A Nonlinear Programming Based Power Optimization Methodology for Gate Sizing and Voltage Selection

V. Mahalingam and N. Ranganathan
Department of Computer Science & Engineering
Nanomaterial and Nanomanufacturing Research Center
University of South Florida, Tampa, FL - 33620
{mvenkata, ranganat}@cse.usf.edu

Abstract

In this paper, we investigate the problem of power optimization in CMOS circuits using gate sizing and voltage selection for a given clock period specification. Several solutions have been proposed for power optimization during gate sizing and voltage selection. Since the problem formulation is nonlinear in nature, nonlinear programming (NLP) based solutions will yield better accuracy, however, convergence is difficult for large circuits. On the other hand, heuristic solutions will result in faster but less accurate solutions. In this work, we propose a new algorithm for gate sizing and voltage selection based on NLP for power optimization. The algorithm uses gate level heuristics for delay assignment which disassociates the delays of all the paths to the individual gate level, and each gate is then separately optimized for power with its delay constraint. Since the optimization is done at the individual gate level, NLP converges quickly while maintaining accuracy. Experimental results are presented for ISCAS benchmarks which clearly illustrate the efficacy of the proposed solution.

1. Introduction

In deep submicron technologies, minimizing power has become the dominant concern as it directly affects battery life, reliability and cooling costs [1]. There is a wide range of techniques for dynamic power minimization such as varying voltage, frequency and load capacitance. Minimizing these parameters could lead to increase in delay. However, with careful optimization, power consumption can be reduced without penalizing performance by only increasing the delay of gates in the non-critical paths. Here, we investigate the problems of gate sizing and voltage selection in this context.

Several works [2-6, 9, 12, and 17] have appeared in the literature on power optimization through gate/transistor sizing and voltage selection. A sensitivity-based engine is used to select the transistor to size by the authors in [3-5, 9, and 12]. Static timing analysis is required in every step of the optimization, for identifying the most sensitive transistors in the critical paths. These approaches lack accuracy due to the use of Elmore delay models and also have false path problems due to static timing analysis [6]. Since the formulation of the circuit optimization problem is strictly nonlinear, nonlinear programming (NLP) techniques need to be investigated for better solutions.

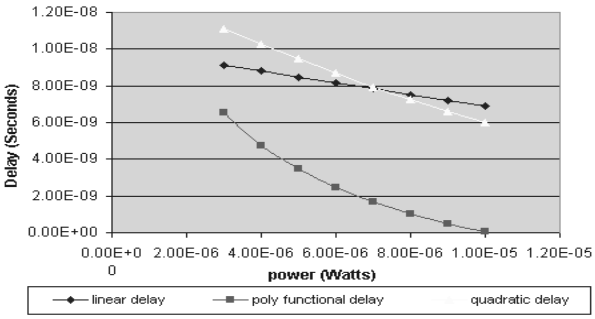

Figure 1. Power Vs Delay - NAND gate

Let us look at a NLP simulation of the 2 input NAND gate. Figure 1 shows the results of a delay constrained power minimization for NAND gate. Linear, quadratic and poly-functional fitting of delay as a function of the gate size is shown. (Poly-functional models are linear combination of polynomial, logarithmic and exponential functions). It can be seen from the graph that, for the same delay constraint, optimization with accurate delay models produce better optimized circuits. For example, with a delay constraint of 7ns, an accurate delay model minimized

Proceedings of the IEEE Computer Society Annual Symposium on VLSI
New Frontiers in VLSI Design
0-7695-2365-X/05 $20.00 © 2005 **IEEE**

the power to 3µw, compared to 8µw for delay modeled linearly. The work in [6] also acknowledges this fact and optimizes circuit using a nonlinear formulation; however, the solution does not converge well for large circuits due to solving the problem as a whole.

Hence, in this work, we optimize the problem using a NLP approach at individual gate level after a heuristic delay assignment. The heuristic based delay assignment effectively disassociates the optimization problem to the level of the individual gates. The problem is then solved efficiently at the gate level using NLP formulation. Thus, the main contribution of this work is the combining of NLP technique with the heuristic based delay assignment procedure which provides for both accuracy and speed. The combination captures the efficiency of NLP while getting rid of the convergence issue by formulating at the gate level. Experimental results on ISCAS benchmarks indicate that the combination yields a better solution and also converges faster. The remainder of this paper is organized as follows. Section 2 explains some of the previous works in circuit optimization. The problem formulation and the proposed approach are given in sections 3 and 4 respectively. Results are presented in section 5, followed by conclusions in Section 6.

2. Related Work

Circuit optimization techniques can be broadly classified into three categories; 1) Dynamic tuning, 2) Static tuning and 3) heuristic based tuning. Figure 2 shows a taxonomy diagram of related works.

Dynamic tuning [6, 17] uses time domain simulation combined with adjoint sensitivity computation. Jiffy-tune [6] from IBM is a popular dynamic tuning tool. Jiffy-tune uses a fast circuit simulator (SPECS) and a gradient based nonlinear optimizer (LANCELOT) for circuit optimization. Dynamic tuning is highly accurate but requires the specification of input signals and is slow due to full circuit simulation needed in every iteration. Hence, these are mostly limited to circuits of a few thousand transistors.

Static tuning [3, 4, 9, 12 and 16], on the other hand, uses static timing analysis in every step of the optimization. All critical paths are enumerated and the most sensitive transistors in these paths are sized for delay and/or power optimization. Recently, similar optimization procedures have also been proposed for dual voltage optimization. However the inaccuracy of the delay models employed by static tuners often limits their utility in real time [6].

The third category is the heuristic based circuit tuning. Ranganathan et.al. in [5], proposed an economic theory based solution to the gate sizing problem. The gates in the path bid for delay values and the optimization problem is solved using Nash equilibrium [5]. Pant et al, in [2], present a heuristic based tuning method for optimizing total power. The authors here use a heuristic, similar to path balancing, to assign maximum possible delays to all the gates in the circuit. Binary search is then used to find optimized values for V_{dd}, V_{th} and gate sizes.

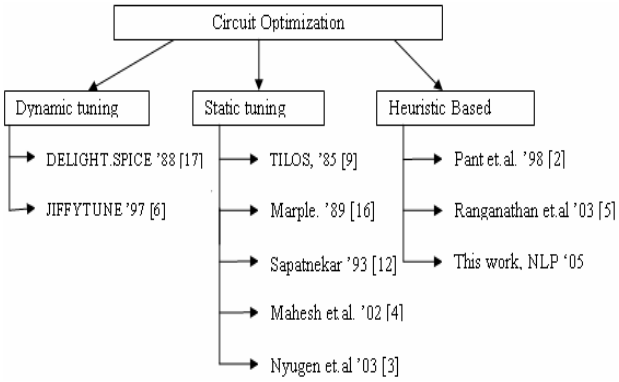

Figure 2. Circuit Optimization: Taxonomy

In this paper, we propose an approach in which the delay assignments for each gate are obtained using heuristics similar to that in [2, 14] and then NLP is applied to optimize each gate in order to perform gate sizing and voltage selection. Using NLP at the individual gate level provides for a fast and accurate solution. We use primal dual interior point NLP described in [7].

3. Problem Formulation

In this section, we describe our power and timing models to be used in the formulation of the gate sizing and voltage selection problems. This is then followed by the maximum delay assignment procedure.

3.1. Power and Timing Model

The power and timing models constitute important components in the modeling and formulation of the optimization problem. The dynamic power consumption of a gate i is given as,

$$P_i = f * V_{dd}^2 * E_i * (C_i + C_{wire}) + P_{sc} \qquad (1)$$

where, P_i is the total dynamic power consumed by gate i, f is the clock frequency, V_{dd} is the input supply voltage for the gate, E_i is the average transition rate of the gate, C_i is the intrinsic gate capacitance and C_{wire} refers to the load from the fanout count of the gate. The static power consumption of the gate is also directly

Proceedings of the IEEE Computer Society Annual Symposium on VLSI
New Frontiers in VLSI Design
0-7695-2365-X/05 $20.00 © 2005 **IEEE**

proportional to the gate size and to the supply voltage and is given by,

$$P_s = \frac{V_{dd} * GS_i * I_{off}}{f} \quad (2)$$

where P_s is the static power consumption, GS_i refers to the gate size and I_{off} refers to the off current. The delay of the gate with respect to its size is given as,

$$d_i = gd_i + C_i \frac{C_{out\,i}}{GS_i} \quad (3)$$

where d_i refers to the delay of the gate, gd_i is the intrinsic gate delay of gate, c_i is a constant, $Cout_i$ is the fanout load of gate and GS_i is the gate size. Reducing the size of the gates reduces the intrinsic gate capacitance of the gate, thus, reducing power consumption and fan-in load capacitances of the gate. Hence if optimizing power is the only objective, using minimum sized gates would achieve it. However, since the objective is to minimize power with a specified timing constraint, the gate sizes must be selected properly. The relation between the supply (V_{dd}) and threshold voltage (V_t) to the gate delay d_i is given as

$$d_i = \frac{K * V_{dd}}{(V_{dd} - V_t)^{\alpha}} \quad (4)$$

where, α is a technology related parameter and is fixed to 1.4 in this paper. Similar to GS, decreasing V_{dd} reduces power, but increases the delay of the circuit. This is often compensated by reducing V_t. But reducing V_t increases the leakage power. Hence all the three parameters V_{dd}, V_t and GS hold inverse relationship with respect to power and delay of the circuit. Optimal values for these parameters are needed to build low power high performance circuits. The optimization problem for minimizing power with delay as a constraint can be formally stated as,

$$\begin{aligned} &\text{Minimize : Power} \\ &\text{Subject to : Delay} \le T_{spec} \end{aligned} \quad (5)$$

V_{dd}, V_t and GS are the variables in this optimization process. Further for simplicity in fabrication, it is important that all gates in the network have identical threshold and supply voltage. Increasing number of distinct voltages requires additional implant masks, multiple tub biases [8]. Hence the whole logic network is limited to a single V_{dd} and V_t value.

3.2. Maximum delay assignment

The delay assignment is a preprocessing step to the heuristic based optimization. The approach is similar to path balancing in a directed acyclic graph considering the paths with least slack first [2, 14]. The delay assignment procedure to each gate is based on the simple reasoning that slower gates consume less power. In other words, to drive a large load a gate should be given more time to complete switching, to enable low power consumption. Hence, the delays to all the gates in the non-critical paths are increased to maximum possible, without affecting the specified critical path delay. The analytical equation for maximum delay assigned to a gate in the non-critical path in these works, [2, 14] is given by,

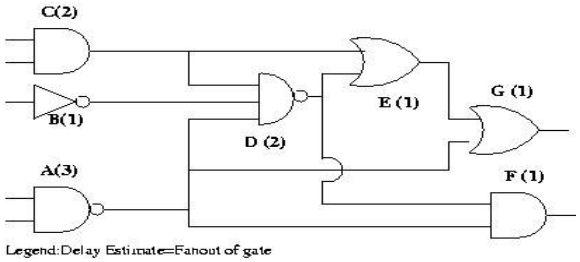

Figure 3 Example Circuit: Fan-out Delay Estimate

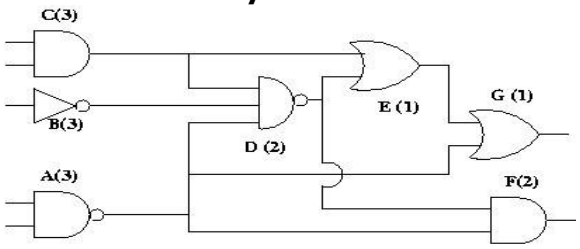

Fig 4 Maximum Delay Assigned Circuit

$$t_{Max\ gi} = \frac{T_M - T_p}{D_p} * d_{gi} \ge d_{gi} \quad (6)$$

Where T_M is the most critical delay of the circuit given by,

$$T_M = \max_{p \in circuit} (D_P) \quad (7)$$

d_{gi} is the un-optimized delay estimate of the gates in the circuit calculated using the fan-out values of the gate in this example. T_p is the sum of the initial delay estimates of all the gates in the path P that has already been assigned. D_P is the sum of the delay estimates of remaining gates. The sum $D_p + T_p$ gives the critical delay of that path. The paths in the circuit must be sorted in decreasing critical delay for this procedure to produce an acceptable delay assignment.

An example illustration of the maximum delay assignment procedure is given in Figures 3 and 4. The critical path delay of the circuit using fan-out count as the initial delay estimate is 7 units, given by the path A-D-E-G. The remaining paths are ordered in decreasing order of critical delay and each gate in these paths is assigned the maximum possible delay using

equation 6. Here, we explain this maximum delay assignment procedure for the second most critical path, C-D-E-G alone. The delay values of gates D-E-G, already in the most critical path remain unaltered and are marked as delay assigned gate. Hence, by definition, the values of $T_P = 4$, $D_P = 2$ and $T_M = 7$ substituted in equation 6 for gate C with an initial delay estimate $d_{gi} = 2$, produces a new delay value of 3 units for gate C. After this delay assignment all the gates whose delays have been changed are searched for a common V_{dd}, V_t and separate GS values, for maximum power reduction. The proposed nonlinear optimization approach is explained in the next section.

4. Proposed Nonlinear Optimization Approach

Input: Circuit C, Minimum possible gate size $minsize_i$
Output: Power optimized circuit, C_{pow}

% perform static timing analysis and identify the most critical paths;
P ← critical paths covering all gates (Table1);
D_{gi} ← Initial Delay estimate based on load driven;
% sort the critical paths in decreasing order of delay;
P_{new} ← ordered set of paths for delay assignment;
for each path ε P_{new} **do**
  for each gate g_i ε C_G **do**
    Calculate $Tmax_{gi}$,using equation 6;
    $D_p \leftarrow D_p - D_{gi;}$
    $T_p \leftarrow T_p - D_{gi;}$
    if ($Tmax_{gi}$!= D_{gi})
%gate is a candidate for sizing;
      nonlinear_optimize (power, $Tmax_{gi}$, GS);
%optimize power (equation 1) with $Tmax_{gi}$ as constraint for (equation 3) varying gate size (GS);
      endif;
    endfor;
endfor;

Algorithm 1: NLP Based Gate Sizing

The objective in our approach is to nonlinearly optimize power with the given clock cycle time as a constraint. The technology mapped circuit for gate sizing and voltage selection are first assigned the maximum delays. After the delay assignment to each gate in the circuit as shown in Figure 4, we optimize each gate individually for minimum power. The timing and power models described earlier constitute important components in this optimization. Since, more accurate these models are, more closely the power-delay curve is followed, resulting in efficient power minimization. The proposed NLP approach is given in algorithm 1. The inputs to the gate sizing algorithm are the given circuit C, and the minimum possible gate size *minsize*. The output from the algorithm is a power optimized circuit, where the gates in the non critical path are optimally sized without affecting the critical delay. The procedure starts by performing a static timing analysis, using pathmill, to identify the most critical paths. The critical paths are then sorted, before assigning maximum delays to create an acceptable assignment. This decreasing criticality order ensures no gate in the circuit is assigned zero or negative delay.

Another important attribute for this heuristic delay assignment is the number of paths considered, as the path count in a circuit is exponential to the number of gates. Hence sorting them would take a long time to complete. We here consider only K paths (Refer to Table 1 for value of K), which are required to cover 98% of the gates at least once in the circuit. The gates in these critical paths are then assigned maximum delays. The D_p and T_p values for path p is updated after each gate delay assignment. If the gate's $Tmax_{gi}$ value is different from D_{gi}, it is selected for optimization.

Table I Circuit Characteristics

Circuit	# Gates	# Paths
C880	441	190
C1355	552	191
C1908	773	260
C2670	1190	424
S510	264	98
S526	183	84

```
var x{j in 1...1} >= 0.18*10^(-6)
minimize obj;
2.081*10^(-6) + 2.509*10^(1)*x[1];
s.t. c1: 1*10^(-16)*x[1]^(-1) – 5.8*10^(-11)
*log(x[1]) + 3.489*10^(-13)*x[1]^(-0.5)
-5.789*10^(-10) <= 1*10^(-9)
s.t c2: x[1] <= 20*10^(-6)
data;
let x[1] := 2*10^(-6);
```

Figure 5. AMPL model: 2-input NAND Gate

The nonlinear optimization is done using the KNITRO package. KNITRO [7] is a primal dual interior point NLP solver. It is trust region based optimization solver, which uses sequential quadratic programming method to treat barrier sub problems. The optimization problem is fed to KNITRO in AMPL format. AMPL [13] is a popular modeling language

used for representing inputs in several optimization packages. A sample illustration of AMPL input modeling for power optimization of a 2 input NAND gate is shown in Figure 5.

The optimized power value for each gate is then multiplied by the associated transition density. Transition density for each gate in the circuit is calculated using the procedure described by Najm in [15]. The algorithm takes the transition density of primary inputs (0.5 is used in this paper) and propagates the values through logic modules to calculate the value in all the internal and output nodes. The summation of all these products is the optimized power value.

4.1. Gate Sizing and Voltage Selection

In this section, the gate sizing and voltage selection algorithm is given. Gate sizing and voltage selection is dependent on each other. Thus, combining these problems yield to better optimization. After the assignment of maximum delays to each gate in the circuit, we optimize each gate individually for minimum power. The strategy used in this paper is to find iteratively, the optimal combination of voltage and gate size values for each gate that meets the delay condition while minimizing power consumption. This strategy is based on the observation that power consumption and delay are monotonic functions of V_{dd}, V_t and gate sizes individually, other parameters being fixed. The algorithm optimizes the the V_{dd} and V_t value first for a low power value and then sizes the gate for these voltage value. The combination with the lowest power consumption is assigned to the circuit.

5. Experimental Results

In this section, the details regarding the implementation of the proposed algorithms and experimental results for the benchmark circuits are presented. The experiments were conducted on a set of combinational and sequential ISCAS benchmark circuits. The overall simulation flow followed in this paper is shown in Figure 6. The ISCAS benchmark circuits are converted to Berkeley Logic Interchange format (BLIF) and then mapped to user defined technology library using Sequential Interactive Synthesis (SIS). The BLIF file generated by SIS is then converted to a spice netlist, for enumeration of critical paths. Timing and power estimates were determined using Path-Mill and Prime-Power. The generated critical paths are then filtered to K paths (Refer Table 1) covering most of the gates. These paths are then sorted in decreasing order of criticality. Maximum delays are assigned to each gate separately using equation 6. A KNITRO [7] simulation is done to optimize power for all these gates with these delay constraints. The primary input activity factor of 0.5 is used for the simulations and is propagated using the procedure in [15].

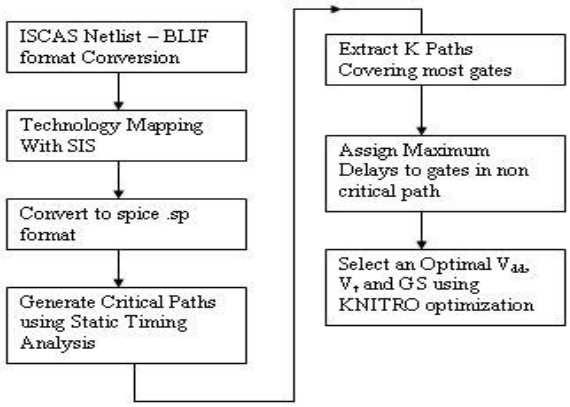

Figure 6. Simulation Flow

Table 2. Power Reduction with Gate Sizing only

Circuit	Nguyen 03 [3] (percent)	This work NLOP (percent)
C17	0	7.910
C1355	32.64	39.792
C432	41.1	54.996
C499	19.7	33.425
C880	58	72.451
C1908	62.16	62.196
C2670	69.51	83.117
C3540	66.25	70.455

The results for gate sizing and voltage selection have been presented and compared with the heuristic based binary search [2] approach and a linear programming based slack assignment [3] approach in Tables II & III respectively. The V_{dd} and V_t values for the gate sizing optimization are set to 1.8V and 0.45V respectively, for a fair comparison with results presented by Nyugen, in [3]. The percentage reduction in power obtained by gate sizing optimization alone is shown in Table II. It can be seen that our method achieves better savings for all circuits. Optimization with V_{dd}, V_t and sizing is compared in Table III with values presented by Pant 98, in [2], which uses a binary search procedure, for finding optimal values for V_{dd}, V_t and gate sizes. Data in Table III is represented with the amount of power reduction from the un-optimized circuit. Here 68x in column 2 refers to a reduction of

68 times from the original circuit. The proposed method achieves better savings in this case too. Further, in addition to providing a more efficient solution, the proposed method also converges faster than the binary search procedure [2]. The gate sizing optimization using KNITRO on the average required 4.3 iterations to converge to an optimal solution compared to 10 iterations in [2]. Similarly the V_{dd} and V_t values also required only 3.2 iterations to converge for most circuits. Thus, the gate sizing and voltage selection problem formulation using NLP technique with a heuristic delay assignment improves the efficiency and also the time required to converge to the optimal solution.

Table 3. Power Reduction for Benchmarks using Gate sizing and Voltage selection

Circuit	Pant 98 [2]	This work NLOP
S298	68 x	146 x
S382	77 x	87 x
S386	135 x	179 x
S510	102 x	139 x
S526	67 x	96 x

6. Conclusion

In this work, we have formulated the gate sizing and voltage selection problems as strict nonlinear problems and solved them using an interior point NLP for power minimization under delay constraints. The algorithm assigns maximum delays to all gates in the circuit and then selects optimum V_{dd}, V_t and gate sizes for power minimization. Experimental results on ISCAS benchmark circuits clearly illustrate the efficacy of the proposed approach. The algorithm also converges faster than the heuristic based binary search approach. Further investigation of NLP is needed for other optimization problems in VLSI design automation.

Acknowledgements

This work was supported in part by an exploratory grant from Semiconductor Research Corporation.

References

[1] "International Technology Roadmap for semiconductors": Semiconductor Industry Association, 2003

[2] Pankaj Pant, Vivek K. De, and Abhijit Chatterjee, "Simultaneous Power Supply, Threshold Voltage, and Transistor Size Optimization for Low-Power Operation of CMOS Circuits", IEEE TVLSI, vol. 6, no. 4, Dec 1998

[3] David Nguyen et.al, "Minimization of Dynamic and Static Power Through Joint Assignment of Threshold Voltages and Sizing Optimization", *ISLPED'03*, pp 25-27

[4] Mahesh Ketkar and Sachin Sapatnekar, "Standby Power Optimization via Transistor Sizing and Dual Threshold Voltage Assignment", ICCD 2002

[5] N. Ranganathan and Ashok K. Murugavel, A Microeconomic Model for Simultaneous Gate Sizing and Voltage Scaling for Power Optimization, ICCD 2003

[6] Chandu Visweswariah, Optimization techniques for high-performance digital circuits, ICCAD 1997, Pages: 198 - 205

[7] Richard H. Byrd, Mary E. Hribar, Jorge Nocedal, "An Interior Point Algorithm for Large Scale Nonlinear Programming", SIAM Journal on Optimization 99, 877 – 900

[8] J.Burr and J.Shott, "A 200 mv self-testing encoder-decoder circuit using Stanford ultra low power CMOS", in ISSCC: Tech Dig, Feb 1994, pp 84-85

[9]J.P.Fishburn and A.E. Dunlop. TILOS: A posynomial programming approach to transistor sizing. ICCAD 85, 326–328.

[10] A. R. Conn, N. I. M. Gould, Ph L. Toint, "LANCELOT: A FORTRAN Package for Large-Scale Nonlinear optimization", Springer-Verlag NY, Inc., Secaucus, NJ, 1992

[11] D. Gay, M. Overton, M. Wright, "A primal-dual interior method for nonconvex nonlinear programming, in: Y. Yuan (Ed.), Advances in Nonlinear Programming", Kluwer, Dordrecht, 1998, pp. 31-56.

[12] S. S. Sapatnekar, V. B. Rao, P. M. Vaidya, and S. M. Kang. An exact solution to the transistor sizing problem for CMOS circuits using convex optimization. IEEE TCAD, 1621-1634, November 1993.

[13] Robert Fourer, David M. Gay, and Brian W. Kernighan, "The AMPL Book *AMPL: A Modeling Language for Mathematical Programming",* Duxbury Press, 2002

[14] Chunhong Chen and Xiaojian Yang and Majid Sarrafzadeh, Potential slack: an effective metric of combinational circuit performance, 2000 IEEE/ACM ICCAD, Pages 198—201

[15] F.N Najm, "Low-pass filter for computing the transition density in digital circuits",IEEE TCAD, 1123 – 1131, Sep 94

[16] D. Marple, Transistor size optimization in the tailor layout system, 26th ACM/IEEE DAC, 1989, Pages 43 – 48

[17] W. Nye, D. C. Riley, A. S. Vincentelli, and A. L. Tits. DELIGHT.SPICE: An optimization-based system for the design of integrated circuits. IEEE TCAD, 501-519, Apr 88