**Lab 11: Worst Case and Monte Carlo Analysis**

1. **Background Information**

As an engineer, you will be designing devices and systems that are themselves made of smaller devices and systems. For instance, you might be an electrical engineer designing a circuit using resistors, capacitors, and batteries or a civil engineer designing a bridge using different types of concrete and steel beams. Each of these elements is a device in and of itself, with a variety of different considerations that went into their design and fabrication. Understanding the capabilities and limitations of the components you use in your design is a critical skill to develop.

In this lab, you will be investigating the effect of tolerance on a simple two resistor series circuit. While this circuit is very simple, it is actually used very often to help reduce a larger voltage to a more useful level. From the basic electricity experiment in Engineering Foundations, we know that because of Kirchhoff’s voltage law, in a series circuit each element will have a part of the source voltage applied across them. For instance, in the circuit shown below, 2 V will be seen on the 1kΩ resistor and 4V will be seen on the 2kΩ resistor. This can be calculated using Ohm’s Law and the facts that series resistors can be added together to form an equivalent resistance and the current through series elements is the same.

|  |  |
| --- | --- |
| 2kΩ  1kΩ  6V | **Ohm’s Law:**  V = I\*R  **Series Resistors:**  Req = R1 + R2 |
| **Figure 1: Series Resistor Circuit** | |

However, what happens to the voltage on the 2kΩ resistor if the resistance values are not actually what they claim to be? This is a real concern as for most components, be they electrical, mechanical, or material, will have a tolerance specified for their performance. In the case of a resistor, the tolerance will specify the range of resistance values in which any given component might fall. For a hydraulic piston, the tolerance might specify the range of force that could be expected when the piston is activated. For a steel beam, the tolerance might specify the range of maximum sheering force the beam can experience before deforming. If you do not take into account the variation inherent in any type of component, you may experience catastrophic failure of your device.

1. **Necessary Calculations**

As mentioned above, we will be dealing with a simple two resistor series circuit and calculating the voltage on different elements based on the values of the resistors. While you have done these calculations during the Basic Electricity experiment in Engineering Foundations, we’ll remind you of the calculations here. Please refer to Figure 2 below when reviewing the calculations.

The voltage on the first resistor:

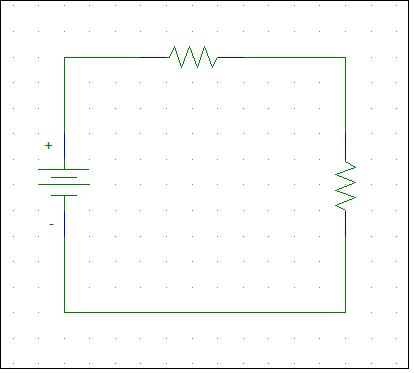
The voltage on the second resistor:

The tolerance of a resistor is usually expressed as a percentage. Therefore, if resistor R1 has a nominal value of 1kΩ and a tolerance of 10%, the resistor might actually fall anywhere within a range of 900Ω to 1100Ω.

R1min = R1nominal – R1nominal\*tolerance = 1000 – (1000\*0.1) = 900 Ω

R1max = R1nominal + R1nominal\*tolerance = 1000 + (1000\*0.1) = 1100Ω

+ V1 -



+

V2

-

R2

R1

Vsource

**Figure 2: General Series Resistor Circuit**

In order to understand how tolerance affects the performance of this simple circuit, we will perform two different analyses. First, a worst case analysis will be performed where we will explore the full range of possible values for each resistor. Second, we will perform a Monte Carlo analysis, during which we simulate the construction of a large number of circuits with different components in order to understand what we would expect to if we built the circuit with real components.

1. **Worst Case Analysis**

To conduct the worst case analysis, create a script that performs the following actions:

1. Prompts the user for necessary values:
   * Value of resistor 1
   * Value of resistor 2
   * Tolerance of resistor 1
   * Tolerance of resistor 2
   * Source voltage
   * Acceptable voltage tolerance on resistor 1
   * Acceptable voltage tolerance on resistor 2

***NOTE: pay attention to how you enter the tolerances for the resistors and the resistor voltages. Do you want the user to type in the value as a percent or a decimal?***

1. Compute the ranges for the resistor values as well as the acceptable resistor voltages
2. Set up a nested for loop structure to do the following:
   * The outer loop should go through the range of values for resistor 1 in 500 steps
   * The inner loop should go through the range of values for resistor 2 in 500 steps
   * For every pair of resistor values, compute the resistor voltages and determine whether the voltages are acceptable
   * Store a value of 255 into a 2-D array if the voltages are acceptable and a value of 0 into the array if the voltages are not acceptable  
       
     The structure of the array is as follows:

Minimum R1

Maximum R2

Minimum R1, R2

|  |  |  |  |
| --- | --- | --- | --- |
|  |  |  |  |
| Maximum R1  Minimum R2 |  |  |  |
|  |  |  |  |
|  |  |  | Maximum R1, R2 |

***NOTE: you will likely need to use separate variables as indexes into the 2-D array as your resistor values may not be whole numbers***

1. Display your results as an image using the following command:  
     
   **imshow(uint8(results))**  
     
   where you should replace results with the name of your 2-D array

When you run your script, what you will see is a black and white image. The white areas (values of 255) indicate where the circuit performed within acceptable ranges. The black areas (values of 0) indicate where the circuit did not perform within acceptable ranges.

Once you have completed your code, run it for the following sets of values and paste your resulting images below:

* Trial 1:
  + R1 = 1000 Ω
  + R2 = 2000 Ω
  + Voltage Source = 6V
  + Tolerance of R1 = 10%
  + Tolerance of R2 = 5%
  + R1 Voltage Tolerance = 1%
  + R2 Voltage Tolerance = 1%
* Trial 2:
  + R1 = 1000 Ω
  + R2 = 2000 Ω
  + Voltage Source = 6V
  + Tolerance of R1 = 1%
  + Tolerance of R2 = 1%
  + R1 Voltage Tolerance = 1%
  + R2 Voltage Tolerance = 1%

**MATLAB Plot of Results for Trial 1:**



**MATLAB Plot of Results for Trial 2:**

1. **Monte Carlo Analysis**

To conduct the Monte Carlo analysis, add code at the end of your script to do the following:

1. Set up a for loop that:
   * repeats 10000 times
   * computes random values for resistor 1 and resistor 2 using the randn function as follows:  
       
     ***resistor\_value* = *nominal\_value* + *range*/2\*randn(1);**  
       
     where ***resistor\_value*** is the randomly computed value for the resistor, ***nominal\_value*** is the expected value for the resistor, and ***range***is the product of the tolerance for the resistor and the nominal resistor value.
   * stores the voltages on resistors 1 and 2 in vectors
   * counts the number of circuits that do not meet the voltage tolerances
2. Create histograms for the computed voltages on the two resistors using the **hist** command:  
     
   **figure; hist(*voltage\_1*,100);**  
     
   This command creates a histogram of the voltages in ***voltage\_1*** (replace this with your variable) with 100 different sections in a new figure window  
     
   ***Don’t forget to label and title your histogram!***
3. Compute and display the probability of having a bad circuit using the following formula:
4. Run your script for the two trials listed in part C and include your histogram plots and probabilities below

|  |  |
| --- | --- |
| **Trial** | **Probability of Bad Circuit** |
| 1 | 78.66% |
| 2 | 3.4% |

**MATLAB Histogram Plots for Trial** **1:**



**MATLAB Histogram Plots for Trial 2:** 



|  |  |
| --- | --- |
| **Question:** Based on these results, what effect does decreasing the tolerance have the performance of the circuit? Why would you not always use components with the smallest possible tolerance?   |  | | --- | | Decreasing the tolerance increases the performance of the circuit. One would not always use the smallest possible tolerance because the price would increase for the lower tolerance resistors. | |

**Paste your final script below:**

%Models Lab 11

clear; clc;

rng('shuffle');

%input

r1 = input('Resistance of resistor 1 (Ohm): ');

r2 = input('Resistance of resistor 2 (Ohm): ');

voltage\_source = input('Source voltage (V): ');

r1\_tol = input('Tolerance of resistor 1 (%): ');

r1\_tol = r1\_tol/100;

r2\_tol = input('Tolerance of resistor 2 (%): ');

r2\_tol = r2\_tol/100;

r1\_vtol = input('Acceptable voltage tolerance of resistor 1 (%): ');

r1\_vtol = r1\_vtol/100;

r2\_vtol = input('Acceptable voltage tolerance of resistor 2 (%): ');

r2\_vtol = r2\_vtol/100;

voltage\_check = zeros(500,500);

%analysis

r1\_min = r1 - r1\*r1\_tol;

r1\_max = r1 + r1\*r1\_tol;

r2\_min = r2 - r2\*r2\_tol;

r2\_max = r2 + r2\*r2\_tol;

r1\_nomvoltage = r1/(r1 + r2)\*voltage\_source;

r2\_nomvoltage = r2/(r1 + r2)\*voltage\_source;

r1\_vmin = r1\_nomvoltage - r1\_nomvoltage\*r1\_vtol;

r1\_vmax = r1\_nomvoltage + r1\_nomvoltage\*r1\_vtol;

r2\_vmin = r2\_nomvoltage - r2\_nomvoltage\*r2\_vtol;

r2\_vmax = r2\_nomvoltage + r2\_nomvoltage\*r2\_vtol;

R1 = linspace(r1\_min,r1\_max,500);

R2 = linspace(r2\_min,r2\_max,500);

for p = 1:500

for k = 1:500

r1\_voltage = R1(p)/(R1(p) + R2(k))\*voltage\_source;

r2\_voltage = R2(k)/(R1(p) + R2(k))\*voltage\_source;

if r1\_voltage >= r1\_vmin && r1\_voltage <= r1\_vmax && r2\_voltage >= r2\_vmin && r2\_voltage <= r2\_vmax

voltage\_check(p,k) = 255;

end

end

end

figure(1)

imshow(uint8(voltage\_check))

r1\_values = zeros(10000,1);

r2\_values = zeros(10000,1);

count\_bad = 0;

for l = 1:10000

rand\_r1 = r1 + r1\*r1\_tol/2\*randn(1);

rand\_r2 = r2 + r2\*r2\_tol/2\*randn(1);

r1\_volt = rand\_r1/(rand\_r1 + rand\_r2)\*voltage\_source;

r2\_volt = rand\_r2/(rand\_r1 + rand\_r2)\*voltage\_source;

r1\_values(l) = r1\_volt;

r2\_values(l) = r2\_volt;

if r1\_volt < r1\_vmin || r1\_volt > r1\_vmax || r2\_volt < r2\_vmin || r2\_volt > r2\_vmax

count\_bad = count\_bad + 1;

end

end

figure(2)

hist(r1\_values,100);

xlabel('\bfVoltage (V)','FontSize',14);

ylabel('\bf# of Circuits','FontSize',14);

title('\bfResistor 1 Voltages','FontSize',20);

figure(3)

hist(r2\_values,100);

xlabel('\bfVoltage (V)','FontSize',14);

ylabel('\bf# of Circuits','FontSize',14);

title('\bfResistor 2 Voltages','FontSize',20);

probability\_bad = count\_bad/10000\*100

1. **To be turned in:**

* You will need to upload this word document with all tables, questions, and figures included and the m-file for your script.