INDIAN JOURNAL OF SCIENCE AND TECHNOLOGY

ORIGINAL ARTICLE



OPEN ACCESS

**Received:** 07/08/2025

**Accepted:** 22/08/2025

**Published:** 08/09/2025

**Citation:** Prajapti PP, Maru RM, Sheth KR, Patel AM, Kshatriya AJ, Bathani NJ, Patel KV (2025) Implementation of Teaching Learning Based Optimization (TLBO) Algorithm to Optimize CMOS Based Analog Circuits. Indian Journal of Science and Technology 18(34): 2784-2793. https://doi.org/10.17485/IJST/v18i34.1373

* **Corresponding author**.

pankajprajapati@vgecg.ac.in

**Funding:** None

**Competing Interests:** None



Published By Indian Society for Education and Environment (iSee)

**ISSN**
Print: 0974-6846
Electronic: 0974-5645

# Implementation of Teaching Learning Based Optimization (TLBO) Algorithm to Optimize CMOS Based Analog Circuits

**Pankaj P Prajapti[1]*, Rahul M Maru[2], Kinjal R Sheth[1], Alpesh M Patel[3], Anilkumar J Kshatriya[3], Nitin J Bathani[1], Kirit V Patel[3]**

**1** Department of Electronics and Communication Engineering, Vishwakarma Government Engineering College, Chandkheda, Gujarat, India
**2** Senior Data Analyst, Rawalwasia, Surat, Gujarat, India
**3** Department of Electronics and Communication Engineering, Government Engineering College, Gandhinagar, Gujarat, India

## Abstract

**Objective:** To leverage the Teaching Learning-Based Optimization (TLBO) algorithm for efficient circuit design and optimization of CMOS-based analog circuits by addressing complex and time-consuming design challenges. **Method:** The TLBO method is used to optimize a number of CMOS analog circuit parameters. Its performance on several CMOS circuit parameters is thoroughly examined. **Findings:** Strong optimization capabilities for CMOS-based analog circuit design are demonstrated by TLBO. The width of transistor and value of resistor achieved 1.45006 $\mu$m and 1.737417 K$\Omega$ respectively for the targeted specification for the CS amplifier. For Triple cascoded current mirror (TCCM), this work achieved width of transistor as 70.89 $\mu$m and value of resistor as 72.52 K$\Omega$ and the $I_{ref}$ value equal to 10 $\mu$A using TLBO approach. **Novelty:** To the best of our knowledge, this is the first time the TLBO algorithm has been used to automatically construct CMOS-based analog circuits. The technique efficiently explores the design space while cutting down on calculation time compared to manual design of analog circuits.

**Keywords:** Optimizing parameter; TLBO; TCCM; CSVCO; NgSpice

## 1 Introduction

In many everyday electronic devices, interfacing between digital and analog circuits is essential. However, optimizing analog-digital circuits is a complex task. When designing an analog circuit, it is beneficial to use an evolutionary optimization algorithm [(1)]. But with so many available algorithms, it can be difficult to determine which one will yield the expected results and meet the desired specifications [(2)], [(3)].

Evolutionary computation paradigms comprising genetic algorithms, particle swarm optimization [(4)], differential evolution, and their hybridized derivatives are having long underpinned metaheuristic strategies in the analog CMOS circuit optimization

domain [5] [6]. These population-based mechanisms leverage multi-agent collaboration and adaptive operator selection to perform global search across a complex design space, thereby achieving a judicious trade-off between convergence efficacy and solution diversity. These foundational approaches have seeded numerous contemporary hybrid frameworks optimized for advanced analog topologies.

In contrast, Teaching Learning-Based Optimization (TLBO), models the analog of an educational process via "teaching" and "learning" phases [7]. TLBO distinguishes itself by eliminating user-dependent algorithmic parameters, simplifying implementation while maintaining robust global search characteristics. Despite its conceptual elegance and domain-agnostic applicability, TLBO often incurs heavy computational overhead, predominantly because each candidate solution evaluation mandates expensive circuit simulation—rendering TLBO comparatively slower, especially in high-dimensional analog sizing problems where specification constraints are stringent.

Differential Evolution (DE) has been systematically leveraged for transistor dimension optimization in Miller-compensated OTAs, enabling finely tuned trade-offs among key metrics such as open loop gain, unity gain bandwidth, phase margin, and the integrated power area figure-of-merit specially in mature 0.18 $\mu$m and 0.35 $\mu$m CMOS nodes. Concurrently, Genetic Algorithm (GA) frameworks, often augmented with adaptive crossover/mutation schemes or supplemented by expert-guided heuristics, have been successfully applied to two-stage op-amp architectures, producing demonstrable improvements in operating-point stability, input-referred noise floor, and linearity performance under stringent design constraints [8]. Hybrid optimization strategies such as the combination of Particle Swarm Optimization (PSO) with Nelder–Mead simplex methods have been demonstrated to drastically accelerate convergence velocity and effectively reduce dimensionality in low-voltage CMOS op-amp sizing while preserving high performance through SPICE-level evaluations [9]. Similarly, multi-objective variants like SADE—a specialized DE variant emphasizing performance-speed trade-offs have enabled the simultaneous evolution of both parameter values and circuit topologies (inclusive of passive, active, and BiCMOS classes), offering enhanced flexibility in design space exploration [10].

Despite their efficacy, such heuristic and hybrid approach commonly incur high computational costs due to repeated SPICE evaluations. And their applicability often hinges on manual tuning or domain-specific insights when transferred across distinct circuit families or technology nodes thus constraining scalability and generality. Recent methodologies as elaborated in [11] that integrate Bayesian surrogate models with evolutionary search such as Gaussian-process–assisted DE (GDE3+GP) offer greater efficiency by directly exploring Pareto fronts in high-dimensional design spaces, significantly reducing simulation count while preserving diversity of feasible trade-offs.

Another notable advance is the batch-constrained Bayesian optimization ensemble (MACE) discussed by S. Zhang *et al.* [12], which accelerates multi-objective analog circuit synthesis by up to 74× compared to conventional DE in unconstrained scenarios and up to 15× in constrained cases, while maintaining Pareto diversity. Additionally, surrogate-assisted constrained multi-objective optimization strategies, such as those proposed by S. Yin et al. [13], have demonstrated fast convergence and constraint handling for analog sizing problems through self-adaptive incremental learning techniques. An ML-assisted evolutionary approach is presented for analogue circuit sizing, in order to combine the benefits of machine learning with conventional evolutionary algorithms [14] [15].

Recent IEEE research underscores complementary approaches that address TLBO's limitations while maintaining global search benefits. A TLAGO: A Transfer Learning-Assisted Global Optimization is proposed that incorporates neural transfer learning within surrogate-assisted evolutionary algorithms. TLAGO reduces simulation cost and accelerates convergence by reusing learned models across related circuit configurations, yielding performance gains of over eight percent against state-of-the-art methods while substantially lowering runtime [16].

The TLBO algorithm, developed by Dr. R. V. Rao [17], is an evolutionary algorithm that simulates the teaching-learning process. In this approach, students learn from their teacher through communication, and students also benefit from interacting with one another. The key components of the TLBO algorithm involve both educators and learners.

Analog circuit design techniques pointed out here mostly focus on hybrid and evolutionary optimization techniques. Despite their encouraging outcomes, these methods frequently have limited scalability, a heavy reliance on manual heuristics, and poor generalization between circuit types. Automated techniques that can reduce manual intervention and produce effective, high-quality designs are still needed. In line with this ongoing research direction, the present work explores the automatic design of CMOS analog circuits using the Teaching–Learning Based Optimization (TLBO) algorithm. The objective is to establish TLBO as a more reliable, scalable, and efficient alternative for CMOS circuit design. To validate this, three benchmark circuits like the common-source amplifier, triple cascaded current mirror, and three-stage current-starved VCO are implemented and analyzed in a 0.18 $\mu$m CMOS process.

Section I provides an introduction, while Section II explains the TLBO algorithm. The content and flowchart of the TLBO algorithm are presented in Section III. Section IV discusses the optimization algorithm for analog circuits and the flowchart

for automatic analog circuit design. Section V details the experiments conducted on several analog circuits and their results. Finally, Section VI presents the conclusion.

## 2 Methodology

### 2.1 Teaching Learning Based Optimization (TLBO) Algorithm

This mutative algorithm is encouraged by the teaching learning (TL) process and was established by Dr. R. V. Rao [17]. The TLBO algorithm is a population-based method that draws from the idea of students learning from both a teacher and their peers. The core components of the TLBO algorithm include the teacher and learners, with a single cycle of the algorithm consisting of two distinct phases:

- **Teacher's Phase** – In this phase, the personal best value is found among all learners, which is then set as the teacher's value. Following this, the students interact with the teacher and learn from this interaction.
- **Learner's Phase** – Here, learners interact with one another to find the globally best solution out of all learners.

The performance of teaching-based optimization algorithms is influenced by the students' outcomes, which are dependent on the efficiency of the instructor. In this context, the learners improve their grades through the guidance of the teacher. Additionally, interactions among students help further enhance their performance.

The first step in the optimization process involves defining the initialization criteria, which include:

1. **Population size** – This refers to the number of learners. In the present experiment, three different population sizes were tested: 10, 30, and 50.
2. **Number of design variables** – This represents the subjects or areas of knowledge that each learner specializes in. In TLBO, each student is expected to learn across different variables, with interactions occurring during the learner phase.
3. **Range of variables** – To estimate the algorithm's performance, various benchmark functions are tested. In this experiment, thirteen benchmark functions were used for evaluation.
4. **Objective function** – The performance of the algorithm was evaluated using these thirteen benchmark functions.
5. **Termination criteria** – This criterion checks if the algorithm has achieved the expected result. In this experiment, the termination criterion was set to D = 1.00e-07, indicating the stopping condition for the algorithm.

Through these initialization criteria, the TLBO algorithm is set up to test its effectiveness and optimize the design of analog circuits.

### 2.2 Teacher Phase

In the current phase of the algorithm, the goal is to identify the personal best solution for each learner and select the student with the best solution to act as the teacher. Once the teacher is selected, all learners interact with the teacher to gain knowledge and improve their solutions based on the teacher's expertise. The interaction with the teacher helps the learners to refine their own solutions and progress towards a more optimal outcome. Let, k is the objective function, and there are A and B numbers of learners and subjects respectively. In the first cycle, we are using C code to construct a random matrix

$$[K_x \times K_y] \tag{1}$$

$$\text{Where } x = 1, 2, \ldots, A \text{ and where } y = 1, 2, \ldots, B \tag{2}$$

Following the creation of random matrices, each learner's performance is assessed by computing their objective function. The objective function measures the degree to which each learner's solution satisfies the required standards or requirements. We choose each learner's personal best solution, which is the best solution they have come across during the optimisation process, after the objective function has been determined for each learner. We needed two distinct random factors, Ri and TF, in order to find the best answer as a teacher using the difference means formula.

$$TF = round[1 + rand(0,1)(2-1)] \tag{3}$$

Here, TF falls between 1 and 2. By choosing TF, the anticipated outcome or solution finding for this research project is 2. Although we can use C code to choose random integers, we are choosing TF = 2 for simplicity, best results, and estimation. In the context of Ri, selecting interval [0,1], values are chosen randomly each time using C code. **Equation (4)** is being used to calculate the differential formula after TF and Ri have been chosen. It is crucial to remember that TLBO's constraint is the teaching component in this case. We are creating an additional matrix for the differential mean formula in this stage.

$$Diff_{mean_{x,y,i}} = R_i \left(K_{x,ybest,i} - TfM_{j,i}\right) \tag{4}$$

In the following phase, we will use **equations (3), (4), and (5)** to update the primary matrices. But as previously stated, the teaching factor is choosing number two.

$$K'_{x,y,i} = K_{x,y,i} + Diff\_Mean_{x,i} \tag{5}$$

Where is the most recent value? If K is inside the accepted range or criterion, then the objective function value is acceptable.

## 2.3 Phase of Learners

This phase, which is referred to as the learners' phase, is the following one that involves two learners interacting with one another and learning from each other. Here, we need to know whether we are going to use a minimisation or maximisation procedure. For instance, it's crucial to remember that **equations (6a) and (6b)** must be used if we are going to minimise the CMOS parameters.Here, any of the two students, A and B, will be selected automatically.The modified function values for student A and student B are specified at the conclusion of the learning process, in brief.

This is the most recent value of the interaction between two randomly chosen students. It's crucial to remember that **equations (6a) and (6b)** must be used if we are to carry out the minimisation process. **Equation (7a) and (7b)** are utilized for the process of maximization.

$$K^{"}_{x,a,i} = K'_{x,a,i} + R_i \left(K'_{x,a,i} - K'_{x,b,i}\right), \quad if K'_{total-a,i} < K'_{total-b,i} \tag{6a}$$

$$K^{"}_{x,a,i} = K'_{x,a,i} + R_i \left(K'_{x,b,i} - K'_{x,a,i}\right), \quad if K'_{total-b,i} < K'_{total-a,i} \tag{6b}$$

$$K^{"}_{x,a,i} = K'_{x,a,i} + R_i \left(K'_{x,a,i} - K'_{x,b,i}\right), \quad if K'_{total-a,i} > K'_{total-b,i} \tag{7a}$$

$$K^{"}_{x,a,i} = K'_{x,a,i} + R_i \left(K'_{x,b,i} - K'_{x,a,i}\right), \quad if K'_{total-b,i} > K'_{total-a,i} \tag{7b}$$

## 2.4 Flow of Automatic Circuit Design

There are primarily three components in this automatic circuit design process. a) the primary scripting file; b) the error file; and c) the output representation graph file. Flow chart of the automatic circuit design optimization is shown in figure 1.

**Step 1:** First, initialise circuit.cir, pso.c, parameter.dat, and input.dat.

**Step 2:** Run the pso.c C program, which takes input from input.dat and optimises the input value, as part of the initial scripting/optimization step. The specs data in Obtainedspec1.dat are provided by this file, which runs in Ng-spice. After that optimised value has been added to the parameter.dat and the data from the parameter.dat has been added to the circuit.

**Step 3:** Check the obtainedspec1.dat file with the expected specifications data file, desispec.dat, and obtain the perror.dat file by compiling it in the error.c.

**Step** 4: The error file perror.dat is produced during the initial optimisation of the error phase when the criteria of Obtainedspec1.dat match the Desiredspec.dat loop.

**Step 5:** Creating an error.m file in MATLAB's second phase. Code for reading perror.dat data and producing simulation results is included in this file.

**Step-6:** The plot.m file displays a simulation of the data and graph from the optimisation process at the end.

# 3 Result and Discussion

## 3.1 Experiments of Optimization of Analog Circuit Using TLBO Algorithm and Result

The TLBO method is used to automatically design the following five basic circuits.

1. Triple cascoded current mirror (TCCM) [18].
2. Common source amplifier. (CSA)
3. Three stage current starved VCO (CSVCO) [19], [20].
4. CMOS voltage divider (VD) [21], [22].

All the circuits are constructed using MOSFET models in 180 nm technology. Ng-spice is used to simulate the circuit by interacting with an algorithm for optimisation.

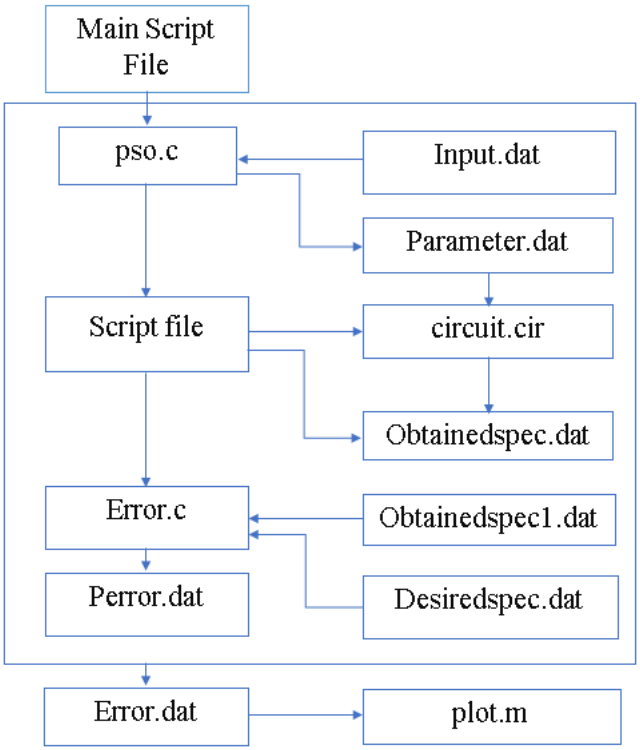


**Fig 1.** Flowchart of Automatic Circuit Design Optimization

## 3.2 Triple Cascoded Current Mirror (TCCM)

Triple cascoded current mirror (TCCM), shown in figure 2, is an exceptionally important building block for twin CMOS design. The main components of this circuit are M1 to M6 NMOS transistors and R1 resistor. the identical length ($L$) chosen for every transistor is as L = 180 nm and the width ($W$) for each transistor as $W1$. So, for this circuit we have to find two design parameters namely $W1$ and $R1$. The range and obtained values of the components by the TLBO algorithm for this circuit are shown in Table 1. The desired specifications are reference current ($I_{ref}$) and output current ($I_{out}$), and the obtained values are shown in Table 1. Figure 3 shows the plot of specified error accomplished after using the Ngspice simulator to create the circuit file.

## 3.3 Common Source Amplifier

The common-source amplifier as depicted in figure 4 is used as a voltage or transimpedance amplifier. The range and obtained values of the components by the TLBO algorithm for this circuit are shown in Table 2. Figure 5 shows the plot of specified error achieved when the compilation of the circuit file is done using the Ngspice simulator.

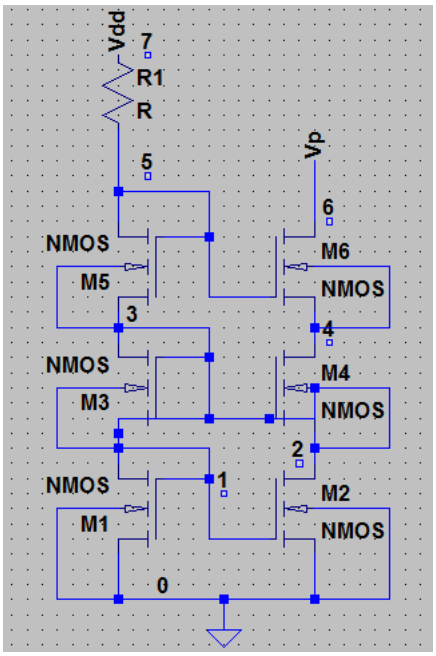


**Fig 2.** Circuit Diagram of TCCM CIRCUIT [15]

**Table 1.** Circuit Components, Required Parameters, and Obtained Values Using TLBO for TCCM

| Sr. No. | Components | Range of Components value | Obtained Values |
|---|---|---|---|
| 1 | $W_1$ | 1 to 1000 ($\mu$m) | 70.893778 ($\mu$m) |
| 2 | $R_1$ | 1 to 1000 (K$\Omega$) | 72.522823 (K$\Omega$) |
| | Parameters | Required Value ($\mu$A) | Obtained Values ($\mu$A) |
| 1 | $I_{ref}$ | 10 | 10.000000 |
| 2 | $I_{out}$ | 10 | 10.000010 |

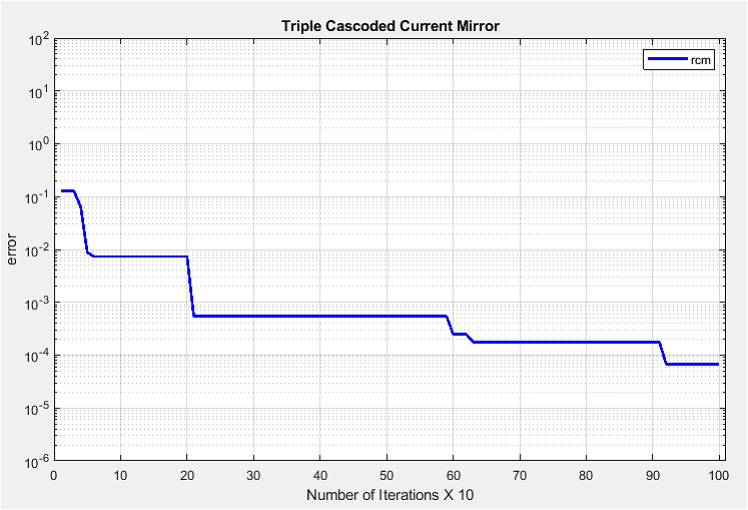


**Fig 3.** Plot of Generations vs. error in Triple Cascoded Current Mirror Circuit

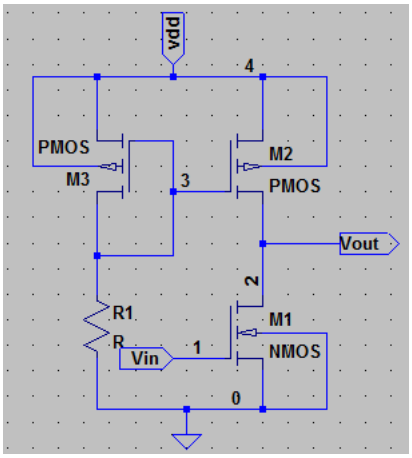


**Fig 4.** Common source amplifier [21]

**Table 2.** Circuit Components, Required Parameters, and Obtained Values Using TLBO for Common Source Amplifier

| Sr. No. | Components | Range of Components value | Obtained Values |
|---|---|---|---|
| 1 | $W_1$ ($\mu$m) | 1.0 - 1000 | 1.45006 |
| 2 | $R_1$ (K$\Omega$) | 1.0 - 1000 | 1.737417 |
| | | | |
| | Parameters | Required Values | Obtained Values |
| 1 | $A_V$ | > 25 dB | 26.93 (dB) |
| 2 | UGB | > 50 MHz | 93.04 (MHz) |
| 3 | PM | > 45° | 98.89° |

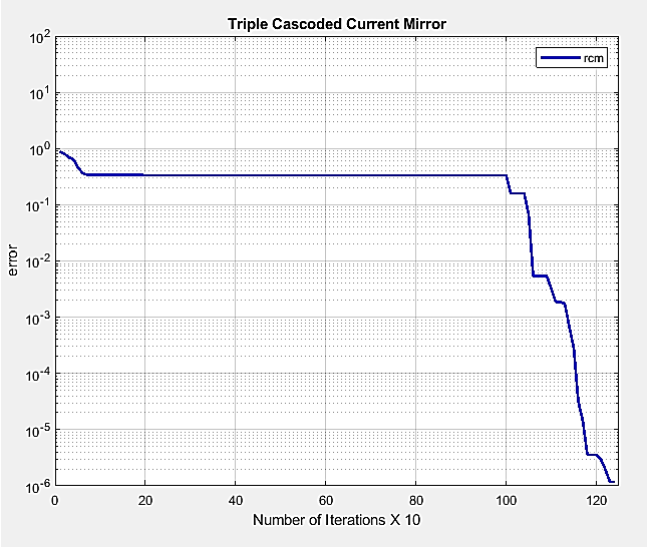


**Fig 5.** Plot of Generations vs. error in Common Source Amplifier

## 3.4 Three stage Current Starved VCO

The three-stage current – starving VCO is shown in Figure 6. Its operating process is close to that of a ring oscillator. The purpose of this work is to develop and optimize the three-stage starved Voltage-Controlled Oscillator (VCO) circuit. In this research, the VCO is designed and optimized using the TLBO algorithm. The TLBO algorithm is applied to optimize the various components of the VCO circuit to meet the desired specifications. The range and obtained values of the components for the three-stage starved VCO circuit, as optimized by the TLBO algorithm, are presented in Table 3. These optimized component values ensure that the circuit performs efficiently within the desired operational parameters.

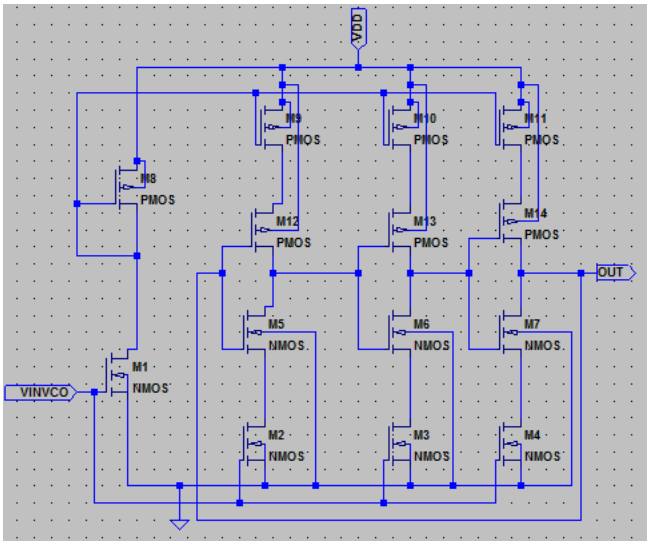


**Fig 6.** Schematic of three-stage starved Voltage-Controlled Oscillator (VCO) circuit

**Table 3.** Circuit Components, Required Parameters, and Obtained Values Using TLBO for Three-stage starved Voltage-Controlled Oscillator (VCO)

| | Components | Range of Components value ($\mu$m) | Obtained Values | Achieved Parameters (Oscillation Frequency) MHz |
|---|---|---|---|---|
| 1 | $W_1$ | | 646.4364 | 400.0301 |
| 2 | $W_2$ | | 655.3027 | 400.0301 |
| 3 | $W_3$ | | 624.1227 | 400.0301 |
| 4 | $W_4$ | 1 to 1000 | 572.8479 | 400.0301 |
| 5 | $W_5$ | | 648.5861 | 400.0301 |
| 6 | $W_6$ | | 655.1839 | 400.0301 |
| 7 | $W_7$ | | 668.6649 | 400.0301 |
| 8 | K | 1 to 5 | 5.0000 | 400.0301 |

## 3.5 CMOS Voltage Divider

In analog CMOS circuit design, biasing requirements generally fall into two main categories: voltage references and current references. Voltage references are crucial for applications such as cascade stages and comparators, while current references are used for biasing amplifiers through multiple current mirror schemes D-A convertor and more. For certain applications, simple biasing arrangements are sufficient. One such example is the CMOS voltage divider, which is commonly used as a reference voltage in CMOS IC design [21], [22]. The schematic of the CMOS voltage divider is depicted in Figure 7. The key components of this circuit include four MOS transistors: **M1, M2, M3, and M4**. These components work together to provide the necessary voltage reference for the design. There are four design parameters in the circuit namely $W1$, $W2$, $W3$, and $W4$ as detected from table 4. This table lists the range and obtained values of the components by the TLBO algorithm. Achieved specified error is plotted and shown in Figure 7 when the circuit file is connected with the Ngspice simulator.

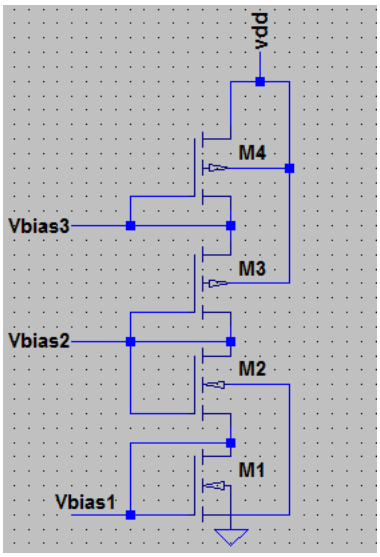


**Fig 7.** Schematic of CMOS Voltage Divider

**Table 4.** Circuit Components, Required Parameters, and Obtained Values Using TLBO for CMOS Voltage Divider

| Sr. No. | Components | Range of Components value ($\mu$m) | Obtained Values |
|---|---|---|---|
| 1 | $W_1$ | 0.1 to 1000 | 0.5366 |
| 2 | $W_2$ | 0.1 to 1000 | 906.0000 |
| 3 | $W_3$ | 0.1 to 1000 | 222.2993 |
| 4 | $W_4$ | 0.1 to 1000 | 368.0000 |
| | Parameters | Values | Obtained Values |
| 1 | $V_{bias1}$ | 0.8 V | 7.7174e-01V |
| 2 | $V_{bias2}$ | 1.2 V | 1.2374e+00V |
| 3 | $V_{bias3}$ | 1.7 V | 1.6714e+00V |

## 4 Conclusion

Assessing the TLBO algorithm's performance in CMOS circuit optimization was the aim of this work. In this research, four fundamental analog circuits have been designed using the TLBO algorithm, showcasing its potential as a novel approach in analog circuit design. TLBO's appropriateness for automated analog circuit design was confirmed by its overall high accuracy in satisfying the required parameters while optimizing within the specified component ranges. Although upper bound values were attained in the CMOS Voltage Divider—likely to accomplish exact voltage division—experimental results indicated that the resultant component values were often well within the lower bound of the permitted ranges.

This work concentrated on simple circuits as a precursor to more intricate systems. This work concentrated on basic circuits as a starting point for developing more intricate systems. Future research could focus on creating an intuitive platform that enables circuit designers—even individuals with little experience in electronics—to use evolutionary algorithms to create circuits that are customized to meet particular needs. The design process would be more accessible and efficient with such a tool, and the method might be expanded to optimize more intricate CMOS circuits.

## References

1) Rashid R, Raghunath G, Badugu V, Nambath N. Performance evaluation of evolutionary algorithms for analog integrated circuit design optimisation. *Microelectronics Journal*. 2023;141:105983. https://doi.org/10.1016/j.mejo.2023.105983.
2) Kwon M, Lee Y, Song I. Circuit-centric genetic algorithm (CGA) for analog and radio-frequency circuit optimization. *arXiv preprint arXiv:240317938*. 2023. https://doi.org/10.48550/arXiv.2403.17938.
3) Prajapati PP, Kshatriya AJ, Bharvad SL, Upadhyay AB. Performance analysis of CMOS based analog circuit design with PVR variation. *Bulletin of Electrical Engineering and Informatics (BEEI)*. 2023;12(1):141–148. https://doi.org/10.11591/eei.v12i1.4357.

4) Joshi D, Dash S, Reddy S, Manigilla R, Trivedi G. Multi-objective hybrid particle swarm optimization and its application to analog and RF circuit optimization. *Circuits, Systems, and Signal Processing*. 2023;42(8):4443–4469. https://doi.org/10.1007/s00034-023-02342-1.
5) Prajapati PP, Kshatriya AJ, Patel DN, Gonsai SK, Tank HB, Sheth KR. Comparative analysis of meta heuristics algorithm for differential amplifier design. *Bulletin of Electrical Engineering and Informatics (BEEI)*. 2023;12(6):3395–3401. https://doi.org/10.11591/eei.v12i6.6153.
6) Bharvad SL, Prajapati PP, Kshatriya AJ. Meta Heuristic Optimization Approach for CMOS Based Analog Circuit Design and Performance Evaluation of Evolutionary Algorithms. *ICTACT Journal on Soft Computing*. 2023;13(2). https://doi.org/10.21917/ijsc.2023.0402.
7) Zhai Z, Dai Y, Xue Y. A novel teaching-learning-based optimization with Laplace distribution and experience exchange. *Mathematical Problems in Engineering*. 2022;2022:4177405. https://doi.org/10.1155/2022/4177405.
8) Das P, Jajodia B. Design automation of two-stage operational amplifier using multi-objective genetic algorithm and SPICE framework. 2022. https://doi.org/10.1109/ICICT54344.2022.9850461.
9) Maji KB, Saha A, Kar A, Bhattacharyya A. Design of low-voltage CMOS op-amp using evolutionary optimization techniques;vol. 1. Singapore. Springer. 2018. https://doi.org/10.1007/978-981-13-2550-8_25.
10) Chakraborty S, Saha AK, Sharma S, Sahoo S, Pal G. Comparative performance analysis of differential evolution variants on engineering design problems. *Journal of Bionic Engineering*. 2022;19. https://doi.org/10.1007/s42235-022-00190-4.
11) Vișan C, Pascu O, Stănescu M, Grosu-Șandru ED, Diaconu C, Buzo A, et al. Automated circuit sizing with multi-objective optimization based on differential evolution and Bayesian inference. *arXiv preprint arXiv:220602391*. 2022. https://doi.org/10.48550/arXiv.2206.02391.
12) Zhang S, Yang F, Yan C, Zhou D, Zeng X. An efficient batch constrained Bayesian optimization approach for analog circuit synthesis via multi-objective acquisition ensemble. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 2021;p. 1–1. https://doi.org/10.1109/TCAD.2021.3054811.
13) Yin S, Wang R, Zhang J, Liu X, Wang Y. Fast surrogate-assisted constrained multi-objective optimization for analog circuit sizing via self-adaptive incremental learning. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*. 2023;42(7):2080–2093. https://doi.org/10.1109/TCAD.2022.3221694.
14) Chen YY, Fang SY. Optimizing Analog Circuit Design Through a Machine Learning-Assisted Evolutionary Algorithm. *Electron Lett*. 2025;61:e70331. https://doi.org/10.1049/ell2.70331.
15) Agwekar A, Singh L. CMOS-Based Process-Scalable Analog Circuits for Machine Learning: A Comprehensive Review and Future Directions. *Journal of VLSI Design Tools and Technology*. 2025;15(01):8–17. Available from: https://journals.stmjournals.com/jovdtt/article=2025/view=198472.
16) Díaz KYG, Rodríguez YAL, Flores-Tena A, Martínez-Castro D, Márquez-Velázquez CT. Teaching–learning-based optimization algorithm applied in electronic engineering: A survey. *Electronics*. 2022;11(21):3451.
17) Rao RV, Savsani VJ, Vakharia DP. Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems. *Computer-Aided Design*. 2011;43(3):303–315. https://doi.org/10.1016/j.cad.2010.12.015.
18) Prajapati PP, Shah MV. Sizing of CMOS two-stage Operational Amplifier using Cuckoo Search Algorithm. Allahabad, India. 2015. https://doi.org/10.1109/UPCON.2015.7456700.
19) Mishra A, Sharma GK. Design of Power Optimal, Low Phase Noise Three Stage Current Starved VCO. New Delhi, India. 2015. https://doi.org/10.1109/INDICON.2015.7443417.
20) Yasmin F, Basak A, Mahin M, Amin MT. MOS Capacitance Based 3-Stage Current Starved Ring VCO for Wireless Applications. Cox'sBazar, Bangladesh. 2019. https://doi.org/10.1109/ECACE.2019.8679192.
21) Prajapati PP, Shah MV. Automatic Sizing of CMOS-Based Analogue Circuits Using Cuckoo Search Algorithm. *International Journal of Intelligent Systems Technologies and Applications*. 2020;p. 207–217. https://doi.org/10.1504/IJISTA.2020.107217.
22) Venkatesh A, Sanyal A. A Machine Learning Resistant Strong PUF using Subthreshold Voltage Divider Array in 65nm CMOS. Sapporo, Japan. 2019. https://doi.org/10.1109/ISCAS.2019.8702525.