

On the Linearization of MOSFET Capacitors

Mohammad Danaie^{1,2}, Hamed Aminzadeh² and Sasan Naseh²

¹ Computer and Communications Research Center, Ferdowsi University of Mashhad

² Electrical Engineering Group, Engineering Department, Ferdowsi University, Mashhad, Iran

Email: mo_da90@stu-mail.um.ac.ir

Abstract — In this paper, a heuristic methodology for design of highly linear MOSFET capacitors (MOSCAPs) has been presented. For a certain amount of accessible chip area, the proposed algorithm intends to find the structure which has the least C-V variation. It uses a modified version of the genetic programming to optimize the capacitor topology and transistors' dimensions. To test the performance of the algorithm, it was utilized for obtaining a highly linear 1 pF capacitor, i.e. with less than 3% variation of capacitance versus 1 V variation across the MOSCAP structure terminals, in a commercial 0.18 μm standard digital CMOS technology. This level of linearity can be otherwise achieved only with MIM capacitors, which are not available in digital CMOS processes.

I. INTRODUCTION

The CMOS technology is a digital-oriented technology, i.e. has been developed with little or no analog applications in mind. Being able to integrate the digital and analog portions of a system on a single chip provides many advantages one of which is lower fabrication costs. Hence it is very desirable to integrate the analog part of a mixed-signal system in standard digital CMOS technologies.

The digital circuitry in mixed-mode CMOS ICs usually requires only a single poly-Si layer for the gates; however in order to achieve linear capacitors needed in many analog applications, a second poly-Si layer or extra metal layers intended specifically for the purpose of implementation of metal-insulator-metal (MIM) capacitors are introduced into the process, resulting in a significant increase in the fabrication costs.

Although the available metal layers in the standard CMOS technology can be utilized for implementation of MIM capacitors, due to the high thickness of the oxide between those metal layers, the achieved capacitance per area would be small. Also, a parasitic capacitance will exist between the bottom plate of these MIM capacitors and the substrate.

One of the solutions to avoid these disadvantages is to use MOSFET gate as a capacitor. As the gate oxide is thin, MOS capacitors (MOSCAP) have large capacitance per unit area. However, the main obstacle for utilizing MOSFET capacitors in most of analog applications is that their value is highly dependent on their terminal voltages. In other words, MOSCAPs show a strong nonlinear behavior. In most cases,

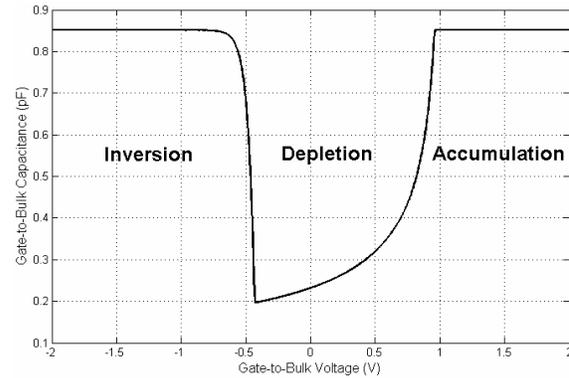


Fig. 1: C-V characteristic of a p-channel MOSFET ($W=L=10\mu\text{m}$)

this nonlinear behavior degrades the performance metrics of analog circuits. The performance degradation in many applications depends on the level of the capacitor nonlinearity. In a MOS transistor, this nonlinear behavior is caused by different charge distributions in the accumulation, depletion and inversion regions of operation. As it is shown in Fig. 1, the gate-to-bulk capacitance in depletion region is very different from those of the inversion and accumulation regions.

Several attempts have been made to linearize MOSCAPs. Serial compensation and parallel compensation depletion-mode techniques (SCDM and PCDM, respectively) are two commonly used solutions found in literature [1]-[3]. Simulations in a 0.18 μm standard digital CMOS technology for terminal voltages between [0V, 1V] show about 34% and 8.6% variations in the capacitance value for PCDM and SCDM techniques, respectively; the capacitor increases for larger voltages. While SCDM is much more linear than PCDM, the nonlinearity of this structure is still not tolerable for some applications. This amount of nonlinearity is almost independent of the capacitance value.

In this paper, a new method for linearizing the MOSCAP capacitance is proposed. Using a novel genetic programming approach, a CAD has been developed which finds the optimum solution with regard to both linearity and the available area for implementation on chip. In the next section a mathematical description of the procedure is given. Section

3 explains the optimization procedure. In section 4 the fitness function is defined. The simulation results are presented in section 5 and finally, section 6 is the conclusion.

II. MATHEMATICAL DESCRIPTION OF THE PROCEDURE

In this paper a global algorithm has been proposed for achieving precise capacitor structures from the nonlinear Gate-bulk capacitances of MOS transistors; however as it was said before the algorithm is global and by changing the performance index of the optimization process, the designer could use the proposed algorithm for obtaining capacitors, whose C-V characteristics obey a predefined pattern. Answering the following two simple questions forms the main idea in this paper: (1) "Can connection of a couple of nonlinear capacitors result in a more linear capacitor?" and (2) "For a certain chip area, what is the structure whose voltage-capacitance variation is the least?"

In order to answer these questions, the capacitive structures have to be modeled mathematically. A graph is used to model the structure. Fig. 2 shows example of a graph having two different types of nonlinear capacitors whose value is dependent on the voltage across its nodes. Since these two types of capacitors are nonlinear their polarity is also important; assuming that the on-chip areas of all the nonlinear capacitors are equal, we will have four different types of capacitors known by C_1 to C_4 .

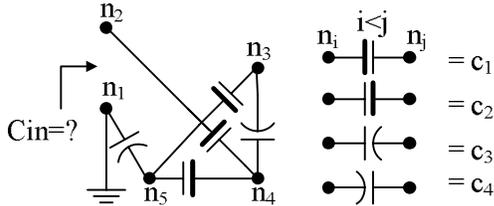


Fig. 2: An example of a capacitive structure.

Equation (1) defines a matrix for circuit shown in Fig. 2. This matrix is referred to as INDV matrix (standing for individual). Assuming that all the four capacitors in Fig. 2 have the same size, or on-chip areas, the INDV matrix represents the compound capacitor structure's graph thoroughly. In case the sizes of the components were different, another matrix would be needed to store those values.

$$INDV = \begin{bmatrix} 0 & 0 & 0 & 0 & c3 \\ 0 & 0 & 0 & c1 & 0 \\ 0 & 0 & 0 & c4 & c2 \\ 0 & c1 & c4 & 0 & c1 \\ c3 & 0 & c2 & c1 & 0 \end{bmatrix} \quad (1)$$

Using this graph enables the designer to identify and deal with the structures mathematically. Fig. 3 shows the three basic capacitors used in this optimization procedure. Using these components, various structures could be formed. The

length and width of the gate of the PMOS transistors were chosen equal to $L_0=20\mu\text{m}$ and $W_0=10\mu\text{m}$, and in order to change the aspect ratio, their m (number of fingers) was chosen as one of the optimization variables. Another matrix called $INDV_M$ was used to store these aspect ratios. (In switched capacitor circuits the NMOS transistors could be replaced by switches turning on at ideal instants.)

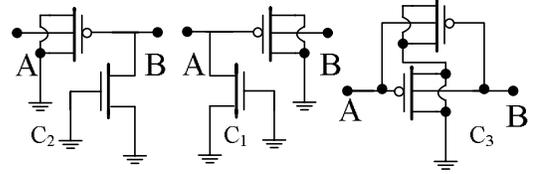


Fig. 3: Basic building blocks.

For each capacitive structure which has n nodes and three basic components, there exists a matrix defined by Equation (2):

$$INDV = \begin{pmatrix} a_{11} & \cdots & a_{1n} \\ \vdots & \ddots & \vdots \\ a_{n1} & \cdots & a_{nn} \end{pmatrix} \quad (2)$$

$$\forall i, j \rightarrow a_{ij} = a_{ji}, a_{jj} = 0$$

If no connection exists between nodes n_i and n_j , then $a_{ij}=0$, otherwise $a_{ij}=1, 2$ or 3 depending on which one of C_1, C_2 or C_3 is used. Fig. 3 shows different possible connections between two nodes. Since changing the name of the nodes does not change the structure of the circuit, node n_1 is always assumed as the input node and node n_2 is assumed to be connected to the ground.

III. OPTIMIZATION PROCEDURE

The flowchart of the optimization process is shown in Fig. 4. Except for some differences which will be discussed later, the algorithm used for obtaining linear capacitance structures is very much similar to genetic programming. In the proposed algorithm a group of individuals are generated as the first generation. The number of these individuals is a user-defined parameter. In the case study presented in section V, each generation contained 80 individuals. The individuals of the first generation are $INDV(j)$ and $INDV_M(j)$ matrixes of the structures that contain only two nodes, in which j denotes the number of individual. Second generation contains circuits with two and three nodes. The k^{th} generation contains circuits with less than $k+1$ nodes if $k < N_MAX$ (where N_MAX is the maximum allowed number of nodes), and circuits with less than N_MAX nodes if k is greater than or equal to N_MAX .

The first generation is chosen in a complete random basis. Then Mutation and Crossover are performed on the first generation and the results are fed to a circuit evaluator which determines how good the results are [5]. It chooses next generation according to the fitness of individuals in a random basis so that every individual has the chance of being chosen, with superior individuals having a higher chance. The fitness function used in this case will be discussed in section IV.

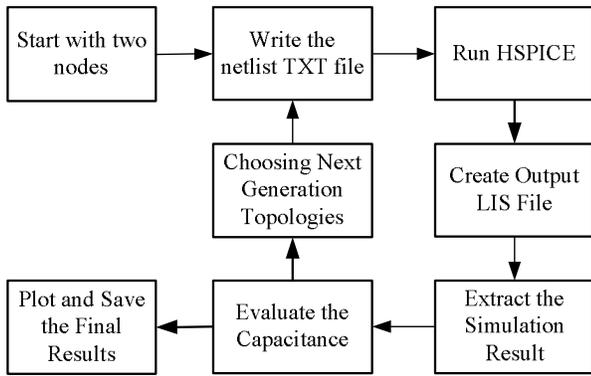


Fig. 4: Optimization flowchart.

$$\begin{bmatrix} 0 & a_{12} & a_{13} & a_{14} \\ a_{21} & 0 & 0 & a_{24} \\ a_{31} & 0 & 0 & a_{34} \\ a_{41} & a_{42} & a_{43} & 0 \end{bmatrix} \rightarrow \begin{bmatrix} 0 & a_{12} & a_{13} & a_{14} \\ a_{21} & 0 & x & a_{24} \\ a_{31} & x & 0 & a_{34} \\ a_{41} & a_{42} & a_{43} & 0 \end{bmatrix}$$

$$x \in \{1, 2, 3\} \quad \forall i, j \quad a_{ij} = a_{ji} \in \{0, 1, 2, 3\}$$

Fig. 7: An example of connecting two nodes randomly together.

3. Change the value of m randomly. (The parameters m are elements of INDV_M matrix).
4. Probabilistically add a new node and connect it randomly to two of the existing nodes. Fig. 8 shows an example of this.

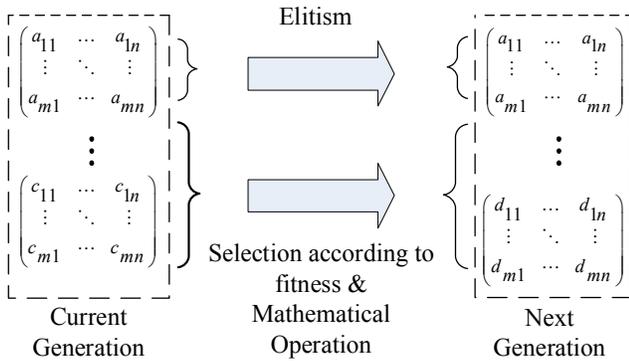


Fig. 5: Optimization algorithm.

$$\begin{pmatrix} 0 & 1 & 2 & 0 \\ 1 & 0 & 1 & 3 \\ 2 & 1 & 0 & 0 \\ 0 & 3 & 0 & 0 \end{pmatrix} + \begin{pmatrix} y \\ 0 \\ x \\ 0 \end{pmatrix} \Rightarrow \begin{pmatrix} 0 & 1 & 2 & 0 & y \\ 1 & 0 & 1 & 3 & 0 \\ 2 & 1 & 0 & 1 & x \\ 0 & 3 & 0 & 0 & 0 \\ y & 0 & x & 0 & 0 \end{pmatrix}$$

$$x, y \in \{1, 2, 3\}$$

Fig. 8: An example of adding a new node.

5. Probabilistically add a new node and connect it randomly to two of the existing nodes, and if there is a connection between those two nodes omit it. It could be shown that using the above mutation rules, the circuit graph remains fully connected and all possible structures have the chance to be created.

A fixed number of the best individuals called ELITES are also moved to the next generation. This causes the next generation to improve compared to the current generation [4]. This process is shown in Fig. 5.

Ordinary crossover and mutation cannot be used in this case since it might lead to unacceptable answers. Fig. 6 shows some of the unacceptable answers. The line between each two nodes represents a capacitor. A new methodology for generating new individuals has been proposed which ensures no unacceptable circuit will be generated. The basics of the method are as follows:

1. Connect two nodes randomly, choosing a random connection type. This connection can be achieved, using one of the three proposed building blocks C1, C2 or C3 shown in Fig. 3. Fig. 7 shows an example of this process.
2. Change the type of connection between two nodes randomly from 1 to 2, from 3 to 1, etc.

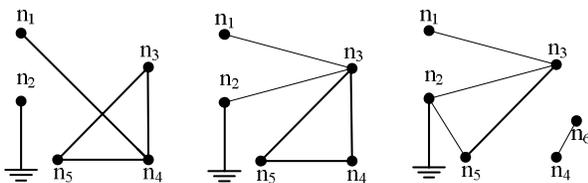


Fig. 6: Examples of unacceptable structures.

IV. FITNESS MEASUREMENT

As it was said in the previous section, the individuals are chosen probabilistically for the next generation according to their fitness. This is the principle which helps the genetic programming algorithms to converge to optimum solutions [5]. Usually fitness is the function which has to be minimized, and having a smaller value of fitness means that the specified individual is better than the other ones. In this paper two constraints are important. The first is the relative percentage of linearity achieved, and the second is the chip area needed to implement these capacitors. Assuming that the maximum allowed area for the capacitor is $area_{max}$ and the actual area is $area_{act}$, the following fitness function was defined in this case:

$$\text{Fitness}(\cdot) = \text{Max}\{CU, CD\} / C_{DESIRED} + \gamma AF$$

$$C_U = |C_{Max} - C_{DESIRED}|$$

$$C_D = |C_{Min} - C_{DESIRED}|$$

$$AF = \exp(-area_{max} / area_{act})$$
(3)

where γ is a user defined parameter which determines the weight put on the second term. The first term indicates the relative capacitance deviation from the desired value. The required area can be easily calculated according to the W and L of the transistors. A function is written which estimates the area using INDV_M and INDIV matrixes for this purpose. It is obvious that a structure with a higher linearity and smaller area will have a smaller (i.e. better) fitness value.

V. CASE STUDY

The circuit simulations were performed with HSPICE using BSIM3 models of a commercial 0.18 μm CMOS technology. Fig. 9 shows the result obtained for linearization of a 1pF capacitance with a maximum area of $4500\mu\text{m}^2$ and $\gamma = 0.01$. It shows 1.7% variation for a voltage range of 0 to 1 volt. This linearity is better than that obtained using SCDM at the expense of more chip area. For instance, the 1 pF capacitor is implemented in an estimated area of $4500 \mu\text{m}^2$ using the proposed method while it was $2300 \mu\text{m}^2$ and $650 \mu\text{m}^2$ for SCDM and PCDM cases, respectively. However for large values of γ the algorithm will converge to SCDM structure. Fig. 10 is the output of the graphical section of the developed CAD which draws the circuit's graph, and Fig. 11 shows the actual circuit topology. The resulting INDV_M matrix is:

$$\text{INDV_M} = \begin{bmatrix} 0 & 0 & 5 & 0 & 3 \\ 0 & 0 & 0 & 2 & 0 \\ 5 & 0 & 0 & 2 & 2 \\ 0 & 2 & 2 & 0 & 0 \\ 3 & 0 & 2 & 0 & 0 \end{bmatrix} \quad (2)$$

The initial W/L ratio is $L=20 \mu\text{m}$ $W=10 \mu\text{m}$ for the building block MOSCAPS and $W=2 \mu\text{m}$ $L=2 \mu\text{m}$ for the parasitic transistors. The total simulation time for a maximum node number of $n=15$, using a Pentium III computer with 512MB RAM was 1502 seconds. The algorithm converges in 15 generations in which each generation has 80 individuals. Fig. 12 shows the best obtained fitness in each generation.

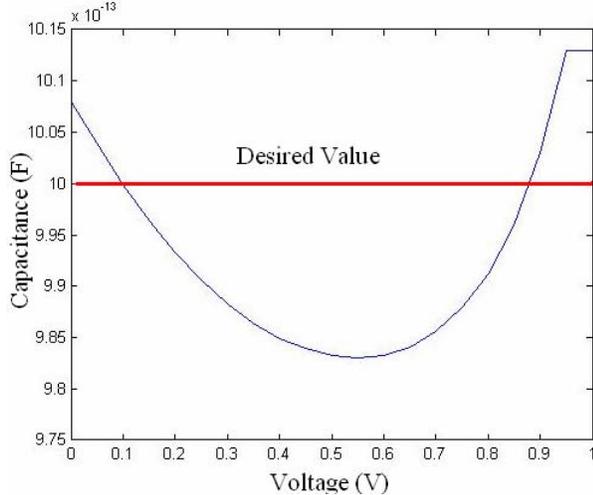


Fig. 9: Output capacity per voltage

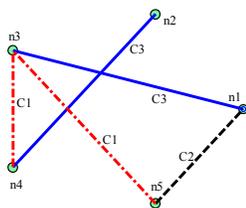


Fig. 10: The resulted circuit's graph.

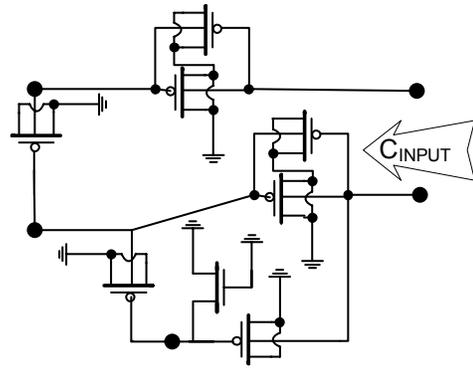


Fig. 11: The resulted linear capacitor circuit.

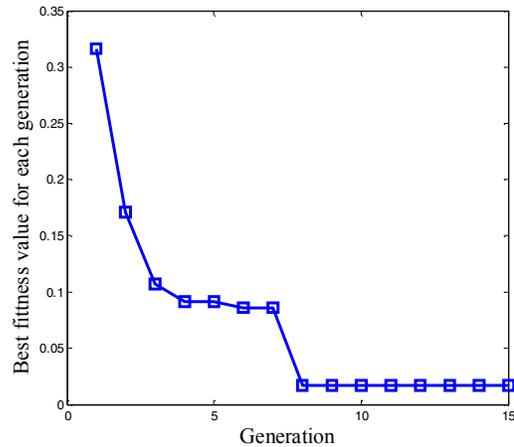


Fig. 12: Best fitness in each generation.

VI. CONCLUSION

In this paper a CAD tool for designing high linearity MOSCAPs was proposed. For a certain amount of available chip area, the proposed algorithm finds the best possible structure which has the minimum C-V variation. The mentioned methodology was used to obtain a highly linear capacitor in a commercial 0.18 μm standard digital CMOS technology. The obtained results show the ability of this method in finding the optimum desired MOSCAP structures.

REFERENCES

- [1] T. Tille, J. Sauerbrey, M. Mauthe, and D. Schmitt-Landsiedel, "Design of Low-Voltage MOSFET-Only Sigma-Delta Modulators in Standard Digital CMOS Technology," IEEE Transactions on Circuits and Systems, VOL. 51, NO. 1 January 2004.
- [2] H. Yoshizawa, Y. Huang, P.F. Ferguson, and G.C. Temes, "MOSFET-Only Switched-Capacitor Circuits in digital CMOS Technology," IEEE Journal of Solid-State Circuits, VOL. 34, NO. 6, June 1999.
- [3] K. Leelavattananon, C. Toumazou, and J.B. Hughes, "Balanced Compensation for Highly Linear MOSFET Gate Capacitor Branch," Electronics Letters Online, NO. 19990992, 6 July 1999.
- [4] K. John H. Holland, "Genetic Algorithms - Computer programs that evolve in ways that resemble natural selection can solve complex problems even their creators do not fully understand," Scientific American, pp. 66-72, July 1992.
- [5] J. Koza, "Genetic Programming - on the programming of computers by means of natural selection", MIT Press, 1992.