

基于粒子群算法的 3D NoC 测试优化方法

许川佩^{1,2}, 李克梅^{1,2}

(1 桂林电子科技大学 电子工程与自动化学院, 广西 桂林 541004;

2 广西自动检测技术与仪器重点实验室, 广西 桂林 541004)

摘 要: 三维片上网络(3D NoC)中资源内核的测试问题日益突出, 本文采用带分复用策略并结合改进离散粒子群算法对 IP 核测试数据分配及调度方案进行快速寻优。算法通过群体多样性的监控实现对粒子前行速度的动态调节, 增强全局探索能力, 并自适应调整惯性权重, 以加强粒子的局部开发能力, 进而改善粒子搜索的停滞现象。以 ITC02 测试标准电路作为实验对象, 仿真结果表明, 本文方法能有效地完成 3D NoC 资源内核的最大化并行测试, 减少了测试时间, 提高了资源利用率。

关键词: 三维片上网络; 带分复用; 离散粒子群算法; 测试优化

中图分类号: TP306

文献标识码: A

文章编号: 1000—7180(2017)09—0026—06

Optimization of 3D NoC Test Based on Particle Swarm Optimization

XU Chuan-pei^{1,2}, LI Ke-mei^{1,2}

(1 School of Electronic Engineering and Automation, Guilin University of Electronic Technology, Guilin 541004, China;

2 Guangxi Key Laboratory of Automatic Detection Technology and Instrument, Guilin 541004, China)

Abstract: The IP core test problem in three dimensional network-on-chip (3D NoC) is becoming increasingly obvious. In this paper, the improved discrete particle swarm optimization algorithm and a bandwidth division multiplexing method are used to optimal the test data assignment and scheduling for IP core. In order to enhance the global exploration ability and improve the stagnation phenomenon, the algorithm realizes the dynamic adjustment of the forward speed of particle by monitoring the population diversity, meanwhile, adjust the inertia weight adaptively to increase the local development ability. Taking ITC'02 standard circuit as the test object, experiment was conducted; and the experiment results show that the proposed method can effectively maximize the parallel test of 3D NoC, thus shortened the test time as well as improved the resource utilization.

Key words: 3D NoC; bandwidth division multiplexing; DPSO; test optimization

1 引言

随着三维集成电路(3D IC)技术的迅速发展, 三维片上网络(Three-Dimensional Network-on-Chip, 3D NoC)应运而生。3D NoC 基于三维封装技术将多个具有二维结构的芯片封装成一个芯片, 芯片间通过硅通孔技术(Through-Silicon-Via, TSV)实现互连, 具有集成度高、互连线短、可扩展性好、电路抗

噪声能力强等优势^[1]。然而, 随着芯片集成度的提高, 测试时间越来越长^[2]; 此外, 高测试功耗是片上网络测试面临的问题之一。在测试过程中, 多 IP 核同时工作, 使得测试功耗堆积, 引起芯片延时和串扰等问题, 从而降低测试芯片的可靠性, 甚至损坏。因此, 如何在带宽、功耗双重约束下, 对 IP 核进行合理地测试优化, 提高资源利用率, 缩短测试时间, 是 3D NoC 测试面临的一大挑战。

KARMAKA R. 等人^[3]结合粒子群算法,对TAM进行优化设计,从而优化了测试时间,却忽略了测试过程中的功耗约束问题. Mukesh A. 等人^[4]采用多播路由对基于NoC的同构多核SoCs的测试时间进行优化,利用ILP算法及动态规划有效地提高了测试效率. 文献^[5]通过限制TSV的数量,采用基于云模型的进化算法对3D NoC进行测试优化,既提高了TSV的利用率又缩短了测试时间,但没有充分利用TAM带宽.

本文采用重用NoC作为测试存取机制(Test Access Mechanism, TAM),结合带分复用测试策略对3D Mesh结构的NoC进行并行测试,并基于离散粒子群算法对3D NoC测试资源分配方案及IP核的调度顺序进行研究,以提高资源利用率,优化测试时间,同时,探讨功耗约束对测试时间的影响.

2 问题描述

本文针对规则的3D Mesh结构NoC开展测试优化方法研究,3D Mesh体系结构如图1所示,包括资源节点、路由节点、互连线和TSV^[2].

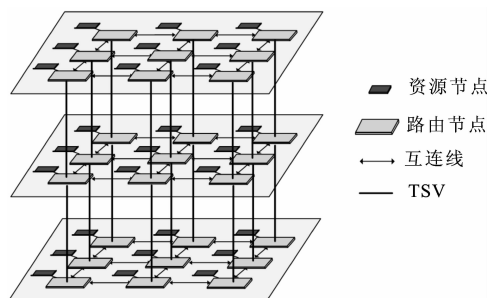


图1 3D Mesh结构NoC

在并行测试过程中,多个IP核同时测试,系统测试功耗增大,造成器件的损坏. 因此,本文研究的测试优化问题可描述为:给定 N_c 个待测核Core, B 条固定带宽的TAM,研究如何将各IP核的测试数据合理有效地分配至各条TAM,并且在功耗、带宽约束下,无冲突地调度各IP核的测试顺序,以实现IP核的最大化并行测试,使得总测试时间最短,可将问题表示为如图2所示的三维装箱问题.

针对这一NP难问题,须探求一种快速而有效的算法来完成测试. 粒子群算法(Particle Swarm Optimization algorithm, PSO)是一种智能优化算法,该算法基于迭代模式,具有大范围全局搜索能力、收敛速度快以及鲁棒性好等优点,本文将其应用在3D NoC测试优化中,以实现测试资源的合理分配.

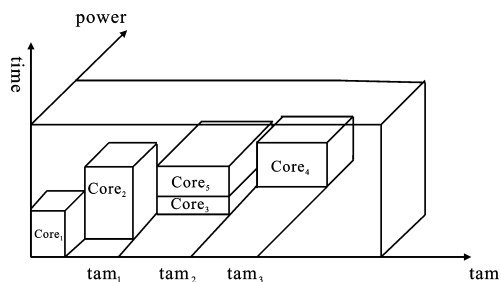


图2 3D NoC测试优化问题

3 3D NoC测试优化分析

3.1 测试策略

3D NoC数据吞吐量大,支持大量数据的传输,为并行测试提供了有利条件;为了节省并行测试过程中的硬件开销,本文采用重用片上网络的测试访问机制^[6],即复用NoC的路由节点和通信链路等资源作为TAM,以节省硬件开销. 同时,为了充分利用TAM带宽,提高资源利用率,本文采用带分复用测试策略^[7],即将单一TAM根据需要动态划分为多条子TAM,实现多核的测试数据能同时共享同一TAM进行并行传输,从而尽快完成测试. 如图3所示,图3(b)中的 t_1 、 t_2 称为测试时序,图3(a)是未复用带宽的测试优化方案,图3(b)是带分复用下的测试优化方案. 图3(a)、(b)所示规划方案的测试时间分别为 T_a 、 T_b ,其中 $T_b < T_a$,规划(b)优于(a),可见进行带分复用能充分利用TAM带宽,从而提高并行测试效率.

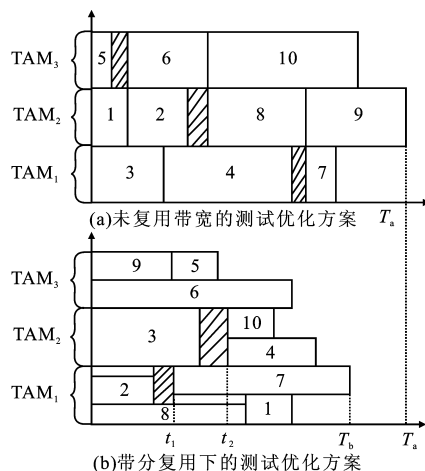


图3 不同测试策略的优化结果

3.2 适应度值与测试约束函数

(1) 适应度值

本文以系统总测试时间作为3D NoC测试的适应度值,由测试结束时间最长的被测核决定,即:

$$T = \max(T_{\text{end}}(\text{core}_i)), 1 \leq i \leq N_C \quad (1)$$

式中, N_C 为待测核数目, $T_{\text{end}}(\text{core}_i)$ 为核 i 的测试结束时间, 由核 i 测试开始时间与其总测试时间组成, 测试开始时间由上一个被测核的测试结束时间决定, 而核 i 的总测试时间包括测试数据经由 xy 通道及 TSV 的时间和核本身的测试时间, 即

$$T_{\text{end}}(\text{core}_i) = T_{\text{core}} + T_{\text{pass}} + T_{\text{start}}.$$

(2) 功耗约束

在并行测试过程中, 多个 IP 核同时工作, 使得芯片瞬时功耗迅速增大, 可能损坏在测芯片. 为提高测试稳定性、正确性, 本文对总功耗进行约束, 同时, 考虑到在异构 NoC 系统中每层功耗的不同, 须对每一层的功耗进行约束.

① 层功耗约束函数

$$P_l = \sum_{i=1}^{N_{\text{on}}} P_{\text{test}i_l} \leq P_{\text{max}l} \quad (2)$$

式中, P_l 表示某一时刻, 层 l 上所有正在测试的核的功耗总和, N_{on} 表示层 l 上正在测试的核的数目, $P_{\text{test}i_l}$ 表示某一时刻层 l 上核 i 的功耗, 包含测试功耗和传输功耗, $P_{\text{max}l}$ 为层 l 的最大允许功耗. 单个核的测试功耗包括核本身的测试功耗以及测试数据的传输功耗, 即: $P_{\text{test}} = P_{\text{core}} + P_{\text{pass}}$.

② 总功耗约束函数

$$P_{\text{total}} = \sum_{l=1}^L P_{\text{test}l} \leq P_{\text{max}} \quad (3)$$

式中, P_{total} 为同一时刻在测 IP 核的总功耗, L 表示系统层数, P_{max} 为系统最大允许总功耗, $P_{\text{test}l}$ 为层 l 上核的测试功耗.

(3) 带宽约束

在任一测试时序 t , 分配到 TAM_j 上的 IP 核测试数据占用带宽的约束函数如下:

$$W_{\text{total}j}^t = \sum_{i=1}^{N_{\text{on}t}} W_{\text{core}i_j} \leq W_{\text{TAM}j} \quad (4)$$

式中, $W_{\text{total}j}^t$ 为 TAM_j 上所有在测时序为 t 的核的测试数据占用的总带宽, $N_{\text{on}t}$ 为 TAM_j 上所有在测时序为 t 的核的数量, $W_{\text{core}i_j}$ 为在测核 i 占用 TAM_j 的带宽, $W_{\text{TAM}j}$ 为 TAM_j 的带宽.

4 基于粒子群算法的 3D NoC 测试优化

4.1 PSO 算法原理

PSO 是由 Kennedy 和 Eberhart 共同提出的一种新的基于种群的进化算法^[8], 利用粒子间的自组织性、协作性进行求解. 在 PSO 算法中, 粒子的位置表示待优化问题的潜在解, 粒子的速度表示粒子飞

翔过程中位置变化的方向, 粒子同时具有记忆功能, 共享当前最优位置, 通过信息交互, 逐步提高在求解过程中发现较好解的可能性, 搜索最优解. 每个粒子根据如下公式进行更新:

$$V^{t+1} = \omega V^t + c_1 r_1 (X'_{\text{pbest}} - X^t) + c_2 r_2 (X'_{\text{gbest}} - X^t) \quad (5)$$

$$X^{t+1} = X^t + V^{t+1} \quad (6)$$

式中, V^t 为粒子当前速度, X^t 为粒子当前位置, ω 为粒子保持惯性的权重, c_1 和 c_2 是调整粒子自身经验与群体经验影响粒子运动轨迹的加速因子, r_1 和 r_2 是 $(0, 1)$ 之间的随机数.

PSO 特别适用于变量是连续实数值的非线性函数的优化^[11], 然而, 在解决实际离散问题时需依据其特有的离散表示形式重新定义算法的操作算子, 此时称之为离散 PSO (Discrete PSO, DPSO). 同时, PSO 在搜索的初期收敛速度很快, 但在后期却易陷入局部极优点. 全局最优粒子对于其他粒子过强的导向作用, 加快了粒子间的信息流动, 使粒子相互靠拢, 导致群体多样性下降, 出现搜索停滞现象, 大大降低了算法的全局探索能力. 群体多样性是种群进化的基础, 为此, 本文提出对群体多样性进行监控, 不同的多样性, 对应的粒子速度更新公式不同, 即通过控制粒子的寻优方向及前行速率来实现不同情况下的更新操作, 从而更好的发挥算法的全局探索能力, 提高问题求解的质量.

4.2 算法设计

4.2.1 编码方案

本文根据 3D NoC 的结构特点和带分复用的测试方法设计编码方案, 对 IP 核测试数据分配以及测试顺序调度进行双重优化. 以规模为 $M \times N \times L$ 的 3D Mesh 结构 NoC 系统进行说明, 给定 TSV 数目 N_T , B 条 TAM, IP 核数量 N_C . 将底层称为第 0 层, 其上各层按顺序依次称为第 1 层、第 2 层……. 各层 IP 核的编号依次为 $1, 2, \dots, N_C - 1$. 各条 TAM 编号依次为 $1, 2, \dots, B$.

(1) IP 核测试数据分配编码

$$X_{\text{IP}} = \begin{bmatrix} B_{11} & B_{12} & \cdots & B_{1N_C^1} \\ B_{21} & B_{22} & \cdots & B_{2N_C^2} \\ \vdots & \vdots & \vdots & \vdots \\ B_{L1} & B_{L2} & \cdots & B_{LN_C^L} \end{bmatrix}$$

式中, B_{li} 表示第 l 层的第 i 个 IP 核的测试数据在第 B_{li} 条 TAM 上传输, $1 \leq l \leq L, 1 \leq i \leq N_C^l, 1 \leq B_{li} \leq B, B_{li}$ 为 0 代表此核系统中不存在.

(2) IP 核测试顺序编码

分配在同一条 TAM 上测试的 IP 核顺序不同,整个系统测试调度规划会不同,测试时间也会不同。为此,本文设计 S_{IP} 标记每条 TAM 上 IP 核的测试数据传输的先后顺序。

$$S = \begin{bmatrix} S_{11} & S_{12} & \cdots & S_{1N_C} \\ S_{21} & S_{22} & \cdots & S_{2N_C} \\ \vdots & \vdots & \vdots & \vdots \\ S_{B1} & S_{B2} & \cdots & S_{BN_C} \end{bmatrix}$$

式中, S_{bj} 表示在第 b 条 TAM 上传输的第 j 个测试数据属于该编号的 IP 核,为 0 表示该核不存在。

4.2.2 算子定义

定义 1 群体多样性

群体多样性刻画了个体间的平均差异程度,给定群体 S ,其定义如下:

$$\text{Div}(S(t)) = \frac{1}{n} \sum_{i=1}^n \sqrt{\sum_{j=1}^m (x_{ij}^t - \bar{x}_j^t)^2} \quad (7)$$

式中, n 为种群规模, m 为问题的维数, $\bar{x}_j^t = \frac{\sum_{i=1}^n x_{ij}^t}{n}$,

表示第 j 维的平均取值。

定义 2 位置的减法

给定两个粒子的位置 X_i 和 X_j , 粒子间位置的变化即为速度, $X_i - X_j$ 定义如下:

$$V = X_i \odot X_j = \bigcup_{k=1}^m F(x_{ik}, x_{jk}) \quad (8)$$

式中,每一维的值可以根据位置改变的定义确定,即

$$F(x_{ik}, x_{jk}) = \begin{cases} 1, & \text{if } x_{ik} \neq x_{jk} \\ 0, & \text{if } x_{ik} = x_{jk} \end{cases}$$

定义 3 速度的乘法

粒子速度的更新中,保持当前速度惯性运动的概率,以此得到速度的更新,即:

$$V' = p \cdot V \quad (9)$$

定义 4 速度的加法

粒子位置寻优方向,给定两个粒子的速度 V_i 和 V_j ,表示速度更新过程中,当前历史最优位置与全局最优位置对粒子速度的指导作用, $V_i + V_j$ 定义如下:

$$V = p_i V_i \oplus p_j V_j \quad (10)$$

表示在第 i 维上以 p_i 的概率根据 V_i ,第 j 维上以 p_j 的概率根据 V_j 的速度进行更新。其中, $\sum_{i=1}^m p_i = 1$ 。

定义 5 位置与速度的加法

粒子位置的更新通过速度的作用实现,给定粒子当前位置 X_i ,按照速度 V_i 前行, $X_i + V_i$ 定义如下:

$$X_i' = X_i + V_i = \bigcup_{k=1}^m E(x_{ik}) \quad (11)$$

运算规则为:对应位实行 1 不变,0 变的操作,其

中, $E(x_{ik}) = \begin{cases} x_{ik}, & \text{if } v_{ik} = 1 \\ \varphi, & \text{if } v_{ik} = 0 \end{cases}$, φ 由算法采取的策略随机选取。

定义 6 粒子的运动方程

通过对以上运算的重新定义,得出粒子的运动方程式如下:

$$V^{t+1} = \omega \cdot V^t \oplus p_1 \cdot (X_{pbest}^t \odot X^t) \oplus p_2 \cdot (X_{gbest}^t \odot X^t) \quad (12)$$

$$X_i^{t+1} = X^t + V^{t+1} \quad (13)$$

根据粒子群体的多样性,对公式(12)作如下改进:

$$V^{t+1} = \begin{cases} \omega \cdot V^t \oplus p_1 \cdot (X_{pbest}^t \odot X^t) \oplus p_2 \cdot (X_{gbest}^t \odot X^t), & \text{div}(t) > d_{high} \\ \omega \cdot V^t \oplus p_1 \cdot (X_{pbest}^t \odot X^t) \oplus p_2 \cdot (X^t \odot X_{gbest}^t), & d_{low} < \text{div}(t) < d_{high} \\ \omega \cdot V^t \oplus p_1 \cdot (X^t \odot X_{worst}^t) \oplus p_2 \cdot (X^t \odot X_{gbest}^t), & \text{div}(t) < d_{low} \end{cases} \quad (14)$$

当群体多样性大于预先给定的阈值 d_{high} 时,应当发挥全局最优粒子的导向作用,将当前最优解共享于搜索空间中,使得粒子相互靠拢,逐渐收缩到较优空间;当群体多样性小于 d_{low} 时,将个体最差的失败经验共享于搜索空间,禁止粒子向个体最差靠近,扩展潜在最优解空间,从而增加种群多样性;当多样性介于两者之间时,群体多样性出现下滑现象,要保持自身当前最优解的方向,但也需要向远离当前最优解的方向运动,平衡局部与全局的导向作用,以保证粒子的寻优速度和精度。

同时,PSO 算法的参数中,惯性权重对算法性能有很大的影响^[9],它代表了粒子之前的速度对于粒子飞行轨迹的影响,较大的 ω 可以增强算法的全局搜索能力,较小的则有利于算法进行局部搜索。为均衡粒子在飞翔过程中的全局及局部探索能力,维持群落的稳定性,提出了一种自适应权重调整机制。

$$\omega = \begin{cases} 0.2, & \text{div}(t) > d_{high} \\ (\omega_{max} - \omega_{min}) \times \exp(-a \times \frac{\text{iter}}{\text{iter}_{max}}) + \omega_{min}, & d_{low} < \text{div}(t) < d_{high} \\ \omega_{max}, & \text{div}(t) < d_{low} \end{cases} \quad (15)$$

式中, $\omega_{max} = 0.9$, $\omega_{min} = 0.4$, $a = 15$, iter_{max} 为种群

最大进化代数, $iter$ 为种群当前的进化代数.

4.3 算法流程

3D NoC 的测试优化主要流程如图 4 所示, 包括初始化种群、适应度值计算、最优 / 最差更新、多样性监控、粒子更新等步骤.

(1) 初始化种群

利用随机函数产生 n 个分配序列, n 为粒子群规模, 各粒子包含各 IP 核的测试数据分配情况.

(2) 适应度值计算

根据公式(1) 得到各粒子的测试时间.

(3) 最优、最差更新

若 $f(X_i) < f(X_{pbest})$, 则 $X_{pbest} = X_i$; 若 $f(X_{pbest}) < f(X_{gbest})$, 则 $X_{gbest} = X_{pbest}$; 若 $f(X_i) > f(X_{worst})$, 则 $X_{worst} = X_i$.

(4) 自适应参数调整

根据公式(15)完成惯性权重的自适应调整.

(5) 粒子更新

根据群体多样性, 采用公式(14)完成粒子更新操作.

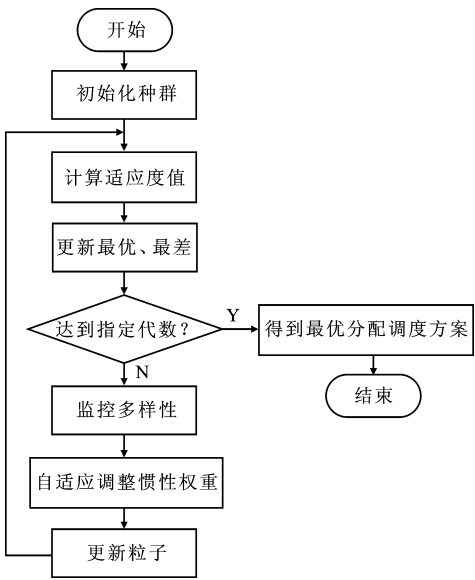


图 4 算法流程图

5 实验仿真与分析

本文选择 ITC' 02 SoC 标准电路作为仿真对象, 采用 Visual C++ 编写程序, 在 Intel(R) Core (TM) i5-2450M CPU 2.50 GHz, 6 GB 内存 Windows 764 位操作系统下运行. 针对 $3 \times 3 \times 3$ mesh 结构的 3D NoC, 选取测试集中经典基准电路 $2 * d695$ 作为研究对象.

(1) 不同 TAM 条数下的测试优化实验. 实验设

定 TAM 带宽为 32, 测试功耗限制为系统额定功耗的 50%, 利用本文算法对 3D NoC 进行测试优化分析. 为验证本文算法性能, 试验中采用本文算法与传统粒子群算法 (PSO) 的研究结果相比较, 比较结果如表 1 所示.

表 1 不同 TAM 条数下的测试结果

基准 电路	TAM 条数	测试时间 (clock cycle)		优化率 /%
		本文	PSO	
$2 * d695$	2	34 962	35 803	2.3
	3	25 004	26 181	4.5
	4	24 788	25 874	4.2
	5	24 678	25 461	3.1
	6	24 641	25 302	2.6

由表 1 知, 随着 TAM 条数的增加, 各算法的测试时间都逐渐减小, 但是, 本文算法的测试时间总是优于 PSO, 可见本文算法在寻优精度上具有明显的优势. 原因是本文对引入了多样性反馈策略, 实时对粒子的前进方向及速度进行指导, 避免算法陷入局部最优, 从而停滞搜索. 实验中所获得的一个较优的测试调度方案如图 5 所示.

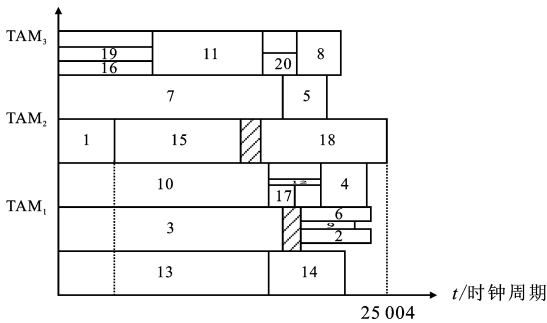


图 5 $2 * d695$ 电路的测试调度结果

图 5 表明, 在测试节点 t_0 处, 核 19、16、7 共享 TAM₃ 进行测试数据的传输, 核 1、10 共享 TAM₂ 进行测试数据的传输, 核 3、13 共享 TAM₁ 进行测试数据的传输. 可见, 通过带分复用策略, 间接增加了 TAM 的条数和测试数据传输通道, 提高了资源利用率. 同时, 由表 1 知, 当 TAM 条数增加至 5 条以上时, 由于系统的测试并行性已足够高, 测试时间并未得到进一步地缩短.

(2) 功耗约束实验. 分别在 30%、50% 以及 80% 的层功耗及总功耗约束下进行测试, 如图 6 所示, 测试结果表明随着最大允许功耗由 80% 降低到 30%, 功耗约束作用不断加强, 测试时间增大. 当最大允许功耗由 50% 降至 30%, 在 TAM 条数为 2 时, 相对测试时间增加了 15.7%, 而在 TAM 条数为 6 时, 相

对测试时间增加了 29.6%;但当 TAM 条数增加到 3 以上时,相对测试时间基本维持稳定.原因是功耗限制加强,能满足可测条件的待测核有限,测试并行性受到约束,故即使增加 TAM 条数也减少不了测试时间.

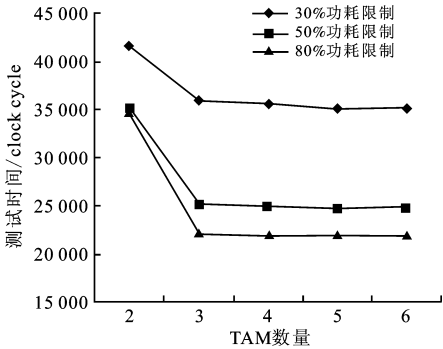


图 6 不同功耗约束的测试结果

6 结束语

本文在带分复用下,采用重用 NoC 作为测试存取机制,并结合离散粒子群优化算法完成对 3D NoC 的测试优化方法研究.选取 ITC'02 测试标准电路进行仿真实验,结果表明,通过监控群体多样性动态指导粒子的前进速度及方向,同时,对惯性权重进行自适应调整,可以有效提高全局搜索能力,应用于 3D NoC 测试优化研究中,能迅速找到最优测试方案,有效缩短了测试时间,提高了资源利用率.

参考文献:

[1] Meena N K, Kapoor H K, Chakraborty S. A new recursive partitioning multicast routing algorithm for 3d network-on-chip [C]//18th International Symposium on VLSI Design and Test. Coimbatore, India, 2014: 1-6.

[2] 张大坤,黄翠,宋治国. 三维片上网络研究综述[J]. 软件学报, 2016,27(1): 155-187.

[3] Karmaka R, Agarwal A, Chattopadhyay S. Testing of 3D-Stacked ICs with hard-and soft-dies-a particle swarm optimization based approach [C]//2015 IEEE 33rd VLSI Test Symposium. Napa CA, 2015: 1-6.

[4] Mukesh Agrawal, Krishnendu Chakrabarty. Test time optimization in NoC-based manycore SoCs using multicast routing[C]//2014 IEEE 32nd VLSI Test Symposium (VTS). California: IEEE Computer Society Press, 2014:18-23.

[5] 许川佩,陈家栋,万春霆. 基于云模型进化算法的硅通孔数量受约束的 3D NoC 测试规划研究[J]. 电子与信息学报,2015,37(2): 477-483.

[6] Han T, Choi I, Oh H, et al. Parallelized network on chip-reused test access mechanism for multiple identical cores [J]. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 2016, 35 (7): 1219-1223.

[7] 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.

[8] Kennedy J, Eberhart R. Particle swarm optimization [C]//International Conference on Neural Networks. Washington: IEEE, 1995:1942-1948.

[9] Chatterjee A, Siarry P. Nonlinear inertia weight variation for dynamic adaptation in particle swarm optimization[J]. Computers&Operations Research, 2006, 33 (3):859-871.

作者简介:

许川佩 女,(1968-),教授.研究方向为自动测试总线与系统、集成电路测试技术.

李克梅(通讯作者) 女,(1992-),硕士研究生.研究方向为集成电路测试. E-mail:401110865@qq.com.