ECSE 343 Project

Circuit Analysis

Electronic circuits are designed to meet the required specifications by choosing a topology as well as nominal values for the component parameters, such as the resistances of resistors. However, when the circuit is manufactured, the actual parameters of the components will vary due to manufacturing variations. Design for Manufacturability (DFM) requires us to assess the impact of this variation on the key figures of merit of the circuit. In general, the random distributions of the parameters are assumed to be known, and the goal is to compute the statistical properties of the key figure of merit using Uncertainty Quantification tools. This is then used to estimate the manufacturing yield.

Your goal is to perform the statistical analysis of the circuit output, V_{out}, for the circuit shown in Figure 1. The input voltage is a sinusoidal signal with an amplitude of 1V and a frequency of 1000Hz. The nominal values of the circuit components are $R_1 = 30k\Omega, R_2 = 18k\Omega, C_1 = 0.01uF, C_2 = 0.0047uF$.

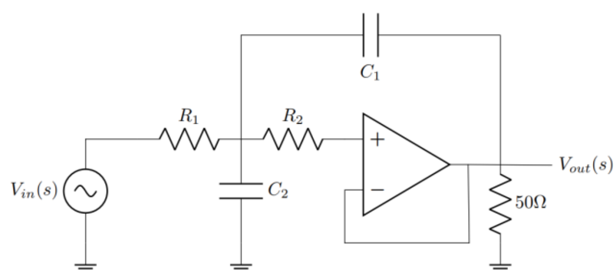

Figure 1: Sallen-Key Filter.

The circuit components (R_1, R_2, C_1, and C_2) have a $\pm$5% uncertainty in their values, which is uniformly distributed around their nominal values. For example, the value of R_1 can be considered to be a uniformly distributed random variable between 28.5kΩ and 31.5kΩ, i.e., (30-5% of 30)kΩ to (30 + 5% of 30) kΩ. Each circuit component can be viewed as uniformly distributed independent random variables. For example, in the case of R1, it is a uniformly distributed random variable with $R_1 \sim U(28.5k\Omega, 31.5k\Omega)$. Since the circuit components are random variables, V_{out} can also be regarded as a random variable. Your goal is to compute the mean, standard deviation and probability distribution of $|V_{out}|$. You will be given a code that computes V_{out} given a set of component parameters.

There are several approaches to performing the statistical analysis of the circuits; one popular method is the Monte Carlo method. Employing the Monte Carlo method directly to a circuit can be computationally expensive, especially in the case of large circuits; this is because Monte Carlo requires us to solve the circuit many times, often thousands of times. Other examples of methods include building the surrogate model of V_{out} as a function of circuit components using multivariable regression approaches such as Polynomial Chaos, Neural Networks or any other method of your choice. One possible choice is to obtain the expression of V_{out} as a function of the circuit components for the given circuit; however, do not take this approach. Instead, use a generic model based on multivariable regression approaches such as Polynomial Chaos, Neural Networks or any other method you choose. Once the surrogate model of the V_{out} is obtained, the Monte Carlo method can be employed on the surrogate model, which is cheaper than solving the circuit.

Your tasks in this project are listed below.

a. Search the literature to identify and evaluate possible approaches for solving this problem, including the Monte-Carlo-based methods.

b. Implement a Monte-Carlo-based approach as well as another approach that you choose based on the project goals above (primarily reducing the number of circuit evaluations required for a given accuracy).

c. Test both the Monte-Carlo-based method that you implemented as well as the other approach that you chose to implement. Vary the number of function evaluations and comment on each approach's accuracy. Compare the mean, standard deviation, and pdf obtained with your method with the Monte-Carlo approach. Comment on how the accuracy changes for different numbers of function evaluations.

Deliverables

Your deliverables will include two items. The first is a written report (maximum five (5) pages, but no minimum as long as you explain the required content). The second is the MATLAB code that you developed. Make sure the code is well commented. Your report should be structured like an IEEE journal paper with an abstract, introduction, body, and conclusion. It must address at least the following points:

1. A description of the theoretical foundation of both methods that you have implemented.
2. A description of the algorithm used for each method in the form of a pseudo-code.
3. A justification for the choice of the second non-Monte-Carlo method.
4. A discussion about the advantages and disadvantages of each method.
5. A discussion of the numerical experiments that were conducted to validate, compare, and evaluate the algorithms used.
6. A description of the contribution of each team member to the project.
7. A final discussion and conclusion.
8. A maximum 3 min video presentation highlighting your methodology and the results obtained.

Grading

The grading scheme is as follows:

- 45% submitted code: Does the code work? Is it efficient? Is it well structured? Is it clear and well-commented? The grade will be split evenly for the Monte Carlo and other approach (es) you choose.
- 45% Project report (see the file “grading rubric” for details on the report grade).
- 10% for a 3 min video presentation (see the file “grading rubric” for details on the presentation grade).

Appendix

Below is an example of the MATLAB function that can compute the values of the Vout for the circuit shown in Figure 1 for a given value of R1, R2, C1, and C2.

```
function [Vout] = simulate_Sallenkey(R1, R2,C1, C2)

% This function will solve the circuit shown in Figure 1
at Vin =1V and Frequency =1000Hz for the input values of
R1, R2, C1 and C2;

% Input: The element values for R1, R2, C1 and C2
% Output: Voltage at output node.

End
```