminals to power a circuit, such as batteries or power supplies.
    a. Attributes:
        i. name (str): Provided by the user as an argument when defining a voltage source.
        ii. bus1 (str): Provided by the user as the bus connected to the voltage source.
        iii. v (float): Provided by the user as the source voltage.
5. Circuit: A circuit is a closed loop or pathway that allows electric current to flow and transport energy from a source to an intended set of loads.
    a. Attributes:
        i. name (str): Provided by the user as an argument when defining a circuit.
        ii. buses (Dict[str, Bus]): A dictionary where each item has a bus name as the key and its corresponding Bus object as the value. The Bus object is created using the add_bus method of the Circuit class.

iii. resistors (Dict[str, Bus]): A dictionary where each item has a resistor name as the key and its corresponding Resistor object as the value. The Resistor object is created using the add_resistor method of the Circuit class.
iv. loads (Dict[str, Bus]): A dictionary where each item has a load name as the key and its corresponding Load object as the value. The Load object is created using the add_load method of the Circuit class.
v. vsource (Vsource): A Vsource object that is created using the add_vsource method of the Circuit class.
vi. i (float): Current flowing through the circuit. It is updated during the power flow calculation.

b. Methods:
   i. add_bus(bus: str): Adds a bus to the circuit.
   ii. add_resistor_element(name: str, bus1: str, bus2: str, r: float): Adds a resistor to the circuit.
   iii. add_load_element(name: str, bus1: str, p: float, v: float): Adds a load to the circuit.
   iv. add_vsource_element(name: str, bus1: str, v: float): Adds a voltage source to the circuit.
   v. set_i(i: float): Updates the i attribute.
   vi. print_nodal_voltage(): Prints voltages at all buses.
   vii. print_circuit_current(): Prints circuit current.

6. Solution:
   a. Attributes:
      i. circuit (Circuit): When creating a solution object, user must pass a Circuit object as an argument.
   b. Methods:
      i. do_power_flow(): Solves the circuit by finding bus voltages and circuit current.

3 Relevant Equations

The equations used in the SCS include Ohm's law, power-voltage relationship, conductance calculations, and KVL to solve for circuit voltages and current.

3.1 Ohm's Law

Ohm's Law is for determining the current flowing through a component in an electrical circuit. It is expressed as:

$$\boldsymbol{I} = \frac{\boldsymbol{V}}{\boldsymbol{R}}$$

Where $\boldsymbol{I}$ is the current through the component (in amperes, A), $\boldsymbol{V}$ is the voltage across the component (in volts, V), and $\boldsymbol{R}$ is the resistance of the component (in ohms, Ω).

3.2 Power-Voltage Relationship

The power consumed by a component in a DC circuit can be calculated using the power-voltage relationship, which is given by:

$$\boldsymbol{P = VI}$$

Where $\boldsymbol{P}$ is the power (in watts, W), $\boldsymbol{I}$ is the current through the component (in amperes, A), and $\boldsymbol{V}$ is the voltage across the component (in volts, V).

By combining Ohm's law and power-voltage relationship, the resistance $\boldsymbol{R}$ of a component can be calculated using its voltage $\boldsymbol{V}$ and power $\boldsymbol{P}$**:**

$$\boldsymbol{R = \frac{V^2}{P}}$$

3.3 Conductance Calculations

The conductance of a component is the inverse of its resistance:

$$\boldsymbol{G = \frac{1}{R}}$$

Where $\boldsymbol{G}$ is the conductance (in siemens, S), and $\boldsymbol{R}$ is the resistance of the component (in ohms, Ω).

3.4 Kirchhoff's Voltage Law (KVL) to Solve for System Voltages and Current

KVL ensures the sum of the voltage drops around the circuit equals the voltage of the source. KVL can be represented as:

$$\boldsymbol{V_{source} = V_1 + V_2 = IR_1 + IR_2}$$

Where $\boldsymbol{V_{source}}$ is the voltage of the source, $\boldsymbol{I}$ is the current, $\boldsymbol{V_1, V_2}$ are voltages of all the other components of the circuit, and $\boldsymbol{R_1, R_2}$ are the resistors**.**

4 Example Case

4.1 Problem Definition

The circuit in **Figure 1** is defined as:

1. Buses A and B are added to the circuit.
2. Voltage source $\boldsymbol{V_a}$ is connected at bus A with **120 V**.
3. Resistor $\boldsymbol{R_{ab}}$ is connected between buses A and B with **50 Ohms**.
4. Load $\boldsymbol{L_b}$ is connected to bus B with power $\boldsymbol{P = 1000}$ **W** and nominal voltage of $\boldsymbol{V_{load} = 100}$ **V**. The load model is constant impedance.

The problem is to solve the current of the circuit, and the voltages of bus A and B.

4.2 Solution Process

The solution details are in the Solution class in solution.py file. The total resistance along the circuit is computed by summing the resistances of the resistor element and the load element:

$$\boldsymbol{R_{total} = R_{ab} + R_{load} = R_{ab} + \frac{{V_{load}}^2}{P}}$$

Using Ohm's Law, the current ($\boldsymbol{I}$) flowing through the circuit is calculated as:

$$\boldsymbol{I = \frac{V_a}{R_{total}}}$$

The voltage at bus A ($\boldsymbol{V_{busA}}$) is set equal to the source voltage since it is directly connected to the voltage source:

$$\boldsymbol{V_{busA} = V_a}$$

The voltage at bus B ($\boldsymbol{V_{busB}}$) is calculated using the current through the circuit and the resistance of the load:

$$\boldsymbol{V_{busB} = IR_{load}}$$

4.3 Expected Output

In the main() function of the main.py file, create a DC circuit instance my_circuit named "Simple DC Circuit" :

```
my_circuit = Circuit("Simple DC Circuit")
```

Use the add_bus() function to add buses to the circuit, named "Bus A" and "Bus B":

```
my_circuit.add_bus("Bus A")
my_circuit.add_bus("Bus B")
```

Use the add_ vsource_element () function to add voltage source to the circuit, named "Va". It is connected to Bus A. Its voltage is 120 V:

```
my_circuit.add_vsource_element("Va", "Bus A", 120)
```

Use the add_ resistor _element () function to add a resistor to the circuit, named "Rab". It is connected to Bus A and Bus B. Its resistance is 50 Ohms:

```
my_circuit.add_resistor_element("Rab", "Bus A", "Bus B", 50)
```

Use the add_ load _element () function to add a load to the circuit, named "Lb". It is connected to Bus B. Its power is 1000 W, and its voltage is 100 V:

```
my_circuit.add_load_element("Lb", "Bus B", 1000, 100)
```

Pass the created circuit my_circuit to the Solution class and call the do_power_flow() function to solve the problem:

```
solution = Solution(my_circuit)
solution.do_power_flow()
```

Then, run the main function:

```
if __name__ == '__main__':
    main()
```

Finally, the expected output would be displayed:

```
Circuit Current = 2.0 A
Bus A Voltage = 120.0 V
Bus B Voltage = 20.0 V
```