The 47th IEEE International Midwest
Symposium on Circuits and Systems

Analog-Circuit-Component Optimization with Genetic Algorithm

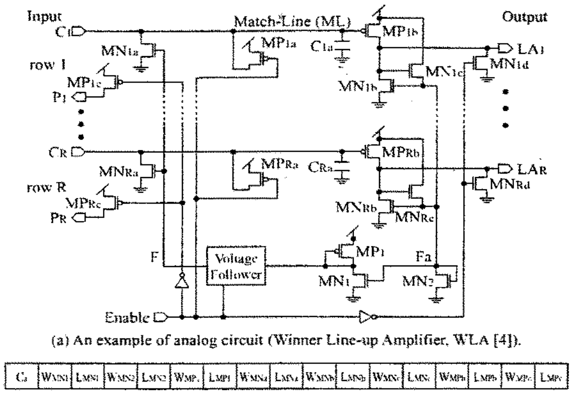
Kazuhiro Takemura† Tetsushi Koide‡ Hans Jürgen Mattausch‡ Toshio Tsuji†
†Graduate School of Engineering ‡Research Center for Nanodevices and Systems
Hiroshima University 1-4-2, Kagamiyama, Higashi-Hiroshima, 739-8527, Japan
Phone: +81-82-424-6265 Fax: +81-82-422-7185 E-mail: {takemura, koide}@sxsys.hiroshima-u.ac.jp

***Abstract*—This paper presents an automatic parameter-optimization method for components in analog circuits, so as to facilitate design tasks such as smallest power consumption or fastest operational speed. For solution space exploration a genetic algorithm (GA) is used. In addition, design know-how of the target circuit is incorporated in the GA to improve effectiveness and time consumption of the solution search. The proposed method enables exploration of a larger solution space than possible for the human designer, while reducing the designer's workload at the same time. The efficiency of the proposed method is verified through its application to a complex analog circuit including feedback, which is used as minimum-distance-search circuit in fully-parallel associative memories.**

I. INTRODUCTION

Recently, in development of ASIC, especially digital circuit design, the efficiency of circuit design is increasing according to sophisticated CAD environments, such as Hardware Description Language (HDL), automatic logic synthesis, placement and routing tools. With recent advances of deep submicron semiconductor technologies, the number of transistors placed on one chip is increasing. Therefore, it becomes possible to implement high-performance applications as SoC (System on Chip), in which digital and analog circuits are integrated in one chip (such as a digital-analog mixed ASIC). However, in analog circuit design, unlike digital circuit design, automatic design tools are not sufficiently developed, and the analog circuit designers suffer from very long design times because of trial and error during design optimization. Therefore, development of the automatic design techniques of analog circuits becomes indispensable [6].

In this paper, we focus on the Genetic Algorithm (GA) [1] method which is known to result in a robust search algorithm for combinatorial optimization problems. A GA-based optimization method for component parameters of analog circuits is proposed so as to automatically obtain circuit design result, which satisfies the performance specifications of the target application. Generally, during the parameter design of circuit components, whenever parameter values of components such as transistor channel width or length are changed, it is necessary to verify the resulting circuit performance by circuit simulation. This process is repeated until the circuit performance satisfies the analog-circuit specification. The iteration speed and design convergence is greatly influenced by the designer's skill, and generally requires significant effort until determine the optimal circuit parameter values are determined. Therefore, by automating the process of circuit-component design, using the powerful search capability of GA, a substantial reduction of the design's workload can be expected.



(a) An example of analog circuit (Winner Line-up Amplifier, WLA [4]).

(b) An example of chromosome representation.

Fig. 1. An analog-circuit example and its chromosome encoding.

II. CIRCUIT-COMPONENT OPTIMIZATION WITH GA

The objective of this paper is to increase the efficiency of analog circuit design by using GA for analog-circuit-component optimization. Figure 1(a) shows the minimum-distance search circuit called Winner Line-up Amplifier (WLA) [4], as an example of an analog circuit. Generally speaking, it is difficult to immediately determine appropriate component-parameter values which realize the optimal circuit performance. Therefore, in this paper, by automating an iterative improvement of circuit parameter values using GA, a set of appropriate parameter values which satisfy the performance specification of the target analog circuit will be determined within a practical computation time.

Genetic Algorithms (GAs) are stochastic algorithms based on the mechanics of natural selection and natural genetics, and deal with the individuals (chromosomes) of candidate solutions (called population) encoded in a problem specific representation [1]. During the genetic process, new candidate solutions are composed by using genetic operators such as crossover and mutation. Since, in general, GAs can generate good quality solutions for complex combinatorial optimization problems, they are used in many engineering fields [1], [3], [5] but they may sometimes require long computation times depending on the optimization problems. In the proposed method, to reduce the computation time and improve the solution quality, design know-how of the target analog circuit is incorporated as a heuristics enhancement technique in the GA.

The flowchart of the proposed GA-based algorithm, as shown in Fig. 2(a), is similar to a typical GA [1]. In the following, the details of the proposed algorithm will be explained.

A. Chromosome Representation

In the proposed method, parameter values of circuit components are represented by real numbers. In general, there are several kinds of component parameters in analog circuits such

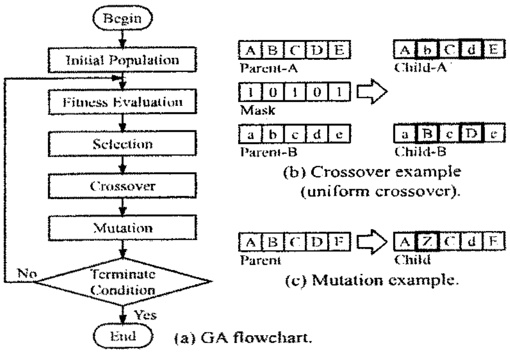


Fig. 2. Proposed GA-based circuit-component optimization.

as transistor (MOSFET) channel width W and channel length L and so on. As shown in Fig. 1(b), we adopt a sequence of these parameters to represent each individual (chromosome) in the population. For the parameter encoding, each parameter is normalized with the minimum unit, which is specified by the corresponding design process, and encoded as an integer. Figure 1(b) shows the chromosome representation of our analog example circuit, the WLA circuit of Fig. 1(a). One capacitance C_{ja}, $(j = 1, \cdots, R)$ and channel width W_i as well as channel length L_i of eight MOSFETs $(i = MN_1, MN_2, \cdots, MP_{jc})$ are encoded as genes of the chromosome.

In the following, let $\boldsymbol{X}_i$ be the i-th individual in the population of N individuals and let its representation be an n-dimensional vector $\boldsymbol{X}_i = (x_{i1}, x_{i2}, \cdots, x_{in})$, where x_{ij} is an integer. Index i $(1 \leq i \leq N)$ refers to the individual number, and index j $(1 \leq j \leq n)$ refers to the gene number within its chromosome. Furthermore, let $x_{ij,min}$ and $x_{ij,max}$ be the minimum and maximum values of x_{ij} $(x_{ij,min} \leq x_{ij} \leq x_{ij,max})$, respectively, which are determined beforehand from the analog circuit constraints, such as performance specification, layout design constraints and so on.

B. Generation of Initial Population

For all genes x_{ij} $(1 \leq i \leq N,\ 1 \leq j \leq n)$, initial values are randomly chosen based on a uniform random distribution satisfying the condition $x_{ij,min} \leq x_{ij} \leq x_{ij,max}$. Thus the initial population is generated. Especially, if an initial circuit-component design is known in advance, then the starting values of each gene x_{ij} are generated by the equation,

$$x_{ij} = x_{ij,init} \times a^b \tag{1}$$

where, $x_{ij,init}$ are the known values of each component from the initial design, a is a constant ($a = 5.0$ is adopted in the simulation experiment) and b is a uniform random number in $[-1, 1]$. By using equation (1), initial values of x_{ij} can be generated based on feasible component values from an actual design, so that an effective search space reduction can be expected.

C. Fitness Evaluation

In GAs, in order to select a set of individuals which should survive in the next generation, the fitness (evaluation function) for each individual is calculated. The larger the fitness of the individual is, the better the quality of the circuit solution it represents is. That means, the set of component values of the corresponding analog circuit satisfies the design specification better. In the proposed method, a circuit simulation is performed based on the component values for each individual generated by the GA, and its fitness is calculated based on a predetermined fitness evaluation function. Figure 3 shows an example of the fitness evaluation flow.

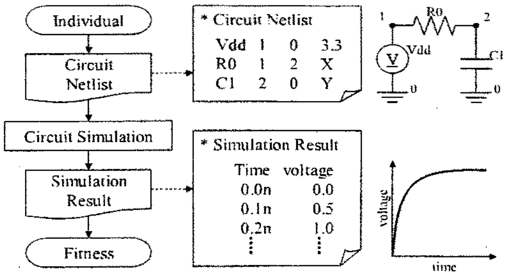


Fig. 3. Fitness evaluation.

First, the circuit netlist for fitness evaluation is generated by using the template of the netlist of the target analog circuit and parameter values of the genes. In the netlist template, the connection among each component is represented beforehand and the component parameters are set to be variables. Next, by decoding each gene of the individual (chromosome), the parameter values of each component are generated, and substituted for variables in the netlist template. Then, a circuit simulation is performed for the completed netlist, and simulation results such as current, voltage or power consumption are obtained for evaluation of the circuit performance. In the application example of the proposed method, power consumption $P(\boldsymbol{X}_i)$ of the whole circuit and operational speed $D(\boldsymbol{X}_i)$ are used for the fitness evaluation function.

The proposed fitness function $F(\boldsymbol{X}_i)$ of the analog-circuit-optimization problem uses the power-delay (PD) product $(P(\boldsymbol{X}_i) \times D(\boldsymbol{X}_i))$, and is defined by the following equation.

$$F(\boldsymbol{X}_i) = PD_{max} - P(\boldsymbol{X}_i) \times D(\boldsymbol{X}_i) - Penal(\boldsymbol{X}_i) \tag{2}$$

Here PD_{max} is the maximum of $P(\boldsymbol{X}_i) \times D(\boldsymbol{X}_i)$ among all the individuals of the previous generation, and $Penel(\boldsymbol{X}_i)$ is a penalty function. Depending on the parameter values of some components, the whole circuit may not operate normally and violate important constraints of the design specification. In such a case, depending on the degree of the violation, an adequate penalty value $Penel(\boldsymbol{X}_i)$ is imposed on the original fitness value as shown by the third term of equation (2) ($Penel(\boldsymbol{X}_i) = 0$ without violation). Thus, the total fitness $F(\boldsymbol{X}_i)$ of other individuals, which have a larger (less good) PD product but satisfy the design constraints, becomes better (i.e. their numerical value becomes larger).

D. Selection

Individuals are selected for the crossover and mutation steps under fixed rules depending on their fitness. Other individuals with low fitness are screened and eliminated from the population. For the proposed analog-circuit-optimization method, we combine the roulette selection and the elitist preserving selection techniques [2]. In the roulette selection, the chance of each individual to survive the selection phase is determined by the ratio of its fitness to the sum of all fitness values. On the other hand, in the elitist preserving selection, the individual with the highest fitness value in the population survives to the next generation. By combining these two selection techniques, the chromosome of the best individual of the present population is not broken by crossover or mutation and survives as it is in the next generation.

E. Crossover and Mutation

The genetic operators of crossover and mutation produce new individuals according to a predetermined probability. Our proposed method adopts the uniform crossover as crossover operator. Two arbitrary individuals (parents) are randomly selected from the population and their genes are rearranged at several crossover points, which are determined randomly, in order to reproduce two new individuals (children). First, an n-dimensional mask pattern is created by randomly generated binary numbers. As shown in Fig. 2(b), Child-A inherits Parent-A's gene at the 1-positions of the mask. Similarly, Child-B inherits Parent-A's genes at the 0-positions of the mask.

As mutation operators, we used the uniform mutation [5] and equation (1) in the same way as for the generation of the initial population. The uniform mutation is carried out by selecting some genes randomly. Their corresponding gene values x_{ij} are changed by generating a uniform random number within the boundaries $x_{ij,min} \leq x_{ij} \leq x_{ij,max}$. In the case of using equation (1), $x_{ij,init}$ is defined as the gene value of the individual in the current generation. In the proposed method, one of these two techniques is selected for each circuit component to reflect the differences in design constraints. This is done because the magnitude of possible parameter values is generally different and depends on the component type as well as on the corresponding layout design rules. For the gene of a component with small range, mutation technique according to equation (1) is applied and for one with large range, the uniform mutation is used. Therefore, new gene values should remain in the suitable range even if mutation is applied.

F. Termination Condition

The proposed method results the operation series of evaluation, selection, crossover and mutation until a predetermined generation number is reached. Individuals of the final population which have the highest fitness values are the candidates for the design solution. The circuit designer finally selects the most suitable design solutions among the individuals, which satisfy the initial analog-circuit specification.

G. Consideration of Design Know-How

Generally, GAs can be expected to converge on a better solution, if individuals have a critical variety in each generation. However, when applied to the component-parameter-optimization problem in analog circuit design, the circuit performance may critically depend on the values of each component. Since especially the fitness of individuals which do not have the normal circuit operation becomes very small, such individuals will not survive in the next generation. As a result, the variety of individuals in the population may become low and convergence to a local optimum may occur.

Therefore, in the proposed method, effective incorporation of the design know-how of the target analog circuit in the GA, is important to keep the variety of individuals in the population high so that the efficiency of the search is increased. For example, when the case of abnormal operation of the circuit is detected during the circuit simulation, the value of the responsible component can be modified so that this problem is corrected. Moreover, when the designer has detailed knowledge about the relation and dependability among circuit components, the corresponding component parameters can be suitably modified based on this knowledge. Thus, the modified individuals will likely survive to the next generation because their fitness becomes higher and in consequence the variety of individuals will remain also high. This heuristic modification is applied after crossover or mutation depending on the fitness of each individual.

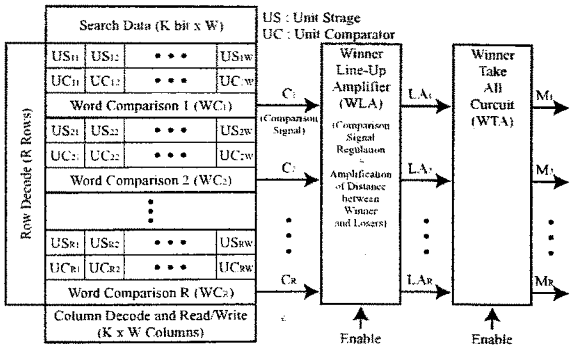


Fig. 4. Fully parallel nearest-distance search associative memory [4].

III. Simulation Experiments

A. Application to a Fully-Parallel Nearest-Distance Search Associative Memory Design

In order to verify the effectiveness of the proposed method, the simulation experiments were carried out with a practically applied analog circuit, namely the Winner Line-up Amplifier (Fig. 1(a)) of a mixed digital-analog fully-parallel nearest-distance search associative memory (Fig. 4) [4]. This associative memory is a functional memory which finds the nearest match for an input data among reference data based on a distance measure (for example, Hamming distance or Manhattan distance). As shown in Fig. 4, it consists of three blocks; memory cell region, Winner Line-up Amplifier (WLA) and Winner Take All circuit (WTA). The memory cell region holds reference data and calculates the distances between search data and reference data in parallel. WLA and WTA circuits search the nearest-distance reference data, called *winner*, using the principle of distance amplification by an analog circuit.

The row which has minimum distance between search data and reference data is called *winner row*, and the voltage on the match-line of the winner row becomes the lowest (Fig. 5(a)). The voltage differences of the match-lines between the winner row and other rows (called *loser rows*) are amplified by the WLA, and it is designed so that the output voltage of WLA at the winner row becomes the highest.

WLA is an especially important analog circuit which determines the performance of the associative memory. Since, it is necessary to amplify even small differences of currents between winner and losers at high speed in the WLA, the optimization of each circuit component is very important.

B. Simulation Experiments

We implemented the proposed GA-based method with C and Perl languages. Execution of the GA, generation of netlists and analysis of a circuit simulation results including fitness evaluation were realized (Fig. 2,3). Nassda's HSIM simulator is incorporated for in the circuit simulation task.

The GA parameters used for the simulation experiments are shown in Table I. The individual (chromosome) representation shown in Fig. 1(b) is adopted. For the fitness evaluation, transient information was acquired by circuit simulation in 1ns steps for the time interval from 50ns (rise time of an enable signal) to 250ns. From the acquired information the power

TABLE I

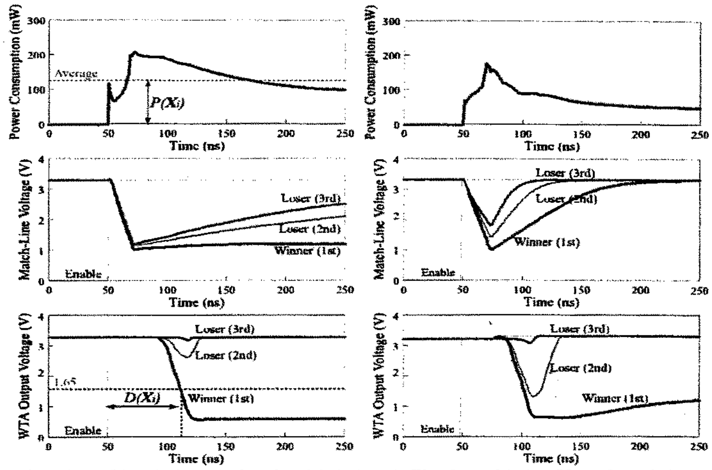
GA PARAMETERS OF SIMULATION EXPERIMENT

Generation limit	500 : Termination condition
Population size	$N = 50$
Chromosome length	$n = 17$ (C x 1, W x 8, L x 8)
Fitness function	PD product
Selection	Roulette selection Elitist preserving selection
Crossover	Uniform crossover (50%)
Mutation	C : Uniform mutation (3%) W, L : Equation (1) (3%)

TABLE II

EXPERIMENTAL RESULTS

Solution	PD product (mW x ns)	Power (mW)	Delay (ns)
Initial (original)	6944	124	56
(A) PD product best	**3388**	**77**	**44**
(B) Power best	5400	54	100
(C) Delay best	4662	111	42



(a) Waveform of the initial circuit (the original design). (b) Waveform of the best PD product solution.

Fig. 5. Waveform of the initial circuit (the original design) and the best circuit obtained with the proposed component-optimization method based on GA.

consumption and the search time were calculated. Power consumption is defined as the average power consumption during the specified time interval (Fig. 5(a)). The search time is defined as the time from the rise time of an enable signal until the search result of the associative memory is determined. For this purpose, the output signal of the WTA (Fig. 4) is analyzed, and the search time is calculated from the time point when the output voltage of the winner row falls bellow 1.65V (threshold voltage of an A/D conversion inverter in the decision circuit).

We also incorporated design know-how about WLA circuit of Fig. 1(a) into the GA processing. As shown in Fig. 1(a), the WLA circuit uses a feedback from its output signals (LA_j) to the match-lines (ML_j). Due to this feedback, the voltages of the match-lines may oscillate, if unsuitable component values are chosen. If such oscillations occur, stable amplification of the WLA is no longer possible. However, the oscillations can be reduced by changing the value of capacitances C_{ja} connected to the match-lines and by adjusting the transistor sizes (channel width or length) in the feedback loop. During the simulation experiments, the genes of the corresponding individuals with oscillations were modified heuristically before selection.

C. Experiment Results

The circuit performance result of the solutions obtained by simulation experiments is compared with the initial solution in Table II. PD product, power consumption and search time of the original circuit, (A) the solution with the minimum PD product, (B) the solution with the minimum power consumption and (C) the solution with the minimum search time are listed, respectively. In addition, the waveforms of the power consumption, the voltages on the match-lines and the output voltages of the WTA are shown for the solution (A) in Fig. 5(b).

The simulation experiment verified that three sets of circuit-component parameters (A)~(C), which improved the performance of the initial circuit design could be obtained by applying the proposed GA-based component-parameter-optimization method. Therefore, application of the proposal method allows the designer to select the best circuit solution from a number of improved design results, which satisfy the constraints given by the design specification.

All simulation experiments were performed on a Sun Blade workstation (Ultra SPARC III 1.4GHz). The computation time of the proposed method was about 20 hours on the condition in Table I. Therefore, the proposed method can achieve the improvement of the circuit performance in relatively short time as compared to the manual circuit parameter design with trial and error, thus substantially reducing the designer's workload. This demonstrates the effectiveness of the proposed GA-based method for circuit-component design.

IV. CONCLUSION

In this paper, we proposed a parameter optimization method of circuit components in analog circuit design. From the experimental results of applying the proposed method to an actual analog circuit design, the effectiveness of the GA-based optimization method could be demonstrated.

Future work includes further the shortening the computation time of the program, applying the proposed method to other analog circuit designs, introduction of other GA types (for example, adaptive GA [3]) and the consideration of the design know-how in a more quantitative formulation.

ACKNOWLEDGEMENTS

HSIM is licensed within an academic collaboration with Nassda Corporation. Part of this work is supported by the 21st Century COE program, Ministry of Education, Science and Culture, Japanese Government.

REFERENCES

[1] D. E. Goldberg: "Genetic Algorithms in Search, Optimization and Machine Learning," Addison-Wesley (1989).

[2] D. E. Goldberg and K. Deb: "Foundations of Genetic Algorithms; A Comparative Analysis of Selection Schemes Used in Genetic Algorithms," Morgan-Kaufmann, pp. 69-93 (1991).

[3] K. Hatta, et al.: "Solving the rectangular packing problem by an adaptive GA based on sequence pair," Proc. Asia and South Pacific Design Automation Conf., pp. 181-184 (1999).

[4] H. J. Mattausch, et al.: "Fully-parallel pattern-matching engine with dynamic adaptability to Hamming or Manhattan distance," 2002 Symposium on VLSI Circuits Digest of Technical Papers, pp. 252-255 (2003).

[5] Z. Michalewicz: "Genetic Algorithms + Data Structures = Evolution Programs," Third, Revised and Extended Edition, Springer-Verlag (1996).

[6] K. V. Noren and J. E. Ross: "Analog circuit design using genetic algorithms," Second Online Symposium for Electronics Engineers (2001).