

DOI: 10.13382/j.jemi.2017.08.003

基于正弦余弦算法的 NoC 测试规划研究*

朱望纯¹ 周甜¹ 胡聪^{1,2} 许川佩¹ 朱爱军¹

(1. 桂林电子科技大学电子工程与自动化学院 桂林 541004; 2. 西安电子科技大学机电工程学院 西安 710071)

摘要:如何实现多约束条件下测试时间优化是目前片上网络(NoC)测试中亟待解决的问题。提出一种基于正弦余弦算法(SCA)的NoC测试规划优化方法。采用专用TAM的并行测试方法,在满足功耗、引脚约束的条件下,建立测试规划模型,对NoC进行测试。通过群体围绕最优解进行正弦、余弦的波动,以及多个随机算子和自适应变量进行寻优,达到最小化测试时间的目的。在ITC'02 test benchmarks测试集上进行对比实验,结果表明相比粒子群优化(PSO)算法,提出的算法能够获得更短的测试时间。

关键词:片上网络;测试规划;正弦余弦算法;优化

中图分类号: TN47 **文献标识码:** A **国家标准学科分类代码:** 510.3040

Test scheduling research for network-on-chip based on sine cosine algorithm

Zhu Wangchun¹ Zhou Tian¹ Hu Cong^{1,2} Xu Chuanpei¹ Zhu Aijun¹

(1. School of Electronic Engineering and Automation, Guilin University of Electronic Technology, Guilin 541004, China;
2. School of Mechano-Electronic Engineering, Xidian University, Xi'an 710071, China)

Abstract: How to optimize the test time under multiple constraints is an urgent problem to be solved in the network-on-chip (NoC) testing. An optimization method of NoC test scheduling based on sine cosine algorithm (SCA) is proposed. A parallel test method using dedicated test access mechanism (TAM) is adopted, and a test scheduling model for NoC is built to satisfy the power consumption and pin constraints. To achieve test time minimization, the population fluctuation with the sine and the cosine function around the optimal solution, and a group of random operators and adaptive variables are adopted. Comparing experiments on the ITC'02 test benchmarks test show that the proposed algorithm can achieve shorter test time than that of the particle swarm optimization (PSO) algorithm.

Keywords: Network-on-Chip; test scheduling; sine cosine algorithm; optimization

0 引 言

为了解决快速上市的问题,芯片生产厂商都采用在单个芯片上集成大量的晶体管和重用IP(intellectual property)核。然而,当SoC(system-on-chip)变得越来越稠密,传统的基于总线结构的SoC受到了很多限制,如全局时钟同步、通信带宽、扩展性以及性能^[1]。为了消除总线结构带来的限制,片上网络(NoC)设计规范应运而生。一个典型的基于包交换互联结构NoC系统,包含IP核、

路由器、资源接口和互联网^[2]。然而,相比于传统SoC,测试NoC系统的遇到了新的挑战^[3-5]。

像传统的基于总线的SoC一样,NoC系统的测试问题包括测试壳设计,测试访问机制(test access mechanism,TAM)结构设计和测试规划方法。根据所用TAM的方式可分成两类,一类是在重用NoC作为TAM的方式进行的,这一类是很多研究人员专注的领域。Cota等人^[6]首先提出了在测试中重用NoC固有网络作为TAM的方法。通过网络重用,测试时间更短了,测试管脚和面积也相应的减小了。所有的冲突在调度过程中

收稿日期:2017-07 Received Date: 2017-07

*基金项目:国家自然科学基金(61561012, 61662018)、广西自动检测技术与仪器重点实验室(YQ16106)、广西高校科学技术研究项目(KY2015YB110)、广西中青年骨干教师基础能力提升项目(2017KY0210)资助

被静态的解决。Richter 等人^[3]采用基于核调度和测试管脚分配给测试访问点的协同优化,通过采用整型线性规划进行 NoC 测试规划。Hu 等人^[7]提出 XY 方向连通子图划分的方法,同时在功耗约束下融合多时钟策略达到测试时间最小化目的。另一类是在专用 TAM 下进行的, Benabdenbi 等人^[8]、Marinissen 等人^[9]和 Iyengar 等人^[10]专注传统 SoC 的针对嵌入式 IP 核测试结构的设计,即测试壳和 TAM 的设计。Larsson 等人^[11]提出一种考虑功耗的可配置测试壳设计,并把它应用到测试规划中,使得测试规划中保证了功耗约束。Chattopadhyay 等人^[12]把遗传算法(genetic algorithm, GA)应用到了测试规划中,相比于整型线性规划(integer linear programming, ILP)取得了更少的 CPU 运行时间。

综上所述, NoC 测试规划在减少测试时间, 提高测试效率中起来关键性的作用, 因此, 如何对 NoC 进行高效经济的测试规划是目前亟待解决的问题。NoC 测试规划问题, 被证明为 NP 完全问题^[3]。

近年来, 具有群智能特征的算法在 NP 完全问题上显示出优越的求解性能, 受到人们越来越多的关注。正弦余弦算法(multi-verse optimizer, SCA)是 Mirjalili^[13]最新提出的一个智能优化算法。通过群体围绕最优解进行正弦、余弦的波动, 以及多个随机算子和自适应变量进行寻优。适于解决组合优化的 NP 完全问题。因此本文提出一种基于正弦余弦算法的 NoC 测试规划方法, 在满足测试引脚约束和测试功耗约束的情况下最小化测试时间。

1 问题描述

本文采用文献[10]提出的测试壳设计方法和专用 TAM。NoC 测试规划问题可描述如下: 在一个 NoC 系统中, 给定核参数集 Co , 每个核在相应的 TAM 宽度下有一个测试时间 $T(c)$, 为 NoC 提供的最大的测试引脚数为 Pin_{max} , NoC 最大功耗约束为 P_{max} , 确定一个测试规划方法, 达到符合 Pin_{max} 约束; 符合 P_{max} 约束; 总的测试时间最小。

嵌入式核在同一条 TAM 上是顺序测试, 在不同的 TAM 上是并行测试。总的测试时间是所有并行测试的 TAM 测试耗时最大的测试时间。

引入一个二进制变量 y_{ij} ($1 \leq i \leq k$ and $1 \leq j \leq m$), 用来确定 NoC 中核到 TAM 的分配。系统中的每一个核都仅分配到一个 TAM 上。

y_{ij} 定义如下:

$$y_{ij} = \begin{cases} 1, & \text{core } i \text{ is assigned to TAM } j \\ 0, & \text{其他} \end{cases}$$

$$\sum_{j=1}^m y_{ij} = 1, 1 \leq i \leq k \quad (1)$$

式中: m 是 TAM 的数量, k 为被测 IP 核数量。测试完 TAM j 上所有的核需要的时间为:

$$\sum_{i=1}^k T_i(w_j) y_{ij} \quad (2)$$

由于所有的 TAM 是并行测试的, 总的测试时间为:

$$T_{sum} = \max_{1 \leq j \leq m} \sum_{i=1}^k T_i(w_j) y_{ij} \quad (3)$$

核本身的测试时间和测试数据传输带宽有关。假设核 i 分配的 TAM 带宽为 w , 那么测试时间 $T_i(w_j)$ 为:

$$T_i(w_j) = \{1 + \max(S_{in}, S_{out})\} \cdot np + \min(S_{in} + S_{out}) \quad (4)$$

式中: S_{in} (S_{out}) 表示一个核的最长 Wrapper 扫入扫描链 (扫出扫描链) 的长度, np 表示测试矢量的个数。 $T_i(w_j)$ 的值通过 Wrapper 设计的 BFD (best fit decreasing) 算法计算所得^[10]。

在整个测试过程中, 被测核使用的总测试引脚不能超过 Pin_{max} 。引入在时间槽 t 中实际使用的测试引脚变量 Pin'_{used} , 其定义如下:

$$Pin'_{used} = \sum_{i=1}^k Pin_i \cdot \lambda_i^t \leq Pin_{max} \quad (5)$$

式中: Pin_{max} 是测试中最大可用测试引脚数。 λ_i^t 定义为:

$$\lambda_i^t = \begin{cases} 1, & TS_i \leq t \leq TE_i \\ 0, & \text{其他} \end{cases} \quad (6)$$

式中: TS_i 和 TE_i 是分别为核 i 的测试开始时间和测试结束时间。

尽管增加 TAM 的数量可以有效的缩短测试时间, 提高测试效率, 但是会导致测试功耗的增加。因此, 为了保证测试的可行性, 在测试过程中必须满足功耗约束。

在任意的时间槽 t , 功耗必须满足:

$$P'_m = \sum_{i=1}^k P_{testi} \cdot \lambda_i^t \leq P_{max} \quad (7)$$

式中: P_{testi} 是核 i 的功耗, P_{max} 是系统允许最大功耗。因此, NoC 测试规划可如下描述:

$$\begin{aligned} \min T_{sum} &= \max_{1 \leq j \leq m} \sum_{i=1}^k T_i(w_j) y_{ij} \\ \text{s. t. } &\sum_{i=1}^k P_{testi} \cdot \lambda_i^t \leq P_{max} \\ &\sum_{i=1}^k Pin_i \cdot \lambda_i^t \leq Pin_{max} \end{aligned} \quad (8)$$

2 SCA 算法

在 SCA 算法中, 位置更新如下:

$$X_i^{t+1} = \begin{cases} X_i^t + r_1 \times \sin(r_2) \times |r_3 W_i^t - X_i^t|, & r_4 < 0.5 \\ X_i^t + r_1 \times \cos(r_2) \times |r_3 W_i^t - X_i^t|, & r_4 \geq 0.5 \end{cases} \quad (9)$$

式中: X_i^t 是当前解在第 i 维第 t 代的位置, $r_1/r_2/r_3$ 是随机数, W_i 是目标点在第 i 维的位置, $|\cdot|$ 表示绝对值。
 $r_4 \in [0,1]$ 是随机数。

SCA 算法中,参数 r_1 表示下一个位置区域在解和目标之内或者之外。参数 r_2 定义为朝向或者远离目标多远。参数 r_3 为目标值给出一个随机权值,为了随机强调 ($r_3 > 1$) 或者忽略 ($r_3 < 1$) 目标值在定义距离的效果。参数 r_4 平等地切换正弦和余弦函数。

由于使用了正弦和余弦函数,所以此算法叫正弦余弦算法。正弦和余弦函数的周期模式允许一个解围绕另外一个解重新定位。这种机制能够保证两个解之间的空间开发能力。为了探索解空间,解应该能够搜索到相应目标值的外部。这可以通过改变正弦和余弦函数的范围实现。一个在 $[-2,2]$ 的正弦和余弦函数的概念模型如图 1 所示。图中显示如何改变正弦和余弦函数范围,需要一个解在自身和另外解的搜索空间的外面或者里面,去更新它的位置。随机定位在里面或者在外面是通过为式(9)的 $r_2 \in [0,2\pi]$ 定义的随机数获得。

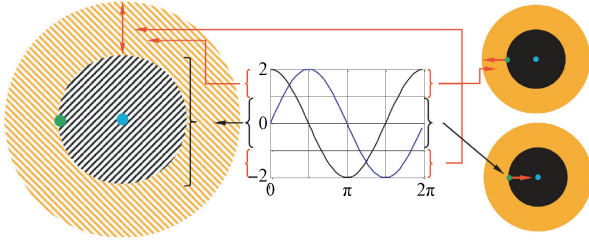


图 1 在 $[-2,2]$ 允许一个解围绕或远离目标值

Fig. 1 Sine and cosine with the range in $[-2,2]$ allowing a solution to go around or beyond the destination

为了平衡算法开发和探索能力,式(9)中正弦和余弦的范围按照式(10)自适应的改变。

$$r_1 = a - t \frac{a}{G} \tag{10}$$

式中: t 是当前迭代次数, G 是最大迭代次数, a 是一个常数。

SCA 算法开始优化过程也是从随机解集开始。然后,算法保存迄今最优解,并把它赋值给目标点,接着更新跟它相关的其他解。同时,为了强调开发能力随着迭代次数的增加,更新正弦和余弦函数的范围。当达到最大迭代次数时,SCA 算法终止。当然也可以设定其他终止条件,比如函数评估次数或者全局最优解精度。

- 1) 初始化种群 X
- 2) do
 - 根据目标函数评估每一个解
 - 更新迄今最优解
 - 更新 r_1, r_2, r_3, r_4
 - 根据式(9)更新解的位置

while($t <$ 最大迭代次数)

3) 返回迄今全局最优值

3 基于正弦余弦算法的 NoC 测试规划研究

3.1 算法编码

定义 1(观测个体编码) $\forall H_z \in OV^k, z \in [1, n], n$ 种群大小, k 是适应度变量的维数, H_i 是一个观测个体, $H_z = (OV_1, OV_2, \dots, OV_k)$ 是一个问题的可行解,代表 IP 核分配到 TAM 的一种分配方案。

定义 2(观测变量编码) 对于观测变量 $OV, \forall OV_j, j \in [1, k], OV_j \in C, C$ 是一个整数集。定义 $C = \{1, 2, \dots, m\}, m$ 是 TAM 的数量。

假定有 k 个被测 IP 核, m 条 TAM, 个体编码表示的一种测试数据分配方案。每一个基因位有 m 种状态,代表哪条 TAM。假定在测 IP 核数量 $k = 11$, TAM 的数量 $m = 3$, 个体为 ‘01022120102’ 意思是核 1, 3, 8, 10 分配到了 TAM0, 2, 6, 9 分配到了 TAM1, 4, 5, 7, 11 分配到了 TAM2。

初始群体生成公式如下:

$$X_{ij} = (Ub_{ij} - Lb_{ij}) \cdot \text{rand}(0,1) + Lb_{ij} \tag{11}$$

式中: $i = 1, 2, \dots, n, j = 1, 2, \dots, m, Ub_{ij}$ 表示第 i 个个体的第 j 维的上界, Lb_{ij} 表示第 i 个个体的第 j 维的下界, $\text{rand}(0,1)$ 表示一个 $(0,1)$ 的随机数。

3.2 算法描述

基于正弦余弦算法的 NoC 测试规划研究描述如下:

- 1) 根据 IP 核的数量, 设置解空间的维数 k ; 设置 TAM 的数量 m , 设置 NoC 功耗约束, 测试引脚约束; 设置种群大小 n ; 设置最大迭代次数 G , 设置自适应常数 a 。
- 2) 利用式(11) 在可行解空间内生成规模为 m 的初始种群。
- 3) 根据式(3) 评估每一个解, 并保存最优解。
- 4) 更新迄今最优解。
- 5) 更新 r_1, r_2, r_3, r_4 。
- 6) 根据式(9) 更新解的位置。
- 7) 判断迭代次数是否达到 G ? 没有达到则 $t = t + 1$, 转步骤 3)。
- 8) 输出最优解和最优目标函数值。

4 实验结果

为了验证本算法的有效性, 实验中, 采用 3 个国际基准电路 ITC '02 test benchmarks^[14] 的 3 个系统: d695, p22810 和 p93791 (表 1)。在 ITC '02 中, d695 是规模最小的系统, p93791 是规模最大的系统。

表 1 基准电路的基本信息

Table 1 Basic information of benchmarks

| 基准电路 | 核数量 |
|--------|-----|
| d695 | 10 |
| p22810 | 28 |
| p93791 | 32 |

为了验证测试结果,让 SCA 和粒子群优化算法 (PSO)^[15] 进行比较,同时为了公平比较,算法设置同样的种群大小(设置为 40)和同样的迭代次数(设置为 500)。运行每一个算法 40 次。算法的其他参数设置在表 2 中给出,这些设置在文献中经常使用。取算法在 d695, p22810 和 p93791 运行的最好结果,如表 3~5 所示。

表 2 算法的参数设置

Table 2 Parameter settings of algorithm

| 算法 | 调整参数 | 值 |
|-----|-------------------------|-----|
| SCA | a | 2 |
| | c1 (Cognitive constant) | 2 |
| PSO | c2 (Social constant) | 2 |
| | w (Inertia constant) | 0.6 |

表 3 不同测试引脚下 d695 的实验结果

Table 3 Experimental results for d695 with different test pins

| $P_{in_{max}}$ | $P_{max}/\%$ | 测试时间/cycles | |
|----------------|--------------|-------------|--------|
| | | SCA | PSO |
| 256 | 100 | 9 869 | 9 869 |
| 256 | 50 | 9 869 | 9 869 |
| 256 | 20 | 20 163 | 20 163 |
| 192 | 100 | 12 663 | 12 663 |
| 192 | 50 | 12 633 | 12 663 |
| 192 | 20 | 20 163 | 20 163 |
| 128 | 100 | 18 869 | 18 869 |
| 128 | 50 | 18 869 | 18 869 |
| 128 | 20 | 21 401 | 21 989 |

表 4 不同测试引脚下 p22810 的实验结果

Table 4 Experimental results for p22810 with different test pins

| $P_{in_{max}}$ | $P_{max}/\%$ | 测试时间/cycles | |
|----------------|--------------|-------------|---------|
| | | SCA | PSO |
| 192 | 100 | 180 962 | 181 992 |
| 192 | 50 | 181 056 | 185 915 |
| 192 | 20 | 181 432 | 183 919 |
| 128 | 100 | 271 331 | 271 474 |
| 128 | 50 | 271 335 | 271 504 |
| 128 | 20 | 271 336 | 272 542 |
| 256 | 100 | 137 712 | 146 221 |
| 256 | 50 | 137 572 | 143 047 |
| 256 | 20 | 137 294 | 145 362 |

表 5 不同测试引脚下 p93791 的实验结果

Table 5 Experimental results for p93791 with different test pins

| $P_{in_{max}}$ | $P_{max}/\%$ | 测试时间/cycles | |
|----------------|--------------|-------------|---------|
| | | SCA | PSO |
| 128 | 100 | 611 744 | 611 760 |
| 128 | 50 | 611 744 | 611 772 |
| 128 | 20 | 611 934 | 657 876 |
| 192 | 100 | 408 790 | 412 776 |
| 192 | 50 | 410 824 | 414 350 |
| 192 | 20 | 553 168 | 585 798 |
| 256 | 100 | 310 027 | 333 948 |
| 256 | 50 | 309 128 | 371 709 |
| 256 | 20 | 551 352 | 567 978 |

2 种算法中最短的测试时间用黑体标出。从表 3 可以看出,在 d695 上 4 种算法在大部分情况下获得了相同的结果,这是由于 d695 在测试基准上有最小的规模。然而,随着规模的增加(表 4 和 5),SCA 算法比 PSO 算法在每一测试项上都产生了最小的测试时间。这得益于 SCA 算法在返回值大于 1 或小于 -1 能探索不同区域的搜索空间,而在 $[-1, 1]$ 内能发掘有前景的搜索空间,并且正弦和余弦函数之间采用自适应范围可以使 SCA 的探索和发掘能力自然过渡,可以很好的平衡探索和发掘能力,使得它相比在得到解的质量方面更优越。

5 结 论

本文对 NoC 测试规划问题进行了描述,建立了 NoC 的测试规划的数学模型,提出了一种基于正弦余弦算法的 NoC 测试规划的方法。通过群体围绕最优解进行正弦、余弦的波动,以及多个随机算子和自适应变量进行寻优,达到最小化测试时间的目的。将实验结果与 PSO 测试方法进行对比,结果表明,本文方法能够取得更短的测试时间,提高了测试效率。

参考文献

- [1] LIU C, IYENGAR V, IYENGAR V, et al. Thermal-aware testing of network-on-chip using multiple-frequency clocking[C]. IEEE Vlsi Test Symposium, 2006.
- [2] 许川佩,姚芬,胡聪. 基于云进化算法的 NoC 资源节点优化测试研究[J]. 电子测量与仪器学报, 2012, 26(3): 192-196.
XU CH P, YAO F, HU C. Optimal test of NoC resource nodes based on cloud evolution algorithm[J]. Journal of Electronic Measurement and Instrument, 2012, 26(3): 192-196.
- [3] RICHTER M, CHAKRABARTY K. Optimization of test pin-count, test scheduling, and test access for NoC-based

- multicore SoCs [J]. IEEE Transactions on Computers, 2014, 63(3): 691-702.
- [4] 许川佩, 刘洋, 莫玮. 带分复用的三维片上网络测试规划研究[J]. 仪器仪表学报, 2015, 36(9): 2120-2128.
XU CH P, LIU Y, MO W. Research on test scheduling of three dimensional network-on-chip with bandwidth division multiplexing [J]. Chinese Journal of Scientific Instrument, 2015, 36(9): 2120-2128.
- [5] 许川佩, 凌景, 胡聪. 动态带分复用的三维片上网络协同优化研究[J]. 仪器仪表学报, 2016, 37(12): 2821-2828.
XU CH P, LING J, HU C. Research on collaborative optimization of three dimensional network-on-chip based on dynamic bandwidth division multiplexing [J]. Chinese Journal of Scientific Instrument, 2016, 37(12): 2821-2828.
- [6] COTA E, KREUTZ M, ZEFERINO C A, et al. The impact of NoC reuse on the testing of core-based systems[C]. Napa Valley, California, USA: IEEE, 2003.
- [7] HU C, LI Z, XU C, et al. Test scheduling for network-on-chip using XY-direction connected subgraph partition and multiple test clocks [J]. Journal of Electronic Testing: Theory and Applications, 2016, 32(1): 31-42.
- [8] BENABDENBI M, MAROUFI W, MARZOUKI M. CAS-bus: A test access mechanism and a toolbox environment for core-based system chip testing [J]. Journal of Electronic Testing: Theory and Applications, 2002, 18(4): 455-473.
- [9] MARINISSEN E J, KAPUR R, LOUSBERG M, et al. On IEEE P1500's standard for embedded core test[J]. Journal of Electronic Testing: Theory and Applications, 2002, 18(4): 365-383.
- [10] IYENGAR V, CHAKRABARTY K, MARINISSEN E J. Test wrapper and test access mechanism co-optimization for system-on-chip [J]. Journal of Electronic Testing: Theory and Applications, 2002, 18(2): 213-230.
- [11] LARSSON E, PENG Z. A reconfigurable power conscious core wrapper and its application to system-on-chip test scheduling [J]. Journal of Electronic Testing: Theory and Applications, 2008, 24(5): 497-504.
- [12] CHATTOPADHYAY S, REDDY K S. Genetic algorithm based test scheduling and test access mechanism design for system-on-chips [C]. IEEE International Conference on Vlsi Design, 2003.
- [13] MIRJALILI S. SCA: A sine cosine algorithm for solving optimization problems [J]. Knowledge-Based Systems, 2016, 96(3): 120-133.
- [14] MARINISSEN E J, IYENGAR V, CHAKRABARTY K. A set of benchmarks for modular testing of SoCs [C]. IEEE International Test Conference, 2002.
- [15] CLERC M, KENNEDY J. The particle swarm: Explosion, stability, and convergence in a multidimensional complex space [J]. IEEE Transactions on Evolutionary Computation, 2002, 6(1): 58-73.

作者简介



朱望纯, 分别在 2000 年和 2003 年于桂林电子科技大学分别获得学士学位和硕士学位, 现为桂林电子科技大学研究员, 主要研究方向为可测性设计、自动测试系统。

E-mail: zhwhc@guet.edu.cn

Zhu Wangchun received B. Sc. and M. Sc. both from Guilin University of Electronic and Technology in 2000 and 2003, respectively. Now he is a professor at Guilin University of Electronic and Technology. His main research interests include design for testability and automatic test system.



胡聪 (通讯作者), 分别在 2003 年和 2006 年于桂林电子科技大学分别获得学士学位和硕士学位, 2017 年于西安电子科技大学获得博士学位, 现为桂林电子科技大学副教授, 主要研究方向为集成电路测试、自动测试系统。

E-mail: hucong@guet.edu.cn

Hu Cong (Corresponding author) received B. Sc. and M. Sc. both from Guilin University of Electronic and Technology in 2003 and 2006, and Ph. D. from Xidian University in 2017, respectively. Now he is an associate professor at Guilin University of Electronic and Technology. His main research interests include integrated circuit test and automatic test system.