SPICE Assignment 2 - EE23B039

evalSpice.py contains a script that parses a SPICE netlist file, extracts the circuit components, and computes the node voltages and currents through independent voltage sources using matrix analysis.

Problem Description

The goal is to analyze a circuit defined in a SPICE file. The netlist includes resistors, current sources, and voltage sources. The code reads this SPICE file, processes it, and then solves for: - Node voltages (excluding ground) - Currents through independent voltage sources

The circuit equations are formulated using Kirchhoff's Current Law (KCL) and solved using matrix methods.

How It Works

Input

- The SPICE file must define a circuit between the `.circuit` and `.end` markers. The components supported are:
  - **Resistors**: `R`
  - **Current Sources**: `I`
  - **Voltage Sources**: `V`
- The circuit must have at least one independent voltage source.

Output

The script returns: - **Node Voltages**: Voltage at each node relative to ground (GND). - **Currents Through Voltage Sources**: The current flowing through each independent voltage source.

Code Structure

- **File Parsing**:
  - The input file is read, and the circuit components are identified between `.circuit` and `.end`.
  - The circuit components are stored in a dictionary, capturing the type of component, the nodes it connects, and its value (e.g., resistance, current, or voltage).
- **Matrix Setup**:
  - The unknown variables are node voltages and currents passing through them. All of them are kept as a single column matrix V. Contents in V till index = number of nodes - 1 are the node voltages(unknown) and from index = number of nodes to (number of nodes+number of independent voltage sources -1) are the unknows currents passing through the voltage sources.

  - A conductance matrix `G` is created to represent the system of equations. The dimensions are `(number of nodes + number of independent voltage sources) x (number of nodes + number of independent voltage sources)`.
  - Initially, G was created as a zero matrix of the above dimensions. Then, it was realised through the KCL equations written manually that G[i][i] is the sum of all the conductances connected to node i. G[i][j] is the negative conductance connected between node i and node j. Based on this G[:number of nodes][:number of nodes] was filled.
  - Now again from the KCL equations, it is seen that if a node is connected to the positive terminal of voltage, by using passive convension current is negative so -1(/+1) is added to the node where it is connected to the same terminal as the positive terminal of voltage(/negative terminal of voltage).
  - By now G[:number of nodes][:] is filled correctly. The remaining rows at the bottom represent the constraints for voltage differences between two nodes when a voltage source is between them. Like suppose a voltage source Vs is connected in between n1 and n2. then V1 - V2 = Vs. So the node which was connected to the positive terminal of source is added 1 and the one connected to the negative terminal is added -1 in the G matrix.
  - Now G matrix is filled with correct values.
  - The current vector `I`(column matrix) is created to represent the known values of currents and voltages. First n(= number of nodes) values of the column matrix contain the source currents entering or leaving a particalar node (+1 or -1) and the remaining contain the voltage sources that were represented in the constraints relation.
- **Solving**:
  - The matrix equation `G * V = I` is solved for `V` (node voltages and voltage source currents).
  - If the determinant of `G` is zero, the system has no solution, and an error is raised.

Error Handling

- If no valid filename is provided, a `FileNotFoundError` is raised.
- If the circuit is malformed (missing `.circuit` or `.end` markers), a `ValueError` is raised.
- Only resistors (`R`), current sources (`I`), and voltage sources (`V`) are supported; any other elements raise a `ValueError`.
- If the system of equations is unsolvable (i.e., the determinant of the matrix is zero), a `ValueError` is raised indicating that no solution exists.
- If any line in the circuit contains less than 4 words, it means something among name, node1, node2 and value is missing so 'Malformed circuit file" error is raised.

References I looked up for:

- ECN notes for writing conductance part of the matrix
- Google for syntax
- My classmates and I had a discussion about filling the G matrix which was very confusing at the beginning and about errors(assersion error for example).