Academic Journal of Electrical and Computer Engineering
Vol. 02, No. 01, August 2026
https://doi.org/10.31272/ajece.47
https://ajece.uomustansiriyah.edu.iq
ISSN 3079-2878
Research Article

Performance Enhancement of MOSFET Circuits Using Evolutionary and Nature-Inspired Optimization Algorithms

Hussein M. Hathal[1*], Riyadh A. Alhilali[2], Reeman M. Hathal[3]

[1,2] Electrical Engineering Department, College of Engineering, Mustansiriyah University, Baghdad, Iraq
[3]Bint Al-Huda Intermediate School for Girls, Second Al-Karkh Directorate of Education, Baghdad, Iraq.
[*]Email: hussein.m.hathal@uomustansiriyah.edu.iq

Received: 01/07/2026 Revised: 12/08/2026 Accepted: 21/08/2026

Abstract— **In this study, a comparative study of Particle Swarm Optimization (PSO), Genetic Algorithm (GA) and Secretary Bird Optimization Algorithm (SBOA) is presented to optimize voltage gain of a common-source MOSFET amplifier. The optimization problem was set up with the design vector $X = [W/L, R_D, R_L]$, with $1 \leq W/L \leq 100$, $500 \leq R_D \leq 5000\ \Omega$, and $100 \leq R_L \leq 5000\ \Omega$. The adopted circuit model used $V_{DD} = 12$ V, $V_{th} = 1.5$ V, and $\mu C_{ox} = 50 \times 10^{-6}$ A/V^2. All algorithms were tested on a population of 30 candidate solutions, 120 iterations and 20 independent runs with the same variable bounds and objective function environments for a fair comparison. The voltage gain of PSO was found to be maximum, i.e. 1378.125, converging to an upperbound solution of W/L = 100, RD = 5000Ω and RL = 5000Ω where the value of standard deviation across the runs recorded was found to be zero. SBOA had the largest gain of 1226.267555, but was more variable from run to run and it failed to grow after a short period of time, while GA had a smaller gain of 1176.766589, with slower step-wise convergence and larger search diversity. The results obtained indicate that PSO is the most reliable and fastest converging method for the problem used here which is a single objective gain maximization problem. The upper-bound solution, however, also shows that future multi-objective circuit optimization is required to take into consideration bandwidth, power, output-swing, noise and operating-region constraints.**

keywords— MOSFET amplifier circuits; Particle Swarm Optimization; Genetic Algorithm; Secretary Bird Optimization Algorithm; swarm intelligence.

I. INTRODUCTION

Optimization rolls an essential way in modern design of electronic circuit since, analog and mixed-signal circuits are usually containing coupled design variables, nonlinear device characteristics, and conflicting performance requirements. In Amplifier circuits - based MOSFET, voltage gain, power consuming, biasing state, and load interaction are impacted simultaneously by resistive components and transistor configurations. Therefore, choosing suitable values of W/L, RD, and RL cannot be considered as a simple linear tuning problem. Otherwise, it is a constrained nonlinear optimization problem that changes any variable may improve gain while degrading stability, feasibility, or operating-point behavior.

When objective functions are smooth, convex and differentiable then, traditional deterministic optimization methods are useful. However, MOSFET circuit optimization often includes device constraints, nonlinear equations, and possible discontinuities which are caused by operating-region violations or simulation limits. For such cases, metaheuristic algorithms -based on population are attractive due to need no gradient information and can search simultaneously multi-areas of the design space. Recent reviews have supposed that metaheuristic algorithms still important for complex engineering optimization due to flexibility mechanisms in balancing exploration and exploitation [1]– [4].

Particle Swarm Optimization (PSO) has the most established swarm-intelligence methods. Each candidate solution updates according to components: inertia, cognitive memory, and social guidance. Inertia component deals with motion tendency, while cognitive component drags into its personal best solution, and the social component drags it to best solution found by the swarm. This methodology gives PSO rapid convergence and simple implementation. Nevertheless, population diversity can be reduced by the same social attraction and lead to premature convergence when the swarm collapses too quickly around a single best point [5], [6]. This behavior is particularly important in MOSFET optimization since, a rapid motion toward high gain may force the solution to the boundaries of W/L, RD, or RL.

Genetic Algorithm (GA) appears an evolutionary search mechanism depends on selection, crossover, mutation, and replacement. Unlike PSO, GA does not move solutions by velocity vectors. Instead, it includes candidate chromosomes through reproduction. Selection permits better solutions with a higher probability of contributing to the next generation, crossover mixes information from two parent designs, and mutation prefers random perturbations that protect the population from losing diversity. GA in general is robust and suitable for global search, but it often asks many generations to near high-quality solutions since improvement is accumulated step-by-step through genetic operators [7], [8].

The Secretary Bird Optimization Algorithm (SBOA) is a modern nature-inspired population-based optimizer proposed by Fu et al. [9]. It employs the survival behavior of secretary birds through two main behavioral mechanisms: hunting and escaping. The hunting stage is consisted from searching for prey, consuming prey, and attacking prey, while escape stage employs camouflage or rapid movement away from predators. The original SBOA paper reports strong performance on CEC benchmark functions and engineering design problems, but later studies also indicate that SBOA performance depends strongly on diversity preservation and update-strategy design [10]– [13]. Therefore, applying SBOA to a MOSFET amplifier design problem provides useful insight into its practical behavior in electronic circuit optimization.

Recent researches of analog circuit design, metaheuristics and surrogate-assisted methods have been employed to optimize analog circuits due to exhaustive manual tuning becomes inefficient as the number of design variables increases [14]– [17].

Metaheuristic- and surrogate-assisted optimization techniques have been proven to be effective in analog circuit optimization in the past. Most of the related research has been dedicated to proposing or application of a single optimization technique, to optimizing the optimizer itself, or to optimizing general analog-circuit parameters, however. Therefore, a controlled comparison of the known PSO and GA methods with a new method (SBOA) for optimizing the voltage-gain of the MOSFET amplifier is still not sufficiently studied. The novelty of the present study is the comparison of PSO, GA and SBOA with the same design-variable ranges, the same number of iterations, the same random initial run conditions and the same objective function. Apart from comparing the maximum voltage gain, statistical repeatability, convergence behavior, exploration–exploitation characteristics, and physical meaning of the optimized MOSFET parameters are explored. This consolidated assessment gives not only an idea of algorithm performance (which results in the greatest gain) but also insight into the search behavior of the algorithms with respect to a specific optimization problem, the MOSFET amplifier.

Thus, the primary contribution of this study is to provide a unified comparative analysis of PSO, GA and SBOA for optimizing voltage gain in MOSFET amplifier. In the present comparison, the best solution, the average solution, the statistical variability, the convergence characteristics, the computational time and the physical interpretation of the optimized variables (W/L), (RD) and (RL) are taken into consideration, as well as studies in which only the final objective value is reported. Special focus is placed on the newly proposed SBOA and its performance compared to the existing PSO and GA algorithms with the same optimization settings. The results accordingly reveal the performance advantages and weaknesses of every algorithm to the single-objective MOSFET design problem studied.

II. METHODOLOGY

This section describes MOSFET optimization model, the selected design variables, the objective function, and the mathematical update rules of PSO, GA, and SBOA. The aim is to make the computational procedure simpler for reproduction and to relate each mathematical stage to its role in the MOSFET amplifier design problem.

To make a fair comparison between PSO, GA, and SBOA, they were all assessed using the same objective function, population size, number of iterations, design-variable bounds, and number of independent runs. The $X = [W/L, R_D, R_L]$was used to represent each candidate solution. The number of candidate solutions was determined as 30 and 120 runs were performed on each algorithm 20 times independently. The search ranges were $(1 \leq W/L \leq 100)$, $(500 \leq R_D \leq 5000\ \Omega)$, and $(100 \leq R_L \leq 5000\ \Omega)$.. The circuit constants were (V_{DD}=12 V), $(V_{th} = 1.5\text{ V})$, and $(\mu C_{ox} = 50 \times 10^{-6}\text{ A/V}^2)$.

The first populations were created with a uniform random distribution between the limits. Since the algorithms were formulated as a minimization procedure, the maximization of the voltage gain was done by the minimization of the negative of the voltage gain, $(-Av)$**.** Solutions that went past the ranges were rounded to the nearest end value of the range. There was no tolerance-based early stopping condition, only 120 iterations were set as a stopping criterion.

For PSO, the inertia weight was fixed to (w=0.7) and the cognitive and social weights were fixed to (c1=c2=1.5). The velocity of the particles was set to zero and there were no explicit velocity limits. GA: Elitism used to select best 20% of the population. The parents were randomly sampled from the best half of the population and arithmetic crossover was employed with a uniformly distributed coefficient ($\alpha \in [0,1]$). Production of mutations at a rate of 0.1, with additive perturbations of $(0.1N(0,1))$.All of the optimization parameters are listed in Table 1.

TABLE I: THE OPTIMIZATION PARAMETERS

Parameter	Value
Design vector	$X = [W/L, R_D, R_L]$
Number of variables	3
Population size	30
Maximum iterations	120
Independent runs	20
$\boldsymbol{W/L}$ **range**	1–100
$\boldsymbol{R_D}$**range**	500–5000 Ω
$\boldsymbol{R_L}$ **range**	100–5000 Ω
Supply voltage, $\boldsymbol{V_{DD}}$	12 V
Threshold voltage, $\boldsymbol{V_{th}}$	1.5 V
Process parameter,$\boldsymbol{\mu C_{ox}}$	50×10^{-6} A/V^2

Initialization	Uniform random initialization
Boundary handling	Clamping to the nearest bound
Stopping criterion	Maximum of 120 iterations
Early-stopping criterion	None
Objective	Maximize $A_v$by minimizing $-A_v$

A. Circuit Topology and MOSFET Model

The circuit that is analyzed in this study is a common source NMOS amplifier as shown in Figure 1. The drain resistance$\boldsymbol{R_D}$ is inserted between the drain terminal and the supply voltage $\boldsymbol{V_{DD}}$and the load resistance $\boldsymbol{R_L}$ is inserted at the output of the amplifier. The transistor's operating point is determined by a DC gate voltage \(V_{GS}\), with source connected to the reference node. The investigated design variables are the width to length ratio of the transistor $\boldsymbol{W/L}$and drain resistance $\boldsymbol{R_D}$ and load resistance $\boldsymbol{R_L}$. The adopted circuit and device parameters are $\boldsymbol{V_{DD} = 12}$ V, $\boldsymbol{V_{th} = 1.5}$ V, and $\boldsymbol{\mu C_{ox} = 50 \times 10^{-6}}$ A/V$^{\mathbf{2}}$.

It is assumed that the device dimensions and biasing condition relate to the transconductance by a first-order square-law MOSFET model. Assuming the transistor is in saturation and the channel-length modulation is ignored; the drain current is given by:

$$\mathrm{I_D} = \frac{1}{2}\mu\mathrm{C_{ox}}\frac{\mathrm{W}}{\mathrm{L}}(\mathrm{V_{GS}} - \mathrm{V_{th}})^{2.} \quad (1)$$

The corresponding transconductance is:

$$\boldsymbol{g_m = \mu C_{ox}\frac{W}{L}(V_{GS} - V_{th})} \quad (2)$$

or, equivalently,

$$\boldsymbol{g_m = \sqrt{2\mu C_{ox}\frac{W}{L}I_D}.} \quad (3)$$

This simplified model is used to take a computational efficient fitness evaluation, so that the optimization algorithms can be compared. However, it does not account for higher order effects, like mobility degradation, body effect, parasitic capacitances, temperature dependence and process variations. These restrictions should be taken into account when interpreting the optimized results.

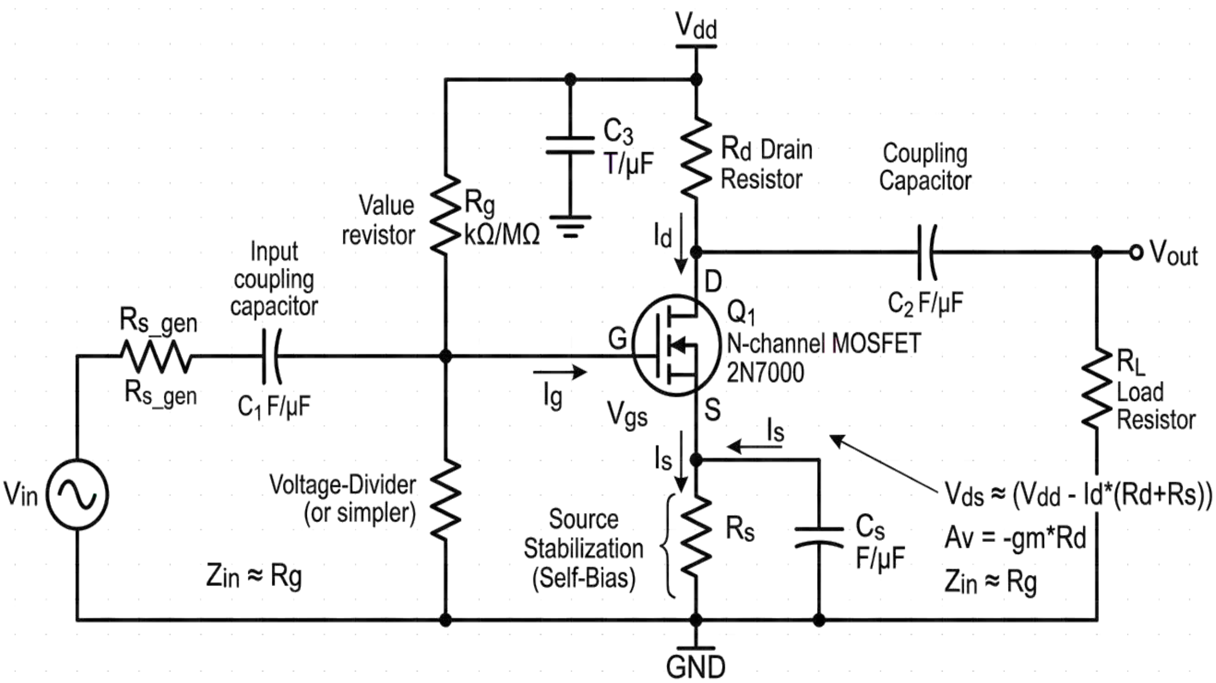

Fig 1: Common Source NMOS Amplifier

B. Small-Signal Equivalent Model

The small signal equivalent model of the common-source amplifier is shown in figure 2. The controlled current source $\mathrm{g_m v_{gs}}$and the Output resistance$\mathrm{r_o}$ represent the MOSFET. The resistances of the drain and the load are in parallel at the output. Thus, the small signal voltage gain is

$$\mathrm{A_v} = \frac{\mathrm{v_o}}{\mathrm{v_i}} = -\mathrm{g_m}(\mathrm{R_D} \parallel \mathrm{R_L} \parallel \mathrm{r_o}). \quad (4)$$

If the channel-length modulation is ignored, r_o is considered to be very high compared to R_D and R_L, and the expression for the voltage gain is reduced to:

$$A_v \approx -g_m(R_D \parallel R_L) \quad (5)$$

$$A_v \approx -g_m\left(\frac{R_D R_L}{R_D + R_L}\right) \quad (6)$$

From this relationship, it can be seen that the voltage gain depends on both the transconductance of the MOSFET and the effective output resistance. Under the adopted biasing model, increasing the value of W/L will increase the value of g_m, whereas increasing the value of R_D and R_L will increase the effective output resistance. This is why it is not possible to solve the optimization problem by changing the three variables individually.

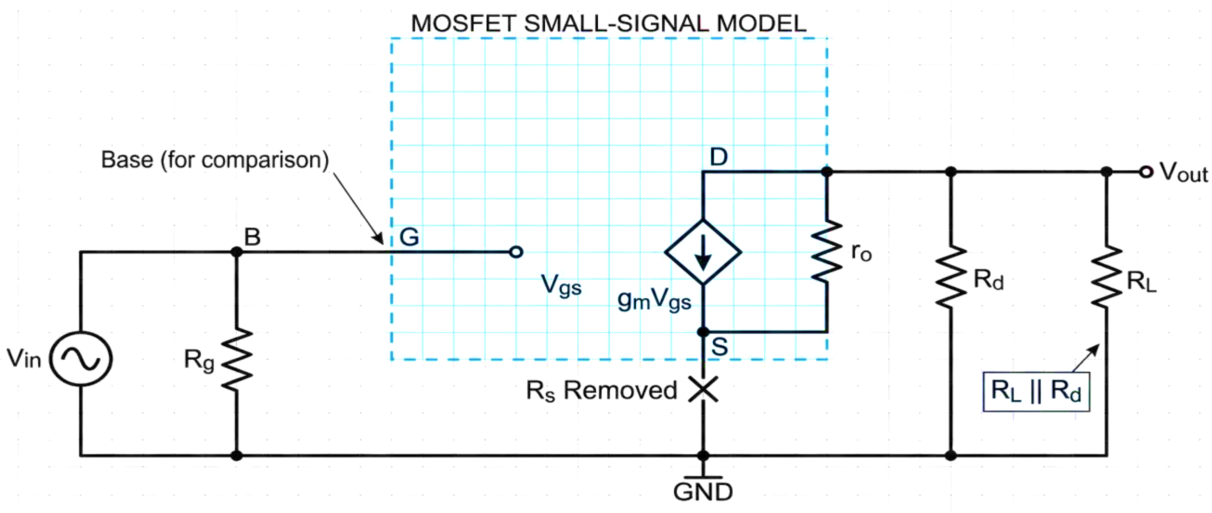

Fig 2: The small signal equivalent model

C Design Constraints and DC Operating-Point Analysis

The numerical search ranges were selected as $1 \leq W/L \leq 100$, $500 \leq R_D \leq 5000\,\Omega$, and $100 \leq R_L \leq 5000\,\Omega$. These limits define the range of space in which the algorithms are allowed to optimize, and will not allow negative or overly large parameters to be produced. But numerical bounds do not necessarily mean a physically feasible amplifier. The operating point of the DC amplifier should also maintain the saturation condition for the MOSFETs and offer adequate headroom for the output voltage.

The DC drain-to-source voltage for the common-source circuit is given by:

$$V_{DS} = V_{DD} - I_D R_D. \quad (7)$$

When it is not otherwise specified, the following condition applies to the MOSFET:

$$V_{GS} > V_{th} \quad (8)$$

and is in saturated condition when:

$$V_{DS} \geq V_{GS} - V_{th}. \quad (9)$$

Thus, any candidate solution that is actually feasible should meet the following criteria:

$$0 < I_D < \frac{V_{DD}}{R_D} \quad (10)$$

As well as meeting the saturation requirement. High values of R_D may result in a larger small-signal gain but can result in a smaller available voltage headroom and/or output swing. Likewise, a higher W/L gives higher transconductance, but also higher transistor area and parasitic capacitances. A large R_L will be beneficial to the output load but may not be the load for a specific application. Thus, the upper-bound solution that resulted from the single-objective optimizer should be interpreted as the mathematical optimal solution of the gain model used, not as a complete circuit design.

Table II. Circuit parameters and optimization bounds

Parameter	Value or range	Description
V_{DD}	12 V	Supply voltage
V_{th}	1.5 V	Threshold voltage
μC_{ox}	50×10^{-6} A/V^2	Process parameter
W/L	1–100	Transistor aspect ratio
R_D	$500 - 5000\ (\Omega)$	Drain resistance
R_L	$100 - 5000\ (\Omega)$	Load resistance
MOSFET region	Saturation	Intended operating region
Objective	Maximize A_v	Single-objective formulation

D Formulation optimization of MOSFET

A constrained maximization task is considered to the optimization problem. Maximizing voltage gain (Av) of a

MOSFET amplifier can be done by selecting an appropriate vector of design variables. The design vector is represented as:

$$X = [W/L, RD, RL] \quad (11)$$

Where W/L is the transistor width-to-length ratio, RD is the drain resistance, and RD is the load resistance. The optimization objective is written as:

$$maximize f(X) = A_v(X) \quad (12)$$

Subjected to each bound (lower and upper) for each design variable:

$$X_{min} <= X <= X_{max} \quad (13)$$

The circuit equations or the MATLAB evaluation function computes the corresponding voltage gain, for each candidate vector X. If a candidate produces a higher feasible gain, therefore it is considered better. The optimizer mission is to investigate the allowed region and recognize the group of W/L, RD, and RL which give maximum gain subjected to the imposed constraints.

Generally, potent transconductance and the output - load resistance interplay is effect on the gain strongly. The voltage gain relation of a common-source MOSFET amplifier is simplified as follows:

$$Av \approx -gm(RD \parallel RL \parallel ro) \quad (14)$$

Where;

g_m is the MOSFET transconductance and,

r_o is the small-signal output resistance.

Relation (4) shows gain improvement is affected by increasing values RD or RL, but it also shows why the response is not purely linear. The drain current and transconductance are affected by changing in W/L value, while the output load and voltage swing are affected by resistance values. Therefore, the optimization process must evaluate the combined effect of all variables rather than tuning them independently.

2.4 Particle Swarm Optimization (PSO)

PSO deals with each MOSFET design as a particle moving through the search space. A particle includes a position vector x_i and a velocity vector v_i. The position contains the circuit variables W/L, RD, and RL. The velocity controls particle changes its variables from one iteration to the next. Each particle saves its own best previous design pbest_i, while the population saves the global best design gbest. The standard PSO velocity equation is:

$$v_i^{t+1} = wv_i^t + c_1r_1(pbest_i - x_i^t) + c_2r_2(gbest - x_i^t) \quad (15)$$

The updated velocity is then used to obtain the next design vector:

$$x_i^{t+1} = x_i^t + v_i^{t+1} \quad (16)$$

Part of the previous motion is preserved by inertia term wv_i^t.Broader exploration is encouraged by a high inertia value because particles continue moving across the design region. The cognitive term $c_1r_1(pbest_i - x_i^t)$ drags particle backward the best design that is personally discovered, which helps each particle exploit its own memory. The social term $c_2r_2(gbest - x_i^t)$drags all particles toward the best design discovered by the population, this attraction extends the fast convergence of PSO, but it can also cause diversity loss if the population collapses too early around a single solution.

PSO procedure can be applied as follows:

(1) Random initial values for W/L, RD, and RL are generated for each particle;

(2) The voltage gain is evaluated;

(3) Pbest and gbest are updated;

(4) Velocity and position are updated by Eqs. (5) and (6);

(5) Boundary constraints are applied; and

(6) The process is repeated until the maximum number of iterations is reached. The PSO flowchart is shown in Figure 3

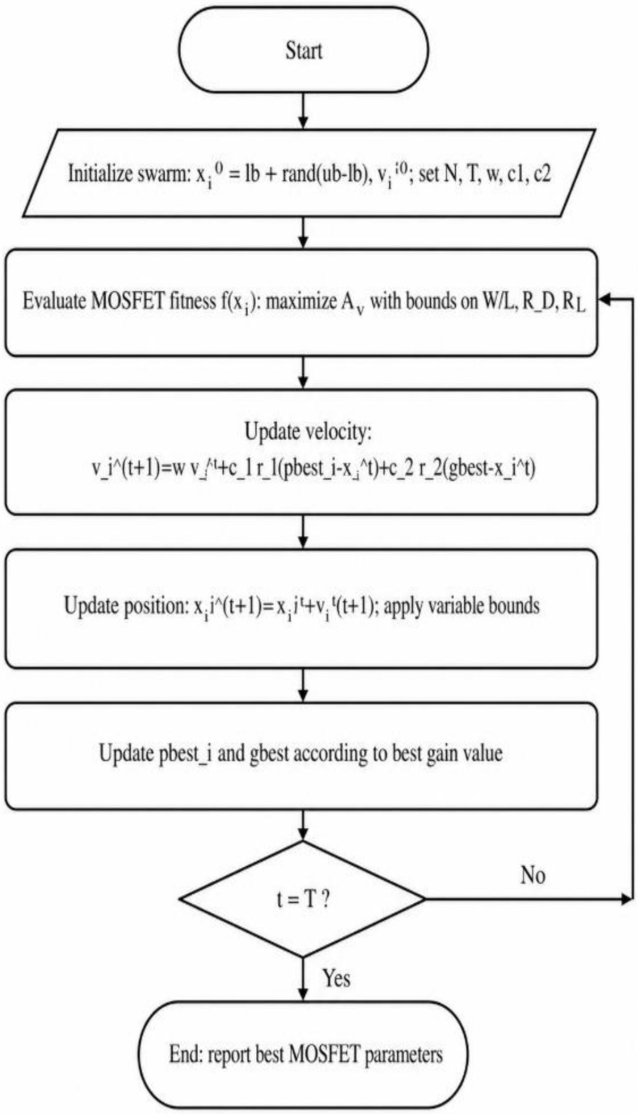

Fig 3. PSO flowchart matched with the velocity and position update equations.

E Genetic Algorithm (GA)

Each MOSFET design represents as a chromosome. The chromosome is composed of the variable's W/L, RD, and RL.

The population evolves over generations through selection, crossover, mutation, and replacement. A common proportional selection probability is given by:

$$P_i = f_i/\Sigma_{j=1}^{N} f_j \quad (17)$$

Where f_i, is the fitness of the ith candidate design.

Since the objective is gain maximization, higher-gain candidates receive higher selection probability. After selection, crossover combines two parent solutions to produce a new design as follows:

$$X_{new} = \beta X_1 + (1-\beta)X_2 \quad (18)$$

Where β, is a random or predefined crossover coefficient.

Equation in (8) produces an offspring located between two parent designs in the continuous design space. Mutation is then applied to introduce additional diversity as follows:

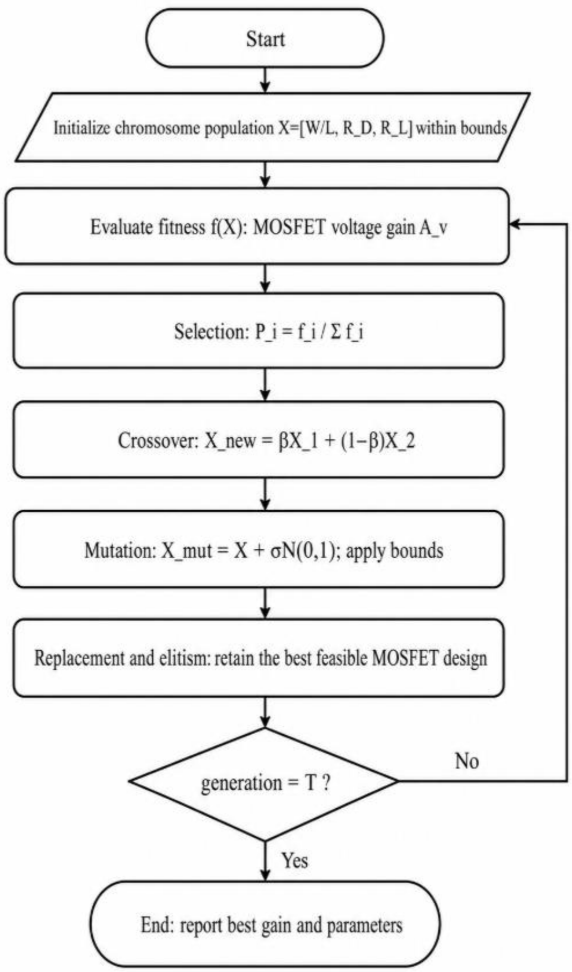

Fig4. GA flowchart matched with selection, crossover, and mutation equations.

F Secretary Bird Optimization Algorithm (SBOA)

SBOA is a modern population-based optimizer gifted by the secretary bird survival approach. The traditional SBOA formulation divides the search into hunting and escape behaviors [9]. The hunting behavior includes three time-dependent survey stages: searching for prey, consuming prey, and attacking prey.

$$X_{mut} = X + \sigma N(0,1) \quad (19)$$

Where σ, controls the mutation strength;
N (0,1), is a normally distributed random value.

Mutation is essential because crossover alone may not generate new regions outside the information which already present in the population. In MOSFET optimization, mutation can change W/L, R_D, or $R_L$slightly, permitting the algorithm to discover gain-improving designs that were not produced by crossover.

In general, GA search process is slower than PSO because generation-level reproduction is occurred through improvement instead of direct attraction to a global best. Therefore, GA is a useful diversity-preserving behavior. The GA steps used in this application can be depicted by a flowchart as shown in Figure 4.

The escape behavior represents exploitation through two possible strategies: camouflage and rapid movement. In the MOSFET application, each secretary bird represents one vector X = [W/L, RD, RL]. The initial population is generated as follows:

$$X_{i,j} = lb_j + r(ub_j - lb_j), i = 1,2,\ldots,N, j = 1,2,\ldots,Dim \quad (20)$$

For all candidate designs and after initialization, the fitness vector is calculated by evaluating the MOSFET gain. At each iteration, the best candidate is updated. During the first third iterations, SBOA uses a differential search rule corresponding to the prey-searching stage as follows:

$$x_{i,j}^{new,P1} = x_{i,j} + (x_{random_1} - x_{random_2})R_1, t < T/3 \quad (21)$$

By using the difference between two randomly selected candidate designs, relation (21) increases diversity. It means that a candidate design may change its W/L, R_D, or R_L according to the difference between two other feasible designs, thereby encouraging global search.

Through second third iterations, the consuming-prey stage uses Brownian motion and the best solution is given by:

$$RB = randn(1, Dim) \quad (22)$$

$$x_{i,j}^{new,P1} = x_{best} + exp((t/T)^4)(RB - 0.5)(x_{best} - x_{i,j}), T/3 < t < 2T/3 \quad (23)$$

Combination of stochastic perturbation with attraction toward the current best solution is done in this stage. It is intended to refine promising areas while still preserving random movement.

In the last third iterations, the attacking-prey stage uses Levy flight and a nonlinear perturbation factor as given below:

$$RL = 0.5Levy(Dim) \quad (24)$$

$$x_{i,j}^{new,P1} = x_{best} + (1 - t/T)^{2t/T} x_{i,j} RL, t > 2T/3(15) \quad (25)$$

Levy flight term permits occasional long jumps, while nonlinear factor decreases over iterations in order to shift the search toward exploitation. The original SBOA formulation defines Levy(D) as given below:

$$Levy(D) = s \times (u\sigma)/\left(|v|^{1/\eta}\right) \tag{26}$$

$$\sigma = \left[\Gamma(1+\eta)sin(\pi\eta/2) /\left(\Gamma\left((1+\eta) /2\right)\eta 2^{(\eta-1)/2}\right)\right]^{1/\eta} \tag{27}$$

Where s = 0.01 and η = 1.5.

Escape strategy is applied as an exploitation step, after hunting stage. The two escape cases are modeled as follows:

$$x_{i,j}^{new,P2} = x_{best} + (2RB - 1)(1 - t/T)^2 x_{i,j}, if rand < r_i; x_{i,j} + R_2\left(x_{random} - K x_{i,j}\right), otherwise \tag{28}$$

$$K = round\left(1 + rand(1,1)\right) \tag{29}$$

Only when each new candidate improves the fitness value, it is accepted. This greedy acceptance rule can accelerate convergence, but if the population loses diversity, it may also lead to stagnation.

This situation is essential for clarifying the results of SBOA - MOSFET optimization state. A flowchart of SBOA can be depicted in Figure 5.

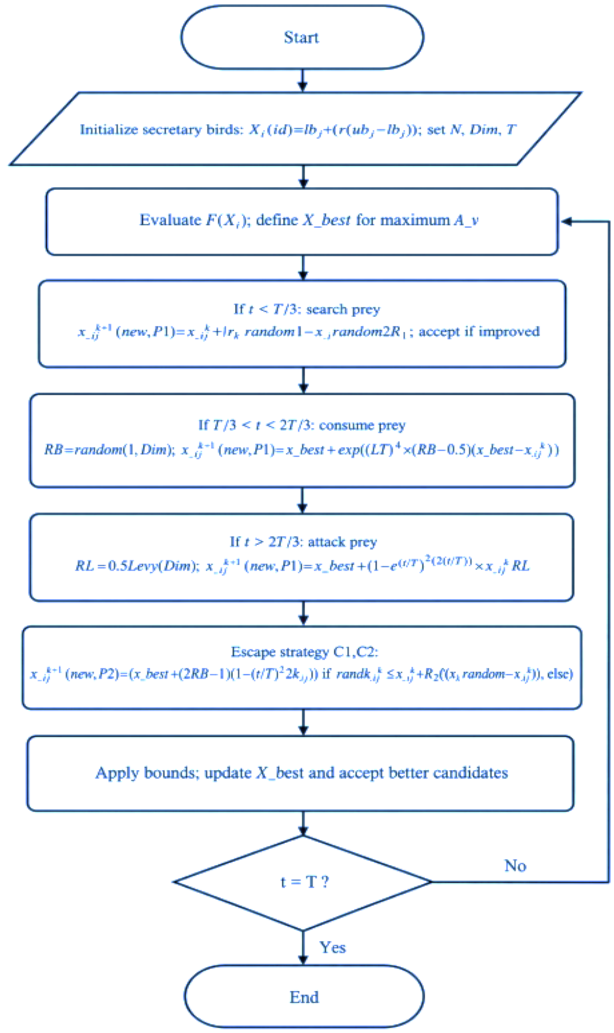

Fig 5. SBOA flowchart description

III. Results and Discussion

In this section, the results of the optimized MOSFET amplifier by PSO, GA and SBOA are presented and discussed. The optimized parameters of the circuits, the statistical performance, the convergence behavior, the computational time and the physical meaning of the designs obtained are taken into consideration when comparing the algorithms.

A. Optimized MOSFET Circuit Parameters

The results of optimizations done with the three algorithms are given in Table 1, showing the optimum values of the circuit parameters. Among the investigated methods, PSO achieved the highest voltage gain of 1378.125 by selecting (W/L=100), (R_D=5000 Ω), and (R_L=5000 Ω). All three values are the maximum values of the search ranges.

Table III: Design of optimized MOSFET circuit parameters

Algorithm	BestGain_Av	Best W/L	Best Rd (Ω)	Best R_L (Ω)	ExecutionTime (s)
PSO	1378.125	100	5000	5000	0.007151675
GA	1176.766589	98.34002573	3902.351478	4892.06084	0.009545175
SBOA	1226.267555	98.69881657	4580.260296	4437.396423	0.007032575

In the numeric sense, PSO was able to achieve the maximum gain of nearly 12.38% more than SBOA and 17.11% more than GA. The differences show the power of PSO to explore the upper-bound region which is preferred by the chosen objective function. The maximum gain is not the only measure of algorithmic performance, however, and convergence behavior, statistical repeatability, and search diversity come into play as well.

The PSO result is consistent with the small-signal gain relationship given in Eq. (4). Assuming the same circuit model and biasing conditions, the larger (W/L) will be, the greater the transconductance of the transistor will be, and the greater the resistance seen at the output will be, due to the increase of (RD) and (R_L). As a result, the objective function will be better the closer these variables get to their maximums.

By all PSO design variables converging to their maximum allowable value, another important characteristic of the optimization problem is also revealed. The objective function focuses on maximizing the magnitude of the voltage gain and does not add any other voltage-gain penalties such as bandwidth, power consumption, output-voltage swing, silicon area, bias sensitivity and operating-region constraints. Thus, the upper-bound solution is mathematically expected for the well-defined single-objective problem. With this solution, the gain is the most significant numerically, but it is necessary to consider the overall circuit specification before physically implementing this solution.

GA got 1176.766589 as its max gain instead. It had an optimal W/L value which was near the maximum, and (RD) and (R_L) were not as large as possible values. This result indicates that GA explored a wider range of intermediate solutions instead of converging immediately to the boundary of the search space. The selection, crossover, and mutation mechanisms preserved more search diversity, but the search diversity downsides are that they slowed convergence and reduced best gain, compared to PSO.

The highest gains that SBOA could obtain was 1226.267555, better than GA's gain but worse than PSO's. The values of optimized (W/L, RD, R_L) were not the highest, but still were not optimized to their maximum values. SBOA identified a competitive high-gain design, but it appears that the algorithm was converging to a high-gain region before it achieved the highest gain region identified by PSO.

Execution times were recorded for all 3 algorithms and the recorded times were small. The shortest execution time was for SBOA, which was only slightly exceeded by PSO, while GA needed slightly more time. However, the differences are slight and might be due to implementation and computational environment. Hence, the convergence behavior of the solution and quality of the solution are more significant than the differences in execution time between the two.

B Statistical analysis

The performance of the three algorithms for the recorded independent runs is summarized in Table 2. PSO had the best mean and maximum gain which was 1378.125 with zero standard deviation. This result shows that PSO was consistently finding the same solution, which is the high gain solution and so has a good repeatability for the objective function investigated.

Table IV: The statistical performance of the optimization algorithms

Algorithm	MeanGain	StdGain	MedianGain	MinGain	MaxGain	Mean W/L	Mean Rd	Mean R_L	MeanTime
PSO	1378.125	0	1378.125	1378.125	1378.125	100	5000	5000	0.0071516
GA	992.159010	153.2312971	1034.218961	685.7939865	1176.766589	88.9026376	3978.962063	4182.742252	0.00954517
SBOA	913.433343	165.783656	868.901502	642.449414	1226.26755	85.2215278	3901.1855	4026.70712	0.00703257

The zero standard deviation found by PSO is a sign of high reliability, but can also signify a fast decrease in diversity of the population. The particles seemed to quickly settle to the same upper bound solution due to the consistent preference towards large values of the three design variables within the gain function. So, PSO proved to be very successful for the current goal with a very strong exploitative search once it had found the boundary region.

The statistical results present an alternative view to the best-run comparison. PSO demonstrated gains around 38.90% and 50.88% higher than GA and SBOA, respectively. SBOA's mean gain and standard deviation were lower than those of the GA, and its maximum gain was higher, suggesting that SBOA had less average reliability and repeatability, but higher maximum gain. Hence, the maximum gain should be read and examined in conjunction with the mean and standard deviation when assessing general performance of each algorithm.

GA achieved a mean gain of 992.159010 and a standard deviation of 153.2312971. Its average gain was less than PSO, and the slight differences in gain between the runs demonstrates the random nature of crossover and mutation. These operators enabled GA to explore various parts of the search space, hence a variety of solutions. The preserved diversity also created variability in the rate and uniformity of GA's approach to the highest gain region.

SBOA achieved a maximum gain of 1226.267555 but had the lowest mean gain, 913.433343, and the highest standard deviation, 165.783656. This window between its best run and its worst run means that SBOA may have found a possible competitive answer in one run, but might not be able to do it again. The chosen SBOA configuration was found to be relatively variable and fast stabilizing indicative of a need for improved parameter control for this optimization problem, or the need for adaptive search mechanisms, or the need for hybridization.

For maximum gain, the algorithms were ranked as PSO, SBOA and GA. But when it comes to mean gain, the rankings are PSO, GA, SBOA. The maximum gain is the maximum possible ability of an algorithm, while the mean gain and standard deviation give a better idea of the average reliability of an algorithm. Thus, PSO showed the best overall performance. GA was more reliable on average, and SBOA was more repeatable than GA, but with better best-case performance.

C Convergence-Curve Analysis

The curves in the figures 4–8 show the convergence clearly, with the search behavior of the three algorithms clearly different.

As can be seen in Figure 6 the congruence curve in the PSO has a rapid rise in the initial few iterations and a virtually steady response thereafter. The velocity-update mechanism is to blame for this behavior, which takes particles' personal best positions and the global best position. When one particle sensed the high-gain BVR, the social part of the system quickly brought the rest of the particles towards it. This is why the recorded runs were found to be at 0 variation and converged at a high speed. The flattening of the curve, though, early on has also shown that there was a loss of diversity rapidly after the boundary solution was reached, as the swarm was the swarm.

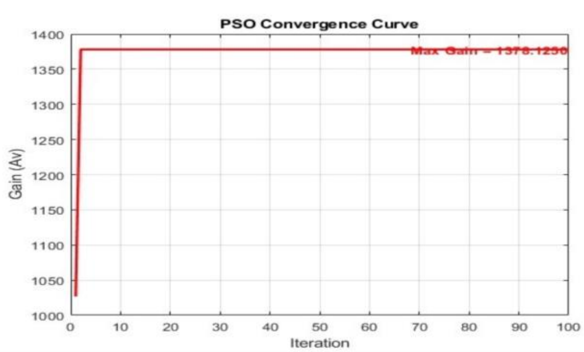

Fig 6. PSO convergence curve.

The curve of the GA convergence in Figure 7 is stepwise improved. An increase is done when an offspring has higher gain than the previously retained solution due to selection, crossover or mutation.

GA does not directly push all candidate solutions towards the global best position, as in PSO. Rather, changes are made over time as positive traits are blended between generations from various candidate designs. As a result, GA had a slower convergence rate than PSO, but a more varied search path.

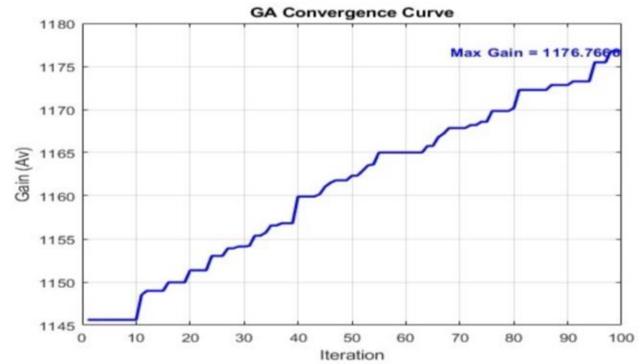

Fig 7. GA convergence curve.

When the SBOA convergence curve shown in Figure 8 reaches a competitive gain value, it turns nearly flat, meaning that it stagnates early on. While the commonly used SBOA contains differential search, Brownian motion, Lévy-flight perturbation, and escape strategies, the chosen parameter configuration did not offer enough explorations to consistently progress towards the upper-bound region found by PSO. This does not necessarily mean that the performance of SBOA is weak overall; it is just that the performance of SBOA could be improved by adaptive control or hybrid search techniques in this problem.

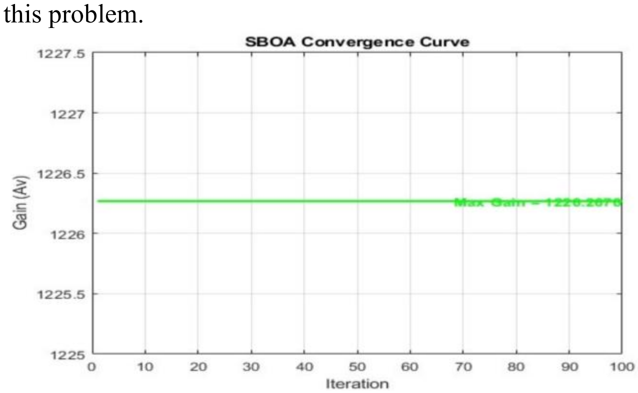

Fig 8. SBOA convergence curve.

Combined curves in figure 9 show the better convergence speed and convergence gain of PSO. While GA gradually increases throughout the search, SBOA even reaches a competitive region but can't improve much after that. These differences are specifically during the initial iterations, which shows a different balance between exploration and exploitation exhibited by each algorithm.

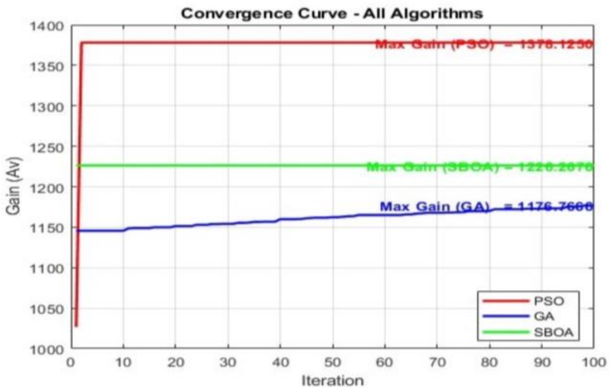

Fig 9. Combined convergence curve of all algorithms.

Overall, PSO had the best exploitation capability and the quickest rate of convergence for the defined gain maximization problem. GA and SBOA demonstrated varying search capabilities, with the former showing slower search but a wider range of variations and the latter having a better best search but being more susceptible to stagnation and run-to-run variability with the chosen configuration.

Convergence curves have to be understood in conjunction with the statistical results. The computational efficiency of the PSO is desirable since it stabilizes rapidly, but this also suggests that the swarm easily got clustered near the same boundary solution. The stepwise GA curve results from the occasional improvement of offspring through crossover and mutation, which is why it is less convergent than the other curves and has more run-to-run variation. This early leveling off of the SBOA curve implies that the search operators failed to do a good job of getting to a competitive area. This behavior is in line with its relative SD and difference between its maximum and its average gains.

D MOSFET design Analysis

Optimized parameters are obtained and it is confirmed that larger voltage gains are linked to relatively large values of (W/L), (RD), and (R_L), as expected from the small-signal gain relationship in Eq. (4). It is assumed that these conditions are biased, and that a larger (W/L) will make transconductance larger, and larger values of (RD) and (R_L) will increase the effective output resistance. PSO took advantage of this relationship the most as it sets all three variables to their maximum values. GA and SBOA also chose relatively high values, though they failed to consistently arrive at the full upper bound solution.

The PSO solution is optimum for the single objective voltage-gain maximization problem mathematically. However, if you want to maximize gain, it doesn't always yield the most practical amplifier design. A large (RD) may result in a decrease in the available voltage headroom, a decrease in the output-voltage swing, and/or greater sensitivity to bias variation. Likewise, as (W/L) increases, device area and parasitic capacitances grow, which may lead to a decrease in the amplifier bandwidth. Noise can also be influenced by high resistance values and so can be the DC operating point.

So, the reported results are considered to be the best solution with the chosen objective function and parameter ranges, and not the best possible MOSFET amplifier designs in general. Implementation would involve demonstration of the selected parameters to ensure the transistor operates in the desired region and meets other requirements, such as bandwidth, power consumption, noise, linearity, stability and output swing. All these thoughts inspire further research on constrained and multiple-objective optimization, in which the voltage gain is optimized along with other relevant circuit performance metrics.

It is possible to understand the reason behind the PSO solution converging to the upper bounds of all three design variables because of the mathematical form of the objective function used. In the limits indicated, the voltage gain calculated is monotonically increasing with W/L,$R_D$and R_L. Furthermore, the single objective formulation is used to optimize voltage gain without adding penalty terms or extra constraints on bandwidth, power consumption, output-voltage swing, device area, noise, and the DC operating region. The optimizer is thus naturally guided towards the highest feasible values of the design variables.

The numbers used for $W/L = 100$ and $R_D = 5000\ \Omega$along with $R_L = 5000\ \Omega$are realistic in the range used in this study, but should not be taken as the most practical for a real amplifier. A large \(W/L\) can make the device area larger and parasitic capacitance larger, which can lead to a reduction in bandwidth. In the same way, a large R_D can contribute to voltage drop, limit voltage swing and impact transistor bias level; and the value of R_L chosen should be consistent with the load to be driven. The reported solution, therefore, is the mathematical optimum solution for the specified voltage-gain objective and not necessarily an optimum circuit design in general. In practice, verification of the operating point of the DC and the introduction of bandwidth, power, noise, linearity and output-swing constraints would be necessary.

On the circuit-design side, the numerical advantage of PSO shall be seen under the constraint of the single-objective model used. High gain was achieved by increasing the value of W/L, R_D, and R_L to their maximum limits. These changes can, however, bring into practice some practical compromise considerations such as bandwidth, parasitic capacitance, voltage headroom, output swing, device area, noise, and bias point stability. So, for the problem of maximum gain, PSO was used to find the best solution, but other circuit constraints and performance parameters have to be taken into account before the optimized values can be deemed as appropriate for practical implementation.

One of the drawbacks of the present study is that only voltage gain is taken as the objective in the optimization problem. This formulation allows the three algorithms to be compared easily, but a high gain solution may not be the "optimal" overall amplifier design. Raising W/L, R_D and R_L may enhance the voltage gain and result in trade-offs in bandwidth, power consumption, parasitic capacitance, voltage headroom, output swing, noise, linearity, and device area. Thus, the reported results must be interpreted in the light of the single objective of voltage gain maximization. Future work will expand on the present formulation to include constrained multi-objective optimization for voltage gain versus bandwidth, power consumption, noise, etc., and other practical circuit performance requirements.

VI. Conclusions

In this paper, a comparative analysis of Particle Swarm Optimization (PSO), Genetic Algorithm (GA) and Secretary Bird Optimization Algorithm (SBOA) has been done for the maximization of the voltage gain of a MOSFET amplifier. The optimization problem was formulated with the design vector $X = [W/L, R_D, R_L]$ and all three algorithms were compared in the same computational conditions, according to the best solution, statistical indicators, convergence behavior, execution time and physical interpretation of the optimized circuit parameters.

PSO had the best overall performance with the maximum voltage gain of 1378.125. It quickly came back to the values of $W/L = 100$, $R_D = 5000\,\Omega$, and $R_L = 5000\,\Omega$, which represent the maximum values of the design variables analyzed. The standard deviation is zero, meaning that the results of the 10 independent runs were exactly the same. It also indicates that when the identification of the high-gain region was given, the swarm converged strongly on the same boundary solution.

GA converged in a gradual, stepwise fashion and had the maximum voltage gain of 1176.766589 when it was implemented through its selection, crossover and mutation mechanisms. GA converged later and had a lower maximum gain than PSO, but had greater population diversity and explored a wider variety of non-boundary solutions. SBOA attained best-run gain of 1226.267555 compared to GA which has best-run gain of 1225.267555 but below PSO. The weak repeatability and early stagnation of the nearly flat curve for convergence and high value of the statistical variability is due to the selected parameter configuration.

In the practical aspect, the results show that the metaheuristic optimization can help the designer of MOSFET amplifiers to obtain an efficient exploration of the combination of the dimensions of transistors and resistances. PSO was the most effective and quickly converged method for optimizing the gain parameters for the adopted single objective formulation. However, the optimized variables will tend to move toward the maximum, suggesting that a maximum voltage gain is not always the best practical amplifier design. There could be tradeoffs with higher values of W/L, R_D, and R_L with respect to bandwidth, parasitic capacitance, voltage headroom, output swing, power consumption, noise, and device area.

Thus, the solutions reported are to be understood as best as possible given the objective voltage-gain, the search ranges and the model of the circuit. The future work is going to further consider the present study for constrained multi-objective optimization, where voltage gain will be included with bandwidth, power consumption, noise, linearity, output swing and device area. Other hybrid and adaptive variants of GA and SBOA can also be explored for better convergence and robustness. Finally, optimized solutions will be verified with circuit-level SPICE simulations as well as process-corner analyses, taking more realistic device modelling, biasing and manufacturing variations.

Acknowledgements

The authors would like to thank the Electrical Engineering Department, College of Engineering, Mustansiriyah University, for providing support during this work.

Conflict of interest:

The authors confirm that the publication of this article has conflict of interest.

References

[1] K. Rajwar, K. Deep, and S. Das, "An exhaustive review of the metaheuristic algorithms for search and optimization: taxonomy, applications, and open challenges," Artificial Intelligence Review, vol. 56, no. 11, pp. 13187–13257, 2023.

[2] B. Morales-Castañeda, D. Zaldívar, E. Cuevas, F. Fausto, and A. Rodríguez, "A better balance in metaheuristic algorithms: Does it exist?," Swarm and Evolutionary Computation, vol. 54, article 100671, 2020.

[3] M. Sarhani, A. El Afia, and R. Faizi, "Initialization of metaheuristics: comprehensive review, critical analysis, and future research directions," International Transactions in Operational Research, Wiley, 2023.

[4] T. Ivanovski, "Recent Advances in Metaheuristic Algorithms," Algorithms, vol. 19, no. 1, article 19, 2025.

[5] J. Kennedy and R. Eberhart, "Particle Swarm Optimization," Proceedings of the IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948, 1995.

[6] M. Clerc and J. Kennedy, "The particle swarm—explosion, stability, and convergence in a multidimensional complex space," IEEE Transactions on Evolutionary Computation, vol. 6, no. 1, pp. 58–73, 2002.

[7] J. H. Holland, Adaptation in Natural and Artificial Systems, MIT Press, 1992.

[8] D. E. Goldberg, Genetic Algorithms in Search, Optimization, and Machine Learning, Addison-Wesley, 1989.

[9] Y. Fu, D. Liu, J. Chen, and L. He, "Secretary bird optimization algorithm: a new metaheuristic for solving global optimization problems," Artificial Intelligence Review, vol. 57, article 123, 2024.

[10] S. Qin, J. Liu, X. Bai, and G. Hu, "A Multi-Strategy Improvement Secretary Bird Optimization Algorithm for Engineering Optimization Problems," Biomimetics, vol. 9, no. 8, article 478, 2024.

[11] X. Mai, Y. Zhong, and L. Li, "The Crossover Strategy Integrated Secretary Bird Optimization Algorithm and Its Application in Engineering Design Problems," Electronic Research Archive, vol. 33, no. 1, pp. 471–512, 2025.

[12] B. Crawford, F. Cisternas-Caneo, R. Soto, C. P. T. Mac-lean, J. Lara Arce, F. Solís-Piñones, G. Astorga, and G. Giachetti, "Binary Secretary Bird Optimization Algorithm for the Set Covering Problem," Mathematics, vol. 13, no. 15, article 2482, 2025.

[13] C. Chen, et al., "A Dual-Mechanism Enhanced Secretary Bird Optimization Algorithm," Biomimetics, vol. 10, no. 10, article 679, 2025.

[14] A. Lberni, M. A. Marktani, A. Ahaitouf, and A. Ahaitouf, "Efficient butterfly inspired optimization algorithm for analog circuits design," Microelectronics Journal, vol. 113, article 105078, 2021.

[15] H. Yang, S. Yang, D. Meng, C. Hu, C. Wu, B. Yang, P. Nie, Y. Si, and X. Su, "Optimization of Analog Circuit Parameters Using Bidirectional Long Short-Term Memory Coupled with an Enhanced Whale Optimization Algorithm," Mathematics, vol. 13, no. 1, article 121, 2025.

[16] A. C. Sanabria-Borbón, R. Soto-Aguilar, J. J. Estrada-López, D. Allaire, and E. Sánchez-Sinencio, "Gaussian-Process-Based Surrogate for Optimization-Aided and Process-Variations-Aware Analog Circuit Design," Electronics, vol. 9, no. 4, article 685, 2020.

[17] A. F. Budak, M. Gandara, W. Shi, D. Z. Pan, N. Sun, and B. Liu, "An Efficient Analog Circuit Sizing Method Based on Machine Learning Assisted Global Optimization," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 41, no. 5, pp. 1209–1221, 2021.

[18] H. Rajabi Moshtaghi, A. Toloie Eshlaghy, and M. R. Motadel, “A comprehensive review on meta-heuristic algorithms and their classification with novel approach,” Journal of Applied Research on Industrial Engineering, vol. 8, no. 1, pp. 63–89, 2021.

[19] Y. Lazo, et al., “Evolution and Trends of the Exploration–Exploitation Balance in Metaheuristic Algorithms,” Biomimetics, vol. 10, no. 8, article 517, 2025.

[20] D. O. Hasan, et al., “Real-World Applications of Metaheuristic Algorithms,” Diyala Journal of Engineering Sciences, 2025.