Development of Simulation Algorithm for Online Electrical Measurements Lab

Submitted
by

Shivam Yadav	**20115135**
Satyam Yadav	**20115127**
Shubham	**20115139**
Rohit Pal	**20115116**
Anjani Kumar Tiwari	**20115015**

Under
Supervision of

Prof. R.S. Anand

Department of Electrical Engineering

Indian Institute of Technology Roorkee

2023

Declaration

I hereby declare that the work which is presented here, entitled, **Development of simulation algorithm for online electrical measurements Lab** submitted for the completion of course EEN 300: Industry Oriented Problem. I also declare that we have been doing our work under the supervision and guidance of **Prof R.S. Anand, Electrical Engineering, Indian Institute of Technology Roorkee**. The matter presented in this report is not submitted for award of any other degree of institute or any other institutes.

Date: April 19, 2023 Signature

Shivam Yadav	**Satyam Yadav**	**Shubham**	**Rohit Pal**	**Anjani Kumar Tiwari**
(20115135)	**(20115127)**	**(20115139)**	**(20115116)**	**(20115015)**

Certificate

This is to certify that the above statement made by the candidates is true to best of my knowledge and belief.

Signature

R. S. Anand (Professor)
Department of Electrical Engineering
Indian Institute of Technology Roorkee

Acknowledgement

We express our sincere gratitude to our guide **Dr. R. S. Anand,** Professor, Electrical Engineering Department (EED), Indian Institute of Technology, Roorkee (IITR) for providing us with the wonderful opportunity to work on Industry Oriented Project (IOP) on the Development of Simulation Algorithms for electrical measurement labs and for his invaluable guidance that enabled us to successfully complete the project.

We would like to express our gratitude to **Mr. Rajeev Kumar,** Senior Research fellow and **Mr. Nipun Jain,** Project Associate Virtual Labs Electrical Engineering Department IIT Roorkee for constantly giving their time, efforts and resources, as well as for their insightful comments and suggestions that considerably improved the quality of the project as a whole.

We would also like to thank our family and friends for their unwavering support and motivation during this process.

The success of this initiative would not have been possible without the assistance of these people. Finally, we thankfully acknowledge every other individual who helped us during the entire project.

Abstract

The aim of this project is to develop simulation algorithms that can provide students with a virtual representation of an electrical measuring lab to enhance their learning experiences. This involves creating a web-based platform that allows students to measure electrical quantities such as resistance, capacitance, and inductance of circuit elements through simulations of experiments like the Wheatstone Bridge, Carey Foster Bridge, and Anderson Bridge. The platform utilizes tools such as the jsPlumb toolkit, HTML, CSS, and JavaScript integration to enable real-time circuit connections, validate user input, and display results in an observation table. The system also includes logic to generate and identify circuit element IDs, as well as a verification algorithm to ensure accurate circuit connections and measurements.

Table of Contents

List of Figures

Chapter 1

Introduction

The Ministry of Human Resource and Development (MHRD) in India, under the National Mission on Education through Information and Communication Technology (NMEICT), has launched a program known as virtual labs. This initiative involves 12 member institutes, with IIT Delhi serving as the coordinating institution. The virtual labs program has brought about a significant shift in the way ICT-based education is delivered, particularly in remote experiments. The program has developed more than 100 virtual labs with around 700 web-based experiments for remote operation and viewing. The virtual lab is expected to transform the science classroom by enhancing the quality of science education.

1.1 Objectives of Vlab

- ❖ The objectives of the virtual labs program is to provide remote access to labs in various science and engineering fields for undergraduate and graduate students, research scholars, and to ignite students' interest in experimentation.

- ❖ The program aims to create On-Demand Labs that enable students to learn at their own pace while exploring both basic and complex concepts.

- ❖ Additionally, the program intends to develop a comprehensive learning management system that integrates virtual labs with various learning resources, such as web content, video lectures, animated examples, and self-assessment quizzes.

- ❖ Lastly, the virtual labs program aims to combine expensive resources and equipment that are typically restricted due to time constraints, making them available to a larger audience.

1.2 Proposed work (motivation and purpose)

Every engineering college should have up-to-date lab experiments and adequate lab facilities. However, limitations such as physical constraints and lack of resources often hinder the ability to conduct tests, particularly when using complex apparatus. Additionally, the scarcity of quality educators further compounds the problem. Although online and video courses have partially addressed the issue of teaching, conducting cooperative experiments and sharing expensive resources between participating universities has been challenging. Despite these restrictions, modern internet and computer technology have now made it possible for students and researchers to overcome these limitations and expand their knowledge and skills. In a country like ours, it is important to promote the sharing of expensive tools and equipment among scholars whenever feasible. The development of web-enabled experiments that can be operated and observed remotely has the potential to foster students' creativity and curiosity. This approach allows for the exploration of both fundamental and advanced concepts through remote experimentation. Many modern pieces of equipment now come with computer interfaces for data storage and management, which can be utilized for conducting meaningful trials to enhance student learning. Internet-based experimentation and the creation of high-quality virtual labs offer the opportunity to leverage web-based resources, knowledge, software, and data, as well as the ability to conduct complex experiments simultaneously at geographically distant locations, and even potentially across different time periods.

1.3 Advantages

- Describing a physical phenomenon using equations and simulating it to predict the outcome of an experiment provides an approximate representation of real-world experiments.
- Providing quantifiable data from virtual lab experiments that align with previously collected measurements from actual systems.
- Initiating and conducting a real lab experiment remotely, with results displayed to the learner through a computer interface, necessitating conducting the experiment online.
- By giving the students additional inputs like complementing audio and video streaming of an actual lab experiment and equipment, virtual labs will become more efficient and realistic.
- For conducting experiments at user premises, virtual labs don't require any additional infrastructure configuration. Remote access to the simulation-based studies is possible over the internet on any PC, laptop, smart phone, or tablet.
- It is free of cost and available for 24*7 usage.
- We have the freedom to make mistakes while performing any experiment as it will not incur any losses and integrated learning is also possible.

1.4 List of proposed experiments:

- To Measure the unknown resistance using Wheatstone bridge
- Measurement of unknown resistance and inductance by Hays bridge
- Measurement of Capacitance by Carey Foster Bridge
- Measurement of Self-Inductance by Maxwell's Bridge
- Measurement of Capacitance by Wien Series Bridge
- Measurement of Self Inductance by Owen Bridge
- Measurement of Capacitance by Schering Bridge
- Measurement of Self Inductance accurately by Anderson's Bridge
- To Study the Wien Robinson's frequency Bridge

Chapter 2

2. Electrical Bridges

Electrical bridges are commonly used in Electrical Engineering. It is used for measuring the unknown resistance, capacitance and inductance up to a certain level of accuracy.

2.1 Types of bridges equations & formulae

There are some bridges as follows, which are designed for measuring specific values:

<u>Wheatstone Bridge:</u>

Wheatstone bridge is a DC bridge, most commonly used for measuring unknown resistance by comparing it with known resistance. The circuit of this bridge consists of four resistance, one DC voltage source and one galvanometer. We vary the one resistance keeping two resistances fixed; after getting null deflection, we can easily calculate the unknown resistance by using the formula of balances condition.

$$\frac{R1}{R3} = \frac{R2}{R4}$$

Now if the condition is satisfied then the current through galvanometer will be zero, else

Voltage difference between terminals of the galvanometer:

$$Vth = \frac{V * r3}{(r1 + r3)} - \frac{V * r4}{(r2 + r4)}$$

Thevenin equivalent across galvanometer:

Vth=$\frac{r1*r3}{(r1+r3)} + \frac{r2*r4}{(r2+r4)}$

$$Ig = \frac{Vth}{(Rg + Rth)}$$

<u>Maxwell’s Bridge:</u>

Maxwell bridge is an AC bridge used for measuring self-inductance in term of calibrated resistance and capacitance. The electrical circuit of this bridge consists of resistor, inductor, detector and AC voltage source. It is generally used for measuring self-inductance having a low Q-value.

$$\frac{R4}{R3} = \frac{R2}{R1} = \frac{L2}{L1(unknown)}$$

Owen's Bridge

Owen's bridge is an AC bridge used for measuring the unknown inductance in term of capacitance. The electrical circuit of this bridge consists of resistor, capacitor, inductor and AC voltage source. It works on the principle of comparison with the standard capacitor. It is used for measuring a wide range of inductance from a few milli Henry to a few Henry.

$$L1 = R2 \times R3 \times C4$$

$$R1 = \frac{R3 \times C4}{C2}$$

Schering Bridge

The Schering Bridge is an AC bridge used for measuring the unknown capacitance of the capacitor, dissipation factor. The electrical circuit of this bridge consists of capacitor, resistor and AC voltage source. It works on the principle of balancing.

$$C1 = \frac{R3 \times C4}{C2}$$

$$R1 = \frac{R3 \times C4}{C2}$$

$D = w \times R4 \times C4$ (D is dissipation factor)

Anderson's Bridge

Anderson's bridge is an AC bridge commonly used to determine the self-inductance of a coil. It comprises resistors, a capacitor, an inductor, a detector, and an AC power source, and is a modified version of Maxwell's inductance-capacitance bridge. By utilizing a standard capacitance, the bridge allows for highly precise measurements of self-inductance over a broad range, from microhenries to several Henries. As a result, this technology is widely employed due to its accuracy and versatility.

$$R1 = \frac{R2 \times R3}{R4}$$

$$L1 = \frac{C1 \times R3}{R4}(R2 \times R4 + R5 \times (R2 + R4))$$

Carey's Foster

The Carey Foster Bridge is an AC bridge that measures capacitance in relation to mutual inductances. It consists of resistors, inductors, unknown capacitors, an AC power source, and a detector, all interconnected. One of the arms of the bridge is intentionally short-circuited, resulting in zero potential drop across that arm. This is an unusual feature of the bridge. For mutual induction to function, negative coupling is necessary, as the potential difference across the opposing arm must be zero to achieve equilibrium.

$$C3 = \frac{M1}{R1 \times R4}$$

$$R3 = \frac{(L1 - M1) \times R4}{M1}$$

Hays Bridge

Hays bridge is an AC bridge used for measuring the resistance and inductance of a coil with high Q-factor. The electrical circuit of this bridge consists of inductor, resistor, capacitor, AC voltage source and detector. It is a modified form of Maxwell's bridge. This bridge gives us a very simple expression for the calculation of inductance.

$$L1 = \frac{R2 \times R3 \times C}{1 + w^2 R1^2 C^2} = \frac{R2 \times R3 \times C}{1 + \frac{1}{Q^2}}$$

For high-quality factor, Q

$$L = R2 \times R3 \times C$$

Wein's Series Bridge

Wein's Series bridge is AC bridge used for measuring unknown capacitance. The electric circuit of this bridge consists of capacitor, resistor, detector and AC voltage source. In this experiment, we express unknown capacitance and resistance in the term of known resistance and frequency.

$$C1 = \frac{C4 \times R3}{R2}$$

$$R1 = \frac{R2 \times R4}{R3}$$

Wein Robinson frequency bridge

Wein Robinson frequency bridge is an AC bridge used for measuring frequency. The electric circuit of this bridge consists of resistor, capacitor, detector and AC voltage source. The frequency can be measured easily by balancing the arms. The electronic circuit of this bridge consists of a series RC combination in one arm and a parallel combination in the other arm. The accuracy of this bridge is 0.1 to 0.5 per cent.

$$\frac{R4}{R3} = \frac{C1}{C2} + \frac{R2}{R1}$$

$$w = \frac{1}{\sqrt{R1R2C1C2}}$$

$$f = \frac{1}{2\pi\sqrt{R1R2C1C2}}$$

If (R1=R2=R) and (C1=C2=C), we have

$$f = \frac{1}{2\pi RC}$$

Electrical Bridges

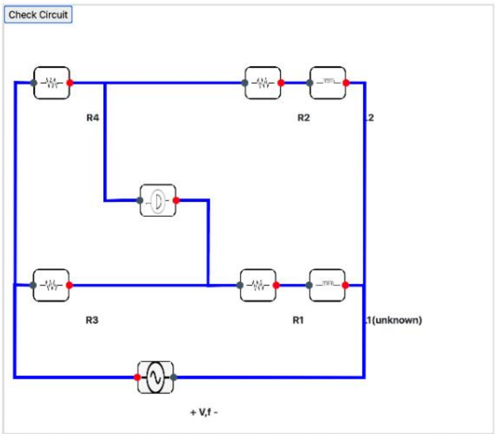


Figure 1:Maxwell Bridge

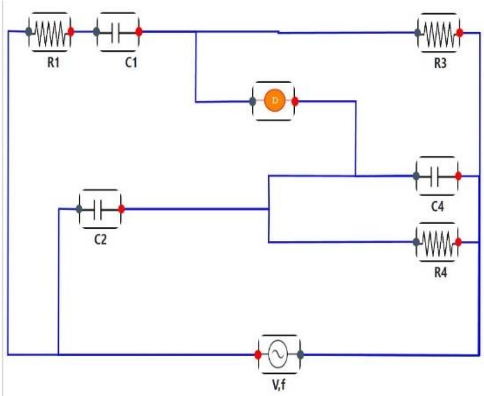


Figure 2: Owen's Bridge

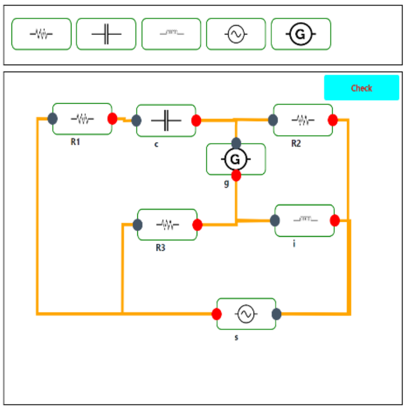


Figure 3: Hays Bridge

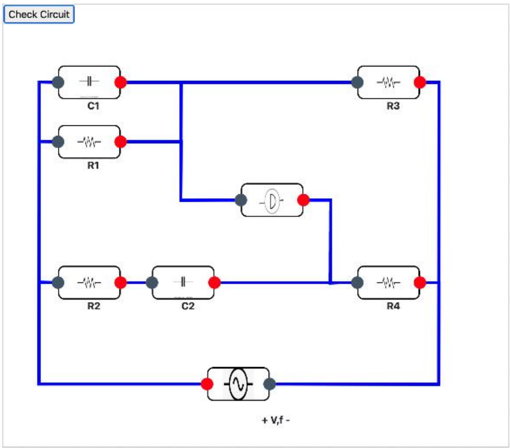


Figure 4: WR Frequency Bridge

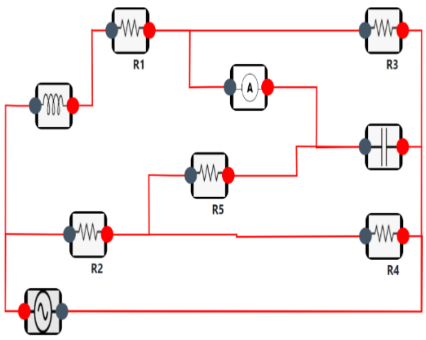


Figure 5: Anderson Bridge

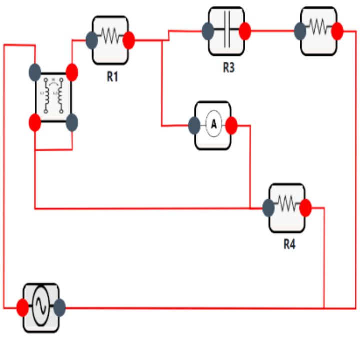
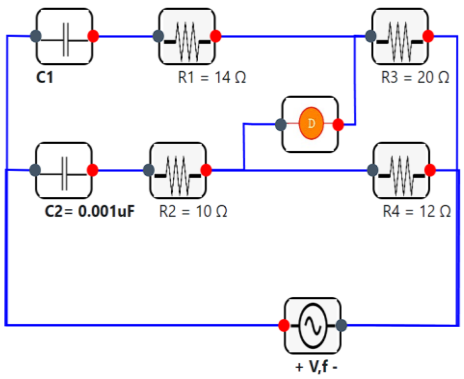
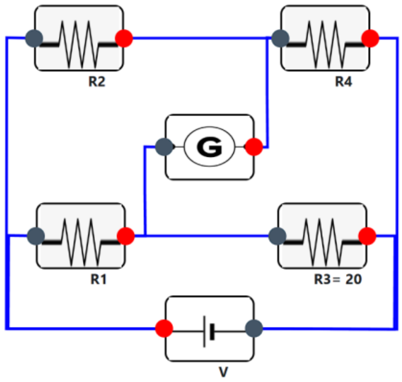
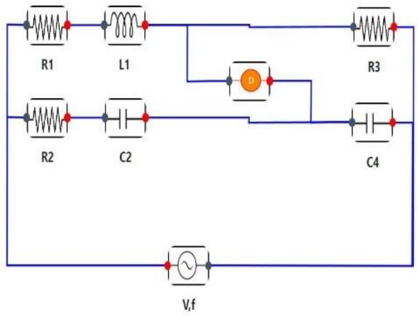


Figure 6: Carey Foster Bridge

 Figure 7: Wien's Series Bridge	 Figure 8: Wheatstone Bridge	 Figure 9: Schering Bridge

Chapter 3

Basics of Code

HTML5, CSS, and JS are the main programming languages used in the development of this project. Other than these, we also utilized some inbuilt JavaScript libraries like jsPlumb and jQuery. We used VS Code (Virtual Studio Code) as Integrated Development Environment to develop our application.

3.1 HTML5

Hypertext Markup Language5 or HTML5 is the latest version of the HTML standard package. It is a markup rather than a programming language for interactive and dynamic web page applications. It provides many features that make the web page more accessible, responsive, and user-friendly. We used it as the base of our software program.

3.2 CSS

Cascading Style Sheets or CSS is a style sheet language used to give the layout and virtual style to web pages. It allows you to control the appearance of the layout made by using HTML. Using the wide range of selectors, it helps change the size, color, and spacing of the fonts, and it also has advanced features which make the web page responsive by changing the content and layout according to the size of the screen or zooming in or out.

3.3 JavaScript

JavaScript or JS is a programming language. Unlike HTML and CSS, it is a high-level dynamic language mainly used in web development applications. Originally it was a scripting language for web page interactivity, but later it became a programming language that can also be used for writing complex and robust applications. It is based on DOM (Document Objective Model) for web page programming interface, which helps the developer to make web pages that can update the content and appearance of the page to the input or response given by the user like a mouse click or keyboard input which can update the content of the webpage in real-time. We can use JS for different tasks, like data processing, validation, animation, and interactive visualizations. It can be used both in frontend and backend development, and it is supported by all major web browsers, including Chrome, brave, edge, etc., making it one of the most widely used languages in the tech industry. Additionally, there are many libraries and framework built using JS, such as React, Angular Query, and JSplumb, which provide additional features to the developer which help in the building of advanced web structures JavaScript also support the OOP(Object-Oriented Programming) paradigms along with other paradigms like functional programming, and procedural programming. In JS, OOPS is implemented using class, object, instance, and inheritance, significantly simplifying the code.

```
class connections {
    constructor(source, target) {
      this.source = source;
      this.target = target;
    }
    connect() {
      console.log(`${this.source}    is    connected    to
${this.target}.`);
    }
}
```

In the above code, there is a class named "connections" that has a constructor and a method called "connect." The constructor takes two parameters, "source" and "target", and assigns them to the class properties "this.source" and "this.target", respectively. The "connect" method logs a message to the console using a template literal.

3.4 jQuery

It is one of the most popular javaScript libraries that provides an easy-to-use API for handling events and manipulating HTML documents. It was first introduced in 2006 by John Resig, and due to its simplicity and efficiency, it gained popularity quickly. It helps the developer perform simple tasks, including handling events and selecting and manipulating HTML events. It also contains a wide range of plugins for additional use. By using Jquery, the user can write efficient and simple code, saving time and effort.
We used jquery UI to clone draggable elements in the toolbox and append them to the droppable container using the draggable and droppable functionality. It is also used to add a custom context menu to the body when the user right-clicks on a control in the diagram and also for removing a custom context menu. It also uses jQuery to call addEndpoint() function on instances, which adds end points to the element with the given id. The code also uses jQuery to select elements with specific ids.

3.5 jsPlumb

JSPlumb is also a JavaScript library like Jquery that provides the developer a method to virtually connects element on a web application. Basically, it provides an API that defines connections between different components and it supports a variety of connections style and types. It also helps in creating drag-and-drop interfaces, interactive diagrams, and different types of flowcharts. It connects the different HTML elements by using various types of connectors by adding endpoints to HTML elements. Every element can have several endpoints which can either be a source, a target, or both. We can also add and remove endpoints for an element dynamically. This library also

has a different method to style the connections, including different paint styles, anchors, and overlays. It is built on HTML5 canvas element, which can be used with most modern browsers
This is a sample example of addEndpoint() function

```
instance.addEndpoint(id, {
          endpoint: "Dot",
          maxConnections: 2,
          anchor: ["Left"],
          isTarget: true,
          // isSource: true,
          connectionType: "red-connection",
```

JSPlumb can be used used together with jQuery to create dynamic and interactive content. like in our project we used JSPlumb to create visual connections between HTML elements, while we used JQuery to update the element's content according to user input and user interactions

These are the sample example of the use of draggable, and droppable functionality in our code, also with removing the custom context menu.

```
$("#toolbox .control").draggable({
      helper: "clone",
      containment: "body",
      appendTo: "#diagram",
    });
```

```
$("#diagram").droppable({
      drop: function (event, ui) {
        var id;
        var clone = $(ui.helper).clone(true);
        var elem = clone.attr("class");
        if (elem[8] == "r") {
          id = elem[8] + count_r.toString();
          count_r++;
        }
     .
     clone.attr("id", id);
      clone.appendTo(this);
```

```
      instance.draggable(id, { containment: true });
    },
});
```

```
$("body").on("click",    ".delete-connection",    function
(event) {
    instance.deleteConnection(window.selectedConnection);
  });
```

Chapter 4

Web-Enabled Simulators

For performing the experiments we have developed web-enabled simulators for each bridge where the circuit can be drawn by dragging and dropping the appropriate elements into the workspace, connections will be checked, and required calculations will be done. In this section, we will discuss the implementation of frontend and backend parts along with the simulation algorithm and results.

4.1 Frontend

The simulator UI is divided into three main parts, the toolbox, the workspace, and the input space, along with the observation table and formula. For dividing the entire window into three parts Bootstrap grid system is used.

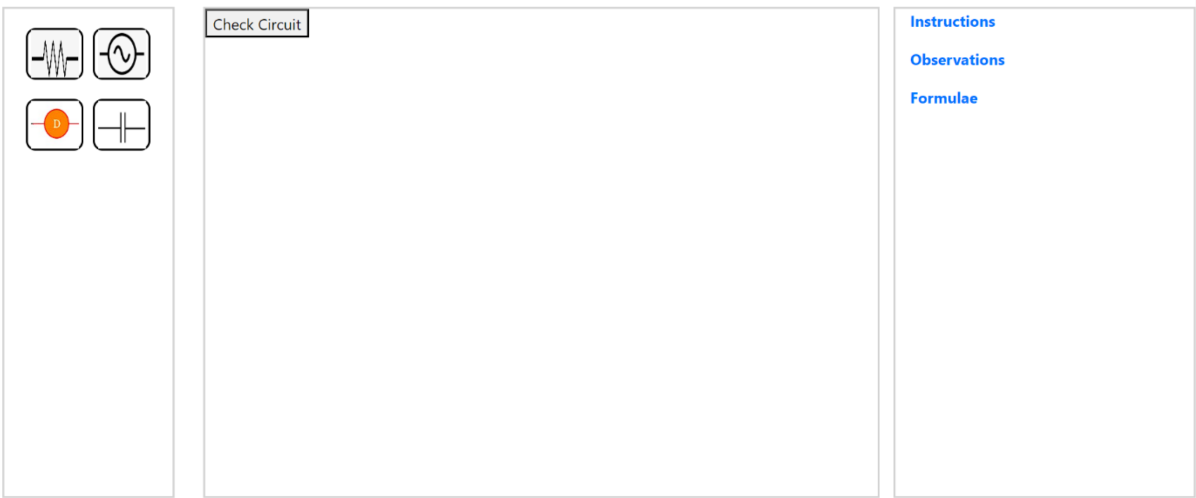


Figure 10: Frontend of Simulator

The toolbox contains the circuit elements like resistors, capacitors, inductors, power sources, null detectors, etc. Each block of the toolbox has been assigned a "control" class that makes it possible to drag it and clone once it drops into the workspace. The drag-drop and cloning properties are implemented using jsPlumb. After cloning the element into the workspace, a unique id is assigned to the element that will help in circuit verification, and terminals are added to it using jsPlumb "addEndpoint( )" function. Among the two terminals, one is the source, and the other is the target terminal, now a connection can originate from the source and terminate at the target point. After making all the connections, hitting the "check circuit" button executes a function in the backend that checks the circuit and enables the input space if connections are found correct. Now this input space uses the HTML tag "<form>" to take values of elements as input from the user. Decimal values can also be entered in the input field using the slider or input box within the permissible range. In the workspace below each element, there is a block that shows the tag for that particular element, and its value gets updated as the input field is changed. After taking input from the user, necessary balancing conditions are checked, and the value of the unknown variable is updated in the table if the bridge gets balanced.

4.2 Backend

The backend part consists of a circuit verification algorithm, functions for showing the tag for circuit elements, cloning elements from toolbox to workspace, and functions for checking the user input and balancing condition of the bridges. All the functions and algorithms are coded in the javascript programming language. As discussed earlier, a connection originates from the source terminal and terminates at the target, so it has two element IDs associated with it. The ids corresponding to each connection can be fetched using the "getConnections( )" method of jsPlumb.

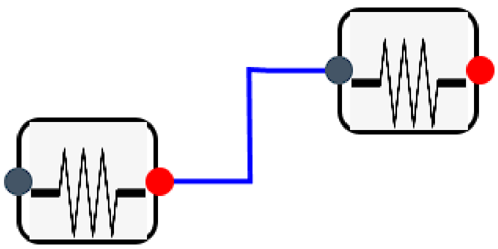


Figure 11: A connection (Red dot: Source, Black dot: Target)

Once all connections are completed, hitting the "check circuit" button will apply the verification algorithm to the received circuit information. It will fetch the source and target ids of all the connections and make a set of source elements and a set of target elements. So, if all the terminals of all elements are connected, then ideally, both sets should be equal. After checking this condition, it will form a map in which there will be a key for each element's id, and the value corresponding to that key will be a vector that contains the id of elements connected to the element working as key also a set will be maintained to keep a record of traversed elements. Now for all possible correct circuit combinations, we know which elements can be connected to a particular element. So, here it checks the same if the vector corresponding to the key has the right combination of elements or not. If it satisfies the condition, then it will move to the next untraversed element of the vector and again checks the same conditions for that element as well and this process is repeated until all the elements of the circuit are not traversed. When all the conditions are satisfied i.e. all the connections are right, the input field becomes active and on submitting the form, a function is executed in the backend that validates the input values and generates the error message accordingly. The value of the unknown variable is generated by the " Math.random( )" function of javascript. After this, the data of input fields are converted to float values, and necessary balancing conditions are checked. If the input values satisfy the balancing condition with the randomly generated value, then the values of all elements are added to the dynamic observation table else a sign is given to the user by the means of headphone sound or current or voltage value to adjust the input values in order to achieve balancing condition. Since the table is dynamic in nature, it adds a row on each balanced condition input combination. The experiment setup has the feature of deleting a particular element. Once an element is right-clicked, a button appears clicking on which executes "instance.remove(window.selectedControl)" internally and removes the selected element. Further, the UI has "Reset" and "Print Report" buttons which can be used to reset the workspace and print the report of the performed experiment.

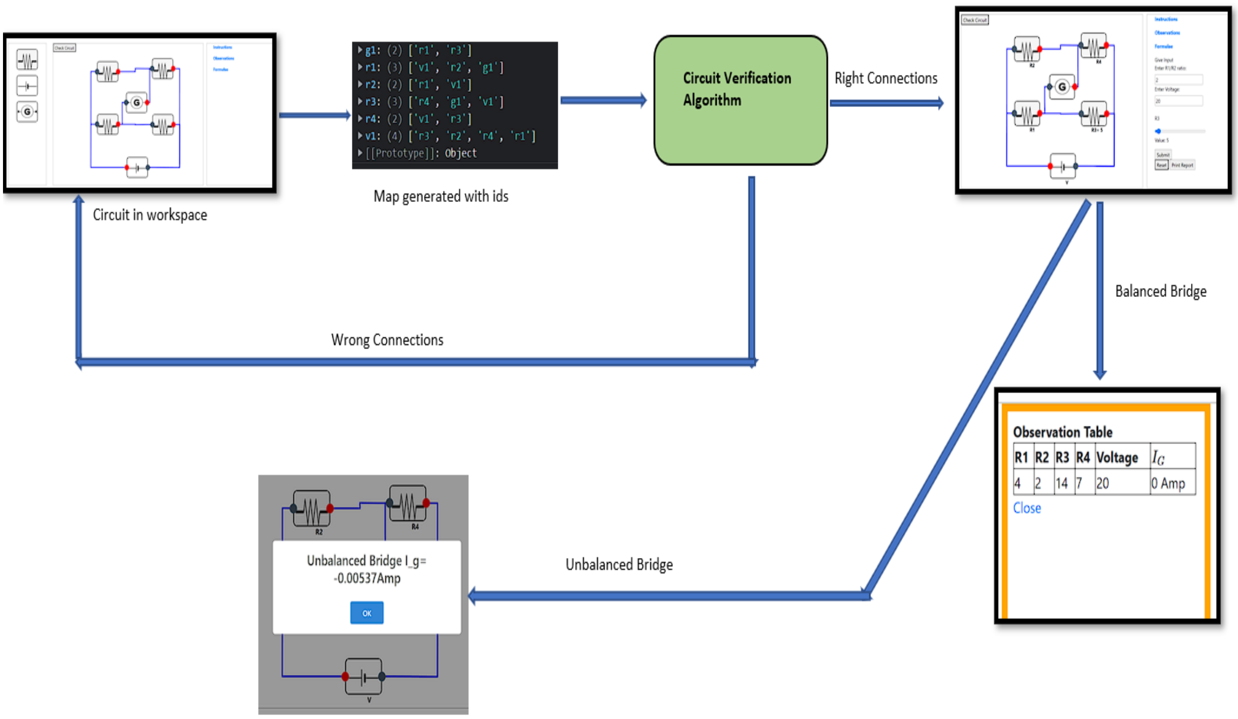
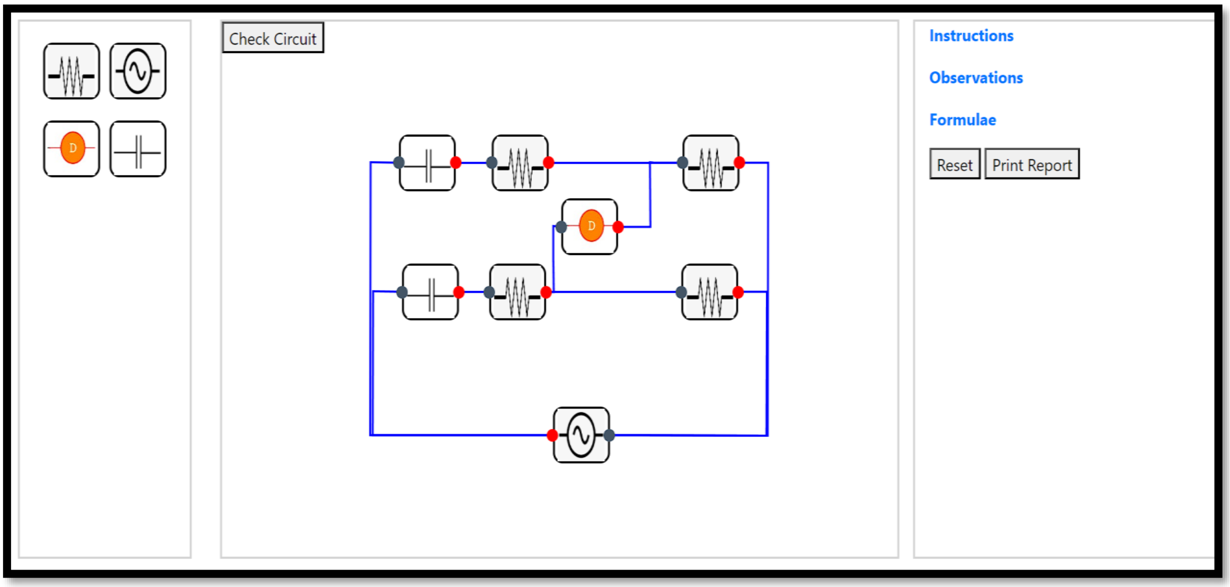
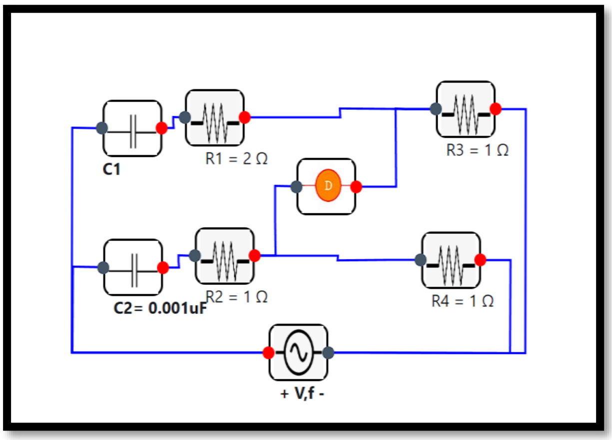


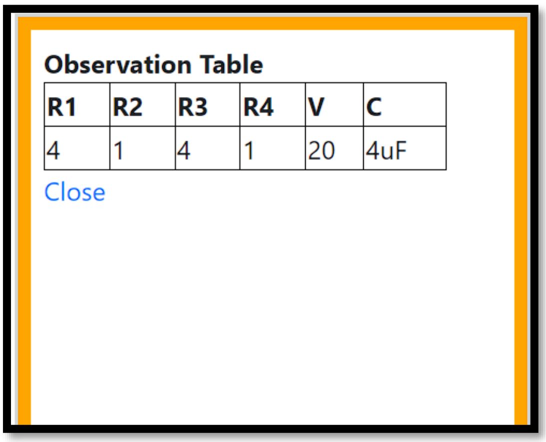
Figure 12: Simulation Algorithm Flowchart

4.3 Simulation Results

The user interface of the simulator is designed with HTML and CSS for the front end, while the back end is powered by JavaScript and jsPlumb to create circuits. Instructions, formulas, and a dynamic observation table can be accessed through three buttons on the left side of the window. Users can create circuits by dragging and dropping components from the toolbox to the workspace and connecting them. After completing the circuit, users can click the "check circuit" button to receive a warning message indicating the verification status of the circuit. Once calculations are done and user input is verified, a new row of values is added to the observation table.







Observation Table

R1	R2	R3	R4	V	C
4	1	4	1	20	4uF

Close

Figure 13: Simulation Results of Wien's Series Bridge

Chapter-5

Conclusion and Future Works

Recent research papers have highlighted the advantages and significance of virtual labs as an alternative approach to overcome challenges posed by traditional laboratory learning settings and enhance student motivation. In the field of electrical engineering, virtual labs have the potential to significantly improve practical learning by overcoming limitations of outdated tools and resources in educational facilities. They offer a practical solution for colleges and universities with limited facilities and resources to provide practical courses. Additionally, virtual labs eliminate risks associated with improper handling of lab equipment and potential electric shocks, resulting in reduced losses. These laboratories are available anytime for the students so they can perform experiments during their exam time also, which will help in their revision and make the concept stronger. Some modifications in this platform can also be used to make models and simulations for any research work.

In future work for this online lab, we can make a knob for inputting the values of elements and a needle-based detector to detect the null point so that it can visually show the null point, which will be more user interactive and give the feel of balancing the bridge. Also, a scope device can also be added so we can compare the waveforms of two quantities. We can also improve the simulation algorithms using advanced data structures and graph algorithms. It can be made more practical by using real-time data from laboratories and training our model using machine learning algorithms because here, we have not taken the resistance of wires. Also, the exact calculated values are not observed in practical experiments. This can give the feeling of having some errors as in actual experiments.

So in this way we can make these labs more practical, more user interactive, and economically viable.

Appendix I

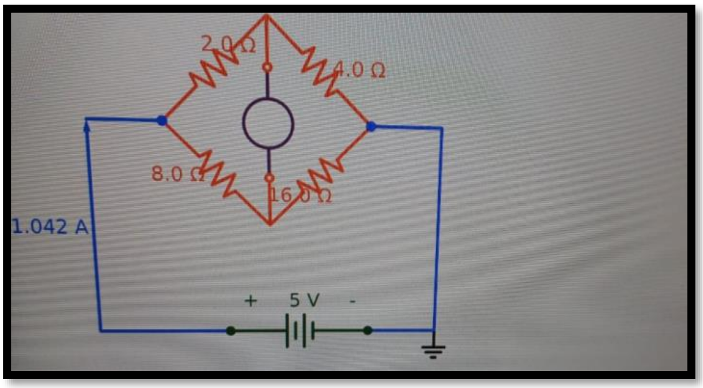
1) **jsPlumb Toolkit Version**:
   It's a paid version of jsPlumb that has the many advanced features like rotating a element at a particular angle and making flexible connections, etc.

2) **Schemdraw:**
   Schemdraw is a python package that is used to produce high quality circuit images. Python code is written to draw a circuit and label it, current directions and complex bridge an rectifier circuits can also be made with the help of it.
   Code for drawing a Wheatstone Bridge circuit:

```
d = schemdraw.Drawing()

W1 = d.add(elm.Wheatstone,vout=True,theta=(90),color=('red'),
    labels=(str(r2)+' \u03A9',str(r1)+' \u03A9',str(r3)+' \u03A9',str(r4)+' \u03A9'))
L1 = d.add(elm.Line().color('blue').left(d.unit*0.5).at(W1.N).idot())
L2 = d.add(elm.Line().color('blue').right(d.unit*0.5).at(W1.S).idot())
L3 = d.add(elm.Line().color('blue').down(d.unit*1.5).at(L2.end))
d.add(elm.Ground)
L4 = d.add(elm.Line().color('blue').tox(L2.start).left(d.unit*1).at(L3.end))
d += (B1 := elm.Battery().color('green').reverse().dot().idot().label('+ '+str(v)+' -'))
d += elm.Wire('-|',arrow='->').at(B1.end).to(L1.end).color('blue').label(str(round(final_cur, 3))+' A')
d += elm.Source().color('purple').endpoints(W1.vo1,W1.vo2)
```

   Circuit Drawn by Schemdraw:



3) **Assigning id to clone element:**
Following code has been used to assign unique id to the element cloned using jsPlumb that can be helpful for circuit verification:

```
drop: function(event, ui) {
  var id;
  var clone = $(ui.helper).clone(true);
  var s=clone.attr("class");
  if(s[8]=='r')
   { id= s[8]+count1.toString();
  count1++;}
  else if(s[8]=='v')
   { id= s[8]+count2.toString();
  count2++;}
  else if(s[8]=='g')
   { id= s[8]+count3.toString();
  count3++;}
  clone.attr("id", id);
  clone.appendTo(this);
```

4) **Code for updating dynamic table:**

```
    for (var i = 1; i <= 5; i++) {
     // Get the value of the current input field
     var data = arr[i-1];
     // Create a new table row element
     // Create a new table data element
     var cell = document.createElement("td");
     cell.innerHTML = data;

     // Add the table data to the table row
     row.appendChild(cell);

     // Add the table row to the table
     document.getElementById("myTable").appendChild(row);
    }


  var cell = document.createElement("td");
  cell.innerHTML = 0.00+" Amp";
  row.appendChild(cell);
  document.getElementById("myTable").appendChild(row);
```

References

[1]https://jsplumbtoolkit.com/community

[2]https://getbootstrap.com/

[3]https://circuitglobe.com/andersons-bridge.html

[4] Yang, Woong Lee, Soo-Hong Zhu, Jin Hwang, Hyun-Tae. (2016). Development of Web-based Collaborative Framework for the Simulation of Embedded Systems. Journal of Computational Design and Engineering. 3. 10.1016/j.jcde.2016.06.004.

[5]https://journals.sagepub.com/doi/pdf/10.1177/0020720918775041

[6]https://www.researchgate.net/publication/366428022_Virtual_Labs_in_Electrical_Engineering%20_Education_During_the_Covid-19_Pandemic_A_Systematic_Literature_Review

[7]https://medium.com/@priyeshayadav9192/creating-a-dynamic-flow-

[8]https://www.electrical4u.com/maxwell-bridge-inductance-capacitance-bridge/

[9]https://circuitglobe.com/wiens-bridge.html