Linearizing the Transconductance of an OTA Through the Optimal Sizing by Applying NSGA-II

Luis Gerardo de la Fraga[1] and Esteban Tlelo-Cuautle[2],
[1] Cinvestav, Computer Science Department,
Av. IPN 2508, 07360 Mexico City, México
fraga@cs.cinvestav.mx
[2] INAOE, Department of Electronics
Luis Enrique Erro No. 1, Tonatzintla, Puebla, México
etlelo@inaoep.mx

***Abstract*—The operational transconductance amplifier (OTA) can be linearized by adding circuit elements as in the source degeneration technique. In this work, we linearize the transconductance of a CMOS OTA by just sizing the MOS transistors and by applying the multi-objective algorithm known as non-dominated sorting genetic algorithm (NSGA-II). The OTA under optimization has a cascode stage and complementary input, and three objectives are chosen: linear transconductance, gain and bandwidth. The sizing is performed by using a CMOS integrated circuit technology of 180 nm. The feasible solutions provided by NSGA-II are presented and compared by selecting 2 and 3 objectives, and at the end, we show the time response of the linearized OTA by implementing a multi-phase sinusoidal oscillator.**

I. Introduction

Analog integrated circuit design has the challenge of mitigating undesirable nonlinear effects that has been done mainly by adding circuitry to a selected topology. For example, in the case of the operational transconductance amplifier (OTA), its transfer function imposes to have a linear transconductance, and in the majority of cases, the linearization is accomplished by adding resistors to the sources of the MOSFETs in the input stage, known as source degeneration technique [1]. Nowadays, linearizing OTAs is a challenge that has different objectives to accomplish according to the application of the final topology, some examples of linearizing techniques are given in [2], [3], [4]. In those topologies, the OTA is linearized by adding circuitry. However, in some cases the linearization can be accomplished by performing an optimal sizing of the MOSFETs. In this case, one can apply metaheuristics as highlighted in [5]. In this manner, we are applying the multi-objective algorithm known as non-dominated sorting genetic algorithm (NSGA-II [6]) to linearize the transconductance of an OTA that is designed with standard CMOS integrated circuit technology of 180 nm.

Our goal is sizing a OTA considering three objectives: gain, bandwidth, and a linear transconductance response. As we are solving a multi-objective problem, we are going to get a set of solutions instead a single one. We have a set of solution because there exist conflicts among the objectives values: it is clear the trade-off between gain and bandwidth, with better gain, lesser bandwidth, and vice versa. In this case we could select the best solutions according to the third objective value. Also, the circuit simulator Ngspice [7] is used to evaluate each

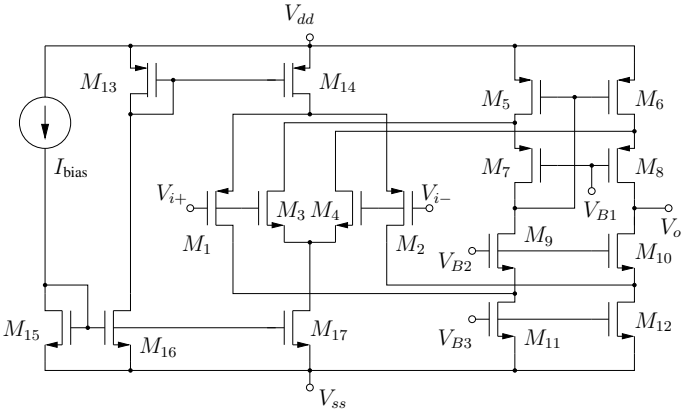


Fig. 1. The cascode amplifier with complementary input taken from [8]

circuit during the search by NSGA-II. NSGA-II is genetic algorithm, it start with random initial solutions which are improved by several iterations of the algorithm. We need to remember that NSGA-II obtains the best solutions, the called Pareto front, very well distributed according to the objectives values.

This remaining of the paper is organized as follows: In Sec. II the optimization problem for the OTA design and its solution is described. In Sec. III the results are presented, and finally in Sec. IV some conclusions and possible future work are given.

II. The Sizing Optimization Problem

Among all the available OTA topologies, our case of study is the one taken from [8], and shown in Fig. 1.

The sizing optimization problem to solve for the feasible sizes of the MOSFETs is defines as:

Maximize gain($\mathbf{x}$), bandwidth($\mathbf{x}$), and linear response($\mathbf{x}$), subject to: all transistors except M_{11} and M_{12} (we have 15 constraints here) must work in saturation mode, phase margin must be greater than $60°$, and amplifier transconductance must be greater than 0.30 mA/V and lesser than 0.33 mA/V.

Then the problem has a total of 18 constraints.

The design variables are encoded into the vector $\mathbf{x} \in \mathbb{R}^5$, is $\mathbf{x} = [v_1, v_2, v_3, v_4, v_5]$. Also, each variable is subject to

978-1-5386-5153-7/18/$31.00

TABLE I. DESCRIPTION OF VARIABLES TO OPTIMIZE FOR THE CASCODE AMPLIFIER. ALL TRANSISTORS HAVE THE SAME CHANNEL LENGTH OF 1 μm.

Variable	Width of transistors
v_1	M1, M2
v_2	M3, M4
v_3	M5, M6
v_4	M7, M8
v_5	M9, M10, M11, M12
20 μm	M13, M14
15 μm	M15, M16, M17

the bounds: $v_i \in [1, 100]$ for $i \in \{1, 2, 3, 4, 5\}$. This is, all transistors lengths are equal to 1 μm, and widths have the meaning described in Tab. I.

Then, the problem to optimize has 5 variables, 3 objectives, and 18 constraints.

The problem is solved by applying NSGA-II [6], which links the free and online available circuit simulator Ngspice [7]. Two aspects that we need to take care of them herein: how the NSGA-II constraint violation index is computed, and the other is related to programming details, i.e. how NSGA-II links with the circuit simulator.

For NSGA-II it is necessary to calculate a value to indicate if a solution (an individual for NSGA-II) fulfill some or all constraints. We do not have in our problem a continuous function to handle each constraint, therefore we decide to handle binary constraint values: let c_i, for $i \in \{1, 2, \ldots, 18$ the constraint value associated to each constraint. Value of c_i is assigned as

$$c_i = \begin{cases} -1, & \text{if constraint } i \text{ is invalid,} \\ 1, & \text{if constraint } i \text{ is valid.} \end{cases} \quad (1)$$

And the constraint violations value is calculated as the sum of all c_i. In (1) a constraint is invalid if, for example, a transistor is outside of its saturation mode.

We have tried in the past using perl [9] or python [10], [11] high level languages to process the output file produced by Ngspice. It certainly is easier for the programmer but also consumes a lot of CPU resources, and increases the execution time. We have found that it it possible to obtain the fastest execution times if we leave to ngpice to process all calculations and to print a single value, and then this single value is read by a C coded function inside NSGA-II. As the NSGA-II source code is freely available in C [1], we decided to use the fork(), execve(), and waitpid() C scheme to launch Ngspice. A little more details will be given in the next subsection.

A. Objectives evaluation

Three runs of the Ngspice simulation must be performed to evaluate an individual: first to calculate the DC operation point and to check if transistors are working in saturation mode, second to analyze the frequency response of the analyzed amplifier and then to calculate its gain and bandwidth, the third to calculate its transconductance and the same time its linearity.

Three simulations must be performed but only a single file is written per individual evaluation. This single file is named as `sizes.lib` and contains the individual parameters in `.param` Ngspice directives. Then, the three associated script files for each simulation remains unchanged along all the simulation execution. Inside of each of these three scripts there is a line `.include ./sizes.lib` to read the sizes values.

[1] In http://www.iitk.ac.in/kangal/codes.shtml

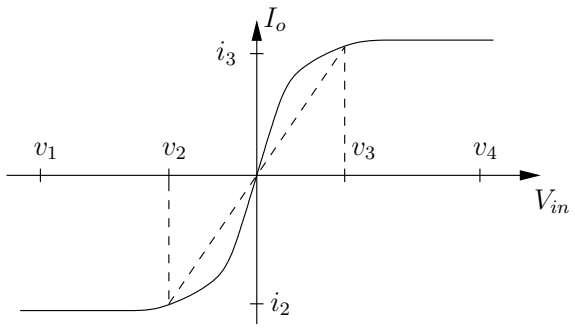


Fig. 2. Error value calculation. This is one example of the current output versus voltage input signal

These conditions are fixed for all the amplifier simulation: polarization voltage of $\pm$ 1.65 V, polarization current Ibias = 30 μV, VB1 = 0.55 V, VB2 = -0.75 V, and VB3 = -0.95V (see Fig. 1).

To check the operation region of each transistor using the first Ngspice script, directly from Ngspice it is calculated using the following lines:

```
.op
.control
run
define constraint(vth, vgs, vds) ( ((vgs > vth) and \
   (vds > (vgs - vth) )) ? 1 : -1 )
let res01 = constraint( @m.Xota1.mp1[vth], \
   @m.Xota1.mp1[vgs], @m.Xota1.mp1[vds] )
print res01
.endc
```

In this example we verify if the MOS transistor M1 is working or not in the saturation region. In this way, from the NSGA-II C code it is only necessary to check the label `res01` and read its subsequent value (1 if M1 is saturated, -1 otherwise).

To measure the DC gain and bandwidth a capacitor load of 10 pF is used.

To measure the value for the third objective, the linearity in the amplifier response, the following idea is used. An error value is calculated as following: to the integral of the response signal from v_2 to v_3 (see Fig. 2) it is substracted the area of two triangles, one formed with points $(\text{offset}, 0)$, $(v_2, 0)$, and (v_2, i_2), and the other triangle with points $(\text{offset}, 0)$, $(v_3, 0)$, and (v_3, i_3). If the error is equal to zero, then the amplifier transconductance is totally linear.

Finally, we also check, as a constraint, if the response signal is symmetrical. In this case, the absolute minimum value of the output signal must be greater than 90% of the absolute maximum signal. This constraint is checked with the following lines:

```
* Symmetry constraint calculation
meas dc iini find i(VL) at=-2.0
meas dc ifin find i(VL) at=2.0
```

```
let v=max(abs(iini),abs(ifin))
let u=min(abs(iini),abs(ifin))
let symmetry = u ge 0.9*v
print symmetry
```

And as before, within the NGSA-II C source code, it is only necessary to check the label `symmetry` and its subsequent value.

The third objective value, the OTA lineality, is measured with the following lines:

```
let amp = ifin - iini
let i1 = iini + amp*0.05
let i2 = iini + amp*0.95
meas dc val1 when i(VL)=i1
meas dc val2 when i(VL)=i2
meas dc offset when i(VL)=0.0
meas dc sum1 INTEG i(VL) from=val1 to=offset
meas dc sum2 INTEG i(VL) from=offset to=val2
let error=sum2-sum1+i1*(offset-val1)/2-i2*(val2-offset)/2
print error
```

III. Results

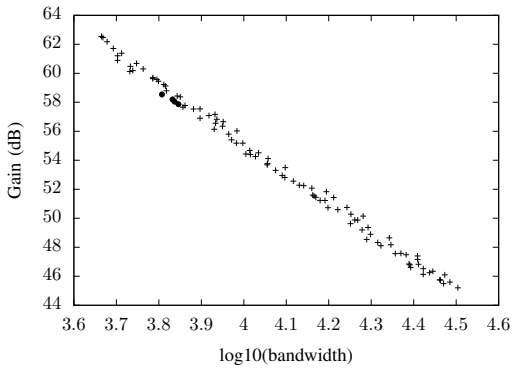
The parameters used during the NSGA-II executions were set to: 100 individuals, 400 generations, crossover probability equal to 0.7, mutation probability equal to 0.25, $\eta_c = 15$, and $\eta_m = 20$.

With this set of parameters, one simulation takes around 145 minutes, on a Mac Pro with 3.2 GHz Quad-Core Intel Xeon, and 8GB of memory.

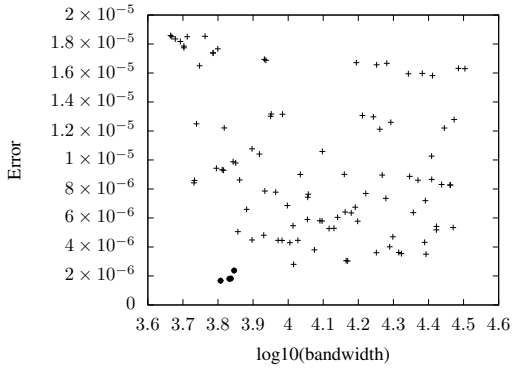
The graphs of the Pareto fronts are shown in Fig. 3. The results of the optimization are listed in Tab. II. Four individuals with the lesser error value are given in Tab. II. We compare these results with the ones obtained by using two objectives: maximizing amplifier's gain and bandwidth. Thus, four individuals with similar gain and bandwidth obtained to the first optimization with three objectives are given in Tab. II. The errors in the linear transconductance are small, but still the optimization with three objectives obtains individuals with two times lesser error, compared with the optimization with two objectives. And a very interesting result is obtained if we compare the estimated areas of the amplifiers in Tab. II (by summing the multiplications of length by widths for all transistors). The optimization with three objectives obtains significantly smaller design solutions. The graphs of the current output vs. the input voltage for the individuals '1's listed in Tab. II, are shown in Fig. 4.

Also with the individuals marked with '1' in Tab. II, the corresponding amplifiers were tested in a multi-phase sinusoidal oscillator application, which has been already introduced in [12]. The output of both oscillators with two outputs are given in Fig. 5. The total harmonic distortion (THD) for both oscillators are less than 1%, THD = 0.99%, THD = 0.49%, for the oscillator signals obtained with the amplifiers optimized with 3 and 2 objectives, respectively. This confirms that the linearized OTA can be used to enhance the response of the oscillator in [12].

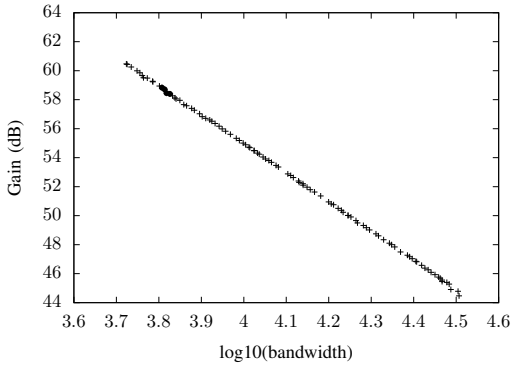
The C code to evaluate one individual is given publicly in http://cs.cinvestav.mx/~fraga/linearizedOTA.tar.gz



(a) Optimization with 3 objectives



(b) Optimization with 3 objectives



(c) Optimization with 2 objectives

Fig. 3. Pareto front of both optimizations, (a) and (b) with the three objectives, and (c) with two objectives. Solutions in Tab. II are marked with •.

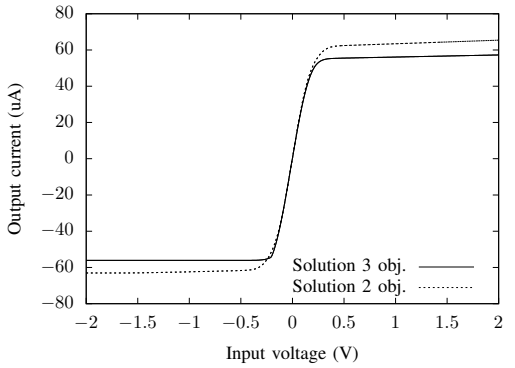
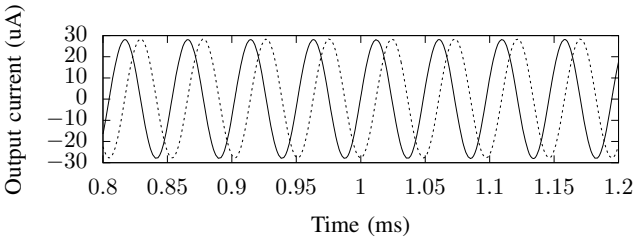


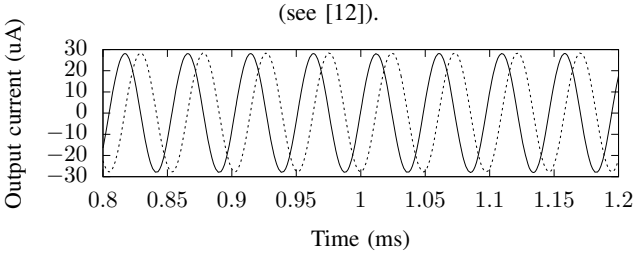
Fig. 4. Response of both solutions '1' from Tab. II. It is possible to observe that the solution using three objectives has a more linear behavior.

TABLE II. RESULTS OF THE OPTIMIZATION USING TWO AND THREE OBJECTIVES

Optimization with three objectives										
Sol.	v_1	v_2	v_3	v_4	v_5	Estimated area	Ao (dB)	BW (Hz)	Error	gm (mA/V)
1	23	4	25	18	8	257	58	6418	1.68e-6	0.324226
2	32	3	42	15	8	301	58	6796	1.80e-6	0.329508
3	31	3	50	15	8	315	58	6866	1.82e-6	0.327777
4	24	4	29	23	9	281	58	7015	2.37e-6	0.328333
Optimization with two objectives										
Sol.	v_1	v_2	v_3	v_4	v_5	Estimated area	Ao (dB)	BW (Hz)	Error	gm (mA/V)
1	48	2	61	37	9	417	59	6417	4.22e-6	0.329212
2	48	2	64	37	9	423	59	6510	4.22e-6	0.329215
3	45	2	32	52	10	387	58	6577	4.35e-6	0.323907
4	48	2	61	86	10	519	58	6699	4.49e-6	0.328472



a) Output corresponding to solution 1 of the optimization with three objectives from Tab. II. $f_{OSC} = 20375$ Hz, C=2.58 nF R=3084 Ω (see [12]).



b) Output corresponding to solution 1 of the optimization with two objectives from Tab. II. $f_{OSC} = 20509$ Hz, C=2.62 nF R=3037 Ω (see [12]).

Fig. 5. Outputs of the multiphase sinusoidal oscillator.

IV. CONCLUSIONS

We have presented the linearization of the transconductance of a CMOS OTA by applying NSGA-II and by considering three objectives: maximize gain and bandwidth, and minimize a error which linearize the transconductance. The feasible solutions were presented in the Pareto Fronts and feasible solutions were selected to test the behavior of the OTA in an application like a multi-phase sinusoidal oscillator, generating the response of two phases. Te results confirm that in some cases, it is quite enough to perform a sizing process to linearize the transconductance of the OTA, as in this work in which we applied NSGA-II.

ACKNOWLEDGMENT

This work has been partially supported by CONACyT/Mexico under grant 237991.

REFERENCES

[1] P. Monsurro, S. Pennisi, G. Scotti, and A. Trifiletti. Linearization technique for source-degenerated cmos differential transconductors. *IEEE Transactions on Circuits and Systems II: Express Briefs*, 54(10):848–852, 2007.

[2] Sheng-Yu Peng, Yu-Hsien Lee, Tzu-Yun Wang, Hui-Chun Huang, Min-Rui Lai, Chiang-Hsi Lee, and Li-Han Liu. A power-efficient reconfigurable ota-c filter for low-frequency biomedical applications. *IEEE Transactions on Circuits and Systems I: Regular Papers*, 65(2):543–555, 2018.

[3] Stanislaw Szczepanski, Bogdan Pankiewicz, Slawomir Koziel, and Marek Wojcikowski. Multiple output differential ota with linearizing bulk-driven active-error feedback loop for continuous-time filter applications. *International Journal of Circuit Theory and Applications*, 43(11):1671–1686, 2015.

[4] Mohamed Mobarak, Marvin Onabajo, Jose Silva-Martinez, and Edgar Sanchez-Sinencio. Attenuation-predistortion linearization of cmos otas with digital correction of process variations in ota-c filter applications. *IEEE Journal of Solid-State Circuits*, 45(2):351–367, 2010.

[5] E. Martens and G. Gielsen. Classification of analog synthesis tools based on their architecture selection mechanims. *Integr. VLSI J.*, 41(2):238–252, 2008.

[6] K. Deb, A. Pratap, and S. Agarwal. A fast and elitist multi-objective genetic algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation*, 6(2):182–197, 2002.

[7] Ngspice is a mixed-level/mixed-signal circuit simulator based on open source software packages, Last date checked March 15th, 2018. http://ngspice.sourceforge.net/.

[8] T. Oliveira Weber and W.A.M. Van Noije. Analog circuit synthesis performing fast Pareto frontier exploration and analysis through 3D graphs. *Analog Integr Circ Sig Process*, 73:861–871, 2012.

[9] L.G. de la Fraga and E. Tlelo-Cuautle. Optimizing operational amplifiers by metaheuristics and considering tolerance analysis. In *2015 16th Latin-American Test Symposium (LATS)*, Puerto Vallarta, Mexico, 25-27 March 2015.

[10] L.G. de la Fraga and E. Tlelo-Cuautle. Optimizing an amplifier by a many-objective algorithm based on r2 indicator. In *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 265–268, Lisbon, Portugal, 24-27 May 2015.

[11] de la Fraga L.G., I. Guerra-Gomez, and E. Tlelo-Cuautle. On the selection of solutions in multiobjective analog circuit design. In O. Schtze, L. Trujillo, P. Legrand, and Y. Maldonado, editors, *NEO 2015*, volume 663 of *Studies in Computational Intelligence*. Springer, 2017.

[12] E. Tlelo-Cuautle, L.G. de la Fraga, K. Phanrattanachai, and K. Pitaksuttayaprot. CDCTA and OTA realizations of a multi-phase sinusoidal oscillator. *IETE Technical Review*, 32(6):497–504, 2015.