

ELEC40006  
Electronics Design Project  
Circuit Simulation

**Imperial College  
London**

# Technical Report

Xin Wang  
CID: 01751352 EIE 1st Year UG

FIRST LAST

FIRST LAST

dd-mm-yyyy

# Contents

<b>1</b>	<b>Overview of the report</b>	<b>3</b>
<b>2</b>	<b>Project Specification</b>	<b>4</b>
2.1	Design Problem . . . . .	4
2.2	Program Requirements . . . . .	4
<b>3</b>	<b>Project Management</b>	<b>5</b>
3.1	Project timeline . . . . .	6
3.2	Management approach . . . . .	7
3.3	Project responsibilities breakdown . . . . .	8
<b>4</b>	<b>Preliminary Designs</b>	<b>9</b>
4.1	Object Orientated Programming . . . . .	9
4.2	Modified Nodal Analysis . . . . .	9
4.3	Eigen Matrix Library . . . . .	9
<b>5</b>	<b>Program Design</b>	<b>10</b>
5.1	Parser module . . . . .	11
5.1.1	Features . . . . .	12
5.1.2	Pseudocode . . . . .	13
5.2	Analysis module . . . . .	14
5.2.1	Features . . . . .	15
5.2.2	Pseudocode . . . . .	16
5.3	Transient module . . . . .	17
5.3.1	Features . . . . .	18
5.3.2	Pseudocode . . . . .	19
<b>6</b>	<b>Testing</b>	<b>20</b>
<b>7</b>	<b>Optimisations</b>	<b>21</b>
7.1	Sparse Matrices . . . . .	21
<b>8</b>	<b>Adding on</b>	<b>22</b>
<b>9</b>	<b>Appendix</b>	<b>23</b>
<b>10</b>	<b>References</b>	<b>24</b>

## **Abstract**

This report describes the design and implementation of a program that is capable of performing a transient simulation by calculating the node voltages at each successive instant in time. This program parses the netlist file into a graph data structure, performs analysis using conductance matrices and outputs the results in a CSV format.

- How accurate is it?
- Comparison to commercial software?

# 1 Overview of the report

This report is the distillation of multiple research documents relating to different components of the program.

Section 2 gives an abstract view of the design of the program, breaking the program down into 3 modules. Section 3 provides a summary of the testing methodologies and a comparison to both handwritten results and results of established circuit simulator software. Section 4 delves into the further work done and some potential ideas to build on. Section 5, the last section, summarises the report and discusses our overall experiences with the development of this project.

*Talk about added functions and anything else.*

## 2 Project Specification

### 2.1 Design Problem

Develop a program that is able to read in a file describing a circuit specified by the user, perform transient simulation on that circuit and output the calculated voltages at each instance in time into a file.

### 2.2 Program Requirements

The main program requirements are listed as follows:

- Program must support basic circuit components listed as follows <sup>1</sup>:
  - Resistors
  - Ideal Capacitors
  - Ideal Inductors
- Input file must adhere to SPICE netlist formats
- The output file must be in Comma Separated Value (.CSV) format.
  - Columns of output file represent nodes in the circuit.
  - Rows of output file represent an instance in the simulation.

---

<sup>1</sup>Advanced component can be supported provided development schedule is not constrained.

### 3 Project Management

### 3.1 Project timeline

## **3.2 Management approach**



### 3.3 Project responsibilities breakdown

## **4 Preliminary Designs**

### **4.1 Object Orientated Programming**

The program architecture was designed with Object Orientated Programming (OOP) in mind. The OOP approach allows the program design to be rapidly and cost-effectively altered to accomodate new features such as additional circuit components or analysis techniques that the client might request later on in the program's lifecycle.

An OOP approach has shown to be slower than preliminary programs designed without OOP to an average of 5% to 10% depending on other optimisations in the program. However, the requirements of the client does not explicitly state a need for extremely fast program execution and a definitive list of circuit components was not provided...

### **4.2 Modified Nodal Analysis**

### **4.3 Eigen Matrix Library**

## 5 Program Design

## 5.1 Parser module

### 5.1.1 Features

### 5.1.2 Pseudocode

## 5.2 Analysis module

### 5.2.1 Features



### 5.2.2 Pseudocode

### 5.3 Transient module

### 5.3.1 Features

### 5.3.2 Pseudocode

## 6 Testing

## 7 Optimisations

### 7.1 Sparse Matrices

## 8 Adding on

## 9 Appendix



## 10 References