

# 一种新的 RFID 标签识别防冲突算法

冯波<sup>1,2,3</sup> 李锦涛<sup>1</sup> 郑为民<sup>1,3</sup> 张平<sup>1,2,3</sup> 丁振华<sup>1,2,3</sup>

**摘要** 在 RFID 网络通信中,当多个标签同时回应阅读器的查询时,如果没有相应的防冲突机制,会导致标签到阅读器的通信冲突,使得从标签返回的数据难以被阅读器正确识别.防冲突算法是阅读器快速、正确获取标签数据的关键.一种被称为基于栈的 ID-二进制树防冲突算法(Stack-based ID-binary tree anti-collision algorithm, SIBT)被提出,SIBT 算法的新颖性在于它将  $n$  个标签的 ID 号映射为一棵唯一对应的 ID-二进制树,标签识别过程转化为在阅读器中创建 ID-二进制树的过程.为了提高多标签识别效率,阅读器使用栈保存已经获取的 ID-二进制树创建线索,用计数器保存标签在该栈中的深度.理论分析和仿真结果表明 SIBT 算法的性能优于其他基于树的防冲突算法.

**关键词** 无线射频识别,防冲突算法, ID-二进制树, 标签识别  
**中图分类号** TP301

## A Novel Anti-collision Algorithm for Tag Identification in RFID Systems

FENG Bo<sup>1,2,3</sup> LI Jin-Tao<sup>1</sup> ZHENG Wei-Min<sup>1,3</sup> ZHANG Ping<sup>1,2,3</sup> DING Zhen-Hua<sup>1,2,3</sup>

**Abstract** In the RFID system, tag-to-reader communication collision occurs when more than one tag responds to a reader's interrogation at the same time. Without any coordination among the reader and the tags, the responses from the tags will become illegible to the reader. Anti-collision algorithms for passive RFID tag identification are significant for fast and correctly obtaining the tags ID. This paper presents a novel anti-collision algorithm which is called stack-based ID-binary tree anti-collision algorithm (SIBT). The novelty of the SIBT algorithm is that the algorithm maps a set of  $n$  tags into a corresponding ID-binary tree, and sees the process of collision arbitration as a process of building the ID-binary tree. In order to efficiently construct an ID-binary tree, the reader uses a stack to store the threads of the construction information while the tag uses a counter to keep track of the stack position where the tag is on. Both theoretic results and simulation performance evaluations show that the SIBT outperforms other tree based tag anti-collision algorithms.

**Key words** RFID, anti-collision algorithm, ID-binary tree, tag identification

RFID (Radio frequency identification technology) 是利用射频技术,在阅读器 (Reader) 与标签 (Tag) 之间非接触地传递信息,进而获取被标识物体信息的识别技术<sup>[1]</sup>.具有非可视传输、快速、穿透性强、较强的抗恶劣环境能力、可重复使用等优点,可以广泛应用于各个行业.在学术界,RFID 被视为连接现实物理世界和计算机虚拟世界的桥梁<sup>[2-3]</sup>,成为实现普适计算的重要技术手段<sup>[4-5]</sup>.借助 RFID 技术,可以构筑一个全新的“物网络”(Internet of things),传统 Internet 网络节点从 PC 延伸向“物”.在“物网络”中,所有现实物体都能被

纳入其中,任何物包括人都可以用 RFID 标签进行唯一标识,成为“物联网”中的一个节点,从而能够自动与网络连接、交互,并被追踪、定位、监视、触发某类事件等等,实现随时、随地、无处不在的计算功能.

与传统网络相似,RFID 阅读器与标签之间的通信同样面临信道共享和访问冲突问题,由于多个标签共享 Tag-to-Reader 的上行信道,当多个标签同时回应阅读器的查询时,如果没有相应的防冲突机制,必然会引起冲突的发生<sup>[6]</sup>,致使标签信息无法被阅读器正确识读或漏读.由于 RFID 通信技术的自身特点,以及受限于标签的计算和存储能力,传统的防冲突技术如 Aloha、Binary tree、CSMA/CD 等,难以直接应用于 RFID.设计高效、鲁棒的 RFID 防冲突算法成为亟待解决的技术难题.

现有的 RFID 防冲突算法可划分为基于 Aloha 和基于二进制树 (Binary tree) 两大类. Aloha 协议族又可分为纯 Aloha、时隙 Aloha (Slotted Aloha)<sup>[7]</sup>、帧-隙 Aloha (Frame-slotted Aloha)<sup>[8-9]</sup>.基于 Aloha 的算法存在不稳定工作区间<sup>[10]</sup>,理论上吞吐量被限制在  $1/e$  以内.更为严重的是 Aloha 协

收稿日期 2007-01-22 收修改稿日期 2007-07-18  
Received January 22, 2007; in revised form July 18, 2007  
广东省重点科技攻关项目 (2005B80406004), 粤港关键领域重点突破项目 (200649813001) 资助  
Supported by the Key Science-Technology Project of the Guangdong Province (2005B80406004) and the Guangdong-Hong Kong Technology Cooperation Funding Scheme (200649813001)  
1. 中国科学院计算技术研究所 北京 100190 2. 中国科学院研究生院 北京 100039 3. 广东电子工业研究院 东莞 523808  
1. Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190 2. Graduate University of Chinese Academy of Sciences, Beijing 100039 3. Guangdong Electronics Industry Institute, Dongguan 523808  
DOI: 10.3724/SP.J.1004.2008.00632

族会导致“标签饥饿问题”,即特定的标签可能会在很长的一段时间内都无法被正确识别.因而以传统二进制树协议为基础,各种变形的二进制树协议在 RFID 领域获得了广泛的应用.典型的如 BBT (Bit-by-Bit tree) 算法<sup>[11]</sup>、Q-tree 算法<sup>[12]</sup>、ABS (Adaptive binary splitting) 算法<sup>[13]</sup>、具有剪裁操作的二进制搜索算法<sup>[14]</sup>等.

本文提出一种新的基于二进制树的标签防冲突算法 (Stack-based ID-binary tree anti-collision algorithm, SIBT), 该算法的特点在于将多个标签的识别转换为构造一棵唯一的 ID-二进制树, 阅读器通过栈存储标签识别过程中所获得的历史信息和构造线索, 加快该 ID-二进制树的构造. 与已有 RFID 标签防冲突算法比较, SIBT 算法能明显减少重复查询次数, 显著降低标签数据传输量和能耗, 算法实现简单, 标签端开销小, 仅需要增加一个栈计数器, 以电子产品编码最长编码 256 位计, 栈的最大深度是 255, 因此附加的计数器最长为 8 位.

本文第 1 节详述 SIBT 算法及其实现, 第 2 节给出 SIBT 算法的理论性能分析, 第 3 节给出 SIBT 算法与已有 RFID 二进制算法的性能仿真结果, 第 4 节总结全文.

## 1 SIBT 算法及其实现

### 1.1 算法设计需解决的关键问题

设计基于二进制树的防冲突算法必须解决好以下三个问题:

1) Tag-to-Reader 信道状态的及时反馈, 即每个标签都必须具备感知信道状态信息的能力, 以便了解信道当前时隙是空闲、冲突或成功, 从而判断在下一个时刻是否尝试重传数据. 反馈信息可分为二元反馈 (Binary feedback)、三元反馈 (Ternary feedback)、多元反馈 ( $N$ -nary feedback) 等. 在 RFID 系统中多采用三元反馈<sup>[15-16]</sup>.

2) 在一个冲突解决期 (Conflict resolution interval, CRI) 内到达的新的标签, 应以何种策略决定是否参与信道的竞争. 通常可以归纳为两类: 一类是阻塞型信道访问策略, 该策略要求在 CRI 内到达的新标签不能加入到当前 CRI 中, 直到当前 CRI 结束, 然后加入下一次 CRI 的信道竞争<sup>[9-12]</sup>; 另一类是自由信道访问策略, 任何新到达的标签都被加入到当前 CRI 竞争中<sup>[13]</sup>. 与阻塞型信道访问策略相比, 自由信道访问策略最直接的优点是, 各标签不必保持严格的同步, 也不必确定 CRI 的界线, 标签只需具备部分感知能力 (Limited sensing), 同时也不存在死锁问题 (Deadlock)<sup>[17]</sup>. 但缺点也很明显, 就是性能不如阻塞型策略.

3) 在 CRI 内标签应以何种策略进行分裂. 一种是确定地址法 (Deterministic addressing), 在 CRI 内, 标签以其 ID 的每一位决定标签是分配到左子树还是右子树; 另一种是随机地址法 (Random addressing), 以一个伪随机数生成器 (或抛硬币) 确定标签是分配到左子树或右子树. 文献 [18] 证明在传统网络系统中, 确定地址分裂算法效率要高于随机地址法, 吞吐量总是大于 1/2, 确定地址算法另一个优点是生成的冲突树层数是一个确定的值.

基于 RFID 的应用实践及以上三个需要关注的问题, 对于 RFID 多标签识别, 我们有以下结论:

1) RFID 冲突解决算法更适于采用确定的分裂算法. RFID 系统的每一次通信, 都是一个阅读器与有限个标签的通信过程, 并且每个标签都有一个唯一的 ID 标识, 这就为采用确定的分裂算法提供了可能.

2) RFID 冲突解决过程有比较明显的间歇性和突发性. 在传统网络中, CRI 内不断有新的数据包以近似于泊松过程到达网络, 每个网络节点需要缓存这些数据包, 并以某种策略让部分数据包参与下一次 CRI 的竞争. 在 RFID 网络中, 通信过程有比较明显的间歇性和突发性, 如超市自动结帐系统、出入库管理系统等每隔一段时间就会有一批数据集中到达, 在短时间内, 阅读器必须读出所有的数据, 然后阅读器停止工作等待下一批数据的到达. 图 1 对 RFID CRI 和传统 CRI 进行了比较, 可以假定 RFID 在一个 CRI 内不会有新的标签到达.

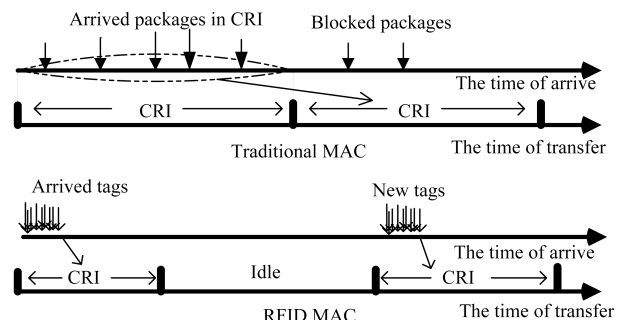


图 1 CRI 比较

Fig. 1 Comparison of CRIs

3) RFID MAC 协议具备一定的受控接入特性. 在 RFID 通信过程中遵循的是“Reader talk first”原则, 所有标签对广播信道的访问都受到阅读器的集中控制, 这就使得标签到阅读器的通信具有较为明显的受控特性. 同时, 阅读器的计算能力和存储能力远远强于标签, 因此, 多标签冲突的解决应该充分利用好阅读器资源优势, 降低冲突解决算法在标签端的复杂度.

## 1.2 算法描述

**定义 1. (ID-二进制树)** 一棵高度为  $k$  且叶节点只出现在第  $k$  层的二叉树,  $k$  为标签 ID 的长度; 在 ID-二进制树中, 每一条左边用 0 标识, 右边用 1 标识; 从根节点到叶节点, 第  $j$  层边对应于标签 ID 的第  $j$  位; 叶节点表示一个标签, 从根节点到该叶节点的路径唯一标识该标签的 ID 号; 从根节点到任意中间节点的路径对应于该标签 ID 号的前缀. 以 ID 长度为 4 的标签集 {0001 0010 1010 1011} 为例, 其对应的 ID-二进制树如图 2 所示.

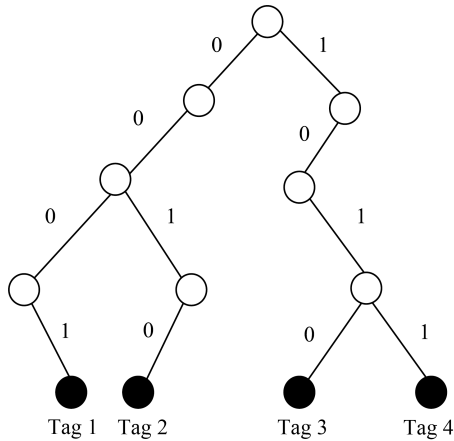


图 2 ID-二进制树

Fig. 2 ID-binary tree

利用 ID-二进制树, 可以将任意包含  $n$  个标签的集合映射到一棵唯一的高度为  $k$ , 叶节点为  $n$  的 ID-二进制树. SIBT 算法将多标签的识别视为在阅读器端构建一棵 ID-二进制树, 阅读器端利用栈保存树的构造线索, 标签利用计数器  $C$  记录它在栈中对应位置,  $C$  的另一个作用是指示标签状态, 如果  $C$  大于零则标签处于非活动状态, 处于非活动状态的标签不响应阅读器的位查询命令. 在一个冲突解决期中, SIBT 算法采用逐位查询方式构建 ID-二进制树, 如果在某一位同时检测到“1”和“0”则表示该位存在冲突, 阅读器保存右子树构造线索, 并继续左子树的查询; 标签利用阅读器发布的命令获取上行信道的状态, 并依据信道状态及查询位的值进行分裂. 算法不考虑在一个冲突解决期中新标签加入的情况, 即采用的是阻塞型策略.

算法在阅读器端的执行过程如下:

**Initialization:** 创建一棵只含根节点的 ID-二进制树  $T$ , 工作串  $S$  置空并入栈;

**Query ( $i$ ):** 查询第  $i$  位, 根据返回的数据判断是否有冲突存在;

**Insert:** 执行分裂操作, 将子树插入  $T$  中;

**Push:** 保存右子树构造线索;

**Pop:** 恢复右子树构造线索.

根据阅读器发布的命令, 标签端进行如下操作:

**Query ( $i$ ):** 如果标签处于活动状态 (栈计数器等于 0), 则返回标签的第  $i$  位值;

**Push:** 表示有冲突存在, 处于非活动状态的标签其栈计数器加 1, 处于活动状态的标签如果当前传送位的值等于“1”, 则栈计数器加 1 转入非活动状态;

**Pop:** 非活动标签栈计数器减 1.

SIBT 算法的伪代码如下所示.

### 标签端伪代码 (Pseudo-code on the tag)

```
Initialize  $count = 0$ ; Switch (receive_command){
  case query ( $i$ ):
    if ( $count == 0$ ) then response with  $i^{\text{th}}$  bit;
    break;
  case push ():
    if ( $count == 0$  and  $i^{\text{th}}$  bit is '1') then
       $count++$ ;
    else if ( $count > 0$ ) then  $count++$ ;
    break;
  case pop ();
    if ( $count > 0$ ) then  $count--$ ;
    break;
}
```

### 阅读器端伪代码 (Pseudo-code on the reader)

```
Initialize ID-binary tree  $T$  to a root node
and push  $\epsilon$  onto the stack  $s$ 
While (Not empty ( $s$ )){
  String  $S = \text{pop}(s)$ ;
  for ( $i = S.\text{length}; i < \|ID\|; i++$ ){
    query ( $i$ ); //query the  $i^{\text{th}}$  bit
    if (query conflict) then{
      insert left branch and node to  $T$ ;
      inset right branch and node to  $T$ ;
      push ( $s, S.\text{append}('1')$ );
       $S = S.\text{append}('0')$ ;
    }else{
      if (query return is '0') then  $S = S.\text{append}('0')$ ;
      if (query return is '1') then  $S = S.\text{append}('1')$ ;
    }
  }
  if ( $i == \|ID\|$ ) then Set.add ( $S$ );
}
Return (Set and  $T$ );
```

注: Set 为容纳  $S$  的集合.

## 2 性能理论分析

本节主要考虑两个性能指标, 一个是阅读器发送的查询次数, 它决定了冲突解决期 CRI 的识别延迟; 另一个是标签通信复杂度, 表示标签在 CRI 中传送的数据量, 它是影响识别延迟以及标签功耗的关键因素.

### 2.1 位查询次数

从 ID-二进制树定义可知, 对于任意的标签集都有一棵唯一的 ID-二进制树与其对应. 查询  $N$  个标签所需查询次数  $Q(N, k)$  等于对应 ID-二进制树的内节点数. 因此, 有下式

$$Q(N, k) = N_1 + N_2 = N + N_1 - 1 \quad (1)$$

其中  $N_1$  和  $N_2$  分别表示度为 1 和 2 的内节点数.

**定义 2. (最大 ID-二进制树)** 在所有包含  $N$  个叶节点的 ID-二进制树中, 拥有内节点最多的 ID-二进制树.

**定义 3. (最小 ID-二进制树)** 在所有包含  $N$  个叶节点的 ID-二进制树中, 拥有内节点最少的 ID-二进制树.

### 2.2 最差查询次数

**引理 1.** 对于高度为  $k$ , 含  $N_2$  个度为 2 的节点的 ID-二进制树  $T$ , 如果  $T$  是拥有最多内节点的最大 ID-二进制树, 则  $T$  中度为 2 的节点层次数限定在  $[0, \lfloor \log_2 N_2 \rfloor]$  中.

**证明.** 假设  $T$  中存在度为 2 的节点层次大于  $\lfloor \log_2 N_2 \rfloor$ , 则根据图 3 可以生成一棵新的 ID-二进制树  $T^1$ , 其内节点数大于  $T$ , 这与  $T$  是最大 ID-二进制树相矛盾.  $\square$

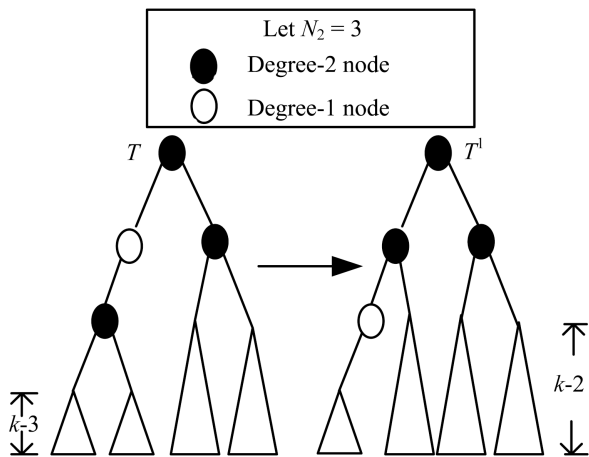


图 3 生成大于  $T$  的最大 ID-二进制树

Fig. 3 Building an ID-binary tree that is larger than  $T$

**引理 2.** 对于高度为  $k$  的最大 ID-二进制树  $T$ ,

如果度为 2 的节点的最大层次数是  $\lfloor \log_2 N_2 \rfloor$ , 则从 0 层到  $\lfloor \log_2 N_2 \rfloor - 1$  层的所有节点的度都为 2.

**证明.** 假设在  $T$  中从 0 层到  $\lfloor \log_2 N_2 \rfloor - 1$  层存在度为 1 的节点, 则根据图 4 可以构造一棵新的 ID-二进制树  $T^1$ , 其内节点数大于  $T$ , 这与  $T$  是最大 ID-二进制树相矛盾.  $\square$

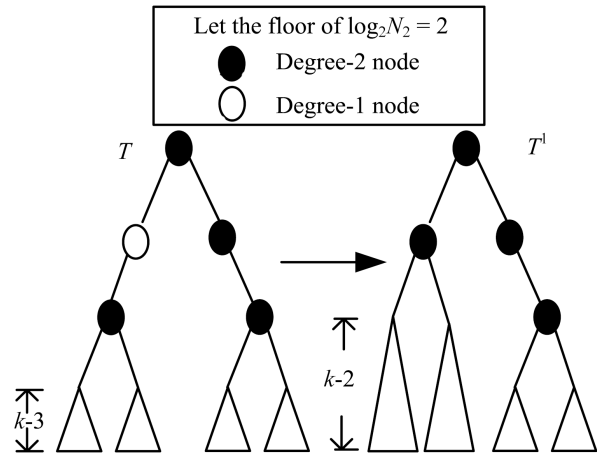


图 4 改变  $T$  成为更大的 ID-二进制树

Fig. 4 Modifying  $T$  into a larger ID-binary tree

**引理 3.** 对于最大 ID-二进制树  $T$  中的任意叶节点, 其度为 1 的祖先节点个数等于  $k - \lfloor \log_2 N_2 \rfloor$  或  $k - \lfloor \log_2 N_2 \rfloor - 1$ .

**证明.** 由引理 1 和引理 2 可知, 由度为 2 的节点构成的树是一棵完全二叉树  $T^2$ , 在  $T^2$  中, 叶节点出现在  $\lfloor \log_2 N_2 \rfloor$  或  $\lfloor \log_2 N_2 \rfloor - 1$  层, 因为  $T$  高度为  $k$ , 所以以  $T^2$  中叶节点为根节点的子树的高度等于  $k - \lfloor \log_2 N_2 \rfloor$  或  $k - \lfloor \log_2 N_2 \rfloor - 1$ , 等于  $T$  中叶节点的度为 1 的祖先节点数.  $\square$

**定理 1.** 查询  $N$  个标签所需最大查询次数为

$$f_{\max}(Q(N, k)) = N(k - \lfloor \log_2(N - 1) \rfloor - 1) + 2^{\lfloor \log_2(N-1) \rfloor + 1} - 1 \quad (2)$$

**证明.** 由引理 2 可知, 在第  $\lfloor \log_2 N_2 \rfloor$  层, 度为 2 的节点数  $M = N_2 - 2^{\lfloor \log_2 N_2 \rfloor} + 1$ , 因而有  $k - \lfloor \log_2 N_2 \rfloor - 1$  个度为 1 的祖先节点的叶节点个数等于  $2M$ , 有  $k - \lfloor \log_2 N_2 \rfloor$  个度为 1 的祖先节点的叶节点数等于  $N - 2M$ , 由式 (1), 可得

$$\begin{aligned} f_{\max}(Q(N, k)) &= \\ &= (N_2 + 1 - 2(N_2 - 2^{\lfloor \log_2 N_2 \rfloor} + 1))(k - \lfloor \log_2 N_2 \rfloor) + \\ &+ 2(N_2 - 2^{\lfloor \log_2 N_2 \rfloor} + 1)(k - \lfloor \log_2 N_2 \rfloor - 1) + N - 1 \\ &= N(k - \lfloor \log_2(N - 1) \rfloor - 1) + 2^{\lfloor \log_2(N-1) \rfloor + 1} - 1 \end{aligned} \quad (3)$$

$\square$

### 2.3 最优查询次数

**定理 2.** SIBT 算法的最优查询次数  $f_{\min}(Q(N, k)) = \sum_{i=1}^k [N/2^i] + k - 1$ , 其中符号  $[\cdot]$  表示上取整, 而  $i$  取值为  $[N/2^i] = 1$ .

**证明.** 根据定义 3 可知, 当  $N$  个标签对应的 ID-二进制树  $T$  是一棵最小 ID-二进制树时, SIBT 算法有最优的查询性能. 因为  $T$  是最小 ID-二进制树, 所以每层内节点数最少, 对于  $k-1$  层最小的内节点数等于  $[N/2]$ , 以此类推, 第  $t$  层最少内节点数等于  $[N/2^t]$ , 得证.  $\square$

### 2.4 标签通信复杂度

在 SIBT 算法中, 标签只有在收到阅读器的位查询命令且处于活动状态时才会发送数据. 对于标签返回的数据, 阅读器将其保存在当前工作串或栈中. 每一个标签都无需重复发送已经发送过的位数数据, 在整个 CRI 过程中, 标签 ID 的每一位只需要发送一次. 所以每个标签总的数据传输量为  $S_t = k$ , 其中  $k$  为标签的 ID 号长度, 由于消除了在其他算法中无法避免的重复发送, 因此 SIBT 算法的标签通信复杂度低于现有防冲突算法.

## 3 仿真结果

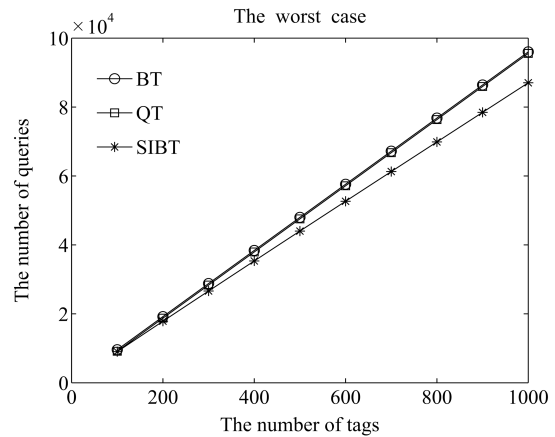
本节我们选择 RFID 二进制树协议中最常用的 BT 和 QT 算法与 SIBT 算法进行比较, 考虑的性能指标包括识别延迟和标签数据通信量. 选取指标的含义如下:

识别延迟用查询次数来衡量, 指阅读器识别  $n$  个标签需要发送的 query 指令总数, 每个查询周期包含 Reader-to-Tag 指令传送时间和 Tag-to-Reader 返回数据传送时间, 发送的 query 指令越少, 识别延迟就越小. 需要说明的是, QT 算法单个查询周期的延迟大于 BT 和 SIBT 算法, 原因是 QT 算法查询指令包含前缀码, 且标签返回的是一串比特位, 而 BT 算法查询指令只附加一位信息, 返回一位数据, SIBT 算法查询指令无附加信息, 返回一位数据. 所以对于一次查询延迟  $T_q$ , 有  $T_{q(QT)} > T_{q(BT)} > T_{q(SIBT)}$ .

标签数据通信量指单个标签在整个识别过程中传送的 ID 数据量, 其值与识别延迟及标签功率成正比.

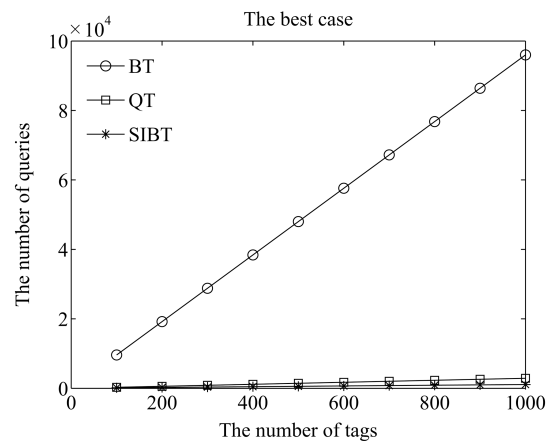
图 5 显示了在最坏、最好和随机三种情况下, 三种算法所需的查询次数. 从图 5 可以看出, 在最坏和最好情况下, SIBT 算法所需查询次数都明显少于 BT 算法, 也好于 QT 算法, 考虑到  $T_{q(QT)} > T_{q(SIBT)}$ , 所以在最坏和最好情况下 SIBT 算法的识别延迟都优于 QT 和 BT. 在随机情况下, 为了缩短

仿真时间且同时不影响可信度, 我们选取的标签长度为 10 位, 标签 ID 号随机生成, 从图 5(c) 可以看出, SIBT 查询次数明显少于 BT 和 QT.



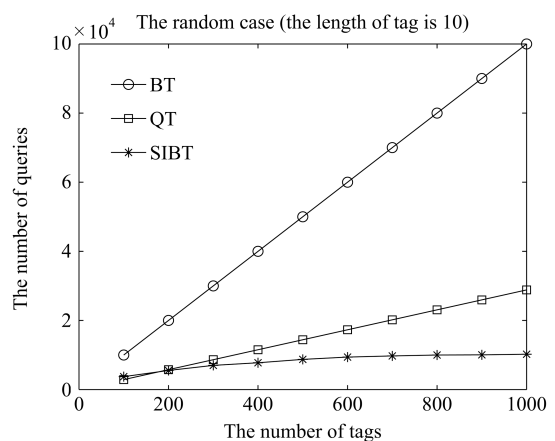
(a) 最坏情况

(a) The worst case



(b) 最优情况

(b) The best case



(c) 随机情况

(c) The random case

图 5 识别延迟

Fig. 5 Identification of delay

图 6 是 BT、QT 和 SIBT 算法单个标签数据通信量的仿真结果, 从图中可以看出 SIBT 算法单个标签数据通信量不受被识别标签数量的影响, 始终保持一个确定的值 (标签 ID 号长度), 说明在 SIBT 算法中, 标签冲突不影响标签的数据通信量. 由于在冲突情况下 BT 和 QT 算法都需要标签不断重传数据, 因此 BT 和 QT 算法随着被识别标签数量的增加, 即随着标签冲突的增加, 单个标签的数据通信量也随之增加.

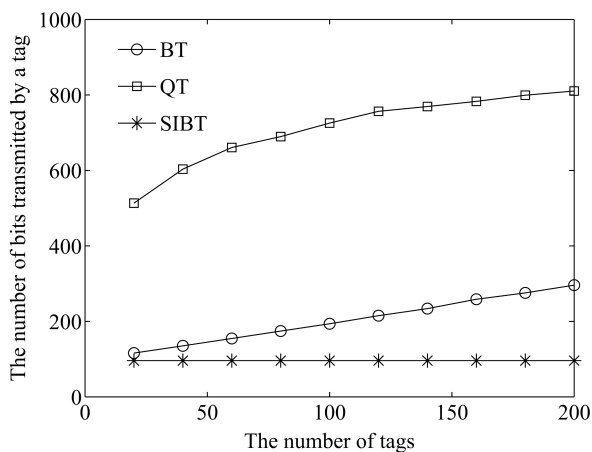


图 6 标签数据通信量

Fig. 6 Tag communication overhead

## 4 结论

RFID 多标签识别面临着与传统网络中信道共享类似的问题, 因此解决 Tag-to-Reader 的信道共享, 设计性能良好的防冲突算法, 协调多个标签与阅读器之间的通信, 就成为提高阅读器多标签识别效率, 减少标签漏读与错读的关键. 在本文中, 我们提出了一种新的 RFID 多标签识别防冲突算法 SIBT, 算法的核心在于通过 ID-二进制树, 将多标签识别过程转化为在阅读器中构建 ID-二进制树的过程. 通过阅读器使用栈保存构建线索, 标签使用栈计数器保存其状态及在栈中的位置等策略, SIBT 能提高 ID-二进制树的构建速度, 即多标签的识别速度, 同时算法使每个标签的数据通信量维持在一个确定的常量, 减低了每个标签在识别过程中所需传送的数据量. 理论分析和仿真结果表明, SIBT 算法优于目前广泛使用的基于二进制树的防冲突算法.

## References

- 1 Finkenzeller K. *RFID Handbook: Radio-frequency Identification Fundamentals and Applications (Second Edition)*. England: John Wiley and Sons, 2003. 6-9
- 2 Stanford V. Pervasive computing goes the last hundred feet

- with RFID systems. *Pervasive Computing*, 2003, **2**(2): 9-14
- 3 Satoh I. Linking physical worlds to logical worlds with mobile agents. In: *Proceedings of IEEE International Conference on Mobile Data Management*. California, USA: IEEE, 2004. 332-343
- 4 Want R. Enabling ubiquitous sensing with RFID. *Computer*, 2004, **37**(4): 84-86
- 5 Romer K, Schoch T, Mattern F, Dubendorfer T. Smart identification frameworks for ubiquitous computing applications. In: *Proceedings of the 1st IEEE International Conference on Pervasive Computing and Communications*. Texas, USA: IEEE, 2003. 253-262
- 6 Vogt H. Multiple object identification with passive RFID tags. In: *Proceedings of IEEE International Conference on Systems, Man, and Cybernetics*. Hammamet, Tunisia: IEEE, 2002. 6-9
- 7 RFID for Item Management—Air Interface, Part 3: Parameters for Air Interface Communications at 13.56 MHz, ISO Standard 18000-3, 2003
- 8 Philips. I · code 1 system design guide [Online], available: [http://www.nxp.com/acrobat\\_download/other/identification/SL048611.pdf](http://www.nxp.com/acrobat_download/other/identification/SL048611.pdf), May 6, 2008
- 9 RFID for Item Management—Air Interface, Part 6: Parameters for Air Interface Communications at 860 MHz to 960 MHz, ISO Standard 18000-6, 2004
- 10 Kleinrock L, Lam S S. Packet switching in a multi-access broadcast channel: performance evaluation. *IEEE Transactions on Communications*, 1975, **23**(4): 410-423
- 11 Auto-ID Center. Draft protocol specification for a 900 MHz class 0 radio frequency identification tag [Online], available: [http://www.epcglobalinc.org/standards/specs/900\\_MHz\\_Class\\_0\\_RFIDTag\\_Specification.pdf](http://www.epcglobalinc.org/standards/specs/900_MHz_Class_0_RFIDTag_Specification.pdf), May 6, 2008
- 12 Ching L, Kayi L, Yeung S K. Efficient memoryless protocol for tag identification. In: *Proceedings of the 4th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*. Boston, USA: ACM, 2000. 75-84
- 13 Jihoon M, Wonjun L, Srivastava J. Adaptive binary splitting for efficient RFID tag anti-collision. *IEEE Communications Letters*, 2006, **10**(3): 144-146
- 14 Pin W T. Enhanced binary search with cut-through operation for anti-collision in RFID systems. *IEEE Communications Letters*, 2006, **10**(4): 236-238
- 15 Hush D R, Wood C. Analysis of tree algorithms for RFID arbitration. In: *Proceedings of IEEE International Symposium on Information Theory*. Cambridge, USA: IEEE, 1998. 107
- 16 Vogt H. Efficient object identification with passive RFID tags. In: *Proceedings of the 1st International Conference on Pervasive Computing*. London, UK: Springer-Verlag, 2002. 98-113

- 17 Tsybakov B. Packet multiple access for channel with binary feedback, capture, and multiple reception. *IEEE Transactions on Information Theory*, 2004, **50**(6): 1073–1085
- 18 Capetanakis J I. The Multiple Access Broadcast Channel: Protocol and Capacity Considerations [Ph.D. dissertation], USA: Massachusetts Institute of Technology, 1978



**冯 波** 中国科学院计算技术研究所博士研究生. 主要研究方向为普适计算和 RFID 技术. 本文通信作者.

E-mail: bfeng@ict.ac.cn

(**FENG Bo** Ph.D. candidate at the Institute of Computing Technology, Chinese Academy of Sciences. His research interest covers pervasive computing and RFID technology. Corresponding author of this paper.)



**李锦涛** 博士, 中国科学院计算技术研究所研究员. 主要研究方向为普适计算和多媒体技术.

E-mail: jtli@ict.ac.cn

(**LI Jin-Tao** Ph.D., professor at the Institute of Computing Technology, Chinese Academy of Sciences. His research interest covers pervasive computing and multimedia technology.)



**郑为民** 博士, 中国科学院计算技术研究所副研究员. 主要研究方向为计算机体系结构和微处理器技术.

E-mail: zhengwm@ict.ac.cn

(**ZHENG Wei-Min** Ph.D., associate professor at the Institute of Computing Technology, Chinese Academy of Sciences. His research interest covers system architecture and microprocessor technology.)



**张 平** 中国科学院计算技术研究所博士研究生. 主要研究方向为普适计算和 RFID 技术. E-mail: zhp@ict.ac.cn

(**ZHANG Ping** Ph.D. candidate at the Institute of Computing Technology, Chinese Academy of Sciences. His research interest covers pervasive computing and RFID technology.)



**丁振华** 中国科学院计算技术研究所博士研究生. 主要研究方向为普适计算和 RFID 技术. E-mail: zhding@ict.ac.cn

(**DING Zhen-Hua** Ph.D. candidate at the Institute of Computing Technology, Chinese Academy of Sciences. His research interest covers pervasive computing and RFID technology.)