

# 基于轮转算法的缓冲交叉开关调度算法的设计与实现

刘 飞<sup>1</sup> 李 冰<sup>2</sup> 张磊磊<sup>2</sup>

(南京泛盈信息科技有限公司 南京 211100)<sup>1</sup> (IC 学院 无锡 214000)<sup>2</sup>

**摘 要** 研究了一种低复杂度、高性能的交换机调度算法——轮转(Round Robin)调度算法及其硬件实现,它是基于缓冲交叉开关交换结构的。缓冲交叉开关交换结构相比于无缓冲的交叉开关交换结构,仅需要简单的调度算法对输入 VOQ 队列和交叉点缓存输出分别进行调度,这大大简化了调度算法硬件实现的复杂程度,减小了系统延迟。因此,结合轮转算法和缓冲交叉开关交换结构各自的优点,可以设计出高性能的交换机。

**关键词** 轮转算法,缓冲交叉开关,调度算法

**中图分类号** TN915.05 **文献标识码** A

## Design and Implementation of Buffered Crossbar Scheduling Algorithm with Round Robin Scheduling

LIU Fei<sup>1</sup> LI Bing<sup>2</sup> ZHANG Lei-lei<sup>2</sup>

(Nanjing Extensivepro Information Technology Co., Ltd, Nanjing 211100, China)<sup>1</sup>

(IC College, Wuxi 214000, China)<sup>2</sup>

**Abstract** In this paper a low complexity, high-performance scheduling algorithm-Round Robin scheduling algorithm and its hardware implementation is given. It is based on buffered crossbar switch fabric. Compared to the bufferless crossbar switch fabric, buffered crossbar switch fabric only requires simple schedulers that operate independently for each output crosspoint queue column and independently for each VOQ queue. This greatly simplifies the scheduling complexity of hardware and reduces system latency. So by combining the advantage of Round Robin scheduling algorithm and buffered crossbar, a high performance switch can be implemented.

**Keywords** Round robin algorithm, Buffered crossbar, Scheduling algorithm

## 1 引言

当前,大容量、高性能的交换机和路由器所采用的核心技术可以总结为交换结构(Switch Fabric)和调度算法(Schedule Algorithm)两个方面。交换结构是解决报文高速转发的主要方式,它的性能直接决定了交换机和路由器的应用性能,因此高性能交换结构的设计对网络核心交换机的研制十分关键。目前,交换结构可以采用总线交换、环形交换、共享内存交换和交叉开关(Crossbar)交换 4 种方式来实现。其中,前 3 种数据交换方式从某种程度上来说都是共享带宽的,因此不同程度地都受交换容量的限制,制约了大容量网络交换的发展。而交叉开关交换方式在交换网络内部没有带宽的瓶颈,不会因为带宽资源不够而产生阻塞,并且扩展性非常好,已经被广泛地应用在高速交换机、路由器中。对于支持高链路带宽和多端口的交换机而言,其调度器的仲裁时间越短越好,这就需要调度算法在仲裁时尽量地缩短仲裁时间,也就是说调度算法的时间复杂度要尽可能的低。因此,在交叉开关交换结构上寻求低复杂度、高吞吐量的调度算法具有非常重要的意义。

目前,交换结构中典型的分组排队机制大致有 3 种:输出

排队(Output queuing)<sup>[1,2]</sup>、输入排队(input queuing)<sup>[3,4]</sup>和联合输入输出排队(combined input output queuing)<sup>[5,6]</sup>。这 3 种交换结构均采用单一中央调度器来寻求输入与输出之间的最大或极大匹配。为了从根本上提高交换机的性能,一种新的方法是将缓存设置在交换矩阵的交叉节点上,这种交换结构被称为缓冲交叉开关结构。本文将研究缓冲交叉开关交换机(Buffered Crossbar Switch)的结构与性能,并研究一种低复杂度、高性能的轮转调度算法(Round Robin Scheduling Algorithm)在缓冲交叉开关交换机上的实现。

## 2 缓冲交叉开系统模型

交叉开关根据其内部的每一个交叉点是否拥有缓存,可以分为有缓冲的交叉开关(Buffered Crossbar)和无缓冲的交叉开关(Bufferless Crossbar)。相对于缓冲交叉开关交换机,无缓冲的交叉开关交换机需要更复杂的调度算法来匹配输入和输出。为了降低算法的复杂度,我们通过在每个交叉点增加一个小容量的缓冲(Buffer)来实现两个阶段的调度<sup>[7]</sup>。第一阶段,每个输入端口选择一个信元到相应的交叉节点。第二阶段,每个输出端口从交叉节点中选择一个信元输出。因此不需要一个复杂的中央调度算法,并且可以使用流水线的

刘 飞(1978—),男,硕士生,主要研究方向为计算机网络、无线通信,E-mail: derekliu@gmail.com; 李 冰(1968—),男,教授,博士生导师; 张磊磊(1982—),男,工程师。

方式来实现高性能的交换机和路由器。

交叉开关结构在理论上内部是不存在阻塞的,但是在实际的应用系统中存在线头阻塞问题(HOL, Head of Line Blocking),从而影响了系统的效率。例如在一个 $N \times N$ 的交换系统中,每一个输入端口的输入信元都存进一个输入 FIFO 队列中,只有当信元到达 FIFO 队列头部时,调度器才会对其进行调度。此时不同输入端口的信元可能会竞争相同的输出端口,调度器会对其进行调度,选择一个输入信元输出到输出端口。调度器只能对各个输入 FIFO 队列的头部进行调度,如果在输入 FIFO 队列中,队首的信元受阻,那么跟在其后的所有信元也都将被阻塞。即使当前时隙该信元指向的输出端口处于空闲状态也将无法实现交换,信元被挂起在输入端口中,而让此时空闲的输出端口处于“饥饿”(starving)状态。因此 HOL 会严重影响交换系统的交换性能,导致系统性能下降到 58.6%<sup>[2]</sup>。由此可见 HOL 阻塞问题将浪费交换系统将近一半的带宽。为克服 HOL 所产生的影响,目前一般采用虚拟输出排队(VOQ)技术来解决。在 VOQ 中,每个输入端对每个输出端维护一个独立的 FIFO 队列,于是一个信元就不会被同一队列中排在它前面的、与它目的输出端不同的信元所阻塞,从而解决了 HOL 问题。

把输入缓存全部设置在交叉节点中,由于交叉节点数是 $N^2$ ,交叉开关中的缓存容量过于庞大,因此难以实现。目前可行的实现方案是在交叉节点上设置小的缓存,把大量缓存设置在输入端的 VOQ 队列中,通过适当的流控机制,使交叉节点中的缓存不会溢出。有研究表明<sup>[8]</sup>,对于一个 $N \times N$ 缓冲交叉开关交换机,交叉点缓存的大小仅为一个信元分组,并且采用 Round Robin 调度算法,在内部加速为 $2-1/N$ 时,系统可以达到 100%的吞吐率。

图 1 所示是一个 $4 \times 4$ 规模的缓冲交叉开关结构模型,它有 $N^2$ 个交叉节点缓存,每个输入端口的缓存分为 $N$ 个 VOQ 队列,每个 VOQ 队列存储从输入端口 $i$ 到达目的端口 $j$ 的信元。交换系统使用 Round Robin 算法来进行输入 VOQ 和交叉节点输出的调度。由于交叉开关内部做了加速,因此需要输出 FIFO 队列来做速率匹配。为了避免交叉节点的缓存溢出,需要产生相应的反压信号来告之 VOQ 队列交叉点缓存是否已满。

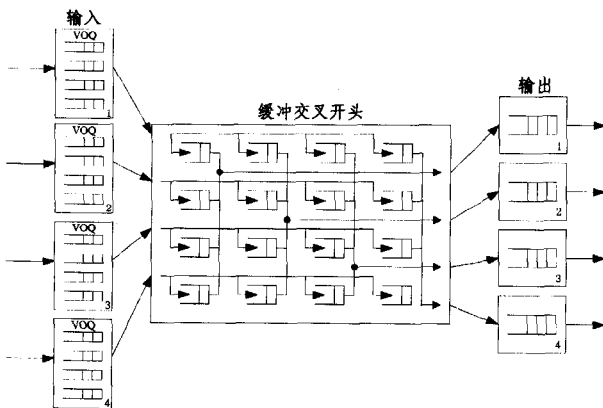


图 1 基于 VOQ 的 $4 \times 4$ 缓冲交叉开关系统模型

### 3 Round Robin 算法分析

近年来国内外已有大量的文献对缓冲交叉开关交换机的调度算法进行了研究。Lin 等人<sup>[9]</sup>从 Crossbar 在每个交叉点缓存一个信元时的角度(输入不采用 VOQ 队列),指出在 Bernoulli 均匀流量下,当交换规模 $N$ 趋向于 $\infty$ 时,缓存 Crossbar 交换机的吞吐率可以达到 100%。Yoshigoe 等人<sup>[10]</sup>、Rojas-Cessa 等人<sup>[6]</sup>对每个交叉点缓存一个信元分组的缓冲交叉开关交换机采用 Round Robin 算法时的性能作了仿真,仿真结果发现缓冲交叉开关交换机采用 Round Robin 分组调度算法时,在均匀流量下可以达到 100%的吞吐率。缓冲交叉开关交换机在调度时无需寻找输入与输出的匹配,即输入和输出端可以分别进行调度。Round Robin 调度算法仅需要简单的指针轮转操作,算法的复杂度为 $O(1)$ ,因此算法实现很简单,减小了系统的延迟。下面介绍 Round Robin 算法的具体调度过程。

缓冲交叉开关交换机的 Round Robin 调度算法是指在输入和输出端分别采用 Round Robin 调度策略的分组调度算法<sup>[11]</sup>。下面以输入端为例介绍 Round Robin 调度算法,在每个时隙每个输入端调度器执行以下 3 个步骤:

1. 请求:输入端 VOQ 队列向每一个相应的交叉点缓存发送请求。
2. 允许:如果一个交叉点缓存收到请求(可能大于一个),则它向最靠近优先级指针的输入端 VOQ 队列发送允许。指向最高优先级的允许指针会指向被允许的输入端的下一个。若未收到相应的请求信号则优先级指针保持不变。
3. 接受:如果一个输入端 VOQ 队列接收到允许,它选取该 VOQ 队列的头分组信元送入交叉点缓存中。

输出端调度算法与输入端的类似,区别仅在于输出时第一步是寻找非空的交叉点缓存,并将其输出到相应输出链路上。Round Robin 调度算法由于只需简单的位操作,因此在高速交换机的设计中得到普遍的关注。

下文说明了 Round Robin 算法的调度过程。图 2、图 3 表示 Round Robin 算法的输入 VOQ 队列和交叉点缓存在连续两个时隙的状态,图 4、图 5 表示 Round Robin 算法的交叉点缓存和输出端口在连续两个时隙的状态。交换机的规模为 $4 \times 4$ 端口,交叉点缓存为 1 个信元分组,为了能够清楚 Round Robin 算法的调度过程,对图 2、图 3 中从 $t-1$ 时隙到 $t$ 时隙输入 VOQ 队列新进来的信元分组未做标示,对图 4、图 5 中从 $t-1$ 时隙到 $t$ 时隙交叉点缓存进来的新信元分组未做标示,从图中可以看到 Round Robin 算法的指针和分组移动过程。图 2 中 $t-1$ 时隙输入端口 1 的指针在 2 位置上,此时假设相应的交叉点缓存为空,VOQ<sub>12</sub>的信元分组将被输入到交叉点缓存中,其它端口的过程类似。经过 1 个时隙的时间,图 3 中 $t$ 时隙输入端口 1、2 和 4 的指针顺时针滑动了一位,输入端口 3 的指针滑动了两位,因为 VOQ<sub>32</sub>队列为空,跳过空的队列。输出端的调度过程与输入端类似。

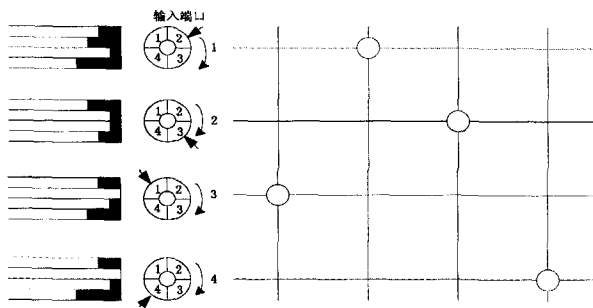


图2 输入端口在  $t-1$  时隙的状态

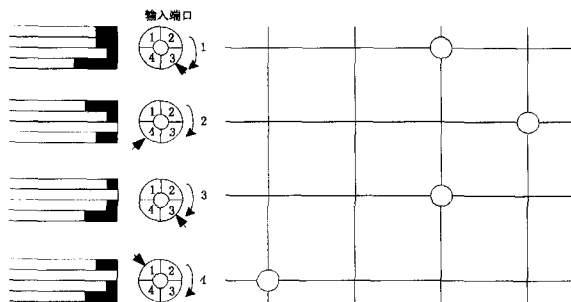


图3 输入端口在  $t$  时隙的状态

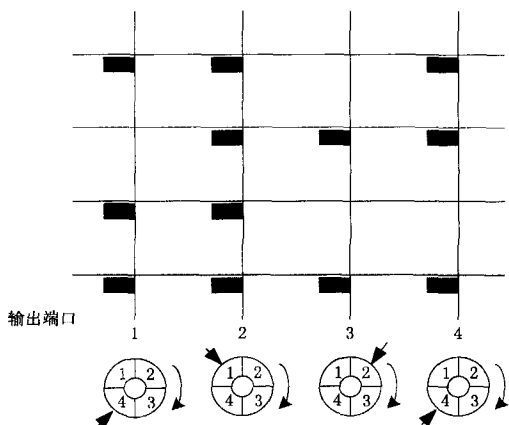


图4 输出端口在  $t-1$  时隙的状态

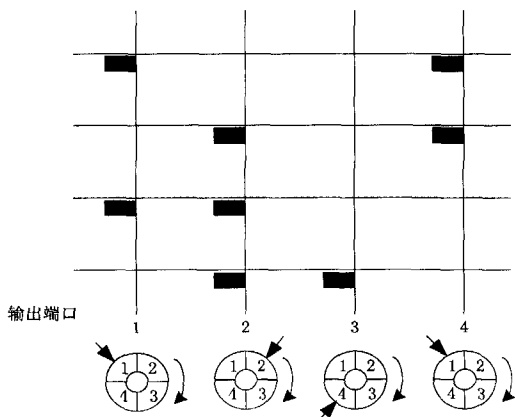


图5 输出端口在  $t$  时隙的状态

#### 4 Round Robin 调度算法的硬件实现

Round Robin 调度算法中请求、允许、接受这 3 个步骤对应着图 6 中的 3 个纵向部分。从图中可以看出,算法中的请

求就是将请求发送至允许仲裁器,因此可以将设计中的请求和允许两部分组合在一起。接受仲裁器的输出作为反馈信号,用来使指向最高优先级的允许指针指向被允许的输入端的下一个。

从图 6 中可以很容易得出,允许仲裁器和接受仲裁器之间的延迟直接决定了整个 Round Robin 算法实现的延迟,从而影响了 Round Robin 算法的实现速度。为了提高 Round Robin 仲裁器的实现速度,需要严格设计 Round Robin 仲裁器的结构。Round Robin 仲裁器的结构是由一个可配置优先编码器(PPE)和优先级指针构成的,优先级指针用来指向优先编码器最高优先级的位置。PPE 相比于普通的优先编码器,其主要特点是 PPE 是由外部输入来决定哪一个输入具有最高的优先级。

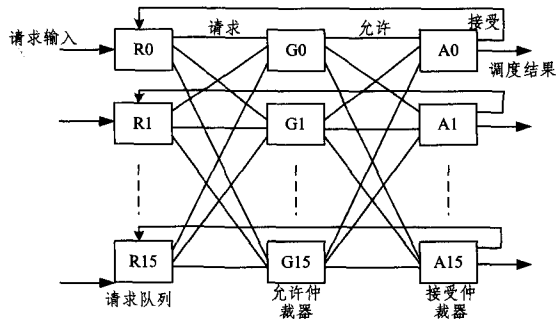


图6 Round Robin 算法抽象图

下面将介绍一种高速 Round Robin 仲裁器的实现方法。

图 7 是一个 Round Robin 仲裁器的结构图。优先级指针 pointer 指向当前输入的最高优先级的位置。在每个时钟上升沿开始的时候,优先级指针 pointer 通过 PPE 在  $N$  比特的请求信号 req 之间进行选择。PPE 模块的输入为  $N$  比特的请求信号 req 和  $\log_2 N$  比特的优先级指针 pointer。它选择从 req[pointer] 开始(包括 req[pointer])的第一个不为零的比特位作为  $N$  比特 grant 信号中的 1 的位置。这一选择过程可以用纯组合逻辑来实现,优先级指针的加 1 操作也可以一并进行。为了进一步提高 Round Robin 算法的实现速度,我们需要减小从请求到允许过程的延迟时间,也就是要减小 PPE 这一纯组合逻辑模块的延迟。

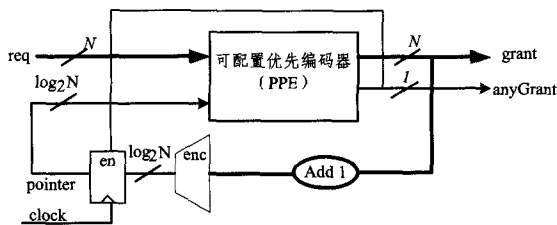


图7 Round Robin 仲裁器结构图

经研究可以发现,通过使用两个简单组合逻辑模块可以减小 PPE 模块中组合逻辑的延迟,将 PPE 改用 PE 和 PPE\_noLoop 来实现,其中 PE 就是普通的优先级编码器,而 PPE\_noLoop 是与 PPE 类似的编码器,不同之处在于它只从 req[pointer] 搜索到 req[N-1],而并不关心从 req[0] 到 req[pointer-1] 的请求信号。因此,PPE\_noLoop 不会绕回去寻找从 req[0] 到 req[pointer-1] 中的请求信号,也就不存在反

馈回路,这就可以大大提高仲裁速度。PPE 仲裁器结构如图 8 所示。

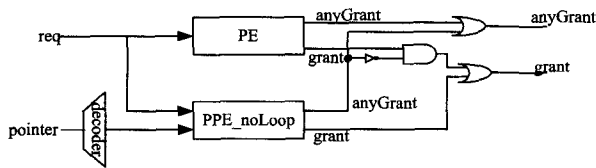


图 8 PPE 仲裁器结构图

PPE 仲裁器工作流程如下:如果在 req[pointer]到 req[N-1]之间没有任何输入请求,那么 PPE 的 grant 输出就等于 PE 的 grant 输出。如果在 req[pointer]到 req[N-1]之间有输入请求,那么 PPE 的 grant 输出就等于 PPE\_noLoop 的 grant 输出。很明显 PPE 的输出 grant 就是一个 2 输入选择器的输出,PPE\_noLoop 的 anyGrant 信号为选择信号,它用来指示是否有仲裁输出,它是通过将各个比特位或起来得到的。PPE 仲裁器由于是通过组合逻辑来实现的,因此延迟很小,可以满足高速 Round Robin 仲裁器设计的要求。

**结束语** 缓冲交叉开关相比于无缓冲的交叉开关具有仅需要较简单的调度算法、降低了输入与输出之间的同步要求、更好的性能等优点。Round Robin 调度算法仅需要简单的指针轮转操作,算法的复杂度仅为  $O(1)$ ,且实现简单,系统的延迟小,因此 Round Robin 调度算法受到了高度的重视。本文根据缓冲交叉开关交换机的特点,结合 Round Robin 算法的优势,设计了一种高速 Round Robin 仲裁器,以满足高速交换机设计的需求。

## 参考文献

- [1] Magill R B, Rohrs C E, Stevenson R L. Output-queued switch emulation by fabrics with limited memory[J]. IEEE Journal on Selected Areas Communications, 2003, 21(4): 606-615
- [2] Karol M, Hluchyj M, Morgan S. Input versus output queueing on a space division switch[J]. IEEE Transactions on Communications, 1987, 35(12): 1347-1356
- [3] McKeown N. Scheduling Algorithms for Input-Queued Switches [D]. Ph D dissertation, University of California at Berkeley, 1995
- [4] McKeown N, Mekkittikul A, Anantharam V, et al. Achieving 100% throughput in an input-queued switch (extended version) [J]. IEEE Transactions on Communications, 1999, 47(8): 1260-1267
- [5] Nabeshima M. Performance evaluation of a combined input- and crosspoint-queued switch[J]. IEICE Transactions on Communications, 2000, E83-B(3): 737-741
- [6] Rojas-Cessa R, Oki E, Jing Z, et al. CIXB-1: Combined input-once-cell-crosspoint buffered switch [C] // IEEE Workshop on High Performance Switching and Routing. Dallas, USA: IEEE Press, 2001: 324-329
- [7] Chuang S T, Iyer S, McKeown N. Practical algorithms for performance guarantees in buffered crossbars [C] // IEEE INFOCOM'05. Miami, FL: IEEE Infocom, 2005: 981-991
- [8] Berger M S. Delivering 100% throughput in a Buffered Crossbar with Round Robin scheduling [C] // High Performance Switching and Routing. IEEE HPSR, Poznan, Poland, 2006: 5-10
- [9] Lin M, McKeown N. The throughput of a buffered crossbar switch [J]. IEEE Communications Letters, 2005, 5: 465-467
- [10] Yoshigoe K, Christensen K J. An evolution to crossbar switches with virtual output queuing and buffered crosspoints [J]. IEEE Network, 2003, 17(5): 48-56
- [11] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [12] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [13] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [14] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [15] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [16] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [17] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [18] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [19] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [20] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [21] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [22] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [23] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [24] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [25] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [26] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [27] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [28] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [29] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [30] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [31] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [32] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [33] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [34] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [35] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [36] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [37] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [38] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [39] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [40] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [41] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [42] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [43] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [44] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [45] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [46] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [47] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [48] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [49] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [50] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [51] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [52] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [53] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [54] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [55] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [56] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [57] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [58] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [59] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [60] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [61] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [62] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [63] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [64] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [65] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [66] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [67] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [68] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [69] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [70] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [71] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [72] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [73] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [74] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [75] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [76] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [77] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [78] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [79] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [80] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [81] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [82] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [83] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [84] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [85] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [86] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [87] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [88] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [89] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [90] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [91] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [92] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [93] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [94] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [95] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [96] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [97] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [98] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [99] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [100] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [101] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [102] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [103] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [104] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [105] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [106] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [107] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [108] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [109] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [110] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [111] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [112] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [113] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [114] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [115] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [116] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [117] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [118] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [119] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [120] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [121] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [122] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [123] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [124] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [125] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [126] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [127] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [128] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [129] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [130] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [131] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [132] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [133] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [134] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [135] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [136] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [137] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [138] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [139] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [140] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [141] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [142] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [143] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [144] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [145] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [146] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [147] 李勇, 罗军舟, 吴俊. 一种交叉点小缓存 CICQ 交换机高性能调度算法 [J]. 计算机研究与发展, 2006, 43(12): 2033-2040
- [148] Cordeiro L, Fischer B, Marques-Silva J. Continuous Verification of Large Embedded Software using SMT-Based Bounded Model Checking [C] // 17th IEEE International Conference and Workshops on the Engineering of Computer-Based Systems. Oxford, England, 2009: 160-169
- [149] Kong Wei-qiang, Shiraishi, Mizushima, et al. An SMT Approach to Bounded Model Checking of Design in State Transition Matrix [C] // Proceedings of 2010 International Conference on Computational Science and Its Applications (ICCSA). 2010: 231-238
- [150] 周宽久, 杨广, 柳朕, 等. 基于拓扑排序的数据竞争方向定位 [J]. 计算机科学, 2012, 39(11): 108-116
- [151] Wang Jie, Cui Kai, Zhou Kuan-jiu, et al. Programmable NoC Scheduling Based on Multi-core Processor [C] // Proceedings of International Conference on Electrical and Control Engineering (ICECE). 2011
- [152] Lai Xiao-chen, Wang Xiao-liang, Zhou Kuan-jiu, et al. Research on method of static analysis for safety of C++ program [J]. International Journal of Advancements in Computing Technology, 2012, 4(21): 337-345

(上接第 86 页)

- [142] Das S, Dill D L, Park S. Experience with predicate abstraction [C] // Computer-Aided Verification. Trento, Italy, 1999: 160-171
- [143] Lahiri S K, Nieuwenhuis R, Oliveras A. SMT techniques for fast predicate abstraction [C] // Proceedings of 18th International Conference on Computer Aided Verification (CAV). Seattle, USA, 2006: 424-437
- [144] Cavada R, Cimatti A, Franzen A, et al. Computing Predicate Abstractions by Integrating BDDs and SMT Solvers [C] // Seventh International Conference on Formal Methods in Computer-Aided Design. Austin, Texas, 2007: 69-76
- [145] 何炎祥, 吴伟, 陈勇, 等. 基于 SMT 求解器的路径敏感程序验证 [J]. 软件学报, 2012, 23(10): 2655-2664
- [146] Clarke E M, Grumberg O, Jha S, et al. Counterexample-Guided Abstraction Refinement [C] // CAV. Berlin, Heidelberg, 2000: 154-169
- [147] Henzinger T A, Jhala R, Majumdar R, et al. Lazy Abstraction [C] // Proceedings of the 29th Annual Symposium on Principles of Programming Languages. Oregon, Portland, 2002: 58-70