

ASSIGNMENT-2 SPICE SIMULATION

SREERAM R EE23B075

Algorithm in brief

- The function `evalSpice()` takes a input file that contains a circuit in a SPICE netlist.
- First we clean the data by only taking the file contents between `.circuit` and `.end` by removing the comments.
- Then the circuit elements are identified as voltage sources, current sources, resistors and also identify the nodes between which these circuit elements are connected. It is done by storing the value of the elements and the nodes between which the element is connected as a list. `voltage`, `current` and `resistance` are the list of lists with the information about each of the components respectively.
- The next step is to declare a coefficient matrix and a constant matrix and populate it with the admittances of the resistors and the effects due to the current sources and the coltage sources in the nodal analysis equations.
- The system of equations is then solved by `numpy.linalg.solve` and the solutions are obtained in the `result` matrix.
- The solutions obtained are stored in the form of two dictionaries `Vsol` and `Isol`. `Vsol` contains the name of the node as the key and the voltage value at the particular node as its value. The `Isol` dictionary contains the name of the voltage source as the key and the current through the corresponding voltage source as it value.

Algorithm in detail

- The `evalSpice()` function begins by opening the input file `in_file` in read mode. An error is raised if a valid SPICE file is not given as an input.
- `lines` is a list of lists ie. It stores the sentences in the input file in the form of lists
- `clean_data` refers to the numpy array which contains the data without comments, baisscally the data is cleansed before processing.
- `current`, `voltage` and `resistor` are list of lists to store the element's value and the nodes between which it is connected.
- It is done by storing the value of the elements and the nodes between which the element is connected as a list. `voltage`, `current` and `resistance` are the list of lists with the information about each of the components in the circuit. Lists `Vlist` , `Rlist`, `Ilist` are maintained by appending the nodes between which a element is connected and its value. Name of the node is also included in the case of voltage sources . Then these lists as appended to the `voltage`, `current` and `resistance` making them a list of lists.

- A list **node** is used to store the nodes in the circuit and a list **node_mapping** is a list declared for mapping the index of the node in the list “node” to the actual name of the node.
- These list of lists are converted into numpy arrays for efficient computation.
- A coefficient matrix **coeff_matrix** and a constant matrix **const_matrix** is created and they are populated with the admittance of the resistors and the effects due to the current and the voltage sources.
- To solve the system of nodal analysis equations **numpy.linalg.solve()** is used and the solutions obtained are stored in the **result** matrix. The outputs used in the result matrix.
- The solutions obtained are stored in the form of two dictionaries **Vsol** and **Isol**. **Vsol** contains the name of the node as the key and the voltage value at the particular node as its value. The **Isol** dictionary contains the name of the voltage source as the key and the current through the corresponding voltage source as its value.
- The **evalSpice** function finally returns the two dictionaries **Vsol** and **Isol**.

Special tests / Border cases

Handling circuit without GND

- raises **ValueError** when given a file with no “GND”

Testing an open circuit.

```
.circuit
V0 1 GND dc 2
R1 1 2 1
R2 GND 3 1
.end
```

- Circuit behaves as expected with voltage of voltage sources at appropriate nodes and GND Voltage at others

Testing circuit with a voltage source of 0 Volts

```
.circuit
R1 n2 n3 1
V0 n2 GND dc 5
R2 n3 GND 1
V1 n3 GND dc 0
.end
```

- Treats it as a short circuit as expected.

Testing circuit with a current source and open circuit

```
.circuit
R1 1 2 1
IO GND 1 dc 1
.end
```

- Throws an error as expected `ValueError: Negative Value of Resistance is not allowed`

References

- Numpy Documentation
- Programming quizzes