

# Evolutionary High Level Synthesis with Predictive Models.

Nidhin. A, D.S Harish Ram.

**Abstract**— In modern day chip design, the pressure for time to market is increasing and recovery time for investment is decreasing. So design times have to be shortened. An Artificial Neural Network (ANN) based predictive model for power to speed-up the cost function evaluation during High Level Synthesis (HLS) is presented here. Genetic Algorithms (GA) have been used successfully to solve HLS problems. Our proposed predictive model can be easily integrated with this evolutionary framework. The accuracy of this predictive model is tested with DFG benchmark circuits.

**Index Terms**— Artificial Neural Network, Cost Function Evaluation, Design Space Exploration, Evolutionary Computation, High Level Synthesis, Power Estimation, Predictive Models.

## 1 INTRODUCTION

High level synthesis (HLS) is the process of synthesizing behavioral specification of a digital circuit into a Register Transfer Level (RTL) description that realizes the behavior with respect to some design constraints.

HLS mainly involves (i) scheduling, (ii) resource allocation, (iii) binding and (iv) controller synthesis [1], [2]. The scheduling phase assigns each operation to one or more clock cycles (or control steps) for the execution. The resource allocation phase assigns the execution of the operations to hardware components. Then it is interconnected using connection elements. Binding is the actual mapping of operations to functional units, data transfer to buses and variables to storage units [3], [4].

Design Space Exploration (DSE) is the process of exploring different design alternatives having various trade-offs and objectives such as power area and delay. It is computationally unfeasible to use an exhaustive exploration strategy. The size of the design space grows as the product of the cardinalities of the variation sets for each objective[5][6]. Evaluation of a single configuration almost always requires the use of simulators or analytical models which are also complex. Another challenge is that the objectives being optimized are often conflicting. DSE flow can be schematically represented as in Fig. 1. The exploration phase uses the results of the evaluation phase to modify the system configuration parameters so as to optimize certain performance indices. The cyclic process ends when a system configuration meets the design constraints. Pareto-optimal configurations for these indices are

optimized and accumulated. These complex explorations are handled successfully by evolutionary algorithms (EAs). The drawback of this approach is the need to evaluate a huge number of design alternatives. The design process needs to be shortened to meet the time-to-market constraint.

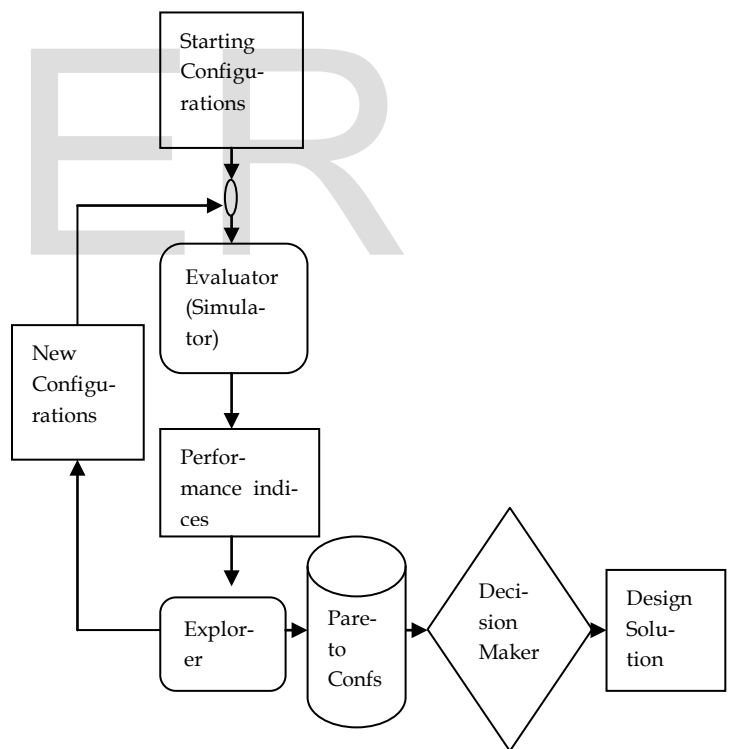


Fig. 1 Design Space Exploration

• Nidhin. A is currently pursuing masters degree program in VLSI Design in Amrita Vishwa Vidyapeetham University, India, E-mail: nidhin.iecee@gmail.com

• Dr D.S Harish Ram is working as Asst. Professor in Amrita Vishwa Vidyapeetham University, India, E-mail: ds\_harishram@cb.amrita.edu

A technique to speed up the cost function evaluation during HLS by applying soft computing and machine intelligence techniques is presented in this paper. Power is an integral component in any multi objective problem in VLSI. Early estimation of power with good relative accuracy is a major challenge requiring expensive simulations and characterizations. Hence power is considered in this work. There are two approaches to reduce this evaluation time. (i) Reducing evaluation time for a single iteration. (ii) Reducing number of true

evaluations [5]. Interpolation techniques are usually less time consuming and are applied to reduce the overall number of true evaluations [7].

This paper is organized as follows. Section 2 describes some works related to speed up the cost function evaluation during high level synthesis. In section 3, three parameters related to the power dissipation of a schedule is discussed. The proposed power prediction model is presented in section 4. The accuracy tested with various benchmark circuits are discussed in section 5 and finally concluded.

## 2 RELATED WORK

In [7], a linear regression model is proposed for area and delay estimation of data paths which reduces the computational effort for cost function evaluation.

The time-consuming logic synthesis step is substituted with a model of area, based on the important features of the structural descriptions obtained by the high-level synthesis step. Two possible cost models for the area are considered. One linearly combines the number of functional units present in the design and their area and counts the memory elements. The other one is a linear regression that also takes into account interconnections.

In [7], the authors also report another approach for the same based on fitness inheritance scheme which on the other hand, seeks to reduce the number of evaluations. Fitness inheritance substitutes some steps of the synthesis by interpolating the fitness of previously evaluated individuals. The fitness of individuals used for this scheme can be evaluated by any approach. In this scheme total number of evaluations is reduced rather than reducing the time required for a single evaluation. Since interpolation techniques are less time consuming, it is used to save some of the time required for a complete synthesis. Fitness inheritance depends only on the definition of the chromosome encoding and the fitness of previously evaluated individuals in the population. This technique is highly dependent on the chromosome and was found to be inadequate for modeling the power cost.

In [5] a Genetic Algorithm (GA) is used as exploration heuristic and a Fuzzy System (FS) as an evaluation tool. It is an intelligent GA approach which has the ability to avoid the simulation of configurations and to give them fitness values as per the fast estimation of the objectives. This system works as follows. When the GA evolves normally, the FS learns from simulations and it becomes expert and reliable. At this moment GA stops launching simulations and uses the FS to estimate the objectives.

A fuzzy rule-based system is chosen as an estimator. The advantages of this system are it has cheap additional computational time requirements for the learning process. It is negligible when compared with simulation time. Another advantage of this system is that fuzzy rules can be easily interpreted by the designer. The proposed approach has been applied on a highly parameterized very long instruction word (VLIW) processor. The integration with fuzzy system saves

great amount of time. However, the system depends heavily on the chromosome encoding used.

## 3 POWER ESTIMATION

In [8] metrics based on the compatibility graph of a schedule have been proposed for estimating the power cost. As the number of edges in the DFG increases the design space of the binding stage expands, increasing the probability of containing the absolute minimum power solution. The edge weights between nodes indicate the switching activity between nodes. Thus a CG with lesser average edge weight indicates a power aware schedule. These dependencies are encoded into the power cost function modelled by equations (1) and (2).

- Metric m1 denotes the number of edges
- Metric m2 considers the sum of the edges weights in the lowest k% of the value range for each node and computes the average of these sums over all nodes. Since the flow algorithm tends to select edges with smaller weights, m2 provides an indication for the quality of the input presented to the flow algorithm. Lower m2 values indicate a higher potential of yielding a lower power binding solution.
- Metric m3 is the generalization of metric m2. It computes the average of all edge weights contained in the DFG. i.e m2 is the value of m3 with k=100%. Metrics m2 and m3 should be minimized for minimum power. These metrics can be formulated as follows:

$$m_1 = \sum_{v \in \forall nodes} n_v \quad (1)$$

$$m_{2,3} = \frac{\sum_{v \in \forall nodes} \sum_{e \in E_v^k} w_{ve}}{m_1} \quad (2)$$

These metrics are also in adopted in our predictive model for power.

## 4 PREDICTION MODEL FOR POWER

GAs are successfully used to solve the HLS problems. Different schedules are encoded as chromosomes in GA [9]. This proposed predictive model can be integrated to this evolutionary framework easily.

The block diagram for the prediction model for power dissipation is shown in Fig. 3. The chromosomes are given as the input to the predictive model. The power metrics are extracted from the chromosomes [8]. These metrics are given as the input to the trained neural network. This network will predict the power for that schedule. The network is trained by

pre-evaluated power for some chromosomes using Synopsys tools. The Artificial Neural Network (ANN) used for predicting power is shown in Fig 4.

fort is achieved.

For IIR and HAL which are smaller bench marks the models achieved 100% accuracy with the training set of 45 chromosomes. In the case of DCT average error is 3.75% for the training set of 30 chromosomes. However it is reduced to 2.06% for the training set of 45 chromosomes.

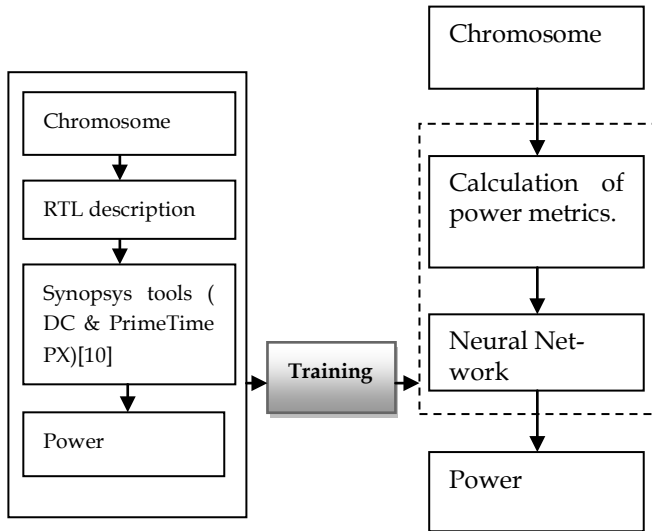


Fig. 3 Block diagram of power prediction model.

The ANN is trained using Levenberg - Marquardt (LM) algorithm[11][12]. The LM algorithm blends the steepest descent method and the Gauss-Newton algorithm. It inherits the speed advantage of the Gauss-Newton algorithm and the stability advantage of the steepest descent method.

### 5 RESULTS AND DISCUSSIONS

The accuracy of the developed model is verified using DFG benchmark circuits. The results on the HAL, IIR and DCT benchmarks are shown respectively in Tables 1, 2 and 3.

For the DCT benchmark, two sets of power models with training sets of 30 chromosomes and 45 chromosome each are developed. It is observed that as number of chromosomes in training phase increases the accuracy also increases.

Table. 1 Predicted power for HAL benchmark

Chromosome No.	Predicted power (in mW)	Actual power (in mW)
1	0.10555	0.10555
2	0.14189	0.14180
3	0.14189	0.14189
4	0.10555	0.10555
5	0.064723	0.06472
6	0.14188	0.14189
7	0.098082	0.09808
8	0.10555	0.10555
9	0.14189	0.14189
10	0.10555	0.10555
11	0.14189	0.14189
12	0.10555	0.10555
13	0.14189	0.14189
14	0.14189	0.14189
15	0.14189	0.14189
16	0.14189	0.14189
17	0.098082	0.09808
18	0.14189	0.14189
19	0.064723	0.06472
20	0.064723	0.06472

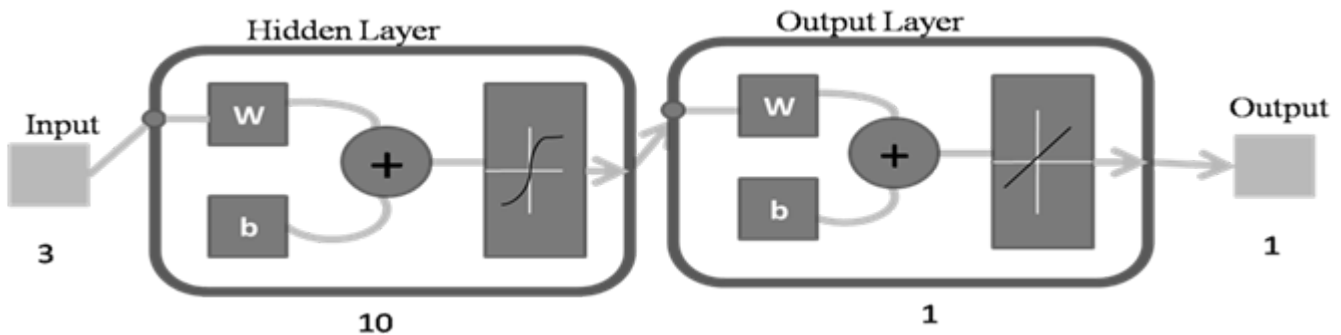


Fig. 4 Artificial Neural Network

The relative accuracy and minimization of computational ef-

Table. 2 Predicted power for IIR benchmark

## 6 CONCLUSION

A predictive model for power estimation during evolutionary high level synthesis by applying soft computing techniques is developed. The accuracy of the proposed model is tested with different DFG benchmark circuits. This model can be easily integrated into an evolutionary framework to solve HLS problems. Since the number of true evaluations is reduced with this model, it leads to reduction in design time.

## REFERENCES

- [1] M. C. McFarland, A. C. Parker, and R. Campasano, "Tutorial on high-level synthesis," in Proc. 25th Design Automation Conf., pp.330-336, June 1988.
- [2] C. Mandal, P. P. Chakrabarti and S. Ghose, "GABIND: a GA approach to allocation and binding for the high-level synthesis of data paths," Very Large Scale Integration (VLSI) Systems, IEEE Transactions on, vol. 8, no. 6, pp.747-750, 2000.
- [3] Krishnan, Vyas, and Srinivas Katkooi. "A genetic algorithm for the design space exploration of datapaths during high-level synthesis." Evolutionary Computation, IEEE Transactions on 10.3 (2006): 213-229.
- [4] F. Ferrandi, P.-L. Lanzi, D. Loiacono, C. Pilato and D. Sciuto, "A Multi-objective Genetic Algorithm for Design Space Exploration in High-Level Synthesis," in Symposium on VLSI, 2008. ISVLSI '08. IEEE Computer Society Annual, 2008.
- [5] Ascia, Giuseppe, et al. "Efficient design space exploration for application specific systems-on-a-chip." Journal of Systems Architecture 53.10 (2007): 733-750.
- [6] C. A. Mandal, P. P. Chakrabarti, and S. Ghose, "A design-space exploration scheme for data-path synthesis," IEEE Trans. VLSI Syst., vol. 7, pp. 331-338, June 1999.
- [7] Pilato, Christian, et al. "Speeding-Up Expensive Evaluations in High-Level Synthesis Using Solution Modelling and Fitness Inheritance." Computational Intelligence in Expensive Optimization Problems. Springer Berlin Heidelberg, 2010. 701-723.
- [8] Kursun, Eren, Rajarshi Mukherjee, and Seda Ogrenci Memik. "Early quality assessment for low power behavioral synthesis." Journal of Low Power Electronics 1.3 (2005): 273-285.
- [9] K. Deb, Multi-objective Optimization using Evolutionary Algorithms, John Wiley and Sons, 2002.
- [10] Prime Time-PX User Guide, Version B-2010.11.
- [11] Moré, Jorge J. "The Levenberg-Marquardt algorithm: implementation and theory." In Numerical analysis, pp. 105-116. Springer Berlin Heidelberg, 1978.
- [12] Roweis, Sam. "Levenberg-marquardt optimization." Notes, University Of Toronto (1996).

Chrom. No.	Predicted Power(in mW)	Actual Power (in mW)
1	0.1041	0.1041
2	0.0982	0.0982
3	0.1065	0.1065
4	0.0712	0.0712
5	0.0651	0.0651
6	0.1102	0.1102
7	0.0821	0.0821
8	0.0965	0.0965
9	0.0788	0.0788
10	0.1205	0.1205

Table. 3 Predicted power for DCT benchmark

No	Actual Power (in mW)	Training 30	with Error	Training 45	with Error
		Pre- dicted Power(in mW)		Pre- dicted Power(in mW)	
1	0.1389	0.1310	5.5%	0.1372	1.1%
2	0.1369	0.1312	4.1%	0.1425	4.0%
3	0.1354	0.13205	2.4%	0.1375	1.4%
4	0.1384	0.13556	2.0%	0.1402	1.3%
5	0.1319	0.12972	1.6%	0.1341	1.6%
6	0.1329	0.13954	4.9%	0.13514	1.6%
7	0.1438	0.1376	4.3%	0.1416	1.5%
8	0.1408	0.1317	6.4%	0.1364	3.2%
9	0.1334	0.1369	2.6%	0.1312	1.4%
10	0.1339	0.1391	3.8%	0.1386	3.5%