Optimization of amplifier circuits by using gradient boosted trees and probability annealing policy

Evelyn de Oliveira Lima¹, André Cavalcante¹, and João Viana da Fonseca Neto¹

¹Evelyn de O. Lima, André Cavalcante and João V. da Fonseca Neto, Departamento de Engenharia Elétrica, Universidade Federal do Maranhão, Brazil
e-mail: evelynlima02@gmail.com

Abstract—Automatic optimization of analog circuits befits a hard and costly optimization problem. This work focus on the optimization of analog amplifier circuits. Here, we propose two contributions to design automation methodologies based on machine learning. Firstly, we propose a probability annealing policy to boost early data collection and restrict electronic simulations later on in the optimization. Secondly, we employ multiple gradient boosted trees to predict design superiority in order reduces overfitting to learned designs. When compared to the state-of-the-art, our approach reduces the number of electronic simulations, the number of queries made to the machine learning module required to finish the optimization.

Index Terms—EDA, machine learning, gradient boosted trees.

I. INTRODUCTION

The design of mixed-signal integrated circuits generally requires human expertise to optimize project-specific analog modules. One step of this optimization consists of properly sizing circuit components to achieve the required specification. In this regard, it is important to notice that practical circuits may have a large number of parameters, and secondly, the quantities that define the specification may compete or demand different values for same circuit parameters. Therefore, analog circuit design befits a hard and costly optimization problem.

Design automation methodologies provide candidate values for the parameters of the circuit, either approximating or achieving the required specification [1–4]. In [2,3], candidate designs are generated by an evolutionary algorithm. In [4], deep reinforcement learning is employed to find an optimal solution in the search space of circuit parameters. Regardless of the optimization framework, all these methodologies leverage the utilization of electronic simulation frameworks for DC, AC and transient analysis, and also to simulate the effect of layout parasitics [5][6]. These simulations are carried interactively to provide realistic estimates of the performance of each candidate circuit and steer the optimization. The shortcoming of such strategy is that electronic simulations can have a high computational cost.

In order to minimize the number of simulations required to generate an acceptable design, [2][3] employ neural networks to learn which patterns of parameter values are likely to yield low and high-performance circuits. The neural network is then used to decide which candidate designs are passed on to the simulator. Specifically, only candidates which are likely to have high performance are electronic simulated. In [2] the machine learning module is based on support-vector machines [7][8]. Recently, [3] proposes to use deep neural networks.

Reference [4] also uses neural networks. However, in that case, the goal of the neural network is to explicitly output parameter values for the circuit along a trajectory in the parameter search space. It is important to notice that electronic simulations are carried out for all candidate designs in the course of the neural network training. Thus, the number of simulations in that approach can be very large. After training, however, optimal parameter values are found at the end of a trajectory which consists of few simulations.

This work focus on the optimization of amplifier circuits. Specifically, we employ evolutionary optimization and the machine learning based decision-making methodology more recently proposed in [3]. However, this work makes the following contributions:

1. We employ multiple gradient boosted trees [9] to evaluate design superiority instead of deep neural networks in order to minimize overfitting to circuit designs represented by low-dimensional real-valued vectors;

2. We propose a probability annealing policy to decide which candidate designs are passed on to simulation, thus boosting data collection for the machine learning module early on in the optimization, and restricting simulation as the evolution progresses and high-performance designs are generated.

II. METHODS

The architecture for circuit optimization based on evolution and machine learning decision making is illustrated in Fig. 1. The processing flow starts by the generation of an initial population of circuit designs. Each design is represented by a real-valued vector which contains the parameter values of the circuit. These values are randomly generated for the initial population. All of these random initial designs are electronically simulated and the specification quantities such as gain, bandwidth and phase characteristics are calculated. Secondly, the evolution module generates a new set candidate designs from the current population using evolutionary operations such as combination and mutations. Thirdly, each candidate is evaluated by the machine learning decision module.

Here, this module consists of multiple gradient boosted trees. In case the candidates are disqualified by the decision module, the evolution module generates a new set of designs.
The total cost is defined as the sum of the individual specification costs which are determined in executed for every pair of designs in the population.

Finally, the annealing policy is executed. This annealing policy consists of two parts. The first decreases the index of the reference design used for comparison against the candidate design. The second increases the probability threshold for simulation. In this way, a candidate design is simulated only if its probability of being superior than the reference design is higher than a certain threshold value. Notice that the maximum probability threshold for simulation in executed for every pair of designs is used to train a new set of gradient boosted trees for the decision module.

### A. Decision module

In the decision module, every new candidate design generated by the evolutionary process is “compared” to a reference design chosen from the current population according to the total critical cost proposed in [3]. The total cost is defined as the sum of the individual specification costs which are defined as

$$
\epsilon_i = \frac{|c_i - c^*_i|}{c_i + c^*_i},
$$

where $c^*_i$ represents the required value for $i$-th specification quantity, and $c_i$ represents the candidate design’s current true value for specification quantity $i$. In this way, Eq. (1) is a normalized error.

If the new candidate is likely to be superior than the reference design regarding each specification quantity then it is approved for simulation. Here, we employ multiple gradient boosted trees for measuring this likelihood. These decision trees are implemented by the XGBoost algorithm [10]. This architecture is illustrated in Fig. 2.

The input data of the architecture consists of a single vector formed by concatenating the parameters of the two designs to be compared. This vector is then shared across the set of trained gradient-boosted trees. Each tree has a single output. Each output models the probability of the event “design A is superior to design B regarding a specification quantity”. For instance, regarding specification quantity $i$, this probability is represented as $P(A > B | \text{specification } i)$. In this way, the architecture has as many trees as the number of quantities defining the specification.

The gradient boosted trees are trained in supervised fashion. The training dataset consists of two parts: input and label data. The input data consists of all vectors formed by concatenating the parameters of every possible pair of designs in the population. For each of these input vectors, label data consists of a N-dimensional binary vector, where N is the number of specification quantities. The $i$-th element of this vector is “1” if the cost $\epsilon_i$ of design A is less than that of design B. And it is “0” otherwise. In order to increase robustness and dataset size, a permutation of the order of designs in executed for every pair of designs in the population.

It is important to notice that gradient boosted trees regarding different specifications are trained fully separately from each other. Furthermore, every time a candidate design is approved for simulation, the current trained trees are destroyed, and new ones are trained with a dataset that includes the new simulated design.

A candidate design A is approved for simulation only if

$$
P(A > B | \text{specification } i) > \alpha, \forall i,
$$

where B represents a reference design current selected at index $j$ in the ranked population [3], and $\alpha$ the current probability threshold for simulation. In this way, a candidate design is simulated only if its probability of being superior than the reference design is higher than $\alpha$ for all specification quantities.

### B. Probability annealing policy

Here, the threshold $\alpha$ has an initial value represented by $\alpha_0$, and it is linearly annealed at every iteration step of the optimization process. This annealing policy is defined as

$$
\alpha \leftarrow \min(\beta, \alpha + \gamma),
$$

where $\gamma$ represents the additive annealing coefficient for the probability threshold value. Notice that the maximum threshold value is not greater than the limit $\beta$.

The initial value $\alpha_0$ for the probability threshold is set very low in order to allow more designs to be simulated at the beginning of the optimization process. This boost data collection for training the decision module early on in the optimization. On the other hand, the annealing policy will increase this value restricting electronic simulation as the optimization progresses and superior designs are generated.

The restriction is also intensified by decreasing the index of the reference design chosen in the ranked population.
Specifically, after every $n$ iterations of the optimization process, the reference index is decremented according to
\[ j \leftarrow j - z, \tag{4} \]
where $z$ is an integer. Notice that the minimum reference index is one, representing the best design in the population. Thus, as the optimization progresses, the offspring designs have to achieve higher standards to be electronically simulated.

### III. RESULTS

#### A. Experiments

The proposed methodology is evaluated for the optimization of two circuits using the schematic simulator NGSPICE [5]. The first circuit is a common source amplifier composed of a NMOS transistor and a resistor. The circuit has five parameters: width, length and multiplier of the transistor, the value of the resistor and the gate to source biasing voltage of the transistor. The specifications quantities consist of the minimum gain, the minimum bandwidth and maximum bias current of the transistor. In this way, the goal is to find a set of parameters that satisfies the specifications taking into consideration the gain to biasing current tradeoff of the transistor. The specific values for these quantities are shown in Table 1.

The second circuit consists of the two stage opamp used in [3]. This circuit consists seven parameters including the size of the transistors and a feedback capacitance. For this circuit, there are eight specification quantities which must be perfectly achieved. These specifications are the minimum gain, minimum bandwidth, the unit-gain frequency, phase margin, settling time, minimum common mode rejection ratio (CMRR), minimum power supply rejection ratio (PSRR), the systematic offset and the maximum bias current. The required values for these quantities are shown in Table 2.

In order to analyze the convergence behavior of the proposed methodology, 50 optimizations processes are carried out for each of the previous described circuits. It is important to notice that each process starts with a different initial population of designs. For all experiments, the configuration parameters of the proposed methodology were set as $N = 5$, $\alpha_0 = 0.1$, $\gamma = 10^{-3}$, $\beta = 0.35$, $n = 5$, $j_0 = 20$ and $z = 1$.

#### B. Measures of performance

In order to compare the performance of the proposed methodology to the state-of-art, the first measure analyzed is the total cost of a candidate design, which is calculated as the sum of individual specification costs given by Eq. 1. The evolution of the total specification cost for the fittest candidate in the design population over the iterations of the optimization process is shown in Fig. 3(a) and 3(b) for the common-source amplifier and two-stage opamp, respectively. In these figures, the green solid line represents the median value of the minimum specification cost for the proposed method “BagNet” proposed in [3]. The inferior and superior limits of the green shaded area represent the first and third quantile of the minimum cost distribution for the proposed method, respectively. Similarly, the orange shaded area limits represent the same statistical quantities for [3].

From Fig. 3(a), it is also possible to observe the same behaviors for the first and third statistical quantiles found for the common-source amplifier experiment. However, notice in Fig. 3(a) that the proposed method is the first to finish the majority of the optimization tasks.

For the two-stage opamp, i.e., Fig. 3(b), the proposed method achieves a zero median cost over 20 iterations of the optimization procedure. The method “BagNet” achieves the zero median cost approximately at 30 iterations. The superior limit of the green shaded area is smaller than that of the orange shaded area. This indicates that the proposed method generates fewer high costs than “BagNet”. For both methods, the median value becomes the inferior limit of the shaded areas after a number of iterations of the optimization. This indicates that the cost distribution is non-Gaussian and positively skewed. This is expected since finding design candidates with smaller costs becomes more difficult as the optimization progresses. For both methods, all 50 optimization procedures find a perfect solution approximately only after 50 iterations, as can be observed by the end of the lines representing the median. However, notice in Fig. 3(a) that the proposed method is the first to finish the majority of the optimization tasks.

For the two-stage opamp, i.e., Fig. 3(b), the proposed method achieves a zero median cost over 40 iterations of optimization. “BagNet” achieves the zero median cost over 50 iterations. Notice that for the two-stage opamp, several optimization procedures finished only after 80 iterations for both methodologies. This is expected since this circuit has more specification quantities to be met. It is important to notice that the maximum number of iterations is smaller for the proposed method in comparison to “BagNet”.

From Fig. 3(b), it is also possible to observe the same behaviors for the first and third statistical quantiles found for the common-source amplifier experiment. However, now, the represented statistical quantities exhibit closer values for both methods. Still, the proposed method produces less higher costs for both circuits since the green shaded area is smaller than orange area.

In order to analyze the computational cost of the proposed methodology, 50 optimizations processes are carried out for each of the previous described circuits. It is important to notice that each process starts with a different initial population of designs. For all experiments, the configuration parameters of the proposed methodology were set as $N = 5$, $\alpha_0 = 0.1$, $\gamma = 10^{-3}$, $\beta = 0.35$, $n = 5$, $j_0 = 20$ and $z = 1$.

#### III. RESULTS

### A. Experiments

The proposed methodology is evaluated for the optimization of two circuits using the schematic simulator NGSPICE [5]. The first circuit is a common source amplifier composed of a NMOS transistor and a resistor. The circuit has five parameters: width, length and multiplier of the transistor, the value of the resistor and the gate to source biasing voltage of the transistor. The specifications quantities consist of the minimum gain, the minimum bandwidth and maximum bias current of the transistor. In this way, the goal is to find a set of parameters that satisfies the specifications taking into consideration the gain to biasing current tradeoff of the transistor. The specific values for these quantities are shown in Table 1.

The second circuit consists of the two stage opamp used in [3]. This circuit consists seven parameters including the size of the transistors and a feedback capacitance. For this circuit, there are eight specification quantities which must be perfectly achieved. These specifications are the minimum gain, minimum bandwidth, the unit-gain frequency, phase margin, settling time, minimum common mode rejection ratio (CMRR), minimum power supply rejection ratio (PSRR), the systematic offset and the maximum bias current. The required values for these quantities are shown in Table 2.

In order to analyze the convergence behavior of the proposed methodology, 50 optimizations processes are carried out for each of the previous described circuits. It is important to notice that each process starts with a different initial population of designs. For all experiments, the configuration parameters of the proposed methodology were set as $N = 5$, $\alpha_0 = 0.1$, $\gamma = 10^{-3}$, $\beta = 0.35$, $n = 5$, $j_0 = 20$ and $z = 1$.

### B. Measures of performance

In order to compare the performance of the proposed methodology to the state-of-art, the first measure analyzed is the total cost of a candidate design, which is calculated as the sum of individual specification costs given by Eq. 1. The evolution of the total specification cost for the fittest candidate in the design population over the iterations of the optimization process is shown in Fig. 3(a) and 3(b) for the common-source amplifier and two-stage opamp, respectively. In these figures, the green solid line represents the median value of the minimum specification cost for the proposed method “BagNet” proposed in [3]. The inferior and superior limits of the green shaded area represent the first and third quantile of the minimum cost distribution for the proposed method, respectively. Similarly, the orange shaded area limits represent the same statistical quantities for [3].

From Fig. 3(a), it is also possible to observe the same behaviors for the first and third statistical quantiles found for the common-source amplifier experiment. However, notice in Fig. 3(a) that the proposed method is the first to finish the majority of the optimization tasks.

For the two-stage opamp, i.e., Fig. 3(b), the proposed method achieves a zero median cost over 40 iterations of optimization. “BagNet” achieves the zero median cost over 50 iterations. Notice that for the two-stage opamp, several optimization procedures finished only after 80 iterations for both methodologies. This is expected since this circuit has more specification quantities to be met. It is important to notice that the maximum number of iterations is smaller for the proposed method in comparison to “BagNet”.

From Fig. 3(b), it is also possible to observe the same behaviors for the first and third statistical quantiles found for the common-source amplifier experiment. However, now, the represented statistical quantities exhibit closer values for both methods. Still, the proposed method produces less higher costs for both circuits since the green shaded area is smaller than orange area.

In order to analyze the computational cost of the proposed methodology, 50 optimizations processes are carried out for each of the previous described circuits. It is important to notice that each process starts with a different initial population of designs. For all experiments, the configuration parameters of the proposed methodology were set as $N = 5$, $\alpha_0 = 0.1$, $\gamma = 10^{-3}$, $\beta = 0.35$, $n = 5$, $j_0 = 20$ and $z = 1$.

#### III. RESULTS

### A. Experiments

The proposed methodology is evaluated for the optimization of two circuits using the schematic simulator NGSPICE [5]. The first circuit is a common source amplifier composed of a NMOS transistor and a resistor. The circuit has five parameters: width, length and multiplier of the transistor, the value of the resistor and the gate to source biasing voltage of the transistor. The specifications quantities consist of the minimum gain, the minimum bandwidth and maximum bias current of the transistor. In this way, the goal is to find a set of parameters that satisfies the specifications taking into consideration the gain to biasing current tradeoff of the transistor. The specific values for these quantities are shown in Table 1.

The second circuit consists of the two stage opamp used in [3]. This circuit consists seven parameters including the size of the transistors and a feedback capacitance. For this circuit, there are eight specification quantities which must be perfectly achieved. These specifications are the minimum gain, minimum bandwidth, the unit-gain frequency, phase margin, settling time, minimum common mode rejection ratio (CMRR), minimum power supply rejection ratio (PSRR), the systematic offset and the maximum bias current. The required values for these quantities are shown in Table 2.

In order to analyze the convergence behavior of the proposed methodology, 50 optimizations processes are carried out for each of the previous described circuits. It is important to notice that each process starts with a different initial population of designs. For all experiments, the configuration parameters of the proposed methodology were set as $N = 5$, $\alpha_0 = 0.1$, $\gamma = 10^{-3}$, $\beta = 0.35$, $n = 5$, $j_0 = 20$ and $z = 1$.

### III. RESULTS

#### A. Experiments

The proposed methodology is evaluated for the optimization of two circuits using the schematic simulator NGSPICE [5]. The first circuit is a common source amplifier composed of a NMOS transistor and a resistor. The circuit has five parameters: width, length and multiplier of the transistor, the value of the resistor and the gate to source biasing voltage of the transistor. The specifications quantities consist of the minimum gain, the minimum bandwidth and maximum bias current of the transistor. In this way, the goal is to find a set of parameters that satisfies the specifications taking into consideration the gain to biasing current tradeoff of the transistor. The specific values for these quantities are shown in Table 1.

The second circuit consists of the two stage opamp used in [3]. This circuit consists seven parameters including the size of the transistors and a feedback capacitance. For this circuit, there are eight specification quantities which must be perfectly achieved. These specifications are the minimum gain, minimum bandwidth, the unit-gain frequency, phase margin, settling time, minimum common mode rejection ratio (CMRR), minimum power supply rejection ratio (PSRR), the systematic offset and the maximum bias current. The required values for these quantities are shown in Table 2.

In order to analyze the convergence behavior of the proposed methodology, 50 optimizations processes are carried out for each of the previous described circuits. It is important to notice that each process starts with a different initial population of designs. For all experiments, the configuration parameters of the proposed methodology were set as $N = 5$, $\alpha_0 = 0.1$, $\gamma = 10^{-3}$, $\beta = 0.35$, $n = 5$, $j_0 = 20$ and $z = 1$.
methodology, the following measures have also been registered for evaluation of the proposed method: the number of queries made to the decision module and the number of electronic simulations carried out in each optimization procedure. These measures are directly related to computational cost since they reflect the number of designs created and simulated during the optimization process. The statistical analysis of these measures is presented in Tables 3 and 4 for the common-source amplifier and the two-stage opamp circuits, respectively.

From Table 3, the proposed method queries its decision module in average less than a third of the number of queries made by “BagNet”. In this regard, the standard deviation of the number of queries made by the proposed method is also smaller in comparison. Notice that for the case of “BagNet”, which employs deep neural networks, querying the decision module in these optimization tasks can be very time consuming [3].

Also, from Table 3, the average number of electronic simulations made by “BagNet” is 145. The proposed method makes only 109 simulations in average. The standard deviation values are very close for both methods.

From Table 4, the average number of queries for the optimization of the two-stage opamp are significantly higher than for the common-source amplifier for both methods. This is also the case for the standard deviation values of the number of queries. The mean and standard deviation values of the number of simulations are is also higher for the two-stage opamp. This is expected since, as shown in Fig. 3, this circuit required a higher number of iterations of optimization. Notice that the difference between the average number of simulations between the proposed method and “BagNet” is approximately the same as observed for the common-source amplifier. However, the standard deviation value was slightly higher for the proposed method in this case.

These results suggest that the proposed methodology is able to search the parameter space as effectively as the state-of-the-art method. In this regard, both methodologies employ a machine learning based decision making to decide what candidate parameters should be simulated. Thus, it is important to compare the performance of the learning machines employed in each method for this decision task. For instance, does the learning machine correctly indicate if a design A is superior than a B in regard to a specification quantity i? On the other hand, its lowest decision accuracy is for the bandwidth specification. Interestingly, the recall and precision values are very close in each specification indicating that the deep neural network employed by “BagNet” does not suffer from class imbalance. For the proposed methodology, the decision module has the highest accuracy for the bandwidth specification and its lowest for the common mode rejection ratio. Recall and precision values are also very close indicating that the gradient boosted trees employed in the proposed method also do not suffer from class imbalance.

Table 3. Statistical analysis of measures related to computational cost of the optimization of the common-source amplifier.

<table>
<thead>
<tr>
<th>Measure</th>
<th>BagNet [3]</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average number of decision queries</td>
<td>3095</td>
<td>975</td>
</tr>
<tr>
<td>Standard deviation of the number of decision queries</td>
<td>1785</td>
<td>1219</td>
</tr>
<tr>
<td>Average number of electronic simulations</td>
<td>145</td>
<td>109</td>
</tr>
<tr>
<td>Standard deviation of the number of electronic simulations</td>
<td>49</td>
<td>48</td>
</tr>
</tbody>
</table>

Table 4. Statistical analysis of measures related to computational cost of the optimization of the two-stage opamp.

<table>
<thead>
<tr>
<th>Measure</th>
<th>BagNet [3]</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td>Average number of decision queries</td>
<td>82562</td>
<td>16395</td>
</tr>
<tr>
<td>Standard deviation of the number of decision queries</td>
<td>55796</td>
<td>15904</td>
</tr>
<tr>
<td>Average number of electronic simulations</td>
<td>277</td>
<td>236</td>
</tr>
<tr>
<td>Standard deviation of the number of electronic simulations</td>
<td>76</td>
<td>85</td>
</tr>
</tbody>
</table>
Notice that the proposed method decision module has a higher accuracy than that of the decision module employed in “BagNet” for all specification quantities. Specifically, the difference in accuracy for each specification quantity is larger than 10%. For instance, there is 14.23% difference in accuracy for the case of bandwidth.

These results suggest that gradient boosted trees can classify low dimensional inputs such as circuit designs represented by real vectors more effectively than deep neural networks. The lower results for the decision module based on deep neural network is not surprising since these learning machines are generally better suited for complex data such as images.

Furthermore, deep neural networks generally require large training datasets in order to avoid overfitting. In fact, the model proposed in [3] employ dropout to minimize overfitting. However, the training dataset for the circuit optimization task is still very small since it consists only of candidate circuit designs generated in the course of the few optimization iterations.

One solution to increase the dataset size would be to increase $n$, i.e., the number of candidate circuit designs that should be simulated and added to the current population at each optimization iteration. Potentially, however, this could significantly increase the total number of electronic simulations in a circuit optimization task, and therefore defeat the general goal of this research.

In regard of absolute time consumption, decision tree algorithms are generally order of magnitudes faster than deep neural networks for both training and inference. However, absolute time consumption depends on the current hardware technology. Thus, it is not straight-forward to analyze these methods in this regard.

It is important to notice the limits of the current research and the full spectrum of analog processing. Specifically, large signal applications generally involve far more complex circuitry and a greater number of specification quantities. Furthermore, the parameters in such analog structures can interact in a highly non-linear manner. These characteristics can drastically change the computational cost of finding an appropriate optimal solution.

### IV. CONCLUSION

This work presents a new methodology for automatic optimization of amplifier circuits based on gradient boosted trees and probability annealing policy. This methodology is evaluated and compared to a state-of-the-art method in the optimization of a common-source amplifier and a two-stage opamp.

The proposed methodology achieves a zero-specification cost in fewer optimization iterations than the state-of-the-art. This method also makes less than one third of queries to its decision module in order to decide which candidate designs should be electronically simulated.

Furthermore, the proposed method required in average fewer electronic simulations to achieve a solution. Finally, it is demonstrated that the proposed gradient boosted tree decision module has at least a 10% higher accuracy than the state-of-the-art deep-neural-network-based decision module in the task of identifying superior designs regarding the specification.

Since this research is restricted to the analysis of amplifier circuits, future works are going to include a larger spectrum of analog structures for full large signal applications.

### REFERENCES


### Table 5. Accuracy, recall and precision of the decision made during the optimization of the two-stage opamp.

<table>
<thead>
<tr>
<th>Measure</th>
<th>BagNet [3]</th>
<th>Proposed method</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>(accuracy</td>
<td>(accuracy</td>
</tr>
<tr>
<td>Gain</td>
<td>84.83</td>
<td>84.84</td>
</tr>
<tr>
<td>Bandwidth</td>
<td>82.35</td>
<td>82.36</td>
</tr>
<tr>
<td>Phase margin</td>
<td>84.00</td>
<td>84.00</td>
</tr>
<tr>
<td>Settling time</td>
<td>82.82</td>
<td>82.82</td>
</tr>
<tr>
<td>CMRR</td>
<td>84.86</td>
<td>84.86</td>
</tr>
<tr>
<td>PSRR</td>
<td>83.60</td>
<td>83.60</td>
</tr>
<tr>
<td>Systematic offset</td>
<td>85.20</td>
<td>85.20</td>
</tr>
<tr>
<td>Bias current</td>
<td>83.05</td>
<td>83.05</td>
</tr>
</tbody>
</table>