

# TAM/wrapper Co-optimization And Test Scheduling For SOCs Based On Hybrid Genetic Algorithm

Xu Chuan-pei

Department of Electronic Engineering,  
Guilin University of Electronic Technology, Guilin, 541004, China  
Email: [xcp@guet.edu.cn](mailto:xcp@guet.edu.cn)

Lu Xue-yun and Hu Cong

Department of Electronic Engineering,  
Guilin University of Electronic Technology, Guilin, 541004, China  
Email: [lx269054135@163.com](mailto:lx269054135@163.com) [HuCong@guet.cn](mailto:HuCong@guet.cn)

**Abstract**—In this paper, a new method is presented for TAM/wrapper co-optimization based on hybrid genetic algorithm and two-dimensional packing problem. In this method, core test is represented by rectangles, and a hybrid genetic algorithm that provides highly optimal solution for two-dimensional packing problem is introduced for TAM allocation and test scheduling. During the scheduling, the TAM width assigned to cores could be adjusted to an appropriate size to minimize the idle time. This HGA based method was implemented in C and applied to ITC'02 SOC Test Benchmark. Experimental results show that lower testing time was obtained by this new method compared to other methods<sup>[1,4,5]</sup>.

**Index Terms**—SOC co-optimization, hybrid genetic algorithm, packing problem

## I. INTRODUCTION

The development of SOC technology reduces the cost of electronic products. However, the highly integrated functional modules make SOC testing more complex at the same time. A lot of IC companies can not afford the expensive cost of testing. There is a contradiction between the testing time and test resources. The scarce test resources cause the difficulties of IP core parallel test. This directly lead to the consumption of very long testing time. To address this issue, researchers must develop effective test programs so as to complete the test tasks in the shortest time with using limited test resource.

Facing The challenge of the SOC test, IEEE and VSIA and other international organization formulated the SOC test criteria diligently. IEEE proposed the P1500 standard, namely embedded essence test criteria (SECT, the Standard for a Embedded Core Test). Basic idea of P1500 standard is to provide standard test structure to Ensure that in SOC, IP core and the Interconnection

structure between the IP cores can be tested. The test architecture of IEEE Consists of four parts: test stimuli source, test response sink, test access mechanism(TAM), core test wrapper(CTW). Test stimuli source generates test stimuli, test response sink analyzes the result of test, TAM provide a means to transmit test data. It transmit the test stimuli from the source to the wrapper of the core and transmit the test response from the wrapper of the core to the sink. Wrapper is the interface between nuclear and TAM. It not only provides the right test access path to load test stimuli and capture test response, but also ensure the isolation between tests of cores. besides above,wrapper provides fore modes to cores: The normal function modes, the core scan test mode, reset mode of core testing, as well as the Interconnect test mode between cores. Wrapper and TAM impact directly to chip area, test time and test data volume,so it is extremely important to optimize them.

Basing on the IEEE P1500 test strategy, SOC test optimization problem can be divided into Wrapper/TAM design and test scheduling optimization, these problems have been researched by scholars, and certain solutions have been proposed. Most prior researches have studied wrapper/TAM design and test scheduling optimization as independent problems<sup>[4,6]</sup>. However, there is close relationship between Wrapper/TAM design and test scheduling optimization, solving these two problems in conjunction will certainly shorten the testing time.

In this paper, Best Fit Decreasing(BFD) algorithm is used to completed the design of wrapper to minimize the testing time of each IP core, and hybrid genetic algorithm is used to solve the problem of TAM design and test scheduling. The method which combine HR algorithm with design of flexible\_width TAMs is used to calculate the fitness of chromosome.

Experimental results show that lower testing time was obtained by our method compared to other methods<sup>[1,4,5]</sup>.

This work is supported by National Natural Science Fund under Grant No. 60766001. Dates of manuscript submission, revision is 2009 oct 03, 2009 oct 04-10 respectively

II. RELATED CONCEPTION AND PRIOR WORK

The P1500 wrapper is a shell around a core, that allows that core to be tested as a stand-alone entity by shielding it off from its environment. Like wise, the wrapper allows the environment to be tested independent from the state of the core<sup>[10]</sup>.

The issue of designing balanced scan chains within the wrapper was addressed in [11]. Fig. 1 shows two connection mode between wrapper I/O unit and scan chain2, 1(a) is an unbalanced wrapper, 14 clock cycles is needed for running a test stimulin. 1 (b) is a balanced wrapper, 8 clock cycles is needed for running a test stimulin<sup>[5]</sup>. This shows that balancing scan chains can reduce the testing time of given core.

TAM is test access mechanisms which transports the stimulin from source to core and from core to sink. A wrapper that amongst other things, connects the TAMs to the core<sup>[10]</sup>. During the scheduling each IP core should be connected to a TAM of certain width. The major issue in test scheduling is determining the TAMs width that assigned to each IP core. Test scheduling for SOCs involving multiple test resources and cores with multiple tests is especially challenging. Even simple test scheduling problems for SOCs have been proved to be NP-hard. Many methods have been used to solve test optimization problem. In prior studies, modus operandi is dividing a test bus of certain width into several groups, the wrapper of IP core was connected to a certain group of bus, each IP core use a fixed TAM width throughout the testing process. This is referred as fixed-width TAM architecture. For example, bus is divided into 2 parts in [12], then Integer Linear Programming (ILP) algorithm was used for test scheduling. In [4], bus is divided into 2 or 3 parts and then ILP algorithm is replaced by Genetic algorithm(GA). GA gives very good results, however there is great room for improvement if flexible-width TAM architecture is adopted.

V. Iyenga introduced a flexible TAM architecture in 2002<sup>[1]</sup>. It was shown in[1] that for a given core, the testing time varies with TAM width as a “staircase” function . This implies that the testing time does not decrease with increasing TAM width until a core-specific

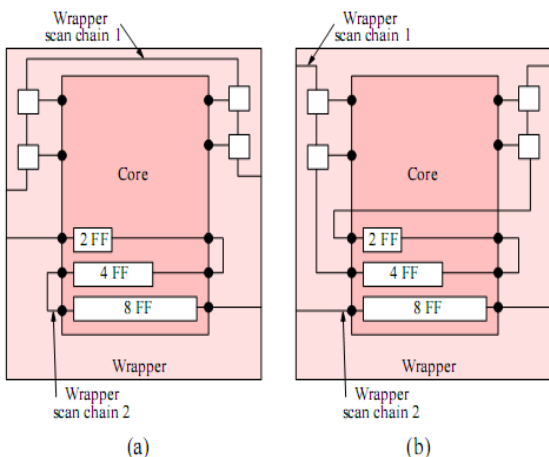


Figure1. Wrapper chains(a) unbalanced, (b) balanced.

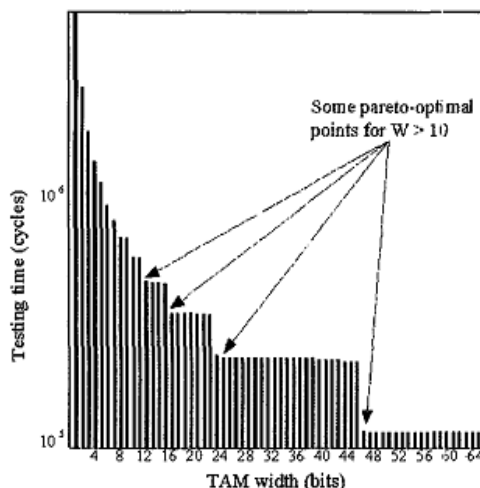


Figure 3. relationship between testing time and TAM width

threshold is exceeded. Hence if a core is connected to a TAM of width  $w$ , the same testing time may actually be obtained using only  $w'$  wires ( $w < w'$ ). The remaining  $w-w'$  wires, which could have been used to transport test data for another core, are not efficiently utilized. So only threshold width should be chosen to assigned to cores. This is the basis on which the V. Iyenga determines the TAM width for each core.

The method in [1] use BFD algorithm to design a set of wrappers for each core to eliminate the mismatch between the core’s test data. A test schedule is then determined by test scheduling algorithm and a temporary TAM width is assigned to each core, this temporary TAM width can be adjusted in accordance with specific needs during the test scheduling. In this method, there is no explicit partition of the total TAM width. Instead, a more flexible TAM architecture is created in which the total TAM width is partitioned effectively among the group of cores being tested during any time interval in the schedule. This partition is allowed to vary with time.

This flexible TAM architecture can significantly reduce the waste of test resources. However, the method in [1] didn’t obtain desired result because ILP algorithm is used for test scheduling. Studies have proved that ILP algorithm is not very effective for solving test scheduling problem.

Addressing all issues above, flexible-width TAM architecture is combined with hybrid genetic algorithm to complete the co-optimization in this paper.

III. SOC TEST OPTIMIZATION MODEL

The co-optimization problem in this paper is as follows:

Given the total TAM width  $W_{max}$  for the SOC, the wrapper and TAM assignation should be determined for each core, and a test schedule should be designed for the SOC, such that:

- a. The total number of TAM wires utilized at any moment does not exceed  $W_{max}$ .
- b. The overall SOC test completion time is minimized.

In this paper, co-optimization is transformed into rectangle packing problem. The rectangle packing problem is described as follows:

Given a collection of rectangles, and a bin of fixed width and unbounded height, pack the rectangles into the bin, such that no two rectangles overlap, and the height of the bin is minimized.

In the co-optimization model, the test of each IP core will be represented by a rectangle. The width of the rectangle corresponds to the TAM width assigned to the core, and the height of the rectangle corresponds to the testing time of the core.

When core test is represented with rectangle, two parameters must be initialized: the width and height of each rectangle, namely the width of bus assigned to each IP core and the test time of the core for this width. The method described in [1] is used to initialize these two parameters. The pseudocode for the initialization is presented in Fig. 4 and the definition of highest Pareto-optimal width can be seen in [1].

The parameters in the pseudocode are as follows:

- $w_{p_i}$ ---preferred TAM width for core  $i$ ;
- $T_{p_i}$ --- testing time of core  $i$  for the width of  $w_{p_i}$ ;
- $T_i(w)$ ---testing time of core  $i$  for the width of  $w$ ;
- $w_i$ ---the width of rectangle for core  $i$ ;
- $h_i$ --- height of rectangle for core  $i$ ;
- $W_{max}$ ---the TAM total width;
- $pc$ ---control parameter, control the value of  $T_i$  and thus control the value of  $w_i$ ;
- $dc$ ---control parameter, control the equivalence probability between  $w_{p_i}$  and highest Pareto-optimal width;

In this paper, BFD algorithm is used to design a set of wrappers for each core. A preferred TAM width is assigned to each core in line 1-9, and then BFD algorithm is used for calculating the test time of each core for the width of  $w_{p_i}$ . Width and height of rectangle  $i$  are made

```

1. compute collection A of cores of SOC
2. Set  $g$ = the quantity of cores belong to C
3. FOR  $i=0$  to  $g$ 
4.   Set  $T_i(W_{max})=BFD(i, W_{max})$ ,
       $T_i(1)=BFD(i, 1)$ 
5.   calculate  $T_i=T_i(W_{max})+(pc / 100) \times (T_i(1)-T_i(W_{max}))$ ;
6.   Set  $w_{p_i}=w$ ,
      such that  $T_i(w)-T_i$  is minimum
7.   Calculate highest Pareto-optimal width  $wh$ ;
8.   IF  $w_{p_i}-wh \leq dc$ 
9.     Set  $w_{p_i}=wh$ ,
        $T_{p_i}=BFD(i, w_{p_i})$ ,
10.  END
11. END
12. FOR  $i=0$  to  $g$ 
13.   Set  $w_i=w_{p_i}$ ,  $h_i=T_{p_i}$ 
14. END
    
```

Figure4 . Initialization for rectangle

respectively equal to the preferred TAM width and test time of the core  $i$ .

A set of rectangles for core test can now be constructed and the test scheduling is transformed into a process of packing, the goal of optimization is to pack all rectangles into a bin of fixed width ( $W_{max}$ ) with using a smallest height of the bin, namely using the shortest testing time.

#### IV. CO-OPTIMIZATION BASED ON HGA

In this paper, hybrid genetic algorithm<sup>[7]</sup> is used to solve the problem of test scheduling. First, the search ability of genetic algorithm is used to find the order in which IP cores are connected to the bus, namely the order in which rectangles are filled into the box. Then fitness value of chromosome is calculated by the method described in section 4.1. After several iterative operations, the best result of HGA is chosen as the solution for test scheduling problem.

Construction of hybrid genetic algorithm is divided into two parts: design of fitness function and design of genetic operators.

##### A. Fitness Function

In this paper, the value of fitness function of chromosome is the height of the box that has been filled, namely the testing time of SOC. The flexible-width TAM architecture is combined with HR algorithm to calculate the value of fitness function.

The rule of HR algorithm is as follows<sup>[6]</sup>:

(1). Create a bounded space. First, rectangles are sorted in the order that generated by GA. Then the rectangle which ranks first is filled into the box. At this point a bounded space  $S$  is created and this space is seen as the current sub-space waiting for filling.

(2).Pack a rectangle into the bounded space, then the space is divided into two subspaces Fig. 6).

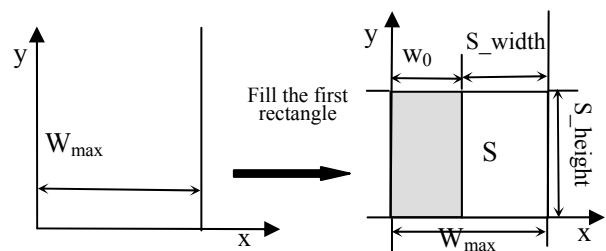


Figure 5. Create a bounded space

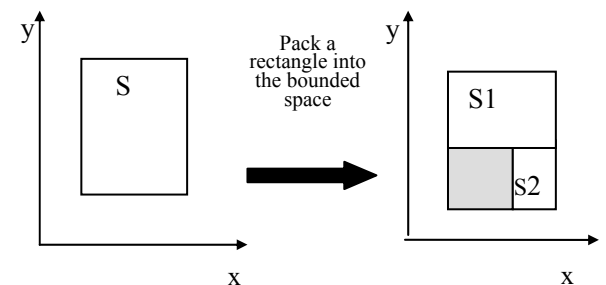


Figure 6. Two subspaces

- (3). Pack each subspace by the method in (2) recursively.
- (4). Create a new space for the remaining rectangles ( Fig. 7).

If all the sub-space has been filled while there are still rectangles remained, a new bounded space will be created with the method proposed in (1) for the remain rectangles, then the remaining rectangles will be packed into the new space by the method proposed in (2) and (3).

Some spaces remain blank because they are too small to be filled with any rectangle during packing. That results in a waste of space. The following method is used to reduce waste:

(1). Insert rectangle to fill idle<sup>[1]</sup>. Width and height of rectangles can be adjusted because flexible-width TAM architecture is adopted. At this time a rectangle is selected to be inserted into the gap after being assigned a new width, then the waste of space can be reduced. In practice, the width of the gap corresponds to the width of the idle bus, and the height of the gap corresponds to the idle time of the bus. The preferred TAM width that selected core obtained before is replaced by the width of idle bus, then this IP core can be connected to the idle bus. After being connected to the bus, the testing time of the selected core will be calculated by BFD algorithm.

Let Q be the set of IP cores meeting the condition  $dif_i > 0$ , the selected core should meet the following condition:

$$dif_{selected} < dif_j \quad (selected \in Q, j \in Q, selected \neq j) \quad (1)$$

Where:

idle\_t --- idle time of bus;

new\_t<sub>i</sub> --- the testing time of core i after core i being connected to bus;

dif<sub>i</sub> ---  $dif_i = idle_t - new_t_i$ ;

(2). Widen rectangle to fill idle<sup>[1]</sup>. If the suitable rectangle can not be found in previous step, a filled rectangle will be selected to obtain an extra width of idle\_w, so that its height is decreased (figure 9). In practice, an IP core that have been connected to bus is selected to obtain an extra width of idle\_w, then test time of this IP core will be shortened. After obtaining an extra width, the testing time of the selected core will be calculated by BFD algorithm.

Let P be the set of IP cores meeting the condition that  $begin_i = idle\_begin$ , the selected core should meet the following condition:

$$D_{sel} > D_j \quad (sel \in P, j \in P, selected \neq j) \quad (2)$$

Where:

idle\_begin --- time when bus enters idle status;

begin<sub>i</sub> --- time when the test of core i begin;

orig\_t<sub>i</sub> --- testing time of core i before it obtaining the extra width;

new\_t<sub>i</sub> --- testing time of core i after it obtaining the extra width;

D<sub>i</sub> ---  $D_i = orig_t_i - new_t_i$ ;

combined With the two methods used to reduce waste, fitness function calculation process is shown in Fig 10. At the end of packing, fitness value will be returned to the

genetic algorithm. The SOC testing time is equal to the height of the box that has been filled.

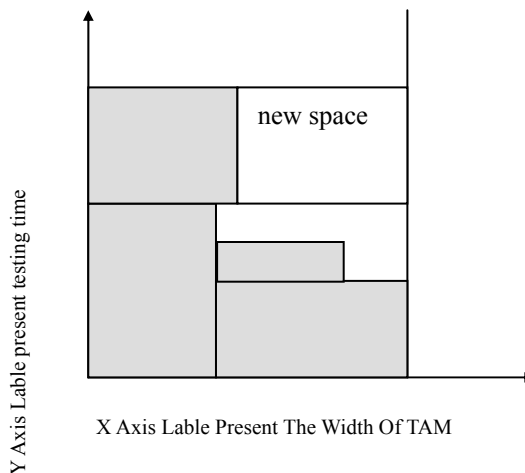


Figure7. Create a new space

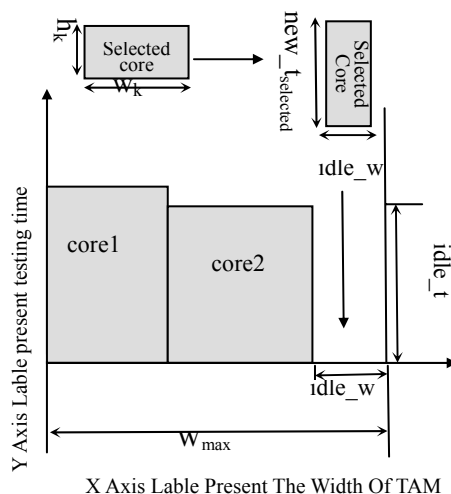


Figure8. Insert rectangle

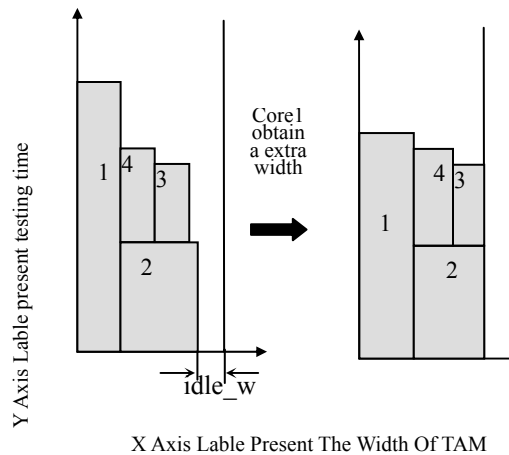


Figure.9 Widen a rectangle

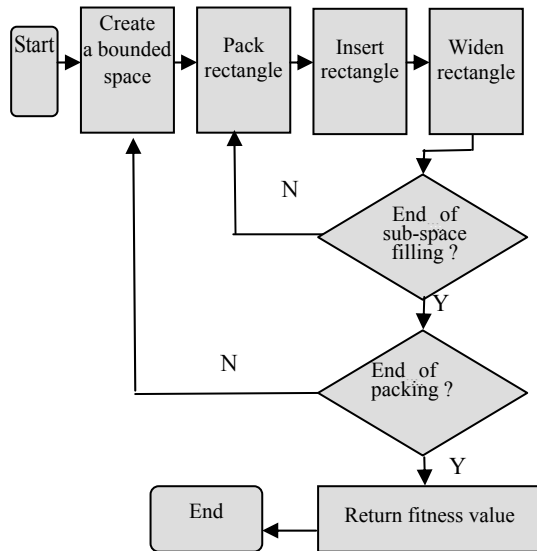


Figure 10. Fitness function

### B. Realization Of HGA

Genetic algorithm is a simulation of genetic selection and natural selection in biological evolution process. It is an algorithm with the function of “survival + detection”. All the individuals in a group can be seen as the object of GA, and an encoded parameter space efficiently searched. Selection, crossover, mutation constitute the genetic operation and parameter encoding, initial population creation, the design of fitness function, the design of genetic operations, settings the control parameter are the kernel of genetic algorithm. Genetic algorithm has been widely applied in various fields because of its strong robustness and search capability. However, GA has its demerits: It is good at global search but lack of capacity of local search, and it is prone to premature convergence. Studies have shown that Hybrid genetic algorithm, which is integrated by GA and other knowledge, can be a good solution for large-scale problems.

Hybrid genetic algorithm is adopted in this paper. It is based on real coding. The value of *i*th gene of the chromosome represents the test sequence of core *i*. Hybrid genetic algorithm is completed by using some genetic operators described in [7]. The fitness value is calculated by the method described in 4.1. Detailed process of HGA is as follows:

Step1: Population initialization. A sequence coded chromosome is used as the first chromosome. Two random breakpoints are selected in this chromosome. The genes on both sides of the breakpoints are exchanged. This method is used to get other chromosomes.

Step2: Calculate the fitness value of individuals of population.

Step3: Crossover. Two individuals are randomly selected from the old generation. Two new chromosomes are generated from the couple of selected chromosomes by using the method of single-point crossover. This operation was executed  $POPSIZE / 2$  times to get the new generation.

Step4: Mutation. For each chromosome, two random

breakpoints are selected and the genes between these two breakpoints are swapped. If the fitness value of the new chromosome is lower than the old chromosome, the new chromosome is accepted as the offspring; otherwise the old chromosome is accepted as the offspring.

Step5: Optimal preservation strategy. Chromosome with the highest fitness value in new generation is replaced by the chromosome with the lowest fitness value in old generation.

Step6: If the number of iterations reaches MAXG, return to step2; otherwise, the shortest time is returned and the corresponding chromosome is selected as the solution of co-optimization.

MAXGENS is the largest number of iterations, POPSIZE is the population size, PXOVER is the crossover probability. In this paper, MAXG=100~400, POPSIZE=50, PXOVER=0.8.

## V. EXPERIMENTAL RESULTS

Our algorithm is implemented in C, and experiments are conducted on Intel Pentium IV, 2.5G Hz processor with 256 MB memory.

In this section, experimental results are presented. These results are for three benchmark SOCs: d695, p93791, p22810. d695 is an academic benchmark which contains 3 memory cores and 8 scan-testable logic cores. p93791 is the largest example SOC which contains 18 memory cores and 14 scan-testable logic cores. SOC p22810 contains 6 memory cores and 22 scan-testable logic cores.

All possible integer values of the parameters  $pc$  and  $dc$  in the range  $0 < pc < 10$ ,  $0 < dc < 4$  are considered and the best results are shown in tables in appendixes.

In the tables, the total TAM width is proposed in the first column, the SOC test time with using our method is proposed in the second column, the SOC test time with using the method in [1,4,5] is proposed respectively in the 3th, 5th, 7th column. The percentage difference between testing time obtained by our method and other methods is calculated by the formula  $\Delta T = (T_a - T_{new}) / T_a$ ,  $T_a$  represents the SOC testing time with using other method and the  $T_{new}$  represents the SOC testing time obtained by our method.

The best results for d695 and 93791 were obtained by GA and Quantum-inspired evolutionary algorithm (QEA) when the bus is divided into three parts, these results are made to compare with our method. For p22810, related literature gives the results only with the bus width partition two, so the comparison between our results and the GA results with the bus width partitions two is made. The best QEA results are obtained when the bus is divided into four parts, these results are made to compare with our method.

It can be seen from the tables that compared with the method which uses the flexible-width TAM architecture<sup>[1]</sup>, the new method proposed in this paper obtains lower testing time in almost all instances. This shows the advantage of hybrid genetic algorithm compared with ILP algorithm. Compared with the GA algorithm and the QEA algorithm, our method obtains lower testing time in most

instances for d695 and p22810. However, for p93791, our method obtain comparable or higher testing time. A careful Analysis reveals reasons for this poor result:

(1). It is difficult to optimize the system testing time with using only a single value of pc and dc for all cores.

(2). The scale of the problem is so large that the algorithms need long computation time, so the iteration number is set to 100, this results that genetic manipulation has been terminated before the best solution obtained. To solve the problem related to these reasons is the direction for further work. The control parameters pc and dc should be distributed for each core According to specific circumstances of the core. The computing time of hybrid genetic algorithm should be reduced and iteration number should be increased. And different selection probability a mutation probability should be tried.

The pie chart in Fig. shows the result after statistics all experimental data. Each part of the pie represents the proportion that one method obtains the best result among 4 methods. For example, In the 21 experiments, Our method 7 times obtains the best results compare with other method and accounts for 1 / 3 in all the results, the proportion corresponds to the black part in the pie chart.

It can be seen from the pie chart that QEA obtains best result in most cases, followed by our method. After the observation, the reasons why QEA performances so great advantages is found: our method obtain higher testing time for p93791. Reasons for the poor results and improved methods for this flaw have been described above and our methods will be improved to achieve better results in future work.

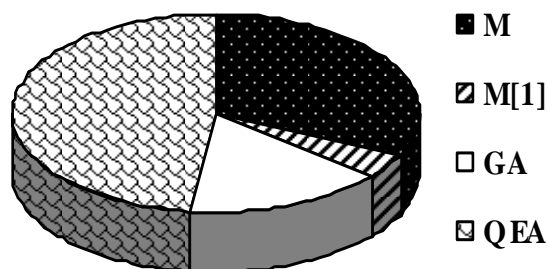


Figure 11 statistics result

## VI. CONCLUSION

A new co-optimization method for SOC test is proposed in this paper. In the new method Wrapper/TAM optimization and test scheduling problem are transformed into rectangle packing problem. And then hybrid genetic algorithm is used to search the best option. At last our co-optimization technique is applied to an academic benchmark SOC and two industrial SOCs. Compared with other method, lower test time can be obtained by our method in. A careful analysis reveals the reasons for the poor results and to solve the problem related to these reasons is the direction for our further work.

## REFERENCES

- [1] V.Iyengar, K.Chakrabarty, and E.J.Marinissen. "On using rectangle packing for SOC wrapper/TAM co-optimization," Proceeding 20th IEEE VLSI Test Symposium(VTS 2002). 2002, pp:253-258.
- [2] V.Iyengar, K.Chakrabarty and E.J. Marinissen. "Efficient Wrapper/TAM Co-Optimization for Large SOCs,"Proceedings 2002 Design, Automation and Test in Europe Conference and Exhibition(DATE'02). 2002, pp:491-498.
- [3] V.Iyengar, K.Chakrabarty, and E.J.Marinissen. "Wrapper/TAM co-optimization, constraint-driven test scheduling, and tester data volume reduction for SOCs," Proceedings 39th Design Automation Conference,2002, pp:685-690.
- [4] S.Chattopadhyay, K.Sudarsana Reddy. "Genetic Algorithm based Test Scheduling and Test Access Mechanism Design for System-on-Chips," Proceedings 16th International Conference on VLSI Design concurrently with 2nd International Conference on Embedded Systems Design. 2003, pp:341-346.
- [5] Xu Chuan pei, Wang Zheng, Li Zhi. "Optimization of SoC test architecture based on quantum-inspired evolutionary algorithm," Chinese Journal of Scientific Instrument. vol.28, No.10, 2007, pp: 1792-1798.
- [6] Zhang Defu, Kang Yan, Deng Ansheng. "A new heuristic recursive algorithm for the strip rectangular packing problem," Computers&Operations Research, vol.33 , No.8, 2006,pp: 2209-2217.
- [7] Chen Sheng da, Zhang De fu, Liu Yanjuan. "An Evolutionary Approximation Algorithm for Strip Rectangular Packing Problem," Computer Engineering. vol.33, No.9, 2007, pp:189-193.
- [8] Y.Zorian, E.J.Marinissen and S.Dey. "Testing Embedded-Core Based System Chips," Computer. Vol.32, No.6, 1999,pp:52-60.
- [9] E.J.Marinissen, et al. "On IEEE P1500's standard for embedded core test," Journal of Electronic Testing: Theory and Applications. Vol.18, No.4, 2002,pp:365-383.
- [10] IEEE P1500 Web Site. <http://grouper.ieee.org/groups/1500>
- [11] T.J.Chakraborty, S.Bhawmik and C.-H.Chiang. "Test access methodology for system-on-chip testing," Proc. Int. Workshop on Testing Embedded Core-Based System Chips, 2000, pp.1.1-1~1.1-7.
- [12] Krishnendu Chakrabarty. "Design of System-on-a-Chip Test Access Architectures using Integer Linear Programming," Proceedings of the 37<sup>th</sup> Annual Design Automation Conference, 2000, pp:432- 437.
- [13] Xu Chuan-pei, DAI Kui, MA Li. "Quantum-inspired evolutionary algorithm based on optimization of hierarchical SOC test architecture," Computer Engineering and Applications. vol.44, No.14, 2008, pp: 96-99.
- [14] Liang Xu , Li Xingshan, Yu Jinsong. "Research on the task schedule algorithm based on the GA," Journal of Electronic Measurement and Instrument. vol.23, No.2, 2009, pp: 19-24.
- [15] E. J. Marinissen et al. "A structured and scalable mechanism for test access to embedded reusable cores," Proc. Int. Test Conf., 1998, pp. 284-293.
- [16] K. Chakrabarty. "Optimal test access architectures for system-on-a-chip," ACMTrans. Design Automation of Electronic Systems, vol. 6,January 2001, , pp. 26-49.
- [17] K.Chakrabarty, V. Iyengar and M. D. Krasniewski. "Test planning for Modular Testing of Hierarchical SOCs," In IEEE Transactions on CAD of Integrated Circuits and

Systems, Vol. 24, No. 3, march 2005.

[18] P.Varma, B.Bhatia. "A structured test re-use methodology for core-based system chips," Proc. Int. Test Conf. 1998,pp:294-302.

[19] T.Ono, K.Wakui, H.Hikima, Y.Nakamura, M.Yoshida. "Integrated and automated design-for-testability implementation for cell-based Ics," Proc. Asian Test Symposium, 1997, pp:122-125.

[20] J. Aerts and E. J. Marinissen. "Scan chain design for test time reduction in core-based Ics," Proc. Int. Test Conf., 1998, pp. 448-457.

[21] J.Pouget, E.Larsson et al. "An Efficient Approach to SoC Wrapper Design, TAM Configuration and Test Scheduling," Proc.Eighth IEEE European Test Workshop(ETW'03),2003,pp:51-56.

[22] S.K. Goel and E.J. Marinissen. "Cluster-based Test Architecture Design for System-on-chip," Proc.20th IEEE VLSI Test Symposium(VTS 2002). 2002, pp: 259-264.

[23] S.Chattopadhyay, K.Sudarsana Reddy. "Genetic Algorithm based Test Scheduling and Test Access Mechanism Design for System-on-Chips," 16<sup>th</sup> International Conference on VLSI Design (VLSI'03), 2003, pp: 341-346.

[24] K.H.Han, J.H.Kim. "Quantum-Inspired Evolutionary Algorithm for a Class of Combinatorial Optimization," IEEE Transactions on Evolutionary computing, 2002, pp: 580-593.

[25] E. Larsson and Z. Peng. "An integrated system-on-chip test framework. Proc. Design Automation Test Eur., 2001, pp. 138-144.

[26] M. Benabdenbi, W. Maroufi, and M. Marzouki. "CAS-BUS: A test access mechanism and a toolbox environment for core-based system chip testing," J. Electron. Testing: Theory Applicat., vol. 18, 2002, pp. 455-473.

[27] I. Parulkar, T. Ziaja, R. Pendurkar, A. D'Souza, and A. Majumdar, "A scalable, low cost design-for-test architecture for UltraSPARC chip multi-processors," Proc. Int. Test Conf., 2002, pp. 726-735.

[28] W. Zou, S. M. Reddy, I. Pomeranz, and Y. Huang, "SOC test scheduling using simulated annealing," Proc. VLSI Test Symp., 2002, pp.325-330.

[29] R. K. Gupta and Y. Zorian. "Introducing core-based system design. IEEE Design and Test of Computers,1997,pp:15-25.



**Xu Chuanpei:** Ms., born in 1968, a doctorate earned from Xi'an University of Electronic Science in 2006.

She is a professor of Guilin University of Electronic Science and Technology. Her research interests are in the area of IC testing theory and technology. She has published over 30 articles in some important magazines,e.g., Acta Electronica

Sinica, Chinese Journal of Scientific Instrument, Journal of Electronics & Information Technology.

Prof.Xu is a member of IEEE.



**Lu Xueyun:** Ms., born in 1984, a bachelor's degree earned from Guilin University of Electronic Science and Technology in 2007.

She is a graduate student of Guilin University of Electronic Science and Technology. Her research interests are in the area of integrated circuit test.



**Hu Cong:** Mr., born in 1981, master's degree earned from Guilin University of Electronic Science and Technology in 2006.

He is an instructor of Guilin University of Electronic Science and Technology His research interests are in the area of Software Testing and validation. He has published several articles in some important magazines, e.g., Microcomputer Information, and Electronic Instrumentation Customer.

APPENDIX EXPERIMENTAL RESULTS FOR 3 SOCS

TABLE 1. experimental results for d695

W <sub>max</sub>	T <sub>new</sub>	T[1]	ΔT	GA	ΔT	QEA	ΔT
16	42703	43723	-2.332 %	42268	+1.029%	42268	+1.029%
24	29432	30317	-2.919%	28389	+3.674%	28292	+4.029%
32	21523	23021	-6.507%	21518	+0.023%	21518	+0.023%
40	18380	18459	-0.428%	17677	+3.977%	17677	+3.977%
48	15493	15698	-1.305%	16975	-8.730%	16975	-8.730%
56	13407	13415	-0.059%	13795	-2.812%	13207	+1.514%
64	11978	11604	+3.223%	12941	-10.332%	12941	-10.332%

TABLE 2. Experimental results for p93791

$W_{max}$	Tnew	T[1]	$\Delta T$	GA	$\Delta T$	QEA	$\Delta T$
16	1842658	1851135	-0.458%	1751423	+5.209%	1743162	+5.708%
24	1244061	1248795	-0.379%	1180948	+5.344%	1173202	+6.040%
32	948601	975016	-2.709%	893313	+6.189%	877066	+8.156%
40	754843	794020	-4.934%	723400	+4.347%	705524	+6.990%
48	593328	627934	-5.511%	593841	-0.086%	590525	+0.475%
56	534997	568436	-5.883%	511001	+4.696%	508459	+5.219%
64	501661	511286	-1.883%	458148	+9.498%	444104	+12.960%

TABLE 3. Experimental results for p22810

$W_{max}$	Tnew	T[1]	$\Delta T$	GA	$\Delta T$	QEA	$\Delta T$
16	384082	452639	-15.146%	461348	-16.748%	438619	-12.434%
24	263303	307780	-14.451%	361326	-27.129%	298914	-11.913%
32	201320	246150	-18.212%	313891	-35.863%	230813	-12.778%
40	170162	197293	-13.752%	287260	-40.764	198184	-14.141%
48	151126	167256	-9.644%	268512	-43.717%	174496	-13.393
56	143325	145417	-1.439%	271429	-47.196%	151886	-5.636%
64	132170	136941	-3.484%	260645	-49.291%	137532	-3.899%