

# A Bayesian Optimization Framework for Analog Circuits Optimization

Shady A. Abdelaal\*, Ahmed Hussein\*, and Hassan Mostafa\*†

\*Electronics and Communications Engineering Department, Cairo University, Giza 12613, Egypt

†University of Science and Technology, Nanotechnology and Nanoelectronics Program, Zewail City of Science and Technology, Giza 12578, Egypt

Email: shady.a.abdelaal@gmail.com, ahmed.hussien60@gmail.com, and hmostafa@uwaterloo.ca

**Abstract**—The growing complexity of analog circuits poses challenging constraints on analog simulation tools. Simulation based optimization approaches have gained a lot of interest to cut down the analog circuit design time and complexity. One of these approaches is the Bayesian optimization (BO) approach, which represents the analog circuit as a black box function, and incorporates optimization goal and constraints aiming to reach the optimum design parameters with the least possible simulation iterations. In this paper, a BO approach for automated sizing of analog circuits is discussed. The proposed approach uses Gaussian Process (GP) as a surrogate model and utilizes SOBOL sampling. The proposed algorithm is validated on a two-stage op amp benchmark circuit and compared to the literature work.

**Index Terms**—Bayesian optimization, Analog circuits optimization.

## I. INTRODUCTION

Challenges in analog circuit design are continuously increasing with the emerging new technology nodes, the growth in operational speed, and the increasing complexity of electronics systems. This results in making manual analog circuit design a very challenging and time consuming process. On the other hand, the demand for high performance and low power designs have been increasing. Which in turn, increases the need for faster and smarter analog circuit design techniques and tools with more degrees of automation. As a result, automated analog circuit design tools attracted the interest of industry as well as academia.

The automation for analog circuit optimization problems can be grouped into two categories: model based and simulation based [1]. Model based approaches focus on generating simplified models that can represent the performance of the circuit, cutting out the long simulation times. Geometric programming is an example of model based approaches, where the circuit metrics are modeled as posynomial and monomial functions. The analog circuit sizing problem can then be represented as a convex optimization problem. Once the posynomial models are obtained, the global optimum could be reached [2].

Alternatively, simulation based algorithms depend on the circuit simulations. They deal with the objective functions and constraints as black box functions, which are evaluated by the circuit simulators. The improvement emerges from the application of various statistical or machine learning approaches to propose new candidates aiming for a more efficient exploration

of the design space to reach the optimum design point. Some examples of these approaches are simulated annealing, evolutionary algorithm, particle swarm intelligence, genetic algorithms, and multi start optimization algorithm. That is in addition to a machine learning technique called Bayesian Optimization (BO), which is a supervised machine learning method, better known for optimizing expensive black box functions in case the closed form expression is unavailable [3]. Previous research applying BO to circuits design proves that it achieves better results, specially in convergence rate, in comparison with other approaches used in analog optimization [4] [5].

In this paper, an efficient BO implementation for automated analog circuit sizing is presented. The implementation relies on an efficient initial random sampling method to build an initial representation of the surrogate model, then iteratively improves a Gaussian Process (GP) model, with an acquisition function used to propose the next design point to evaluate.

This paper is organized as follows. In Section II, the problem definition of analog circuit optimization and its challenges are presented. In Section III, BO theory and components are described. In Section IV, the approach taken to apply it to analog circuit optimization is presented, while the results are provided in Section V, and the work is concluded in Section VI.

## II. PROBLEM DEFINITION

In this section, the problem definition of the analog circuit optimization is presented. Analog circuit optimization - mainly transistor sizing - problem is a non-convex one, like many engineering problems in several fields. Since most of realistic systems do not tend to have linear responses to its control parameters. That is why non-convex optimization attracted the attention of several researchers. Analog circuit optimization problem can be restated as an effort to maximize an expensive-to-evaluate, black box function  $f$  (the circuit performance in our case). Note that, we do not have the functional form of  $f$  and our only approach is to evaluate the function at a sequence of test points. Aiming at reaching a near-optimal design point after a small number of trial evaluations (called arms). This means that our aim is to reach the optimum design point, that is; one that satisfies our design requirements, in the least

number of iterations possible. Formally, it can be represented as an optimization problem with constrains.

$$\begin{aligned} & \text{maximize } f(x) \\ & \text{s.t. : } c_i(x) < 0 \\ & \quad \forall i \in 1 \dots N_c \end{aligned} \quad (1)$$

Where  $x \in R^d$ ,  $R^d$  is the design space, with  $d$  representing the circuit's design variables.  $f(x)$  denotes the main circuit objective (i.e: the DC gain for an op amp or the efficiency for a power amplifier) and the constraints over other performance metrics (i.e: total power consumption or total area) are represented by  $c_i(x) < 0$  [1].

### III. BAYESIAN OPTIMIZATION

In this section, a brief introduction to BO theory using GP, the system parameters and components are presented.

#### A. Bayes' Theorem

BO originated from a well-known equation in probability theory and statistics, called Bayes' theorem. Given a set of training design parameters, the observed metrics are  $D = \{x_i, y_i\}$ . Let us define  $p(h)$  as the prior probability which models the belief about the objective function  $f(x; h)$  prior to the observation of  $D$ , while  $p(D|h)$  is the likelihood function which can represent how probable high efficiency is, given the function  $f(x; h)$  [4]. Bayes' theorem can be applied to evaluate the posterior probability  $p(h|D)$  for the function after observing  $D$  as:

$$p(h|D) = \frac{p(D|h)p(h)}{p(D)} \quad (2)$$

$$p(D) = \int p(D|h)p(h)dh \quad (3)$$

It can be proved that the integrated term is a normalization constant. So, the posterior probability is formulated as:

$$p(h|D) \propto p(D|h)p(h) \quad (4)$$

#### B. Bayesian Optimization Algorithm

In the field of intelligent machine learning algorithms, BO is one of the supervised learning algorithms, best used for the optimization of expensive, non convex objective functions in multi dimensional space. Since the aim is to optimize  $f$  with the least possible number of evaluations, a model has to be built to help us extrapolate and deduce the values of  $f$  at points we have not yet evaluated. In BO, this is called the surrogate model. The surrogate model should also indicate the uncertainty of its predictions in the form of a posterior distribution over function  $f(x)$  at points  $x$ .

In BO literature, the surrogate model is typically a GP, due to its flexibility and tractability. In a GP, the posterior distribution on any finite set of points is a multivariate normal distribution. A GP model is defined by a mean function  $\mu(x)$  and a covariance kernel  $k(x, x')$ , which means that

a mean vector  $(\mu(x_0), \dots, \mu(x_k))$  and covariance matrix  $\Sigma$  with  $\Sigma = k(x, x')$  can be calculated for any set of points  $[(x_1, \dots, x_k)]$ . Using a GP surrogate model for  $f$  means that we assume  $[(f(x_1), \dots, f(x_k))]$  is multivariate normal with a mean vector and covariance matrix determined by  $\mu(x)$  and  $k(x, x')$ . Posteriors are an indication on the "belief" a model has about the values of the function at a point (or a set of points), based on the data it has been trained with till that moment. That is the posterior distribution over the outputs conditional on the data observed so far. The GP posterior is relatively cheap to evaluate, so it is used to suggest points from the search space where the function evaluation is likely to result in an improvement.

Acquisition functions are responsible for proposing sampling points in the search space. They trade off exploitation and exploration and propose a candidate that is believed to achieve the best possible improvement given the current model built so far. Exploitation means sampling where the surrogate model's prediction indicates a high objective, while exploration means sampling where the prediction uncertainty is high. Such two cases would result in a high acquisition function values, and the goal is to maximize the acquisition function in order to decide which point to sample next (i.e, the point that would have the most expected improvement). That point would be ideal to be the next sampling point. To avoid getting stuck in a local maximum (over exploitation) and avoid not making the best of a reached, good value (over exploration), a good balance between exploration and exploitation is required. The objective function  $f$  will next be sampled at

$$x_t = \text{argmax}_x u(x|D_{1:t-1}) \quad (5)$$

$u$  is the acquisition function and  $D_{1:t-1}$  is the  $(t-1)$  samples drawn from  $f$  till this iteration. Examples of frequently used acquisition functions are probability of improvement (PI) and expected improvement (EI) [3]. In our implementation, we will use EI which is the most widely used.

### IV. PROPOSED FRAMEWORK

As mentioned in section III. BO is best used to optimize a black box function. Here, we benchmark our implementation using the op-amp circuit shown in Figure 1. The input parameters to optimize the circuit as well as the main objective and the constraints are shown in Figure 2. The process used is 180nm. The optimization is performed over 11 design variables, which are  $W$ ,  $L$  of all MOSFET devices, the compensation capacitor  $C_c$  and resistor  $R_c$ . The load capacitor  $C_L$  equals 1 pF. The performance metrics used are voltage gain ( $A_v$ ) gain bandwidth product (GBW), and phase margin (PM). The design specifications are listed in Eq. 6:

$$\begin{aligned} & \text{maximize } \text{Gain} \\ & \text{s.t. : } \text{GBW} > 40\text{MHz} \\ & \quad \text{PM} > 60 \end{aligned} \quad (6)$$

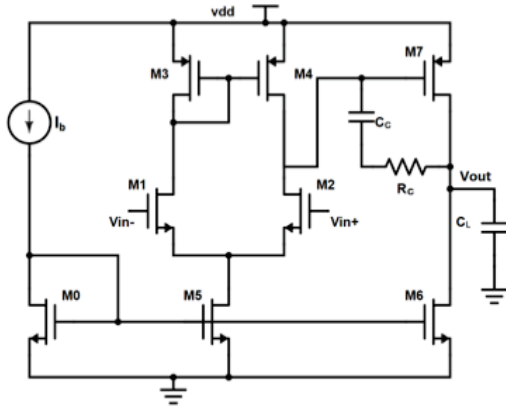


Fig. 1. A two-stage operational amplifier [2]

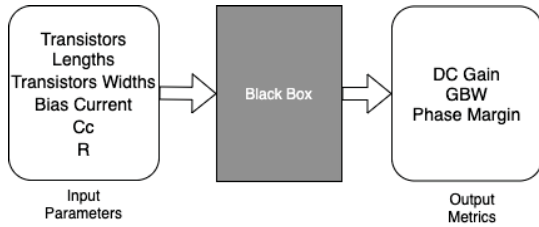


Fig. 2. Black Box system representation

The proposed BO approach is explained in algorithm 1, and the framework flow chart is summarized in Figure 3. First, a random set of training data points is generated using SOBOL. That is a search paradigm for quasi-random search that achieves a more uniform coverage than purely random search. The generated parameter values (arms) are run through the circuit simulator, and the values are used to build the model. Then, based on the acquisition function, new set of arms are selected for evaluation, passed again to the simulator, iteratively optimizing the GPEI model till the stop criteria are reached. Stop criteria can either be a maximum number of trials or a minimum expected improvement value. As an attempt to reduce the time taken to reach the optimum arm, the simulations are performed by the simulator with reduced accuracy. This adds an error term to the mean objective value obtained, which is treated as a noise term in the GP and would not affect the performance of the BO loop. The more accurate simulation values are then obtained only after settling on the optimum arm achieving the optimum metrics.

### Algorithm 1 Bayesian Optimization

- 1: Initialize a set of data points (arms) selected randomly from the search space using SOBOL.
- 2: Simulate the training arms with the SPICE simulator with relaxed accuracy.
- 3: Build the probabilistic surrogate model.
- 4: **for**  $iteration = 1, 2, \dots, N$  **do**
- 5:     Find  $arm_i$  that maximizes the acquisition function.
- 6:     Evaluate  $y = f(arm_i)$  through the SPICE simulator with relaxed accuracy.
- 7:     Update the surrogate model.
- 8: **end for**
- 9: Find the best arm recorded during optimization trials.
- 10: **Return** the best arm, and the best metrics achieved, evaluated by the SPICE simulator with best accuracy.

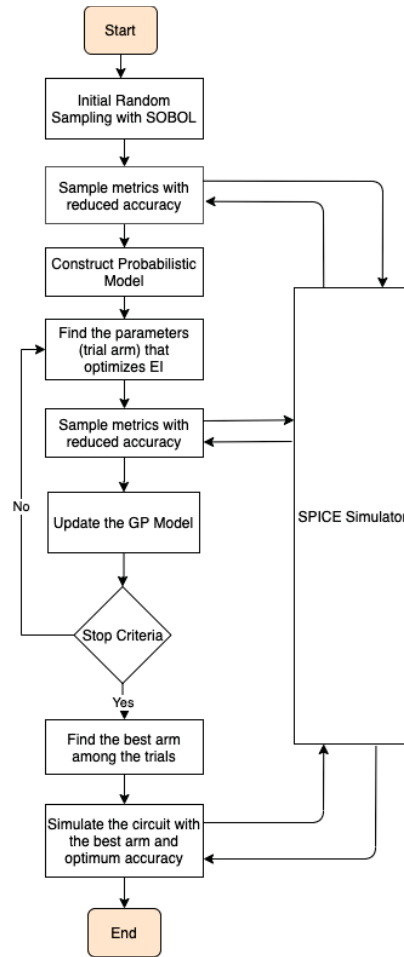


Fig. 3. Framework Flow Chart

## V. RESULTS

In this section, the effectiveness of the algorithm is demonstrated on a sample circuit, showing the algorithm performance in action with plots. The proposed algorithm is implemented in python with BoTorch framework [8]. The test circuit used is a

two stage op amp with miller compensation, which is a widely accepted benchmark in the literature. Let us demonstrate the algorithm performance on a sample run. Figure 4 shows the trace of the accumulated maximum DC gain achieved, which shows that the algorithm quickly reached a maximum DC gain value, approaching the optimum. However, obtaining the maximum gain does not necessarily mean that this is the optimum design point, because other constraints might not have been met. The goal is to find the maximum DC gain possible while still meeting the constraints. Figure 5 shows the achieved DC gain values vs simulation trials, with colour coding showing whether this arm has met the constraints (orange) or not (blue).

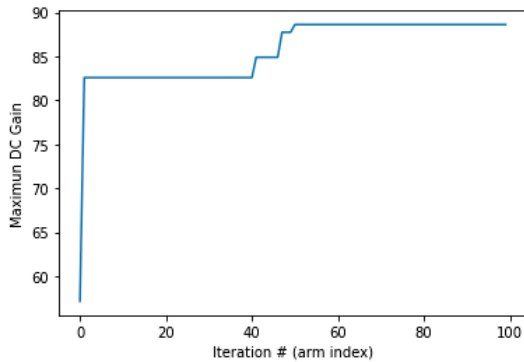


Fig. 4. Maximum DC Gain Achieved vs trial iterations

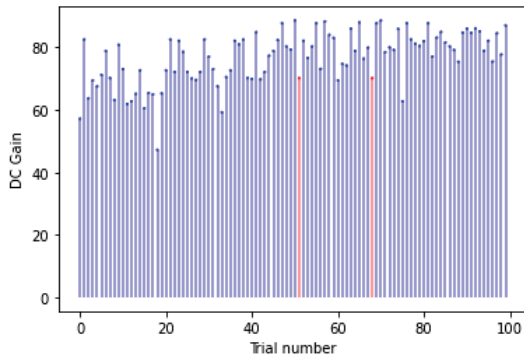


Fig. 5. DC Gain Progress vs trial iterations

The algorithm performance can be evaluated by considering the metrics achieved (gain, GBW and PM) as well as the number of iterations/trials needed to reach these values. The algorithm is run 10 times in order to calculate an average to eliminate random fluctuations. The total number of runs is kept under 100 total trials (including the random initial sampling). The results of the ten runs are shown in table I. While table II shows a comparison between the mean of the values obtained in this work and the values obtained in [2] and [9].

TABLE I  
METRICS OBTAINED FROM 10 RUNS

	Gain	PM(deg)	GBW(MHz)	Power ( $\mu$ W)
Objective	>70	>60	>40	< 170
best arm	75.35	61.24	45.11	123.156
worst arm	68.39	67.09	70.24	336.78
best values	75.35	73.38	67.09	123.156
worst values	68.39	61.66	45.11	336.78
mean	70.91	66.6	60.2	208.143
median	70.19	67.67	62.25	209.34

TABLE II  
TWO STAGE OP AMP OPTIMIZATION COMPARISON WITH [2] AND [9]

Performance Metric	Spec	This work		Obtained in [2]	Obtained in [9]
		Best Arm	Mean		
Gain	>70	75.35	70.91	70.1	69
GBW(MHz)	>40	45.11	60.2	29.1	30.6
PM(deg)	>60	61.24	66.6	60	61
Power ( $\mu$ W)	<170	123.156	208.143	173.5	177

## VI. CONCLUSION

In this paper, a Bayesian optimization framework for analog circuit optimization is presented. A carefully designed optimization loop with appropriate hyperparameters is proposed to achieve a good balance between exploration and exploitation. The proposed algorithm is applied on a two-stage op amp to validate its effectiveness. Finally, the algorithm's performance tracing is illustrated with plots.

## REFERENCES

- [1] W. Lyu et al., "An Efficient Bayesian Optimization Approach for Automated Optimization of Analog Circuits," in *IEEE Transactions on Circuits and Systems I: Regular Papers*, vol. 65, no. 6, pp. 1954-1967, June 2018.
- [2] A. Sayed, A. N. Mohieldin and M. Mahroos, "A Fast and Accurate Geometric Programming Technique for Analog Circuits Sizing," 2019 31st International Conference on Microelectronics (ICM), Cairo, Egypt, 2019, pp. 316-319.
- [3] B. Shahriari, K. Swersky, Z. Wang, R. P. Adams and N. de Freitas, "Taking the Human Out of the Loop: A Review of Bayesian Optimization," in *Proceedings of the IEEE*, vol. 104, no. 1, pp. 148-175, Jan. 2016.
- [4] P. Chen, B. M. Merrick and T. J. Brazil, "Bayesian Optimization for Broadband High-Efficiency Power Amplifier Designs," in *IEEE Transactions on Microwave Theory and Techniques*, vol. 63, no. 12, pp. 4263-4272, Dec. 2015.
- [5] S. J. Park, B. Bae, J. Kim and M. Swaminathan, "Application of Machine Learning for Optimization of 3-D Integrated Circuits and Systems," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 25, no. 6, pp. 1856-1865, June 2017.
- [6] H. M. Torun, M. Swaminathan, A. Kavungal Davis and M. L. F. Bellaredj, "A Global Bayesian Optimization Algorithm and Its Application to Integrated System Design," in *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 26, no. 4, pp. 792-802, April 2018.
- [7] E. Brochu, V. M. Cora, and N. de Freitas, "A tutorial on Bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning," Technical Report UBC TR-2009-23 and arXiv:1012.2599v1, 2009
- [8] M. Balandat, B. Karrer, D. R. Jiang, S. Daulton, B. Letham, A. G. Wilson, and E. Bakshy. BoTorch: A Framework for Efficient Monte-Carlo Bayesian Optimization. *Advances in Neural Information Processing Systems* 33, 2020.
- [9] S. Kundu and P. Mandal, "ISGP: Iterative sequential geometric programming for precise and robust CMOS analog circuit sizing," *Integration, the VLSI Journal*, vol. 47, no. 4, pp. 510-531, 2014.