Circuit Analysis and Design using Evolutionary Algorithms

Marc Thomas, Christian Burwick, Karl Goser
Lehrstuhl für Bauelemente der Elektrotechnik
Universität Dortmund
D-44221 Dortmund, Germany
E-mail: thomas@luzi.e-technik.uni-dortmund.de

Abstract

This paper focuses on electronic design at circuit level. The use of evolutionary algorithms to this application is discussed and a trade off to existing approaches is investigated. The design and analyzing task at this level is described in detail. As example a 1-bit full adder design in static CMOS is inspected with regard to power consumption and delay. In algorithmic scope both, single- and multi-objective optimization are regarded here.

1 Introduction

Development of new concepts to low power design needs a comparison to existing alternatives. Independant of the abstract level an optimal decision in lower, as well as in upper levels is essential to determine the concepts potential. This paper focuses on the circuit level as a parameter adjustment task. Instead of commonly used local meliorating procedure evolutionary algorithms as a global optimum seeking method are used here. A global search could guide the designer to alternative designs far away from usual ones. This becomes more important in using new circuit techniques where experience is missing. Beside simple design tasks analyzing circuit characteristics is in the scope of this work. Simple design can be seen as improving an existing or retrieving a new circuit design for implementation by determining transistor geometries and voltages with respect to given design rules. Circuit analysis focuses on the breadth of design capabilities and its boundaries, e.g. power-delay, with the prospect to gain information about parameter dependencies.

At first a classification of this work in general research activities is given in section 2. Next is an introduction to an evolutionary algorithm and its capabilities in section 3. Following in section 4 and 5 the use of optimization to circuit design and analysis is illustrated. Despite this work is motivated by new low power technologies a usual CMOS circuit is used here as example for better comparison fundamentals to other algorithmic approaches in further work. Finally some concluding remarks are given in section 6.

2 Circuit Optimization

Electronic design at circuit-level is a numeric adjustment process to meet constraints and goals on a fixed structure. Parameter extraction and variation, simulation, and result checking is done interactive in a time consuming process. Experts knowledge allows single parameter adjustment towards given goals step by step. This motivates multiple works to study automated design optimization at physical level.

Several approaches to automated design optimization in circuit level exist [1, 2, 3]. They differ in their specification of the problem as well as in solving aspects. But all dispense from global optimality to comply computation time constraints for practical use. The problem specification can be done analytically or by using a numeric simulator, e.g. SPICE. In analytical formulations known dependencies as in equation 1 ($\overline{P}$: average power consumption, f_{clk}: clock frequency, C_{eff}: effective capacity, V_{DD}: supply voltage, α: switching activity) are used to abstract with less accuracy but more speed. This is adequate in digital circuit optimization with transistors as switches. In contrast to this, a numeric simulator allows a more precise circuit analysis and to handle analog elements as well as digital ones. This can be used to get a circuits evaluation and its gradients.

$$\overline{P} = \frac{1}{2} f_{\mathrm{clk}} C_{\mathrm{eff}} V_{\mathrm{DD}}^2 \alpha \tag{1}$$

The solving approaches can be classified according to their specificy. The common task is power reduction by transistor sizing or scaling the supply voltage. A systematic approach is the iteratively determination of the critical path transistor for power reduction by slowing down [3]. More general approaches use numerical simulators to retrieve an evaluation and its gradients for specifying a problem of linear programming, gradient descend or non-linear optimization with linear constraints [1, 2]. These methods are for local meliorating, special cases, and some allow multi-objective optimization by formulating a min-max problem.

The task of general circuit optimization is still incomplete as some characteristics are left beside. First, analog behavior of the circuits becomes more important as the devices are shrink-ed to sub-μm feature size, i.e. digital elements gain analog aspects [4]. This leads to more complicated models and an increased role to the different transistor operation areas which require numerical simulation. Additionally, the intention of robustness becomes more and more important. Secondly the problem description as linear or non-linear optimization with linear constraints does not cope with general circuit optimization tasks. Parameter domain of correct working circuits is not known a priori. They must be selected after simulation and checking. Third, local methods for solving a circuit optimization task may meliorate manually determined circuits but do not exploit its whole improvement potential. Fourth, discrete optimization is needed to comply design rules. In analog circuits transistor size relations are as important as their absolute geometry. This is the same for digital elements with internal analog states. Continuous optimization with following discretization may fail.

3 Algorithm

In this work the numeric simulator surrounding a certain circuit design is regarded as a black-box model. This means the optimization algorithm gains no further information about the physical characteristics but is therefore not restricted to a special type of parameter sets like geometry values. The circuit structure is fixed, thus a set of parameters specify certain designs for simulation. Given constraints to circuits functionality are subsequently tested by algebraic terms. For optimization an evolutionary algorithm, in particular an evolution strategy is used.

An evolution algorithm is a population based methods using variation and selection of a given set of parameters. A population is a set of elements each representing a dedicated problem's solution (A circuit design as parameter vector collecting transistor sizes and voltages). These elements, called individuums, are attached by an evaluation called fitness. The population evolves step by step. In each step some modification is performed to the individuums. First, they are joined by recombination, a binary operator. Second, is mutation, an unary operator. At last generated individuums, called spring-off, are selected by their fitness to the next generation of population. Recombination is used to obtain and assemble existing properties while mutation allows to achieve new ones. The selection mechanism drives the population towards the requested direction. In this work population size is chosen as 15 and spring-off as 100 which are common values.

Benefits by this approach are the low requirements to problem specification. Any parameters beside the commonly viewed transistor geometries and voltages can be handled, as example threshold voltage or doping concentration in analyzing tasks. Constraints can be any algebraic term, not limited to linear ones. Finally even noisy characteristics as robustness validation can be optimized.

The method allows to start from a previously determined parameter set or with full random initialization. Given constraints can be implemented as hard or soft ones. Hard constraints called restrictions here are essentially needed for circuits behavior. This may be voltage levels or tests whether the numeric simulator is working correctly. Punishments are soft constraints and can be used for requirements which should be fulfilled. This may be used for suppressing hazards or delay constraints. After optimization hard and soft constraints should be fulfilled if the solution set is not empty. This approach is time consuming, applying up to ten-thousands simulations but allows to reach the global optimum which is essential for circuit analyzing subjects.

4 Circuit Design

Electronic design in general is a multi-objective optimization task. This can be managed with single-objective optimization algorithms by linear object weighting or by improving in each step the worst object. The approach used here is to set a constraint for each object except one.

In manual design optimization, rough rules to achieve a low power consumption exist, like the two step design concept proposed in [5]. At first the circuit is designed to minimum delay and afterwards the supply voltage is reduced till delay-constraint is exploited. The result should be a low-voltage, low-power circuit design. Experiments show that this approach results in a significantly higher power consumption in relation to a full adjusted design.

The 1-bit Full Adder

As an example a 1-bit full adder [6] in static CMOS is used here. It consists of 28 transistors each adjustable in width and length. Involving supply voltage 57 parameters are free to choose. The signal levels are assumed to be equal to the supply voltage and ground. But choosing each transistor geometry independently would lead to a too specific circuit design. Evaluation of power consumption by simulation is highly dependent on the given input pattern. To shorten simulation time the input pattern is chosen as simple as acceptable but then transition probabilities relations

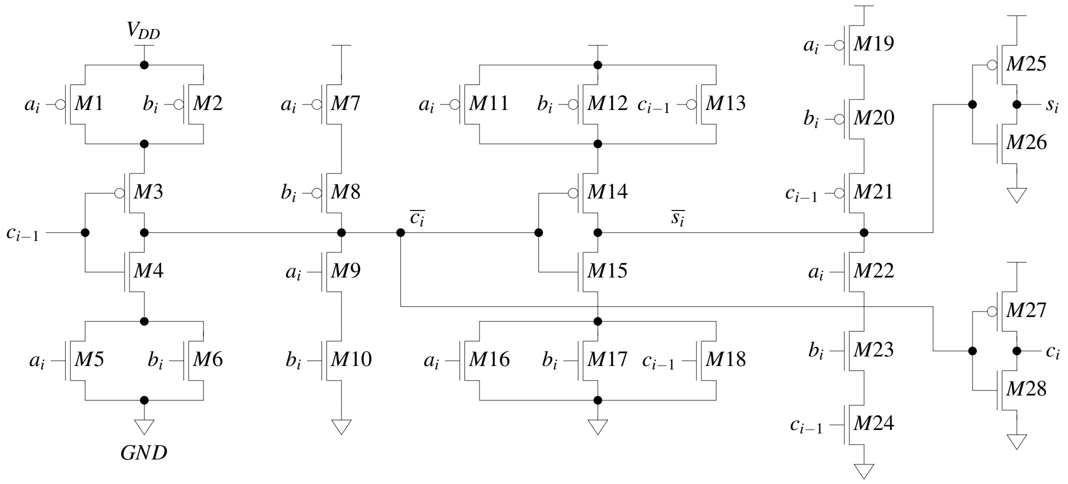

Figure 1: Design of a 1-bit full adder. Parameters are transistor sizes and voltage.

may differ from reality too much. Alternatively, transistor parameters can be grouped to achieve a more general design. This additionally reduces the total number of parameters which normally results in a faster convergence in the optimization process. In this example, shown in figure 1 this implies to assort following transistors: (M1-M2), (M3), (M4), (M5-M6), (M7-M8), (M9-M10), (M11-M12-M13), (M14), (M15), (M16-M17-M18), (M19-M20-M21), (M22-M23-M24), (M25-M27), (M26-M28). Using these dependencies 29 adjustable parameters are left for optimization.

For an initial survey the number of parameters is reduced once more by partitioning the transistors into two groups. The first (M1-M10) for carry evaluation and the second to calculate the sum and the output drivers (M11-M28). In both p- and n-channel transistors have to be specified in width and length. Adding the supply voltage results in nine adjustable parameters for optimization.

The knowledge about a circuit symmetries allows using a fast and simple input pattern initiating a transition in each circuit path. Overlapping impulses on each input line are used here resulting in six different states. The two remaining states of the full adder are symmetric to the involved ones. For complete analyzation of power consumption the different transitions would have to be taken into account but due to complexity it is not considered here (30 transitions in this example). The effort must be justifiable with regard to simulations mismatch to real world. Conforming input pattern, constraints to output must be defined to obtain desired output in a given time limit. The result is a kind of tube for the output signals. Here a delay of 2ns and assurance of 60% of the output level are used. In figure 2

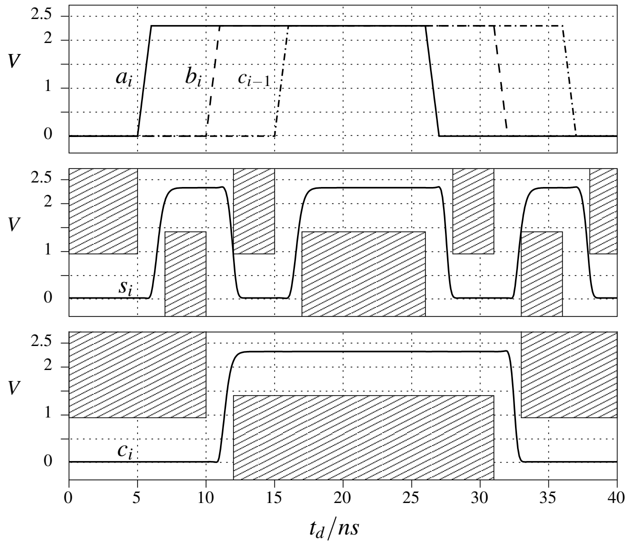

Figure 2: Simulation results of 1-bit full adder. Input and output signal transients with constraints using design B in Table 1

input signals (a_i,b_i,c_{i-1}) and outputs (s_i,c_i) with timing constraints are shown.

Signal transients show how optimization exploits the delay constraints for both rising and falling edge in case of s_i. Transistor grouping detains the carry c_i to reach the limit. The according circuit design has an average power consumption of $\overline{P} = 63\mu W$ in $0.35\mu m$ technology at $50MHz$ (design B in table 1). Alternatively a gradient descend algorithm applied to the same circuit reaches $143\mu W$ as best result. The two-step design concept named above, results in a power consumption of $187\mu W$ at $2.4V$.

5 Circuit Analysis

The task of circuit analysis differs from design as parameter dependencies and circuit features are focused. General characteristics independent to an implementation are observed which implies the needless of parameter discretization and necessity of multi-objective optimization. Determining a functional context a multi-objective optimization algorithm is required which does not result in a single realization but gives an appropriate coverage of the Pareto set.

An algorithm complying these requirements is the predator-prey model [7]. It is an evolutionary algorithm based on a structured population. Individuals (here: circuit parameter sets) correspond to the prey and are locally selected for replacement by different evaluation criteria corresponding to the predators. These perform a random walk and therefore drive the population towards the Pareto set, keeping desired diversity.

This algorithm can be used to investigate certain characteristics among different objectives like power, delay, area, robustness and how design parameters like transistor width and length and supply voltage influence them. As an example the power-delay curve and the dependency of power consumption on the supply voltage for the above mentioned 1-bit full adder is analyzed here. Some expressive designs extracted in this work are listed in Table 1. The designs A, B, C, E and F are Pareto optimal in regard to power-delay characteristics, while design D gives minimal power consumption at a supply voltage of $5V$. Remarkable are the high p- to n-channel ratios in design C and E (2.8–5.7) and also the low ratio in design F (< 1), the fastest one. The both best low-power designs are almost equal in power (A: $62\mu W$, B: $63\mu W$) and delay. Their designs seem to be quite common with p- to n-channel ratios of 2.5 and 3 in design A and 1.9 and 2.0 in design B..

Design:	A	B	C	D	E	F
V_{DD}/V	2.10	2.30	2.87	5.00	3.53	5.00
$\overline{P}/mW$	0.06	0.06	0.16	0.21	0.30	1.26
t_d/ns	1.02	0.99	0.63	1.05	0.48	0.36
$\overline{P} \cdot t_d/fJ$	61	59	101	221	144	454
$W_{p1}/\mu m$	6.85	4.05	16.00	0.95	16.00	16.00
$W_{n1}/\mu m$	2.71	2.05	2.79	0.60	3.85	5.87
$W_{p2}/\mu m$	4.85	2.60	8.95	0.60	8.11	10.60
$W_{n2}/\mu m$	1.61	1.35	1.91	0.60	2.87	12.40

Table 1: Some results for the 1-bit full adder in static CMOS sorted by power consumption. The p- and n-channel transistor widths of transistor group 1 (M1-M10) and 2 (M11-M28) are given in W_{p1},W_{n1},W_{p2},W_{n2}.

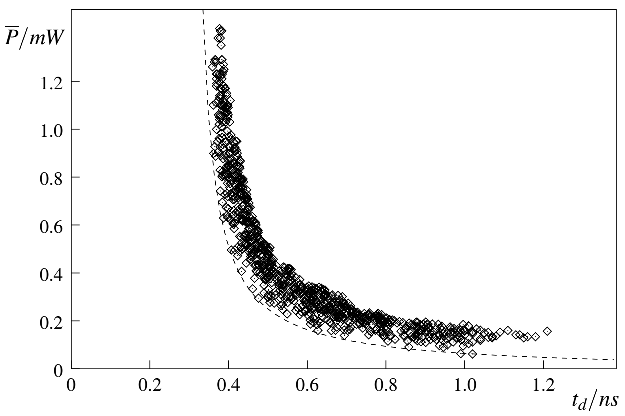

Figure 3: Power-delay characteristics: Simulation results $\diamond$ approximating the assumed power-delay borderline (dotted line).

5.1 Power-Delay characteristics

The most interesting characteristics in circuit design is the power-delay relation which is in lack of suitable analyzing tools simplified to the power-delay product. However it ignores the potential in possible power and delay. A common way to give an idea of the characteristics shape is variation of the supply voltage but its validity is closely limited. In this work multi-objective optimization is used for minimizing power along with delay.

Seeking methods as used here, just give single circuit designs complying the given constraints. There is no guarantee to optimality or even no information about the absolute quality of the solutions. Hence detected results suffer only to gain and sustain hypothesis. A complete derivation of parameter dependencies would require an analytical examination which lacks due to complexity.

The results to the optimization task determined by the predator-prey model are shown in figure 3. A sharp borderline can be determined describing the power-delay limit. This implies that the algorithm approximates the curve very well. The simulation result suffices the analytical form $(\overline{P} - a)(t_d + b) \geq c$ with constants $a, b, c > 0$.

5.2 Voltage-Power characteristics

Low power circuit design is closely coupled to the knowledge of the enormous influence of the supply voltage. Theory substantiates a quadratic voltage influence to power consumption in case of high switching activity (equation 1). Simulation results varying just the supply voltage

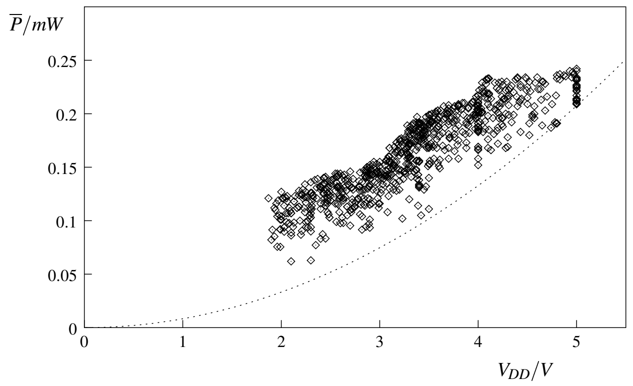

Figure 4: Voltage-power characteristics: Simulation results ⋄ beside theory approved dependency (dotted line) starting at best adjustment for 5V supply voltage.

show an exponent of even more than two. For approximating a voltage-power dependency a monotonic increasing function is assumed for sure. Thus an optimization to maximum voltage and minimum power is appropriate.

Results in figure 4 show less definite characteristics than in case of power-delay which indicate a harder optimization task. It points out that the voltage-power relation is less than quadratic. Especially in the area below 3V a further reduction of the supply voltage seems to become less efficient if full geometry optimization potential is exploited.

6 Conclusion

The evolutionary approach has been presented as a qualified method to circuit design and analyzing tasks. As a global seeking method evolutionary algorithms allow the exploration to the limits. This allows the user to compare technologies independent to designers implementation capabilities. In future works instead of just determining dependencies, detecting the causes to design goals, i.e. the influence of W/L-ratios, p- to n-channel-ratios or supply voltage will be inspected. Evolutionary algorithms are able to cope with noisy data, thus implying robustness by use of noisy input (geometry and signal mismatch) is quite simple. In all, the user gains a more precise imagination to the task of circuit design.

The evolutionary algorithms need improvement, respective specialization towards circuit design to cope with more complex circuits with less simulation effort. Further work will be done in adjustment and performance estimation of this approach.

Acknowledgment

This work is supported by the "Deutsche Forschungsgemeinschaft (DFG)" in the Collaborative Research Center on Computational Intelligence (SFB 531).

References

[1] W. Nye, D. C. Riley, A. Sangiovanni-Vincentelli, and A. L. Tits, "DELIGHT.SPICE: An Optimization-Based System for the Design of Integrated Circuits," *IEEE Transactions on Computer-Aided Design*, vol. 7, no. 4, pp. 501–519, 1988.

[2] C. Visweswariah, "Optimization Techniques for High-Performance Digital Computing," in *Proc. of IEEE Int'l Conf. on Computer Aided Design (ICCAD)*, pp. 198–207, 9-13. Nov. 1997.

[3] P. Girard, C. Landrault, S. Pravossoudovitch, and D. Severac, "A non-iterative gate sizing algorithm for high reduction in power consumption," *Integration, the VLSI journal*, vol. 27, pp. 37–52, 1997.

[4] H. Veendrick, "Digital Goes Analog," in *Proc. of the 24th European Solid-State Circuits Conference (ESSCIRC* (H. G. J. H. Huijsing, A. H. M. van Roermund, ed.), (The Hague, Netherlands), pp. 44–50, 22-24 Sep. 1998.

[5] R. Rogenmoser and H. Kaeslin, "The Impact of Transistor Sizing on Power Efficiency in Submicron CMOS Circuits," *IEEE Journal of Solid State Circuits*, vol. 32, pp. 1142–1145, July 1997.

[6] J. M. Rabaey, *Digital Integrated Circuits*, ch. 7, p. 391. Prentice Hall, 1996.

[7] M. Laumanns, G. Rudolph, and H.-P. Schwefel, "A Spatial Predator-Prey Approach to Multi-Objective Optimization," in *Proc. Fifth Int'l Conf. Parallel Problem Solving From Nature (PPSN V)* (T. Bäck, A. E. Eiben, M. Schoenauer, and H.-P. Schwefel, eds.), vol. 1498 of *Lecture Notes in Computer Science*, pp. 241–249, Springer, 1998.

[8] T. Bäck, *Evolutionary algorithms in theory and practice: evolution strategies, evolutionary programming, genetic algorithms*. Oxford University Press, 1996.

[9] W. Nebel and J. Mermet, *Low Power Design in Deep Submicron Electronics*, vol. 337 of *E - Applied Sciences*. Kluwer Academic Publishers, 1997.