I. INTRODUCTION

The fundamental component of any electronic device that we might use daily, including appliances, industrial automation, communications, automotive, and computing devices, is the analog circuit. Designing any CMOS based analog circuit is a complex design process therefore an extremely essential and decisive task performed by the engineers. Several attempts have been made to improve the structure of analog circuit design [1]. The difficult task of combining digital and analog circuitry on a single chip has become harder as the scale size of CMOS technology has decreased. Even though there are fewer analog circuit components in integrated circuits than in digital circuits, the design process for analog circuits in ICs is more difficult and complex. Because of the limitations imposed by the scaling of CMOS technology, creating analog circuitry has become more difficult and time-consuming, requiring specialized knowledge [2]. Several researchers and technologists have tried numerous times to optimize the architecture of analog circuits. To construct an analog circuit, one typically follows these stages, selecting a topology, optimization using parametric data, and creation of layouts. For the successful design of the analog circuit, it is advisable to follow three phases to attain the required parameters and physical arrangement at the schematic level [3].

Developing sophisticated analog circuits with ultimate performance requires a deep understanding of transistor scaling. The CMOS based automatic analog circuit design method needs stable and consistent optimization strategies achieved by adjusting the aspect ratio of transistor width and length [4], [5]. Circuit design automation systems can be classified into two categories based on their approach to circuit scaling: one category knowledge-based employs a predefined set of rules, while the other employs optimization algorithms to ascertain the most efficient design. In knowledge-based techniques depending upon circuit topology, circuit developer applied design equations based on their understanding of analog circuit design rules. Since a large amount of human work is needed for producing analytical equations and heuristics, the outcomes are often erroneous and poor. As computer power expanded, a different method known as optimization-based synthesis emerged. This method uses search methods to turn the issue into a function optimization problem. The solution is meant and generate specific results.[6], [7].

Optimization techniques are often divided into two categories: deterministic and meta-heuristic techniques. Deterministic optimization techniques were created over several decades for a variety of real-world applications. Analog circuit optimization successfully employs a few well-known mathematical techniques. This strategy has

Abstract: The main objective of the automated circuit sizing technique is to deal with the challenges of design tradeoffs in analog circuit design with improved accuracy and efficacy. Analog circuit design represents a multi-objective optimization challenge, where designers must properly balance various performance factors such as input and output impedance, power dissipation, area, unity-gain bandwidth, slew rate, and open-loop DC gain. Traditional design equations provide a sizing of differential amplifier MOS transistors, further optimization can be achieved through the application of meta-heuristic search techniques. Meta-heuristic search techniques can be used as local optimizers in a smaller search area to improve the optimization of design parameters. The differential amplifier circuit with current mirror load is optimized through the application of the Dung Beetle Optimization (DBO) algorithm. By using an evolutionary algorithm called DBO, all the required parameters were achieved with the least amount of transistor area and power dissipation when compared to the results of the Seeker Optimization Algorithm (SOA), Opposition based Harmony Search Algorithm (OHS), craziness-based particle swarm optimization (CRPSO), and Cuckoo Search (CS) algorithms.

Keywords: Complementary metal oxide semiconductor, Dung Beetle optimization, Seeker Optimization, Particle swarm optimization, Opposition based Harmony Search, Cuckoo Search, Differential amplifier with current mirror load

A Novel Dung Beetle Optimization Approach for Automatic CMOS Analog Circuit Design

1Dhaval N. Patel
2Dharmesh J. Shah

* Department of Electronics and Communication, L.D. College of Engineering, Ahmedabad, Gujarat, India. Email: dnpatel@ldce.ac.in
2 Provost, Indrashil University, Mehsana, Gujarat, India.

Copyright © JES 2024 on-line : journal.esrgroups.org
several drawbacks, including a good starting point, stagnation in the local minimum search area, and reliance on the ongoing targeted function. Swarm intelligence algorithms, also known as population-based metaheuristics optimization algorithms, constitute the second strategy [8], [9].

Their robust, nature-inspired behavior, ease of execution, and great versatility in achieving the best results have drawn a lot of attention to the development of analog circuit design. Novel dung beetle optimization metaheuristic algorithms are investigated for developing differential amplifier circuits with outstanding performance in this article.

II. DUNG BEETLE OPTIMIZER

The outstanding coordination that dung beetles display in their colonies is used as a framework for the DBO algorithm. Designed with several position updating rules ball-rolling, dancing, foraging, thieving, and breeding activities of dung beetles are intended to provide high quality outcomes. The DBO population categorization and boundary selection techniques allow the algorithm to take into account both local convergence and global exploration.

A. Rolling Dung Beetles

It is a well-known fact that dung beetles can mold dung into spherical balls and then roll them to a suitable location. The dung beetle uses celestial cues, such as the direction of the wind or the position of the sun, to move in an accurate, straight route while steering the dung ball. The position modification equation for rolling dung beetles is as follows:

\[ x_i(t + 1) = x_i(t) + \alpha \times k \times x_i(t - 1) + b \times \Delta x \]  

where \( x_i(t) \) indicates the position details of the ith beetle during the t th iteration. \( \alpha \) denotes a natural coefficient has a value of either -1 or 1, \( k \) represents the deviation coefficient in the range (0,0.2), \( b \) represents a constant in the interval (0,1), \( \Delta x \) is employed to imitate variations of light intensity, and \( X_{worst} \) denotes the global worst position.

B. Dancing Dung Beetles

A dung beetle will dance to adjust its direction of walking when it comes across an obstacle that forbids it from moving ahead toward a new path. For a new rolling direction, the tangent function is utilized to recreate the dung beetle’s dancing tendency. Thus, the Dancing Dung Beetle’s position changes as follows:

\[ x_i(t + 1) = x_i(t) + \tan(\theta) \times |x_i(t) - x_i(t - 1)| \]  

Where the angle represented by \( \theta \) varies from 0 to \( \pi \). The position of the dancing dung beetle will not be changed if \( \theta \) is either 0 or \( \pi/2 \).

C. Breeding dung beetles

For the process of laying their eggs, female dung beetles gently move their dung balls to a safe area. This ensures that their young will develop without any danger. She lays eggs in suitable areas to have more babies. A technique for choosing borders that mimics the places where female dung beetles lay their eggs is proposed based on the spawning tactics. The following formula can be used to change your position.

\[ x_i(t + 1) = X^* + b_1 \times (x_i(t) - Lb^*) + b_2 \times (x_i(t) - Ub^*) \]

\[ Lb^* = max(X^* \times (1 - R), Lb) \]

\[ Ub^* = min(X^* \times (1 + R), Ub) \]

where \( X^* \) denotes the current local optimal position, \( R = 1 - t/T_{max} \), \( T_{max} \) represents the maximum number of iterations, \( Lb^* \) and \( Ub^* \) represents the lower and upper bounds of the spawning area of the optimization problem respectively.

D. Small dung beetles

A few adult dung beetles emerge from the ground in looking for feeding. We called them small dung beetles. We need to define the ideal foraging region to direct the small beetles. The following defines the perimeter of the ideal foraging area:

\[ Lb^b = max(X^b \times (1 - R), Lb) \]

\[ Ub^b = min(X^b \times (1 + R), Ub) \]

\[ x_i(t + 1) = x_i(t) + C_1 \times (x_i(t) - Lb^b) + C_2 \times (x_i(t) - Ub^b) \]
where \( x_i(t) \) is the position data of the \( i \)th small dung beetle throughout the \( t \)th iteration, \( C_1 \) is a random number that follows a normal distribution, and \( C_2 \) is a random vector ranging between 0 and 1. The algorithm's global optimal position is denoted \( X^* \), while the lower and upper bounds of the optimization problem's foraging area are represented by \( L^b \) and \( U^b \) respectively.

**E. Thief dung beetles**

A natural behavior of some dung beetles' normal activity involves robbing other beetles of their dung balls. The position information of the thief beetle is updated and can be described as follows:

\[
x_i(t+1) = X^b + S \times g \times (|x_i(t) - x^b| + |x_i(t) - X^b|)
\]  

(8)

Where \( X^b \) represents the ideal location to hunt for food, \( x_i(t) \) represents the position details of \( i \)th thief dung beetle in throughout the \( t \)th iteration, \( S \) denotes a constant value, \( g \) represents a random vector of size \( 1 \times D \) following a normal distribution.

The DBO algorithm's state-of-the-art developments are reviewed in [10], [11], [12], [13], which also covers theoretical background, various DBO algorithm variants, multi-objective optimization engineering applications, and studies to be done for the ongoing advancement of this powerful algorithm. The DBO algorithm has not yet been employed for the design of CMOS based analog circuits. In this paper, the DBO is reviewed for the first time for the automatic design of CMOS differential amplifier. The swarm size of the dung beetle for optimization is taken 30. There are six ball-rolling dung beetles, six brood balls, and seven small dung beetles, and the rest are thieves for optimizing.

**III. CMOS DIFFERENTIAL AMPLIFIER DESIGN CRITERIA**

Leading to the construction of analog and mixed-signal circuits, designers utilized SPICE simulators for data analysis and made predictions regarding the performance of these circuits. The completed circuit goes through such virtual testing to check that it fulfills the design criteria. Circuit design specifications are translated into objective functions and find the optimized design using an optimization algorithm. Through this optimization procedure, we can obtain the finest circuit design parameter [14] [15].

An ideal design of a CMOS differential amplifier circuit has a vast number of design parameters. Specific requirements, such as small-signal differential voltage gain \( (A_v) \), unity gain bandwidth \( (UGB) \), positive and negative power supply rejection ratio \( (PSRR) \), slew rate \( (SR) \), common-mode voltages \( (CMRR) \), and output capacitance \( (CL) \), must be met when constructing differential amplifier circuits. Important design parameters are power dissipation \( (P_{diss}) \) and total transistor area \( (TTA) \). During the design phase of CMOS circuits, variables such as transistor width, length, and input bias current are optimized [16] [17].

Figure 1 shows a circuit diagram for a CMOS differential amplifier with a current mirror load. The given circuit has been thoroughly studied and examined utilizing theoretical and mathematical approaches. [18], [19]. The circuit is optimized using a 0.5 pF load capacitor and a voltage source of \( \pm 1.8 \) volt for CMOS technology of 0.18 \( \mu m \) length [20].

---

**Fig 1. Differential amplifier with current mirror load**
The following parameters and objectives are taken into account when designing a differential amplifier:

- Design of \( \frac{W_1}{L_1} = \frac{W_2}{L_2} \) width-to-length ratio to meet \( Av \) requirements
- Design of \( \frac{W_3}{L_3} = \frac{W_4}{L_4} \) width-to-length to achieve the upper ICMR
- Design of \( \frac{W_5}{L_5} = \frac{W_6}{L_6} \) design to width-to-length to achieve the lower ICMR
- Finding out \( Id5 (I_{ss}) \) range to satisfying both \( SR \) and \( P_{diss} \).

IV. TEST SETUP FOR AUTOMATIC ANALOGUE CIRCUIT DESIGN

![Flow chart for designing automatic circuits](image_url)

This work aims to decrease the total CMOS transistor area under the restrictions of design variables and design requirements using the DBO algorithm. Figure 1 illustrates the automated CMOS analog circuit design test setup. The optimizer designs CMOS-based analog circuits for required specifications using an algorithm and a circuit simulator. The circuit's desired performance parameters are determined. These serve as a target for the optimization process. Through the systematic variation of individual parameters within their pre-defined limitations (upper and lower bounds), the optimization approach finds circuit designs. When creating CMOS analog circuits, the channel length (L) and channel width (W) of every PMOS and NMOS transistor are crucial parameters to consider because they have a substantial impact on circuit efficiency. The fitness function is defined as follows and is calculated by examining the simulation data from the circuit simulator.

\[
F_e = \sqrt{\sum \left( \frac{Spec_{Desired} - Spec_{Sim}}{Spec_{Desired}} \right)^2}
\]  

A fitness function is used to evaluate the circuit's performance [21]. The process ends if this value satisfies the termination requirements. Common criteria include minimizing the fitness function value to a sufficient degree or attaining a maximum number of iterations. If the specified termination requirements are not satisfied, the optimization process will produce an updated set of design parameters that fall within their allowable ranges [22]. To satisfy the termination requirements, the process iterates, developing designs and evaluating them. The algorithm runs until the error value is above 1e-5 or the iteration number is 100. To optimize the DA circuit, in this study, the
performance of the dung beetle optimization (DBO) algorithm is evaluated. A distinct method will be developed to optimize the differential amplifier (DA) design based on predefined criteria. The aim is to examine the degree to which they meet the standards of performance that are required.

V. SIMULATION RESULTS AND DISCUSSIONS

Python is utilized in this research to investigate how well a CMOS circuit of differential amplifier with current mirror load designed optimally and performs on an Ubuntu computer with an Intel Core i5-8265U processor operating at a frequency of 1.60 GHz and RAM size is 8 GB. To find the optimal solution of DA circuit for both the 0.18 µm and 0.35µm CMOS technologies with predefined design specifications simulations were carried out using NGSPICE software. This study’s three main objectives were to decrease power dissipation, optimize amplifier gains, and minimize MOS transistor area overall while keeping the relevant design requirements and standards. The MOS transistors have been sized carefully to lessen the effect of channel length modulation. L5 and L6 MOS transistors have been sized carefully to lessen the effect of channel length modulation, and minimize MOS transistor area overall while keeping the relevant design requirements and standards. The CMOS differential amplifier with current mirror load circuit required specifications are $A_v$, $UGB$, $PM$, positive $PSSR$, negative $PSSR$, $RSR$, $FSR$, $CMRR$, $Pdiss$, and total MOS transistor area (TTA). Minimizing size and power dissipation in analog circuits are both critical for building efficient and compact electronic systems. In this study, our main objective is to minimize these quantities using the DBO algorithm.

Table 1. Optimal design parameters for 0.35 µm technology CMOS differential amplifier circuit with current mirror load

<table>
<thead>
<tr>
<th>No.</th>
<th>Set of design variables</th>
<th>Range of design parameters</th>
<th>Achieved values</th>
<th>CRPSO [25]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$W_1/L_1 = W_2/L_2$ ($\mu$m/µm)</td>
<td>$(W)$ 3.5 to 30</td>
<td>29.4/3.5</td>
<td>24.5/3.5</td>
</tr>
<tr>
<td>2</td>
<td>$W_3/L_3 = W_4/L_4$ ($\mu$m/µm)</td>
<td>$(W)$ 3.5 to 30</td>
<td>11.3/3.5</td>
<td>7.5/3.5</td>
</tr>
<tr>
<td>3</td>
<td>$W_5/L_5$ ($\mu$m/µm)</td>
<td>$(W)$ 3.5 to 30</td>
<td>4.2/1.4</td>
<td>3.4/1.4</td>
</tr>
<tr>
<td>4</td>
<td>$W_6/L_6$ ($\mu$m/µm)</td>
<td>$(W)$ 3.5 to 30</td>
<td>4.2/1.4</td>
<td>4.8/1.4</td>
</tr>
<tr>
<td>5</td>
<td>$I_{bias}$ (µA)</td>
<td>3.5 µA to 30 µA</td>
<td>125</td>
<td>116.8</td>
</tr>
</tbody>
</table>

Applying the suggested DBO method, a differential amplifier with specific design parameters is designed. According to the simulation results, the design parameters effectively satisfy the given design criteria. Table 1 provides the results achieved for the DA circuit employing 0.35 µm CMOS technology with various evolutionary algorithms (EAs) utilized in this study. The best design parameters of differential amplifier from the many EAs were used for validation. Table 2 lists the specifications derived using the Ngspice simulator for the obtained design parameters. Table 2 indicates an improvement over earlier research in the DBO-based design of the differential amplifier circuit.

Table 2. Simulation results and desired criteria for DA in 0.35 µm CMOS technology using various EAs

<table>
<thead>
<tr>
<th>No.</th>
<th>Specifications</th>
<th>Expected value</th>
<th>Obtained Specifications</th>
<th>CRPSO [25]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>$A_v$ (dB)</td>
<td>&gt;40</td>
<td>42</td>
<td>44.02</td>
</tr>
<tr>
<td>2</td>
<td>$UGB$ (MHz)</td>
<td>&gt;10</td>
<td>10</td>
<td>10.48</td>
</tr>
<tr>
<td>3</td>
<td>$PM$ (°)</td>
<td>&gt;45</td>
<td>83.80</td>
<td>83.73</td>
</tr>
<tr>
<td>4</td>
<td>+ve $PSSR$ (dB)</td>
<td>&gt;40</td>
<td>40.10</td>
<td>60.79</td>
</tr>
<tr>
<td>5</td>
<td>-ve $PSSR$ (dB)</td>
<td>&gt;70</td>
<td>68.00</td>
<td>108.6</td>
</tr>
<tr>
<td>6</td>
<td>$RSR$ (V/µs)</td>
<td>&gt;10</td>
<td>22.4</td>
<td>12.28</td>
</tr>
<tr>
<td>7</td>
<td>$FSR$ (V/µs)</td>
<td>&gt;10</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>8</td>
<td>$CMRR$ (dB)</td>
<td>&gt;70</td>
<td>84.2</td>
<td>83.17</td>
</tr>
<tr>
<td>9</td>
<td>$Pdiss$ (µW)</td>
<td>&lt;1,000</td>
<td>1260</td>
<td>559.5</td>
</tr>
<tr>
<td>10</td>
<td>TTA (µm²)</td>
<td>&lt;400</td>
<td>296</td>
<td>235.48</td>
</tr>
</tbody>
</table>
Our cost function aims to reduce the overall component area to below 300 µm² and minimize power dissipation while ensuring all requirements are met. The differential amplifier attains a 45.33 dB gain and a 25.34 MHz UGB. The results presented are compared with the most recent studies, PSO [18], Seeker Optimization Algorithm (SOA) [23], Opposition based Harmony Search Algorithm (OHS) [24], craziness-based particle swarm optimization (CRPSO) [25]. From the comparison, it’s obvious that DBO beats other algorithms in optimizing analog circuits, obtaining a transistor area of 263.06 µm². The power consumption of 35.03 µW is achieved by this method, and its dissipation is about 94% less than that of the best-performing methods from previous studies.

**Table 3. Optimal design parameters for 0.18 µm technology CMOS differential amplifier circuit with current mirror load**

| Sr. No. | Set of design variables | Range of design parameters | Achieved values
<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(W_1 = W_2) (µm)</td>
<td>3.5 to 30</td>
<td>8.78</td>
</tr>
<tr>
<td>2</td>
<td>(W_3 = W_4) (µm)</td>
<td>3.5 to 30</td>
<td>63.47</td>
</tr>
<tr>
<td>3</td>
<td>(W_5 = W_6) (µm)</td>
<td>3.5 to 30</td>
<td>54.16</td>
</tr>
<tr>
<td>4</td>
<td>(I_{bias}) (µA)</td>
<td>1 µA to 200 µA</td>
<td>22.43</td>
</tr>
</tbody>
</table>

**Table 4. Simulation results and desired criteria for DA in 0.18 µm CMOS technology using various EAs**

<table>
<thead>
<tr>
<th>Sr. No.</th>
<th>Specifications</th>
<th>Expected value</th>
<th>Obtained Specifications</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(A_v) (dB)</td>
<td>&gt;40</td>
<td>25.66</td>
</tr>
<tr>
<td>2</td>
<td>UGB (MHz)</td>
<td>&gt;10</td>
<td>-</td>
</tr>
<tr>
<td>3</td>
<td>PM (°)</td>
<td>&gt;45</td>
<td>90.36</td>
</tr>
<tr>
<td>4</td>
<td>+ve PSSR (dB)</td>
<td>&gt;40</td>
<td>-</td>
</tr>
<tr>
<td>5</td>
<td>-ve PSSR (dB)</td>
<td>&gt;70</td>
<td>-</td>
</tr>
<tr>
<td>6</td>
<td>RSR (V/µs)</td>
<td>&gt;10</td>
<td>26.08</td>
</tr>
<tr>
<td>7</td>
<td>FSR (V/µs)</td>
<td>&gt;10</td>
<td>14.61</td>
</tr>
<tr>
<td>8</td>
<td>CMRR (dB)</td>
<td>&gt;70</td>
<td>44.64</td>
</tr>
<tr>
<td>9</td>
<td>(P_{diss}) (µW)</td>
<td>&lt;1.000</td>
<td>-</td>
</tr>
<tr>
<td>10</td>
<td>TTA (µm²)</td>
<td>&lt;400</td>
<td>54.34</td>
</tr>
</tbody>
</table>

The outcomes obtained for the DA circuit using 0.18 µm CMOS technology with the different evolutionary algorithms (EAs) used in this investigation are shown in Table 3. The obtained specifications of the differential amplifier are shown in Table 4. When compared to the results obtained by the DE, PSO, and CSPSO algorithms, the DBO method achieved all specified parameters with the lowest total transistor area (TTA) of 228.52 µm² and the lowest power dissipation \(P_{diss}\) 39.77 µW. Important characteristics like gain, power dissipation, and transistor area are successfully optimized using this method. Compared to the optimal design parameters found in previous articles, it achieves the maximum gain of 41.01 dB while lowering power consumption by around 10% and needing the least amount of transistor area. Overall, the DBO approach appears to be a useful tool for creating tiny, high-performance differential amplifiers that consume little power for 0.18µm and 0.35 µm CMOS technology.
The convergence rates of differential amplifier circuits created utilizing the two distinct CMOS technologies of 0.18 µm and 0.35 µm are probably compared in Figs. 3 and 4. Differential amplifier circuits can be effectively optimized with the help of the DBO method. To maximize circuit performance, it focuses its efforts during the first 20% of the iteration process.

Table 5. Computational performance of DBO algorithm

<table>
<thead>
<tr>
<th>Parameters</th>
<th>CMOS Technology</th>
<th>No. of swarms</th>
<th>No. of Iteration</th>
<th>Fmin</th>
<th>Fmax</th>
<th>Fmean</th>
<th>Fstd</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>0.35 µm</td>
<td>30</td>
<td>100</td>
<td>4.86E-03</td>
<td>5.77E-03</td>
<td>5.05E-03</td>
<td>3.23E-04</td>
</tr>
<tr>
<td></td>
<td>0.18 µm</td>
<td>30</td>
<td>100</td>
<td>3.28E-03</td>
<td>1.08E-02</td>
<td>4.75E-03</td>
<td>2.61E-03</td>
</tr>
</tbody>
</table>

Table 5 provides the computational performance of DBO algorithm for DA circuit employing 0.18 µm and 0.35 µm CMOS technologies. Figure 3 and Figure 4 show the convergence performance of the DBO algorithms of a differential amplifier.
VI. CONCLUSION

Analog CMOS circuit automation is a challenging and time-consuming task, tackled by employing an evolutionary optimization technique based on the DBO algorithm to optimize circuits, such as a CMOS differential amplifier. Furthermore, integration of the DBO algorithm into our Ngspice tool enables optimal sizing of analog circuits through simulation-based optimization in Python. Aiming to achieve all targeted specifications such as higher voltage gain, better frequency response, smaller area, and lower power consumption. The DBO-based design approach is implemented to develop differential amplifier designs with current mirror loads for TSMC 0.35µm and 0.18µm CMOS technologies. These designs not only satisfy all design requirements, but they also use less transistor space and power dissipation overall than the previous approaches.

REFERENCES


