第 27 卷第 4 期　　电 子 与 信 息 学 报　　Vol.27No.4
2005 年 4 月　　Journal of Electronics & Information Technology　　Apr. 2005

使用布尔可满足性的组合电路等价性验证算法

郑飞君　　严晓浪　　葛海通　　杨　军
（浙江大学超大规模集成电路设计研究所　杭州　310027）

摘　要： 该文提出了一种使用布尔可满足性 SAT 的新颖组合电路等价性验证技术。算法是在联接电路（Miter circuit）中进行推理来简化验证问题，推理中使用了“与/非”图结构简化、BDD 扩展、隐含学习多种方法，最后使用有效 SAT 解算器 zChaff 解决验证任务。该算法综合了 BDD 和 SAT 的优点，限制 BDD 构建大小避免了内存爆炸，推理简化减小了 SAT 搜索空间。ISCAS85 电路实验结果表明了本算法的有效性。

关键词： 等价性验证，与/非图，可满足性解算器，隐含学习
中图分类号： TN402　　**文献标识码：** A　　**文章编号：** 1009-5896(2005)04-0651-04

Using Boolean Satisfiability for Combinational Equivalence Checking

Zheng Fei-jun　　Yan Xiao-lang　　Ge Hai-tong　　Yang Jun
(*Institute of VLSI Design, Zhejiang University, Hangzhou* 310027, *China*)

Abstract　In this paper, a new combinational equivalence checking approach using Boolean Satisfiability is proposed. The algorithm uses several methods to reduce the space of the SAT reasoning first, those methods are AND/INVERTER graph transformation, BDD propagation and implication learning, CNF-based SAT solver zChaff is used to solve the verification task. The algorithm combines the advantages of both BDD and SAT, BDD's size is limited to avoid memory explosion problem and structural reduction is applied to reduce the search space of SAT. The efficiency of the proposed approach is shown through its application on the ISCAS85 benchmark circuits.

Key words　Equivalence checking, AND/INVERTER graph, SAT solver, Implication learning

1　引言

检验组合电路的等价性是数字系统设计的一个重要问题，很多技术已被提出用于解决这个问题并被用于验证较大规模设计的正确性。这些技术大致可归结为两大类：功能性验证和结构性验证。功能性方法通过构建两个待验证电路的有序二叉判决图 ROBDD(Reduced Ordered Binary Decision Diagrams)(以下均简写为 BDD)检验两个 BDD 是否同构[1]。结构性方法通过构建两个待验证电路的联接电路(Miter circuit)，即将两个电路的输入共享，对应输出用“异或”门联接，使用 ATPG 等技术来证明联接电路的输出是否是 stuck-at-0 情况[2]。

当前主流等价性验证方法大多数将结构性和功能性技术结合到同一框架里，同时结合多种引擎进行验证。由于 BDD 有一个很好的性质，即布尔函数的 BDD 表示是正则的，因此目前最常用引擎是 BDD。但 BDD 对变量排序（Variable ordering）十分敏感，构建对应电路 BDD 时常常会出现内存爆炸。考虑到两个待验证电路间存在结构相似性，通常的做法是使用局部 BDD,即引进割集(Cutset)[3]来简化验证任务。但如果割集找得不合适，容易导致误判问题(False negative)，也就是说原本等价的两个电路被判为不等价。通常消除误判需花费较多时间。

随着近年来高效可满足性解算器(SAT solver)诸如 zChaff[4]等的提出，SAT 解算器已日益显示出鲁棒的和灵活的推理性能，可满足性解算器也已成为重要验证引擎之一[5]。本文将验证任务转化为 SAT 问题，基于 SAT 解算器 zChaff 进行推理。考虑到 SAT 推理需花费较多的时间在回溯上来解决问题，算法首先在联接电路中进行推理来简化验证任务，使用“与/非”图结构表征[6]和 BDD 扩展[3]来简化电路，采用隐含学习来得到学习子句，减小了 SAT 搜索空间。本文余下部分组织如下：首先给出算法中基本定义和定理；第 3 节介绍本文算法；第 4 节给出实验结果；最后进行小结。

2003-11-04 收到，2004-03-02 改回
国家自然科学基金（90207002）资助课题

2 基本定义和定理

定义 1 “与/非”图（AND/INVERTER Graph, AIG）是有备无患向的非循环图 $G(V=\mathrm{PI}\cup\mathrm{PO}\cup\mathrm{IN},E)$。这里 PI，PO，IN 分别是指对应于原始输入、原始输出及内部节点的节点集，边集 E 描述了节点间互连关系。每个非输入节点 $n\in V$ 均对应于一个二输入“与”运算，并在边上附带“非”运算。

图 1 是 $z=\mathrm{AND}(a,\bar{b})$ 对应 AIG 表征，其中节点 $a,b\in\mathrm{PI}$，$z\in\mathrm{PO}$，边 (b,z) 加黑点表示非运算。

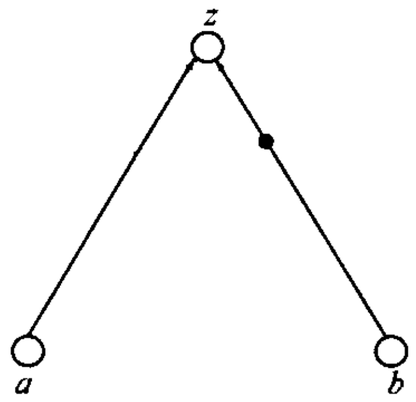

图 1 AIG 表征

定义 2 关于 n 个布尔变量 $x_1,\cdots,x_n$ 的合取范式 (Conjunctive Normal Form, CNF) Φ 是 m 个子句 $\omega_1,\cdots,\omega_m$ 的合取，而每个子句是由一个或多个文字的析取，这里文字是指变量 x 及其非 x’。

定义 3 组合电路的 CNF 公式是指电路中每个门输出对应的 CNF 公式的合取，而每个门的 CNF 公式表示了该门的有效输入-输出赋值关系。

表 1 中给出了 AIG 结构中的简单门对应 CNF 公式，下面给出证明：

证明 因 $p=q\Leftrightarrow(p->q)(q->p)$，又有 $p\to q\Leftrightarrow\bar{p}+q$，综上所得 $p=q\Leftrightarrow(\bar{p}+q)(p+\bar{q})$；同理得 $z=\mathrm{AND}(x,y)\Leftrightarrow(\bar{z}+xy)(z+\overline{xy})$，经转化即：$z=\mathrm{AND}(x,y)\Leftrightarrow(\bar{x}+\bar{y}+z)\ (x+\bar{z})(y+\bar{z})$。其余几种情况类似可证。

证毕

表 1 AIG 中基本门对应 CNF 公式

门函数	CNF 公式
$z=\mathrm{AND}(x,y)$	$(\bar{x}+\bar{y}+z)(x+\bar{z})(y+\bar{z})$
$z=\mathrm{AND}(\bar{x},y)$	$(x+\bar{y}+z)(\bar{x}+\bar{z})(y+\bar{z})$
$z=\mathrm{AND}(x,\bar{y})$	$(\bar{x}+y+z)(x+\bar{z})(\bar{y}+\bar{z})$
$z=\mathrm{AND}(\bar{x},\bar{y})$	$(x+y+z)(\bar{x}+\bar{z})(\bar{y}+\bar{z})$

下面定理我们将组合电路等价性验证问题转化为 SAT 问题。

定理 1 对联接电路对应的 CNF 公式，如果在联接电路输出赋值为 1 时可满足，则两个待验证电路不等价；反之则等价。

不难发现，当联接电路输出赋 1 时，如果至少存在一组赋值，使得联接电路对应 CNF 公式为 1。也就是说，存在一组原始输入赋值，使得两个待验证电路输出值不相等，显然这两个电路不等价。

3 验证算法

本文提出等价性验证算法流程见图 2。在用 zChaff 进行推理前，采用多种方法来简化验证任务。首先以 AIG 结构来构建联接电路，在构建过程中使用哈希表来识别结构等价节点加以优化。如果结构简化后联接电路输出不能得到常数，则接下去将在 AIG 中使用 BDD 扩展来简化 AIG。如在限制 BDD 大小情况下问题未解决，则在 AIG 中隐含学习来获得学习子句（Learned clause），使用当前主流解算器 zChaff 进行推理直至结束。

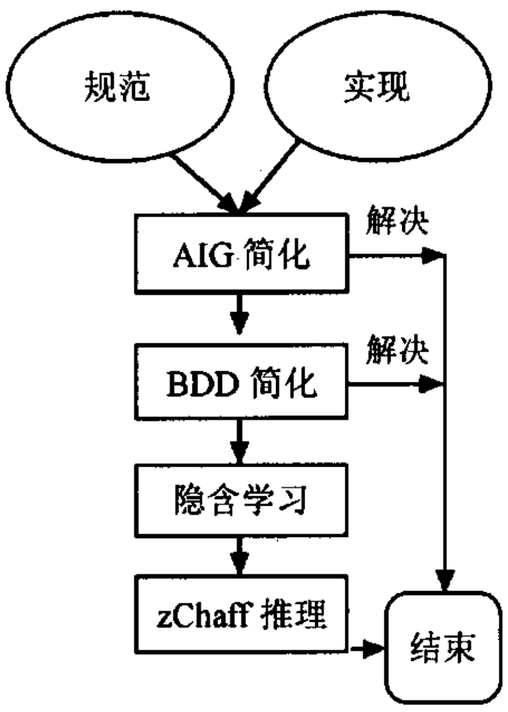

图 2 算法流程

3.1 AIG 结构简化

构建 AIG 的伪代码程序如下：

```
Algorithm create_vertex(p1, p2){
  Special case preprocess;
  If hash lookup(p1, p2)does not find vertex p{
    Creat vertex p;
    If p is the isolated vertex
      replate p with a new variable vertex;
    add p to the hash table;
  }
  return p:
}
```

算法首先以 AIG 形式构建联接电路，构建 AIG 的伪代码见上。p_1,p_2 是待新建节点的两个输入节点，在构建节点 p 前，首先对特殊情况进行预处理，如 p_1,p_2 为常数节点或互相等价等。接下去则进行哈希(hash)查找，判别待构建节点是否已存在。如存在则直接合并同构节点，反之则新建节点 p。每个新建节点都将进入到一个哈希表(hash table)中，

此表使用节点 p 的输入 p_1, p_2 属性作为关键字。

图 3 中给出一个将电路转化为 AIG 结构简单例子，图 3(a)中是两个待验证电路 x 和 y。在构建电路 x 和电路 y 的联接电路时，通过哈希表识别到节点 1 与节点 3 同构、节点 2 和 4 同构，故可分别将它们合并，最后得到 AIG 见图 3（b)。图 3(b)中 x，y，M 分别表示验证电路 x 和 y 以及对应联接电路的 AIG。

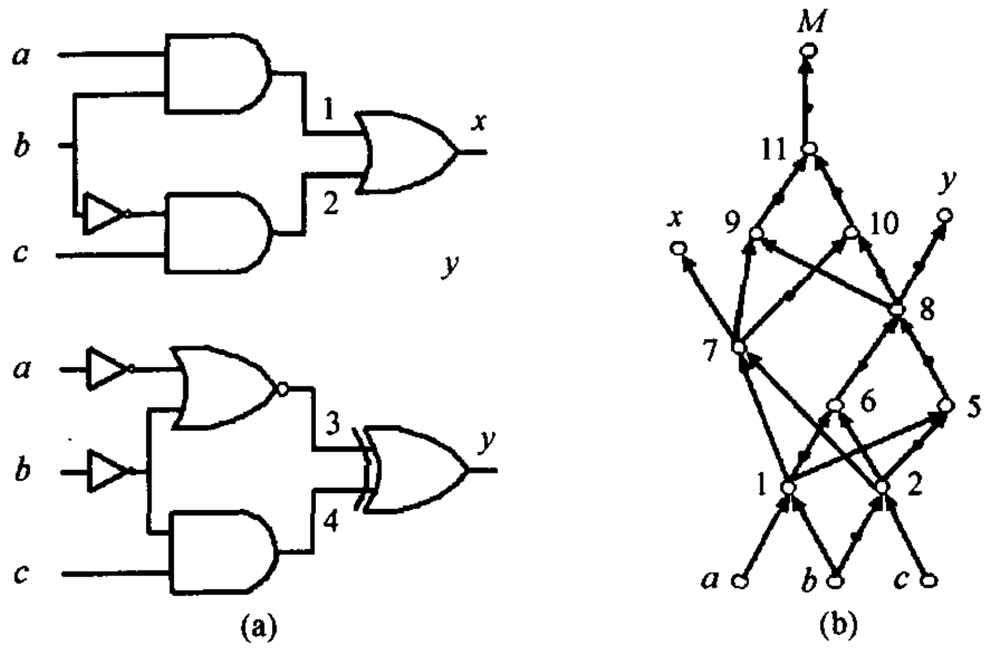

图 3 “与/非”图结构转化

3.2 二叉判决图扩展

在联接电路用 AIG 表示以后，使用 BDD 扩展技术来识别内部等价节点并加以进一步简化。

本文中 BDD 扩展伪代码见图 4。BDD 扩展由一个排序堆（Sorted heap）控制，排序根据是对应节点的 BDD 大小。首先是初始化堆，即对每个原始输入，构建其 BDD 并加入到堆中。然后从堆中迭代地移除具最小的 BDD 的对应节点，并对此节点的所有扇出构建其 BDD，如果新的扇出节点的 BDD 能在限定大小内（算法中限定了最大构建 BDD 数）被构建，则加入到堆中。在构建 BDD 过程中，功能等价节点将被发现并得以合并。如果未能解决问题，以上操作将继续直至堆为空。

图 4　BDD 扩展流程

图 5　隐含电路

3.3 隐含学习

众所周知，在用基于合取范式的 SAT 解算器诸如 zChaff 进行推理时，在合取范式中将所有门用变量替代，这样电路的结构信息就丢失了，这很有可能影响推理的效率。本算法在用 zChaff 推理之前，对 AIG 进行隐含学习，得到学习子句，来进一步减小 SAT 搜索空间。

考虑到电路中隐含是相当有效的(比如在内存为 256M 的奔腾 IV 中每秒可进行上百万次隐含)[7]，算法中对内部顶点也进行隐含。隐含过程如下：对选定某顶点赋值为 1，根据以下查找表 2（只列出部分隐含情况）在 AIG 中进行快速隐含。从隐含结果我们得到学习子句并加到合取范式中。

表 2　查找表

当前状态	1 (0, x)	0 (x, x)	x (0, x)	1 (x, x)	…
下一状态	1 (0, x)	0 (x, x)	0 (0, x)	1 (1, 1)	…
操作	冲突	不定	隐含	隐含	…

假定图 5 中 y_3 是 AIG 中某个内部顶点，在隐含学习中，首先对其赋 1，从查找表中发现 y_1 和 y_2 均需赋 1。进一步对 y_1 和 y_2 隐含发现导致冲突，因为 x_2 从 y_1 中隐含得到赋 1，而从 y_2 隐含得到赋 0。从而可知，只要 $y_3=1$ 就导致冲突，故有学习子句 $\overline{y}_3$。

4　实验结果

在基于 CUDD（Colorado University Decision Diagrams package）[8] 和 zChaff 这两个软件包我们实现了以上算法。所有实验结果在 1.8G 主频，512M 内存的 SUN ULTRA 10 工作站上运行得到。该算法使用实验电路是 ISCAS85 国际标准测试电路。

实验 1　是验证原始电路和无冗余电路(Original and irredundant)的等价性，在实验中限定构建节点对应 BDD 最大结点数为 1000。实验数据如表 3 所示，第 1 栏是被验证电路的名称，第 2 栏到第 4 栏给出分别是文献[5]（主要使用 SAT 进行推理）、文献[9]（仅使用 BDD）和本文算法的验证时间。值得指出的是，表 3 中给出文献[9]实验结果是我们根据文献实现算法并在相同测试环境中运行得到的。从实验结果看，本文算法是相当有效的，尤其是电路 C3540，C5315 和 C7552 这 3 个最困难的电路，其运行时间比其它方法要快得多。

表 3　实验 1 结果（s）

电路	C432	C499	C1355	C1908	C2670	C3540	C5315	C6288	C7552
文献 [5]	0.7	1.17	2.37	3.87	4.46	38.94	6.96	5.04	23.11
文献 [9]	0.65	0.13	0.25	2.30	3.00	3.98	6.10	8.00	9.58
本文算法	0.02	0.04	0.04	0.05	0.08	1.11	0.29	0.50	0.58

表4 实验2结果（s）

电路	C1355	C1908	C2670	C3540	C5315	C6288	C7552
文献 [5]	1.1	5.90	4.93	20.98	27.45	14.52	35.18
本文算法	0.04	0.14	0.52	9.29	0.89	3.23	4.26

实验 2 是验证部分原始电路和优化后电路的等价性。其中优化后电路是用 SIS(a system for sequential circuit synthesis)中优化脚本 Script.rugged 产生，验证任务较之实验 1 相对较难。文献[5]和本文的结果见表 4。不难发现，本文算法在验证较难实例中是极其有效的，对不少电路实验结果比文献[5]要快一个数量级以上。

5 小结

本文提出一种使用 SAT 推理的组合电路等价性验证算法。算法在 AIG 结构中使用 BDD 扩展和隐含学习以简化验证问题，然后利用有效解算器 zChaff 进行推理直至结束。在对 ISCAS85 测试电路的实验结果说明本算法能处理一大类问题。

参 考 文 献

[1] Bryant R E. Graph-based algorithms for Boolean function manipulation. *IEEE Trans. on Computers*, 1986, C-35(8): 677 – 691.

[2] Brand D. Verification of large synthesized designs. ICCAD, San Jose, CA, 1993: 534 – 537.

[3] Andreas Kuehlmann, Florian Krohm. Equivalence checking using cuts and heaps. Design Automation Conference, Anaheim, CA, 1997: 263 – 268.

[4] Moskewicz M, Madigan C, Zhao Y, Zhang L, Malik S. Chaff: Engineering an efficient SAT solver. Design Automation Conference, Las Vegas, 2001: 530 – 535.

[5] Goldberg E I, Prasad M R, Brayton R K. Using SAT for combinational equivalence checking. Design Automation and Test in Europe, UK, 2001: 114 – 121.

[6] Andreas Kuehlmann, Viresh Paruthi, Florian Krohm, Malay K Ganai. Robust Boolean reasoning for equivalence checking and functional property verification. *IEEE Trans. on CAD*, 2002, C-21(12): 1377 – 1394.

[7] Malay K Ganai, Lintao Zhang, Pranav Ashar, Aarti Gupta, Sharad Malik. Combining strengths of circuit-based and CNF-based algorithms for a high-performance SAT. Design Automation Conference, New Orleans, 2002: 747 – 750.

[8] Fabio Somenzi. CUDD: CU Decision Diagram package release 2.3.1(http://vlsi.colorado.edu/~fabio/2001).

[9] Matsunaga Y. An efficient equivalence checker for combinational circuits. Design Automation Conference, Las Vegas, 1996: 629 – 634.

郑飞君： 男，1980年生，博士生，主要从事形式化验证研究.

严晓浪： 男，1947年生，教授，博士生导师，主要从事电子设计自动化、系统级芯片设计等方面的科研和教育工作.