An approximation algorithm for testing memories of an integrated circuit

Lilia Zaourar, Alix Munier-Kordon

LIP6, Université Pierre et Marie Curie CNRS UMR 7606, 4, place jussieu, 75252
Paris Cedex 05, (France)
{Lilia.Zaourar, Alix.Munier}@lip6.fr

Abstract. Memory testing of an integrated circuit is a real industrial challenge, and presents several interesting original combinatorial optimization problems. A simple model for testing a set of memories, taking into account their test time and power, is first presented. Feasible solutions consist on grouping memories into lots of a maximum fixed power. The aim is to minimize the overall test time for a unique sequential tester. The problem includes the classical Bin-Packing problem and thus is NP-hard. A simple lower bound of the overall test time is first obtained using a continuous version of the problem. A simple heuristic issued from a *Next Fit Decreasing* strategy is then presented and its performance ratio is proved to be bounded by 2.

Keywords: Memory test, Batch, Bin-Packing, Approximation algorithm.

1 Introduction

Nowadays, any integrated circuit (SOC in short for System On Chip) leaving a supply chain is tested automatically. In order to reduce the time to market of a product this testing step has to be as cheap as possible with the least time consumption.

For that purpose, an external tester sends data to the circuit, and compares its answers to those normally expected. A set of components dedicated to the realization of this test phase is generally added to the circuit. Part of the processing is then carried out directly by the chip, significantly speeding the industrial process. Those compoments are called BIST (Built In Self Test).

Memory cores usually occupy a significant area of the SOC and are often designed with most aggressive rules. Thus, testing all these memories became a real challenge and presented original combinatorial problems. Two costs are usually considered in this phase: the total testing time of the memories of a chip and power consumption due to the test procedure. The idea here is to group the overall memories of a chip into different lots that are evaluated separately; each lot has its own testing time and required an amount of power to be tested.

Various grouping schemes have been proposed in the literature. In this work, we are interested only in the optimization criteria with respect to both power and time. The pioneering work on memory BIST sharing was presented by [9]. The author proposed a new BIST sharing architecture together with a test scheduling that does not exceed power dissipation limit during test procedure. Several techniques for test scheduling were investigated later. In [7] the authors adressed the issue of test scheduling under power and area constraints. They gave a greedy algorithm with a time complexity of $O(n^3)$ to select grouping memories. Numerical results are given for a SOC with less than 10 memories. In [1] the test scheduling optimization problem is modeled as an open-shop scheduling problem with m machines, and solved by using a Mixed Integer Linear Programming (*MILP*) formulation to obtain the optimal schedule. Based on the same assumption, a power constrained *MILP* model for the precedence and preemptive test scheduling problems has also been presented in [2]. In [5] the authors formulated the problem as a two dimensional Bin-Packing problem by fixing bounds for the maximum test power and test time. Then, they solved it heuristically by using a *Best Fit* heuristic algorithm. A survey of techniques used and numerical results can be found in [8].

The main flaw of all the methods adressed previously is that the complexity of the algorithms developed drastically limits the size of the problem considered. The aim of our work is to present a simple model for a sequential BIST tester and efficient heuristics to solve it.

Our paper is organized as follows. In Section 2, we define the memory grouping problem and formulate it as a generalized version of Bin-Packing. Section 3 is devoted to the determination of an optimal solution for a continuous version of our problem, leading to a nice lower bound of the minimum overall test time. *Next Fit Decreasing* heuristics are studied in Section 4. It is proved that, if the memories are allocated to lots following non increasing powers, the performance ratio of the built solution may be not bounded. However, it is also proved that this ratio is bounded by 2 if memories are considered following non increasing test times. Section 5 is our conclusion.

2 Problem formulation

An instance Π of the problem tackled here is defined by a set L of n memories $M_i, i \in \{1, \cdots, n\}$, each of them defined by a couple of positive integers (t_i, p_i). t_i represents the time needed by the tester to evaluate M_i, p_i is its associated power. An integer value P_{max} corresponding to the power of the tester is also fixed. Lastly, P_{max} should satisfy $P_{max} \geq P$ for $P = \max_{i \in \{1, \cdots, n\}} p_i$.

Memories are grouped into bins (or lots) to be tested. A bin $B \subseteq L$ has a test time $T_B = \max\limits_{M_i \in B} t_i$ and a power $P_B = \sum\limits_{M_i \in B} p_i$. Our problem consists in partitioning the set of memories L into a set of bins $\mathcal{B}$ such that:

- the power of each bin is bounded by P_{max}, *i.e.* $\forall B \in \mathcal{B}, P_B \leq P_{max}$;
- the overall test time $T_{\mathcal{B}} = \sum\limits_{B \in \mathcal{B}} T_B$ is minimum.

M_i	M_1	M_2	M_3	M_4	M_5	M_6	M_7	M_8	M_9	M_{10}	M_{11}	M_{12}	M_{13}
t_i	6	6	6	4	4	4	4	4	3	3	2	2	2
p_i	2	1	1	3	2	2	2	1	3	1	3	1	3

Table 1. Test times and powers of an instance of Π. We also suppose that $P_{max} = 6$.

As example, let consider the instance of Π reported by Table 1. A feasible solution is presented by Figure 1. Corresponding bins are $B_1 = \{1, 2, 3\}$, $B_2 = \{4, 5\}$, $B_3 = \{6, 7, 8\}$, $B_4 = \{9, 10\}$, $B_5 = \{11, 12\}$ and $B_6 = \{13\}$. Their test time are $T_{B_1} = 6$, $T_{B_2} = 4$, $T_{B_3} = 4$, $T_{B_4} = 3$, $T_{B_5} = 2$ and $T_{B_6} = 2$. Total test time is $T_{\mathcal{B}} = 21$.

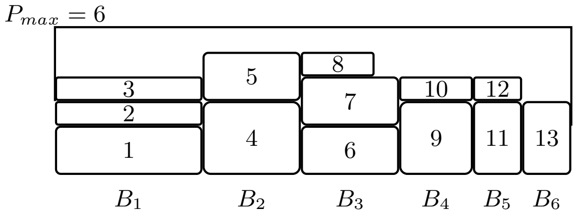

Fig. 1. A feasible solution for the instance of Π given by Table 1.

The sub-problem of Π for which memories are of equal test time ($t_1 = t_2 = \cdots = t_n$) is clearly a *Bin-Packing* problem. We conclude that Π is *NP-hard* [4], and thus the determination of approximation algorithms is of interest. To our best knowledge, Π has never been adressed before. However, several approximation algorithms that have been proposed to solve the classical *Bin-Packing* problem [3] constituted a starting point for our study.

Π can also be formulated as a batch scheduling problem [6] without set-up time on a single machine. Tasks correspond to memories. Each task $i \in \{1, \cdots, n\}$ has a duration equal to the test time t_i. Batches are carried out sequentially. P_{max} is the limit size of a batch.

3 An optimal solution for the continuous problem

Let consider in this section a continuous version Π^{cont} of the initial problem Π where a memory can be placed in several bins. The power of a memory M_i belonging to bins $B_1^{cont}, \cdots, B_\alpha^{cont}$ is split into α positive values $p_j^1, \cdots, p_j^\alpha$ with $\sum_{i=1}^{\alpha} p_j^i = p_j$. This extension of the problem has clearly no application in our industrial context. However, we build easily an optimal solution $\mathcal{B}^{cont}$ of Π^{cont}, providing a nice lower bound for the overall test time of Π.

Let us consider a feasible solution $\mathcal{B}^{cont}$ for the continuous version Π^{cont} built as follows where each memory is split into at most two bins. Let suppose that memories are numbered as $t_1 \geq t_2 \geq \cdots \geq t_n$. For any value $i \in \{1, \cdots, n+1\}$, the sequence u_i is defined as $u_i = \sum_{j=1}^{i-1} p_j$ and we set $k = \lceil \frac{u_{n+1}}{P_{max}} \rceil$ the number of bins B_α^{cont}, $\alpha \in \{1, \cdots, k\}$ built as:

$$B_\alpha^{cont} = \{i \in \{1, \cdots, n\}, (\alpha - 1)P_{max} - p_i < u_i < \alpha P_{max}\}.$$

The corresponding values of the memory power are set as follows:

$$\forall i \in B_\alpha^{cont}, p_i^\alpha = \begin{cases} p_i & if \quad (\alpha-1)P_{max} \leq u_i \leq \alpha P_{max} - p_i \\ u_{i+1} - (\alpha-1)P_{max} & if \quad u_i < (\alpha-1)P_{max} \\ \alpha P_{max} - u_i & otherwise. \end{cases}$$

For the example presented in Table 1, memories are numbered following non increasing test times. A representation of $\mathcal{B}^{cont}$ is pictured by Figure 2. Powers of memories belonging to two bins are $p_4^1 = 2$, $p_4^2 = 1$, $p_7^2 = 1$, $p_7^3 = 1$, $p_{13}^4 = 2$ and $p_{13}^5 = 1$. Test times of each continuous bin are $T_{B_1^{cont}} = 6$, $T_{B_2^{cont}} = 4$, $T_{B_3^{cont}} = 4$, $T_{B_4^{cont}} = 2$ and $T_{B_5^{cont}} = 2$. Thus, the overall test time is $T_{\mathcal{B}^{cont}} = 18$.

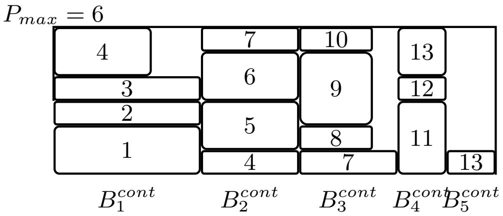

Fig. 2. An optimal solution $\mathcal{B}^{opt}$ for the continuous version Π^{cont} of the instance of Π expressed by Table 1.

Theorem 1. *$\mathcal{B}^{cont}$ is an optimal solution for Π^{cont}.*

Proof. The proof is by recurrence on the number of bins k. $\mathcal{B}^{cont}$ is clearly optimum for $k = 1$ since all the memories are placed in a same bin.

Let consider now an optimal solution $\mathcal{B}^{opt}$ for a minimum number of bin $k > 1$. Corresponding bins are noted $B_1^{opt}, \cdots, B_{k^\star}^{opt}$ with $k^\star \geq k$. Let suppose that B_1^{opt} has a maximum test time, *i.e.* $T_{B_1^{opt}} = \max_{i \in \{1,\cdots,n\}} t_i$. Starting from $\mathcal{B}^{opt}$ and using simple exchange arguments, it is possible to build another optimal solution $\mathcal{B}'^{opt}$ such that, the bin with the maximum test time $B_1'^{opt} = B_1^{cont}$. Remaining memories define another instance I of Π^{cont} with a minimum number of bins equal to $k - 1$. By recurrence, bins $B_2^{cont}, \cdots, B_k^{cont}$ correspond to an optimal solution for I, and thus $\sum_{j=2}^{k} T_{B_j^{cont}} = \sum_{j=2}^{k^\star} T_{B_j'^{opt}}$. We obtained lastly that

$$T_{\mathcal{B}^{opt}} = T_{\mathcal{B}'^{opt}} = T_{B_1'^{opt}} + \sum_{j=2}^{k^\star} T_{B_j'^{opt}} = T_{B_1^{cont}} + \sum_{j=2}^{k} T_{B_j^{cont}} = T_{\mathcal{B}},$$

which concludes the recurrence proof.

4 Performance ratio of Next Fit Decreasing strategies

We observe that classical simple strategies for approximatively solving the *Bin-Packing* problem may have a non-bounded performance ratio. These algorithms only take into account the size of the objects, corresponding in our case to the power of the memories. In the worst case, each new bin will contain a memory with a maximal test time.

Let consider first *Next Fit Decreasing* strategy for which memories are numbered and treated following non increasing powers, *i.e.*, $p_1 \geq p_2 \geq \cdots \geq p_n$. Each new memory M_i is placed in the current bin B. If there is no enough place, B is closed and a new current bin is opened to receive M_i.

Theorem 2. *The performance ratio of the Next Fit Decreasing strategy where memories are considered following their non increasing power is not bounded.*

Proof. Let consider an instance composed by $n_1 = P_{max} > 1$ memories of test time Δ_1 and $n_2 = n_1 \times (P_{max} - 1)$ memories of test time $\Delta_2 = \dfrac{\Delta_1}{P_{max} - 1}$. Powers are supposed to be unit.

A solution of *Next Fit Decreasing* algorithm may be obtained such as each bin contains one memory of test time Δ_1 and $(P_{max} - 1)$ of test time Δ_2. The overall test time equals $T = P_{max}\Delta_1$. Now, if every bin contains only memories of same test time, the overall test time equals $T^{opt} = \Delta_1 + (P_{max} - 1)\Delta_2 = 2\Delta_1$.

The ratio is then equal to $\dfrac{T}{T^{opt}} = \dfrac{P_{max}}{2}$, which is not bounded.

We study in the following the performance of the *Next Fit Decreasing* strategy where memories from L are considered following their non increasing test times. It is supposed thus that they are numbered such that $t_1 \geq t_2 \geq \cdots \geq t_n$.

Figure 1 presents the feasible solution obtained by a *Next Fit Decreasing* strategy applied to our example presented by Table 1.

Theorem 3. *Let the integer* $K = \lfloor \frac{P_{max}}{P-1} \rfloor - 1$. *The performance ratio of the Next Fit Decreasing strategy where memories are considered following their non increasing test times equals* $1 + \dfrac{1}{K}$.

This last bound is asymptotically tight. Indeed, consider an instance of our problem given by n_1 memories of power P, and $n_1 \times (P_{max} - P)$ memories of unit power. Test times of memories are all equal to Δ.

An optimal solution can be found by constructing n_1 bins, each of them containing one memory of power P. Remainings slots are all filled with memories with unit power. The overall test time is here equal to $T^{opt} = n_1 \times \Delta$.

Now, a *Next Fit Decreasing* strategy may lead to a solution where the first n_1 bins are filled with one memory of power P and $P_{max} - 2P + 1$ unit memories. It remains $n_1 \times (P-1)$ memories of unit power that are placed into $n_1 \times \frac{P-1}{P_{max}}$ bins. The overall test time is then equal to $T = n_1 \times \Delta \times (1 + \frac{P-1}{P_{max}})$.

Setting $\alpha = \frac{P_{max}}{P-1}$, we observe that for important values of α that $\frac{1}{\alpha(1-\frac{1}{\alpha})} \approx \frac{1}{\alpha}(1+\frac{1}{\alpha}) = \frac{1}{\alpha} + o(\frac{1}{\alpha})$. Thus, the ratio $\frac{T}{T^{opt}} = 1 + \frac{1}{\alpha} \approx 1 + \frac{1}{\alpha(1-\frac{1}{\alpha})}$, and the performance ratio is tight when $\alpha = \frac{P_{max}}{P-1} \to +\infty$.

5 Conclusion and future works

Theoretical perspectives rely in the study of other approximation algorithms to improve the performance ratio. These algorithms may be implemented and compared experimentally with other more complex algorithms (as example genetic algorithms [10]) to solve real-life instances.

References

1. K. Chakrabarty, Test scheduling for core-based systems using mixed-integer linear programming, *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems* :1163–1174, 2000.
2. K. Chakrabarty, V. Iyengar, Precedence-based, preemptive, and power-constrained test scheduling for system-on-a-chip, *IEEE 19th Proceedings on VLSI Test Symposium* :368–374, 2001,
3. E. G. Coffman, Jr., M. R. Garey and D. S. Johnson. Approximation Algorithms for Bin Packing: A Survey. In *Approximation algorithms for NP-Hard Problems*, ed. Dorit S. Hochbaum,46-93, University of California, Berkeley, CA 94720-1777, 1996.
4. M.R. Garey, D.S. Johnson. *Computers and intractability. A guide to the theory of NP-completeness.* WH Freeman and Company, San Francisco, Calif.
5. Y. Huang, W.T. Cheng, C.C Tsai, N. Mukherjee, O. Samman, Y. Zaidan, S.M. Reddy, Resource allocation and test scheduling for concurrent test of core-based SOC design. *IEEE Test Symposium, 2001. Proceedings. 10th Asian* :265–270, 2001.
6. C.N Potts, M.Y Kovalyov. Scheduling with batching: a review. *European Journal of Operational Research* :228–249, 2000.
7. CP. Ravikumar, G. Chandra, A. Verma, Simultaneous module selection and scheduling for power-constrained testing of core based systems. *IEEE Thirteenth International Conference on VLSI Design* :462–467, 2000.
8. Q. Xu, N. Nicolici, Resource-constrained system-on-a-chip test: a survey. *IEEE Proceedings Computers and Digital Techniques* :67–81, 2005.
9. Y. Zorian, A distributed BIST control scheme for complex VLSI devices. *Proc. 11th IEEE VLSI Test Symposium* :4–9, 1993.
10. L. Zaourar, Y. Kieffer, A.Wenzel. A Complete methodology for determining memory BIST optimization under sharing constraints. *IEEE Asia Symposium on Quality Electronic Design,pages* :46–53, 2011.