
Azam Beg\textsuperscript{1}, Rashad Ramzan\textsuperscript{2}, and Amr Elchouemi\textsuperscript{3}
\textsuperscript{1}College of Information Technology, United Arab Emirates University, Al-Ain, United Arab Emirates
\textsuperscript{2}College of Engineering, United Arab Emirates University, Al-Ain, United Arab Emirates
\textsuperscript{3}Walden University, USA

Abstract—In this paper, we propose a technique for concurrent optimization of CMOS logic gates for power-and-noise-margin and energy-and-noise-margin. The role of progressive sizing for performance enhancement of different gates has been expanded to cover other figures of merit, such as reliability, power, and energy. By using the examples of three- and four-input logic gates, we have demonstrated how multiple, yet conflicting design goals can be achieved. For example, one of our high-performance gates exhibited power savings of more than 30\% while reducing the gate area by 39\%. An important step of balancing the rise- and fall-times of output was also incorporated into the optimization setup. Our proposed methodology is scalable and can be used for optimizing larger logic blocks.

Keywords: CMOS technology, logic circuits, combinational circuits, genetic algorithms, multi-objective optimization, transistor sizing

1. Introduction

With the explosive growth of mobile electronic devices in the recent years, the integrated circuit designers must pay special attention to power and energy efficiency, rather than just the performance (clock frequency, \(f_{\text{clk}}\)). Although continuous downsizing of transistors has helped the chip manufacturers maintain Moore’s Law, the dynamic and the leakage powers have not been able to keep up with the pace of transistor-scaling. Yet another important challenge for the modern devices is their reliable operation.

This paper for the first time proposes the use of genetic algorithms (GAs) for multi-objective optimization of CMOS logic gates based on 22 nm technology. The technique aims to help a circuit designer find transistor sizes that meet a logic gate’s design criteria for power/energy dissipation, static noise margin (SNM) (one way of representing reliability), and balanced rise- and fall-times (\(\tau_{\text{rise}}\) and \(\tau_{\text{fall}}\)), etc.

This paper is organized as follows: Section 2 covers basic information about the logic gates’ metrics and optimization. Section 3 covers the previous work related to this paper, and Section 4 explains our experimental setup. Results and analysis are presented in Section 5. Finally, Section 6 concludes the paper.

2. Preliminaries

2.1 The Metrics for CMOS Logic Gates

The power dissipation of a CMOS gate constitutes two parts, dynamic and static. The static power consumption can be represented with a simple equation: \(P_{\text{static}} = I_{\text{off}}V_{\text{DD}}\), where \(I_{\text{off}}\) is the non-ideal and undesired current flowing between the supply rails (\(V_{\text{DD}}\) and ground) even when the gate is not switching. With a high \(f_{\text{clk}}\), most of the dynamic power consumption (\(P_D\)) is due to the charging and discharging of the load capacitance \((C_L)\): \(\alpha_{0\rightarrow1}f_{\text{clk}}C_LV_{\text{DD}}^2\), where \(\alpha_{0\rightarrow1}\) is the probability of 0\(\rightarrow\)1 transition. The second component of \(P_D\) occurs while both the N- and P-transistors are on. For example, in a matched inverter, the peak current flows when \(v_{\text{in}} = v_{\text{out}} = V_{\text{DD}}/2\). As power and delay can be adjusted separately, it is common to consider them together by using the power-delay product (PDP or energy) (at a given \(f_{\text{clk}}\)) as a figure-of-merit (FOM) [1].

An obvious method for reducing \(P_D\) is to lower either or both the \(f_{\text{clk}}\) and the \(V_{\text{DD}}\). However, lowering power consumption while ignoring \(f_{\text{clk}}\) may not be desirable. This is because reduced frequency increases the time taken to complete a task. Therefore, for an energy-efficient design, it is not enough to minimize just the power; what matters more is the saving in energy per cycle of operation [1].

The ability of a gate to withstand noise has been traditionally defined as the noise margin (NM). The allowed voltage range for logic-low is called the low NM and the range for the logic-high is high NM. The lower of the two NMs is referred to as SNM. Properly balancing the transistor sizes in the N- and P-stacks of a CMOS gate can help improve the SNM [2]. In this paper, SNM is used as a measure of reliability.

2.2 Circuit Design and Optimization

In modern nanometeric-device-based circuits, the design FOMs include not only the power, energy, and delay, but also the reliability. Such characterization generally involves determining proper sizes for the transistors. In most cases, a designer has to find the sizes that balance conflicting objectives, such as power, delay, reliability, etc [3]. Therefore, we can consider the sizing optimization to be multi-objective
in nature; the GAs are just the right tool to handle such optimization [4], [5].

The GAs are based on the principle of natural selection that is derived from the concept of biological evolution. A GA begins the optimization process using a randomly generated population of parents; the GA then uses the principles of random crossover and mutation to find the next generation of 'fitter' children (or offsprings). This is in contrast to a classical algorithm that would select the next points in the search space deterministically. A GA relies on the search of a wide but finite solution space. A GA's input-set includes the fitness function (also called cost function), while the output is considered the cost of fitness. A candidate solution to a GA problem is a chromosome. The chromosomes are made up of genes; the genes represent different design parameters as strings of 0's and 1's [4], [5].

The process of multi-objective optimization using GAs often yields multiple optimal solutions which can be quite different from each other, but they are all deemed 'as good as possible.' Dealing with multiple design objectives means that a designer has a choice to pick the best optimal solution based on his own intuition and/or experience.

3. Related Work

Power consumption and reliability are listed among the 'design difficult challenges' in the current International Technology Roadmap for Semiconductors [6]. The main techniques for power reduction, in the past, have included: transistor optimization [7], near-VTH or sub-VTH region operation [8], use of MOS current mode logic [9], dynamic voltage scaling [10], using thin-body fully-depleted SOI [7], auto-substrate biasing [11], and transistor stacking [12].

The transistor sizing methods have been revisited, from time to time, with the purpose of addressing not just the power consumption but also energy dissipation, performance, and/or reliability [3], [13]. The traditional sizing techniques used analytical models [14], [15], [16], [17] or circuit simulations, while the non-conventional methods have included the use of constraint-based optimization [18], geometric programming [19] and single-objective GAs [20], [21]. Use of set-oriented numerical method for multi-objective sizing of 90 nm gates operating at nominal $V_{DD}$ using GAs was proposed in [22]. Sub-VTH operation of a few basic 65 nm gates (INV, NAND2, NOR2) was considered in [23] and Minority-3 function in [24]. The modern nodes such as 22 nm are much more prone to failure due to variations than the earlier ones, and have to be investigated anew.

One of the methods of sizing the transistors in a CMOS gate is progressive sizing [1] and it entails widening the transistors that are closer to the supply rails. By doing so, both the resistances and capacitances of the transistor stacks decrease, and hence the delay of large fan-in gates. To the best of our knowledge, we are the first ones to investigate multi-objective GAs for progressively sizing the 22 nm transistors for logic gates, with the purpose of reducing not just the delay and power/energy consumption but also for improving the SNM (or reliability).

4. Experimental Setup

In this paper, we investigate multi-objective GA-based optimization of progressively-sized transistors for four different logic gates: NAND3, NAND4, NOR3, and NOR4 (see Figs. 1 and 2). The lower and the upper bounds for all transistor widths ($w_i$’s) were set as $4\,\text{nm} \leq w_i \leq 1000\,\text{nm}$, and the lengths ($w_i$’s) were fixed at $L_{\text{min}} = 22\,\text{nm}$. In order to ensure that the parallel transistors have the same widths during optimization, we used the equality constraints. Progressive sizing of series transistors was enforced using the inequality constraints. The two constraint sets are shown in Table 1.

We set up experiments to minimize these two-objective (fitness) functions: (1) power and the difference between $\tau_{\text{rise}}$ and $\tau_{\text{fall}}$; and (2) PDP and the difference between $\tau_{\text{rise}}$ and $\tau_{\text{fall}}$. We aimed to match $\tau_{\text{rise}}$ and $\tau_{\text{fall}}$ as practical as possible, in order to improve the SNM and hence the operational reliability. The optimization process results in a Pareto-front that represents multiple optimum solutions, from which a designer can select the one that meets yet another criterion, such as the SNM or the area.

In our optimization problem, the genes represent the widths ($w_i$) of the transistors of units-under-test (UUTs). The chromosomes made from the genes are utilized in two separate SPICE circuit files created automatically by a Perl script. One circuit is for measuring the current (and power),

![Fig. 1: Gate schematics: (a) NAND3, and (b) NAND4.](image)

Table 1: GA constraints for transistor $M_i$’s ($w_i$ = width of $M_i$, as shown in Figs. 1 and 2)

<table>
<thead>
<tr>
<th>Gate Type</th>
<th>Equalities</th>
<th>Inequalities</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAND3</td>
<td>$w_0 = w_1 = w_2$</td>
<td>$w_3 \leq w_4 \leq w_5$</td>
</tr>
<tr>
<td>NAND4</td>
<td>$w_0 = w_1 = w_2 = w_3$</td>
<td>$w_4 \leq w_5 \leq w_6 \leq w_7$</td>
</tr>
<tr>
<td>NOR3</td>
<td>$w_3 = w_4 = w_5$</td>
<td>$w_6 \geq w_1 \geq w_2$</td>
</tr>
<tr>
<td>NOR4</td>
<td>$w_4 = w_5 = w_6 = w_7$</td>
<td>$w_9 \geq w_1 \geq w_3$</td>
</tr>
</tbody>
</table>
delay, and $\tau_{\text{rise}}/\tau_{\text{fall}}$, and the other circuit for SNM. It is worth mentioning that the difference between $\tau_{\text{rise}}$ and $\tau_{\text{fall}}$ is highly dependent on the input vectors. Ref. [25] includes an example of set of input vectors for a large fan-in gate. As it would not be practical to perfectly match the two $\tau$'s for all $(C)$ different vectors, we defined an FOM called average error ($\epsilon_{\text{avg}}$):

$$\epsilon_{\text{avg}} = \left| \frac{1}{C} \sum_{i=1}^{C} \left( \frac{V_{\text{DD}}}{2} - v_{\text{out}_i} \right) \right|,$$

(1)

where $v_{\text{out}_i}$ is the gate output voltage in response to the $i$th input vector.

For all gate types, we measured total power, delay, $\epsilon_{\text{avg}}$, and SNM. We also recorded the gate area which was defined as $L_{\text{min}} \times \sum_{i=0}^{M-1} w_i$; here $M =$ number of gate transistors, and $L_{\text{min}} = 22 \text{nm}$. The gates were built from 22 nm PTM HP v2.1 (high-k/metal gate and stress effect) MOS transistor models [26] and BSIM4v4.7 level 54 [27]. The circuits operated at 2 GHz. As we focused only on the high performance operation of the gates, we set $V_{\text{DD}}$ to be the same as the nominal voltage, i.e., 0.8V [26]. In other words, near- or the sub-$V_{\text{TH}}$ modes were not considered.

Our experimental setup can be represented by the flowchart of Fig. 3. The setup is made up of a set of custom Matlab [28] and Perl scripts running under Mac OS X ver. 10.10.5. The GA population size is set to $15 \times N$. The crossover fraction has a value of 0.8. The generation count is set to 500 [28]. The starting point of the design process is a randomly generated set of chromosomes used to create SPICE circuits. The circuits are simulated using NGSpice [?] and the log files are saved. We then parse the log files for current, delay, SNM, etc., in order to evaluate the fitness function. We continue to evolve offsprings for creation and simulation of circuits, until the preset fitness criteria (power-and-$|\tau_{\text{rise}}-\tau_{\text{fall}}|$, or PDP-and-$|\tau_{\text{rise}}-\tau_{\text{fall}}|$) are met or when maximum number of generations have been created. Typically, it took 4–5 compute-hours to optimize a gate. From the list of the optimal solutions, we only considered the ones that had SNM $\geq 25\%$ of $V_{\text{DD}}$ and $\epsilon_{\text{avg}} \leq 10\%$ of $V_{\text{DD}}$ (whenever feasible). Lastly, we sorted the solution-sets on the gate areas.

5. Results and Analysis

We performed multi-objective optimization of four gates, viz., NAND3, NAND4, NOR3, and NOR4, using the setup of Fig. 3. For each gate type, we ran two sets of experiments, one for finding the Pareto-optimals for the power-and-$\epsilon_{\text{avg}}$ balance, and the other for the PDP-and-$\epsilon_{\text{avg}}$. The optimization yielded many combinations of transistor sizes as listed in Table 2.

In the following paragraphs, we have included the data and analysis only for NAND4 (power and $\epsilon_{\text{avg}}$, and SNM) and NOR4 (PDP, $\epsilon_{\text{avg}}$, and SNM), for the sake of brevity (see Tables 3 and 4).

The power-$\epsilon_{\text{avg}}$-SNM plots for the Pareto-optimal NAND4s are shown in Fig. 4, while Table 3 lists the $w$’s and Table 2: Number of optimal points in the Pareto-fronts.

<table>
<thead>
<tr>
<th>Gate Type</th>
<th>Minimizing and $\epsilon_{\text{avg}}$</th>
<th>Minimizing PDP and $\epsilon_{\text{avg}}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>NAND3</td>
<td>32</td>
<td>32</td>
</tr>
<tr>
<td>NAND4</td>
<td>42</td>
<td>27</td>
</tr>
<tr>
<td>NOR3</td>
<td>32</td>
<td>13</td>
</tr>
<tr>
<td>NOR4</td>
<td>42</td>
<td>42</td>
</tr>
</tbody>
</table>
the corresponding data. The progressively-sized gate (PSG) with the smallest area (row 2) dissipates 33% less power than the uniformly/conventionally-sized gate (USG) with an area saving of 39%. Much larger power savings are possible but they incur higher cost in terms of delay and $\epsilon_{avg}$. The SNM is also marginally higher (1%) but it costs more in terms of $\epsilon_{avg}$. For the PSGs, the power-consumption also shows an uptrend as the SNM increases.

The PDP-$\epsilon_{avg}$-SNM data for the Pareto-optimal NAND4s (plots and data are omitted) reveals that the PSG gate with similar area as a USG consumes less PDP than the USG. PSGs’ low gains in PDP are attributed to higher delays. Although we observed very low values of $\epsilon_{avg}$, none of the PSGs were able to meet the SNM target (of 25%); this warrants another look at the experimental data.

The power-$\epsilon_{avg}$-SNM data for the Pareto-optimal NOR4s demonstrates that the NOR4-PSGs have 40% power-advantage over the USG, while the SNM is 6% higher. (Due to the paucity of space, the plots and data are not included). These gains come with the added advantage of lesser gate area, i.e., 21%. The power and $\epsilon_{avg}$ show an inverse relationship (just like the PSG-NANDs did).

The PDP-$\epsilon_{avg}$-SNM plots for the Pareto-optimal NOR4s are in Fig. 5, while Table 4 provides the $w_i$’s and the related data. The smallest-area PSG has 28% PDP-saving and 47% smaller area than the USG; the SNMs are quite comparable, specifically PSG's SNM is 0.7% lower than USG. To gain the SNM-advantage, the gate area has to be increased, which slightly increases the PDP but it still is much lower than the USG.

6. Conclusions and Future Directions

Although progressive gate sizing scheme was originally intended to improve performance, the current work reveals many other advantages of the scheme. Using multi-objective GAs, we demonstrated that we were also able to improve power-and-reliability and energy-and-reliability; yet another benefit was the reduction in gate area. The scalability of our scheme and the encouraging results prompt us to continue our investigation with larger CMOS circuits. Looking into the effects of manufacturing-related variations, for example, in the channel length and the threshold voltage, is another venue for further research.

7. Acknowledgment

This work is partially supported by ADEC Award for Research Excellence (A^2RE) 2015.

References

Table 3: Comparing uniformly-sized NAND4 gate with a few of 'power-\(\epsilon_{\text{avg}}\)'-optimum NAND4 gates (\(w_i = \text{width of } M_i\), as shown in Figs. 1 and 2).

<table>
<thead>
<tr>
<th>Sizing Scheme</th>
<th>(w_0 - w_3) [nm]</th>
<th>(w_4 - w_7) [nm]</th>
<th>Gate Area [(\text{nm}^2)]</th>
<th>Power [W]</th>
<th>SNM [V]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform</td>
<td>88</td>
<td>176, 176, 176, 176</td>
<td>23232</td>
<td>3.10E-07</td>
<td>0.198</td>
</tr>
<tr>
<td>Progressive</td>
<td>44</td>
<td>66, 90, 124, 190</td>
<td>14212</td>
<td>2.07E-07</td>
<td>0.201</td>
</tr>
<tr>
<td>Progressive</td>
<td>44</td>
<td>66, 103, 126, 209</td>
<td>14960</td>
<td>2.15E-07</td>
<td>0.202</td>
</tr>
<tr>
<td>Progressive</td>
<td>44</td>
<td>72, 103, 164, 237</td>
<td>16544</td>
<td>2.33E-07</td>
<td>0.206</td>
</tr>
</tbody>
</table>

Table 4: Comparing uniformly-sized NOR4 gates with a few of 'PDP-\(\epsilon_{\text{avg}}\)'-optimum NOR4 gates (\(w_i = \text{width of } M_i\), as shown in Figs. 1 and 2).

<table>
<thead>
<tr>
<th>Sizing Scheme</th>
<th>(w_0 - w_3) [nm]</th>
<th>(w_4 - w_7) [nm]</th>
<th>Gate Area [(\text{nm}^2)]</th>
<th>PDP [W-sec]</th>
<th>SNM [V]</th>
</tr>
</thead>
<tbody>
<tr>
<td>Uniform</td>
<td>352, 352, 352, 352</td>
<td>44</td>
<td>34848</td>
<td>5.50E-17</td>
<td>0.192</td>
</tr>
<tr>
<td>Progressive</td>
<td>182, 163, 163, 158</td>
<td>45</td>
<td>18612</td>
<td>8.11E-18</td>
<td>0.190</td>
</tr>
<tr>
<td>Progressive</td>
<td>213, 190, 189, 184</td>
<td>45</td>
<td>21032</td>
<td>8.39E-18</td>
<td>0.195</td>
</tr>
<tr>
<td>Progressive</td>
<td>223, 202, 202, 197</td>
<td>45</td>
<td>22088</td>
<td>8.51E-18</td>
<td>0.198</td>
</tr>
</tbody>
</table>


