

## 基于二叉树的 RFID 防碰撞算法的研究

王雪, 钱志鸿, 胡正超, 李奕男

(吉林大学 通信工程学院, 吉林 长春 130025)

**摘 要:** 在二叉树算法的基础上提出了锁位后退防碰撞 (BLBO) 算法, 增加了锁位寻呼指令, 阅读器根据译码结果判断发生碰撞的比特, 发送锁位寻呼指令锁定发生碰撞的比特, 寻呼过程采用后退策略, 每次识别一个标签之后返回到上一个发生碰撞的节点。算法充分考虑了阅读器寻呼次数、传输时延、标签能耗以及吞吐量 4 个重要性能指标, 仿真结果表明, BLBO 防碰撞算法较其他二叉树算法性能有明显提高, 更适用于 RFID 防碰撞协议。

**关键词:** RFID; 锁位; 二叉树; 防碰撞

中图分类号: TN92

文献标识码: A

文章编号: 1000-436X(2010)06-0049-09

## Research on RFID anti-collision algorithms based on binary tree

WANG Xue, QIAN Zhi-hong, HU Zheng-chao, LI Yi-nan

(College of Communication Engineering, Jilin University, Changchun 130025, China)

**Abstract:** The bit-locking backoff (BLBO) anti-collision algorithm was proposed on the basis of binary algorithm, which puts forward the concept and orders of bit-locking. A reader recognizes the bits where there are collisions according to the results of decoding. Then the orders of bit-locking are transmitted to lock the bit collided, after which backoff strategy is adopted. When the reader recognizes one tag, it returns to the previous collided tag. The proposed algorithm fully takes the time of request into account, as well as transmission delay, power consumption and throughput of the system. The analysis on simulation result indicates that BLBO performs significantly better than the existing binary tree algorithms. It is suitable for the RFID anti-collision protocol in a greater deal.

**Key words:** RFID; bit-locking; binary tree; anti-collision

### 1 引言

RFID(radio frequency identification)利用射频信号通过空间耦合(交变磁场或电磁场)实现无接触信息传递并通过所传递的信息达到识别目的。RFID 技术是 20 世纪 90 年代兴起的自动识别技术, 较其他技术明显的优点是电子标签和阅读器无需接触便可完成识别<sup>[1]</sup>。

射频识别技术的一个主要优点就是多目标识别。在系统工作的时候, 阅读器周围可能会有多个标签同时存在, 当多个标签同时向阅读器传送数据的时候就产生了冲突问题。目前存在的 RFID 防碰撞算法主要有 2 种: 一种是基于 ALOHA 的不确定性算法, 另一种是基于二叉树<sup>[2]</sup>(BT, binary tree)的确定性算法。基于 ALOHA 的不确定性算法有个致命的缺点是标签容易出现“饿死”情况(即标签存

收稿日期: 2009-11-16; 修回日期: 2010-04-12

基金项目: 国家自然科学基金资助项目(60940010); 吉林省科技发展计划项目基础研究基金资助项目(20080524)

**Foundation Items:** The National Natural Science Foundation of China (60940010); The Basic Research of Science and Technology Development Program of Jilin Province (20080524)

在不能被识别的可能), 基于二叉树的确定性算法虽然解决了这种“饿死”情况, 但也存在着识别周期长、标签能耗大的问题。

文献[3]提出了基于动态帧时隙 ALOHA(DFSA)协议与正交可变扩频因子(OVSF)码作为扩频码的码分多址技术相结合的超高频 RFID 系统。文献[4]通过增加校验位来增强系统的防碰撞性能。BONUCCELLI M A 等人提出了时隙 ALOHA 算法<sup>[5]</sup>, 电子标签只能在规定的同步时隙内传输数据包, 对所有电子标签的同步由阅读器控制, 时隙 ALOHA 算法较 ALOHA 算法可能出现的碰撞时间只有一半。CHAJR 等人提出了动态帧时隙 ALOHA 算法<sup>[6]</sup>, 该算法根据阅读器周围标签数目动态调整帧的大小。PENG Q S 等人提出了增强型动态帧时隙 ALOHA 算法<sup>[7]</sup>, 这种算法是把标签分成多个簇, 每次只有一个簇与阅读器进行动态帧时隙防碰撞算法, 该算法较动态帧时隙算法性能上有较大提高。FINKENZELLER K 在 RFID 手册中提出二叉搜索树(BS, binary search)算法<sup>[8]</sup>, 标签根据碰撞信号的译码结果发出寻呼, 每识别出一个标签就返回到起始点, 这是二叉树算法中的经典算法之一。YU S S 等人在二叉搜索树算法的基础上提出了动态二叉搜索树(DBS, dynamic binary search)算法<sup>[9]</sup>, 在二叉搜索树算法的基础上将阅读器寻呼和标签响应所发送的比特数减少一半, 大大减少了传输时延。文献[10~13]分别也是在二叉搜索树算法基础上的改进算法, 它们在算法性能上各有利弊。LAW C 提出了无记忆寻呼树(QT, query tree)<sup>[14]</sup>算法, 标签除了记忆自己的 ID 信息外不需要记忆其他任何信息。ZHOU F 等人在文献[14]的基础上通过减少发生碰撞标签的响应次数来对寻呼树算法进行改进<sup>[15]</sup>, 改进的算法减少了标签的能耗。

针对目前二叉树算法中存在着识别周期长、标签能耗大的问题, 本文提出了锁位的概念, 通过锁位寻呼指令锁定碰撞发生比特位置, 在锁定的碰撞位上进行防碰撞运算, 采用后退策略识别碰撞节点下一个分支内的所有标签。本文组织结构如下: 第 2 节介绍锁位后退防碰撞算法的原理、指令与算法步骤; 第 3 节从阅读器寻呼次数、传输时延、标签能量以及吞吐量 4 个性能指标分析算法性能; 第 4 节对锁位后退算法进行了仿真及分析; 第 5 节为本文的结束语。

## 2 锁位后退防碰撞算法

假设阅读器周围有 3 个标签 A、B、C, 它们的 ID 分别为: 10110101、11110101、10100101。阅读器发送总寻呼 REQUEST(1111111111)指令之后, 译码结果为: 1X1X0101。二叉搜索树算法识别过程如图 1 所示, 动态二叉搜索树算法识别过程如图 2 所示。

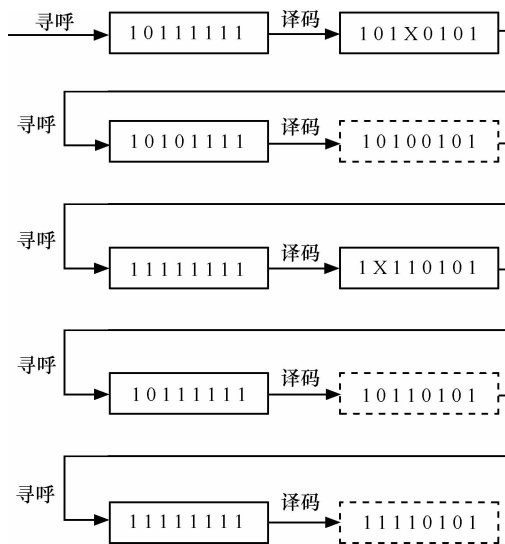


图 1 二叉搜索树算法识别过程

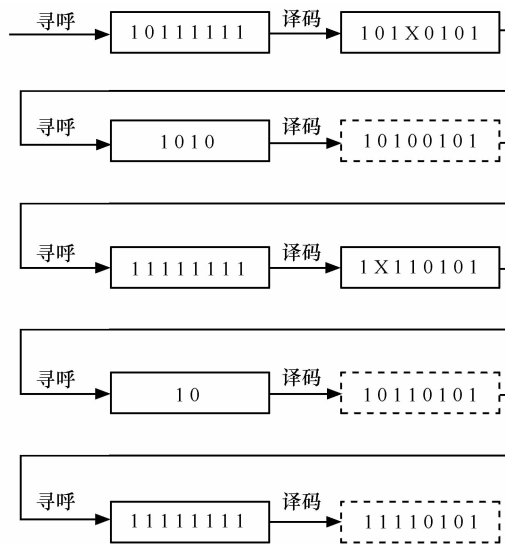


图 2 动态二叉搜索树算法识别过程

锁位后退防碰撞算法在以下 2 个方面进行改进。

### 1) 减少数据冗余位

阅读器在发送寻呼指令之后, 阅读器工作区域内的所有电子标签对此寻呼做出应答, 如果阅读器译码得到有  $h$  个位发生冲突, 显然只有这  $h$  个

比特对于阅读器来说是未知的，其他的比特对于标签是已知的。由图 1 所示的 BS 算法中，阅读器和电子标签每次发出的寻呼是整个序列号，含有的冗余信息太大，DBS 算法在 BS 算法的基础上减除了一半的冗余信息，但也没有达到最优化。BLBO 算法就是在此基础上继续减除寻呼中信息冗余位，以减少传输时延和能耗。例如上述所举的例子中，阅读器得到译码结果为 1X1X01010101，显然只有这 2 个 X 比特对于阅读器来说是未知的，其他的比特都是已知的，锁位后退防碰撞算法就是将防碰撞处理限制在这 2 个 X 比特上，不传输其他的比特，这样就在动态二叉搜索树算法的基础上进一步减少了数据冗余位。

## 2) 减少碰撞发生次数

如图 3 所示，假若阅读器识别电子标签 4 后，二叉搜索树和动态二叉搜索树算法都要返回到根节点去发送寻呼识别其他的电子标签，本算法采取的是后退策略，即识别标签 4 之后，返回到上一次发生碰撞的节点 3 去产生新的寻呼识别标签 5，这样就大大减少了碰撞发生的次数。

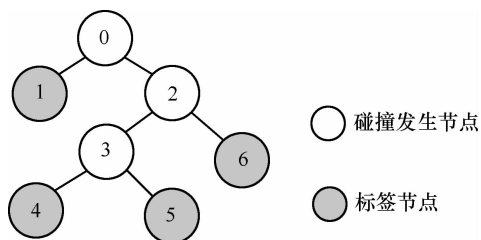


图 3 二叉树算法构成的树结构

## 2.1 锁位后退防碰撞算法的相关指令

为了实现这个算法，需要一组指令，这组指令由电子标签处理。此外，每个电子标签拥有一个唯一的序列号<sup>[8]</sup>。具体指令如下：

REQUEST (UID) ——请求 (序列号)；

SELECT (UID) ——选择 (序列号)；

READ-DATA ——读出数据；

UNSELECT ——去选择；

下面介绍 BLBO 算法需要新加的指令。

REQUEST (UID, 0) ——锁位寻呼，UID 代表阅读器在第一次寻呼之后，根据译码结果所得到的下一次寻呼的序列号，UID 的取值约定为：阅读器在判断出数据发生碰撞的准确比特位置之后，将碰撞发生的几个位置提取出来，并将几个碰撞比特置“1”，未发生碰撞的比特置“0”，组成新的锁定

寻呼指令的序列号。阅读器在发送这个寻呼指令之后，电子标签的响应为：标签在接到这个锁位命令之后，将自己 ID 中的数据位与接收到的阅读器发出的序列号进行比较，与阅读器发出的 UID 比特中值为“1”所对应的比特进行锁定，在接下来的防碰撞处理中，参与数据发送和比较的仅仅是这几个被锁定的比特。只有电子标签锁定的所有比特中最高比特的值为 0 的回送自己的 ID 给阅读器，并且返回锁定的比特中除最高位的其他几比特，最高比特与 X 值相同的不响应。

## 2.2 锁位后退防碰撞算法的工作流程

图 4 为 BLBO 算法的防碰撞处理流程，该流程分为 3 个小流程：阅读器发送 REQUEST (1111…1111) 命令时标签无碰撞发生时的流程、有碰撞发生时“0”分支处理流程和有碰撞发生时“1”分支处理流程（锁定的比特中最高比特为“0”的所有标签处在“0”分支，锁定的比特中最高比特为“1”的所有标签处在“1”分支）。BLBO 算法的主要步骤如下。

1) 阅读器发送 REQUEST (1111…1111) 命令，所有 ID 码值小于或者等于 (1111…1111) 的电子标签对此命令做出应答，然后所有应答标签将自己的 ID 码发送出去。

2) 阅读器检测收到的信号，如果没有信号，表示阅读器周围没有电子标签，则转到步骤 1)，否则转到步骤 3)。

3) 阅读器对所有电子标签做出的应答信号进行译码，根据译码结果判断是否有碰撞发生，如果没有碰撞发生，阅读器发送 SELECT 和 READ-DATA 指令，对标签进行读写操作之后，阅读器发出 UNSELECT 命令，使该标签进入无声状态；如果译码结果判断出有碰撞，则转到步骤 4)。

4) 阅读器根据步骤 3) 中的译码结果判断碰撞发生在哪几个比特上，阅读器将这几个碰撞的比特置“1”，未发生碰撞的比特置“0”，接着阅读器发送 REQUEST (UID, 0) 指令，标签在接到此命令之后将 UID 与自己的 ID 进行比较，将发生碰撞的比特锁定，锁定比特中最高比特为“0”的标签对此命令做出应答，将自己锁定比特中剩下的几比特发送给阅读器。阅读器判断是否有碰撞发生，如果没有碰撞发生，阅读器发送 SELECT 和 READ-DATA 指令，对标签进行读写操作之后，阅读器发出 UNSELECT 命令，使该标签进入无声状态。如

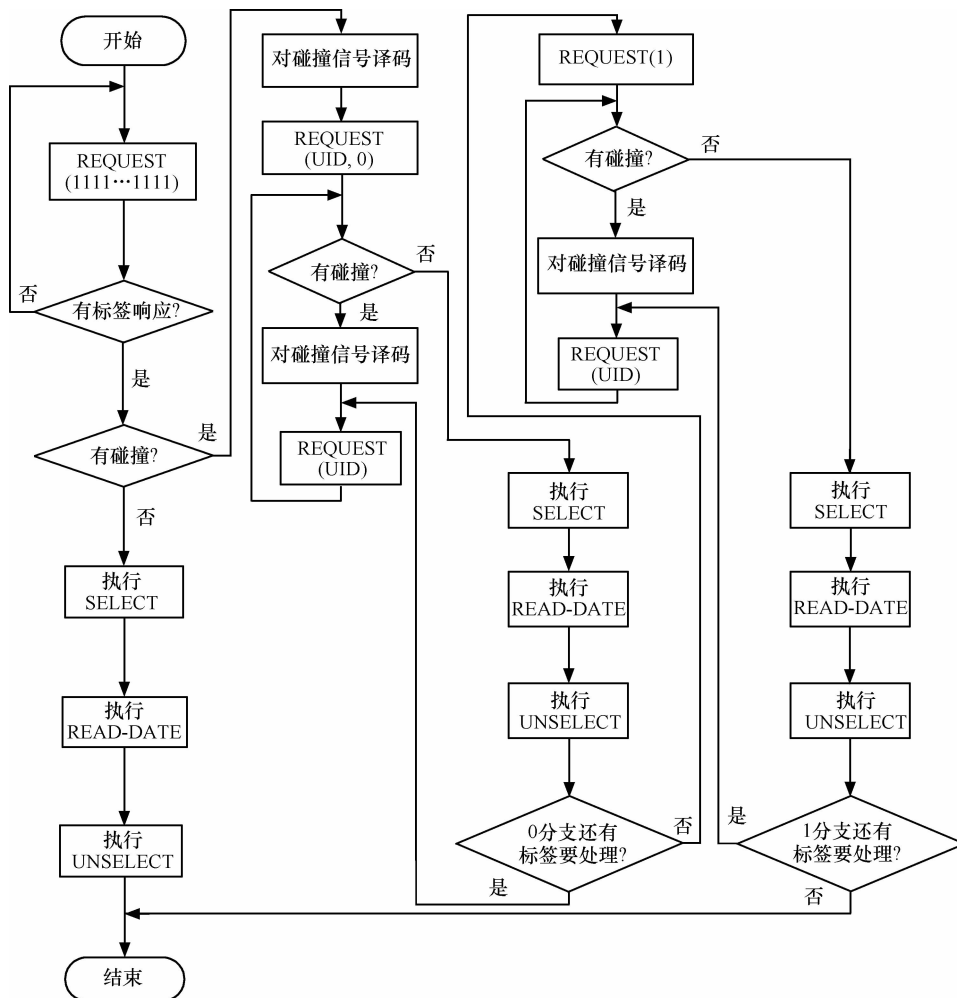


图 4 锁位后退防碰撞算法的工作流程

果有碰撞发生，阅读器对接收到的信号再进行译码，判断出发生碰撞的准确比特，将碰撞发生的最高比特置“0”，高于该比特的值不变，低于该比特的值舍去，在发生碰撞的这些标签中再次执行 REQUEST (UID) 命令。每次顺利读取某个标签之后，采取后退策略，返回到上一次发生碰撞的节点，识别此节点的另外一个分支，这样不断重复操作，直到把锁定的比特中最高比特为“0”这个分支内产生碰撞的所有标签识别完以后，转到步骤 5)。

5) 阅读器发送 REQUEST (1) 这个指令，锁定比特中最高比特为“1”的标签对此命令做出应答，将自己锁定比特中剩下的几比特发送给阅读器。阅读器判断是否有碰撞发生，如果没有碰撞发生，阅读器发送 SELECT 和 READ-DATE 指令，对标签进行读写操作之后，阅读器发出 UNSELECT 命令，使该标签进入无声状态。如果有碰撞发生，阅读器对接收到的信号再进行译码，判断出发生碰

撞的准确比特，将碰撞发生的最高比特置“0”，高于该比特的值不变，低于该比特的值舍去，在发生碰撞的这些标签中再次进行 REQUEST (UID) 命令。每次顺利读取某个标签之后，返回到上一次发生碰撞的节点，识别此节点的另外一个分支，这样不断重复操作，直到把锁定比特中最高比特为“1”的这个分支内产生碰撞的所有标签识别完以后，转到步骤 6)。

6) 待所有电子标签都被识别出来，识别过程结束。

### 3 算法性能分析

#### 3.1 阅读器的寻呼次数

若有  $n$  个电子标签在阅读器的工作区域内，为了识别  $n$  个电子标签阅读器所要发出的寻呼次数设为  $Q(n)$ ，识别过程中发生数据碰撞的次数设为  $C(n)$ ，则可以得到：

$$Q(n) = C(n) + n \quad (1)$$

$$Q(n) = 2n - 1 \quad (2)$$

下面证明式 (1) 和式 (2)。

**证明** BLBO 算法所形成的二叉树中所有节点的数量代表的是寻呼的总数  $Q(n)$ ，非叶子节点的总数代表数据碰撞的次数  $C(n)$ ，叶子节点的数量即为标签的个数  $n$ ，所以有  $Q(n) = C(n) + n$ ，即式 (1) 成立。

下面证明式 (2)，这里采用数学归纳法来证明。

1) 当  $n=1$  时，表示阅读器工作区域内只有一个电子标签，所以不会产生碰撞，很显然  $Q(1)=1$  成立。

2) 当  $n=2$  时，表示阅读器工作区域内有 2 个电子标签，此时标签至少有 1 个比特发生碰撞。无论有几个比特发生碰撞，由此算法得到，在发生碰撞的第一个比特分别置“0”和“1”就可以把 2 个电子标签区别开来，因此  $C(2)=1$ ，则  $Q(2) = C(2) + 2 = 3$ ，满足公式。

3) 假设有  $n$  个电子标签时结论成立，即  $Q(n) = 2n - 1$ ，那么当阅读器工作范围内有  $n+1$  个电子标签时，有如下结论：当第  $n+1$  个电子标签进入到阅读器的工作范围时，在本算法的二叉树上将增加一个分支，即多增加一个碰撞节点，也就是说碰撞的次数为  $C(n+1) = C(n) + 1$ ，于是可以得到：

$$\begin{aligned} Q(n+1) &= C(n+1) + n + 1 = C(n) + n + 2 \\ &= Q(n) + 2 = 2n + 1 = 2(n+1) - 1 \end{aligned} \quad (3)$$

由此得到有  $n+1$  个电子标签结论也成立，即式 (2) 成立。

### 3.2 传输时延

假设有  $n$  个电子标签的 UID 的长度为  $k$  bit，且冲突的比特数为  $x$ ， $x$  为阅读器第一次接收到来自电子标签的信号所译码判断出的碰撞的比特数，取值范围为  $[\text{Integ}(\text{lbn}), k]$ <sup>[16]</sup>。

BLBO 算法中 Type1 型节点个数为  $n-1$  个，Type2 型节点个数为  $n$  个。对树的搜索过程中，当  $n \geq 2$  时，阅读器第一次发出的寻呼为  $k$  bit，电子标签第一次回送的也是  $k$  bit，以后的寻呼中阅读器和电子标签发送的比特数为  $[0, x]$  bit，阅读器的第一次寻呼需要处于 Type1 型标签去经历  $[2k + 21]$  个时钟周期，第二次锁位指令需要处于 Type1 型标签经历  $[k + x + 21]$  个时钟周期，剩下的寻呼需要处于 Type1 型标签经历  $[x + 21]$  个时钟周

期。如果通信顺利未出现碰撞，阅读器额外需要 2bit 周期来登记电子标签 ID 信息，因此 Type2 的节点需要经历  $(x + 23)$  bit 周期。BLBO 算法在防碰撞处理过程中标签和阅读器需要传输的所有比特长度之和  $L_{\text{NEW}}$  为

$$\begin{aligned} L_{\text{NEW}} &= (2k + 21) + (k + x + 21) + \\ &\quad (x + 23)n + (x + 21)(n - 3) \\ &= 3k - 2x - 21 + (2x + 44)n \end{aligned} \quad (4)$$

当  $n=1$  时，BLBO 算法在防碰撞处理过程中标签和阅读器需要传输的所有比特长度之和  $L_{\text{NEW}} = 3k + 23$ 。

传输时延取决于阅读器发出寻呼的次数和每次发出寻呼的 UID 长度。BLBO 算法中要识别  $n$  个电子标签所需要的寻呼次数为  $Q_{\text{NEW}} = 2n - 1$ ，假设数据传输速率为  $v$  bit/s，则二叉搜索树算法识别  $n$  个电子标签的传输时延为

$$\tau_{\text{NEW}} = \frac{L_{\text{NEW}}}{v} = \frac{3k - 2x - 21 + (2x + 44)n}{v} \quad (n \geq 2) \quad (5)$$

$$\tau_{\text{NEW}} = \frac{L_{\text{NEW}}}{v} = \frac{3k + 23}{v} \quad (n = 1) \quad (6)$$

式 (5) 对  $x$  求导可以得到：

$$\tau'_{\text{NEW}} = \frac{2n - 2}{v} \quad (7)$$

由式 (7) 得出总的传输时延  $\tau_{\text{NEW}}$  对于变量  $x$  是增函数。

### 3.3 标签能耗

在文献[15]中，作者将等效电路的整流电路称为“Shottky 探测电路”。假设  $T$  为系统的时钟周期，对于任何一个  $(k_2 - k_1)T$  过程内， $k_1$  和  $k_2$  分别代表这个过程的开始和结束经历的时钟周期数，可以得到消耗的功率为

$$P(k_2, k_1) = \frac{V_{\text{dd}}^2 C_{\text{load}}}{(k_2 - k_1)T} \left( \sum_{i=k_1}^{k_2} A_i - \frac{C}{C_{\text{load}}} \frac{V_{\text{drop}}}{V_{\text{dd}}} \right) \quad (8)$$

其中， $A_i$  代表电路中第  $i$  时钟周期内“0”和“1”传输， $C_{\text{load}}$  为电路的平均负载电容， $V_{\text{dd}}$  为剩余功率， $V_{\text{drop}}$  为最大电压降。

为了公平比较，设时间周期  $T_{\text{INQ}}$ ，定义  $T_{\text{INQ}} = T \times \text{Cyc}$ ， $\text{Cyc}$  代表完成一次整体事件所需要的全部时钟周期，式 (8) 改写为

$$P(k_2, k_1) = \frac{C_{yc} V_{dd}^2 C_{load}}{(k_2 - k_1) T_{INQ}} \left( \sum_{i=k_1}^{k_2} A_i - \frac{C}{C_{load}} \frac{V_{drop}}{V_{dd}} \right) \approx \frac{C_{yc}}{(k_2 - k_1)} \left( \sum_{i=k_1}^{k_2} A_i - \frac{C}{C_{load}} \frac{V_{drop}}{V_{dd}} \right) \quad (9)$$

与 3.2 节所述的搜索过程类似，可得本算法的整个识别过程共需要的时钟周期为

$$C_{yc_{NEW}} = (2k + 3) + (k + x + 3) + (x + 3)(n - 3) + (x + 5)n = 3k + 8n - 3 + (2n - 2)x \quad (10)$$

$$k_2 - k_1 = 3k + 6 + mx + 4m \quad (11)$$

在本算法中，每个“接收”和“发送”操作分别产生 7 次和 8 次传输，2 个“NULL”操作需要产生 14 次和 9 次传输，译码操作产生 7 次传输，因此可以得到：

$$\begin{aligned} \sum A_i &= (14 + 9 + 8k + 7k) + (7k + 1) + \sum_{i=1}^m [14 + 9 + 7i + 8(x - i) + 7] + [14 + 9 + 7(m - 1)] \\ &= -\frac{1}{2}m^2 + \left(\frac{73}{2} + 8x\right)m + 40 + 22k \end{aligned} \quad (12)$$

由此可以得到当  $n \geq 2$  时，BLBO 算法的标签能耗为

$$P_{NEW} = \frac{C_{yc_{NEW}}}{(k_2 - k_1)} \left( \sum_{i=k_1}^{k_2} A_i - \frac{C}{C_{load}} \frac{V_{drop}}{V_{dd}} \right) = \frac{3k + 8n - 3 + (2n - 2)x}{3k + 6 + mx + 4m} \left[ -\frac{1}{2}m^2 + \left(\frac{73}{2} + 8x\right)m + 40 + 22k - \frac{C}{C_{load}} \frac{V_{drop}}{V_{dd}} \right] \quad (13)$$

对式 (13) 两边对变量  $x$  求导得到：当  $n \geq 2$  时，BLBO 算法的能耗  $P_{NEW}$  对于变量  $x$  是增函数。当  $n = 1$  时，标签的能耗为

$$P_{NEW} = \left( 23 + 15k - \frac{C}{C_{load}} \frac{V_{drop}}{V_{dd}} \right) \quad (14)$$

### 3.4 吞吐量

BLBO 算法的吞吐量为

$$S_{NEW} = \frac{n}{2n - 1} \times 100\% \quad (15)$$

## 4 仿真及分析

本文参考 ISO18000-6 标准，不计控制、前后缀、校验冗余等开销，在理想信道(error-free)条件

下进行仿真。标签 ID 均匀分布，长度固定为 64bit，标签数量在 0~90 间动态变化，比特率为 100kbit/s，30 次仿真取均值，对阅读器的寻呼次数、传输时延、标签能耗以及吞吐量 4 个性能指标进行分析，与 BS 算法和 DBS 算法进行了比较。

图 5 为 BLBO 算法与 BS 算法和 DBS 算法的阅读器寻呼次数比较，由图可以看出 BLBO 算法的阅读器寻呼次数明显小于 BS 算法和 DBS 算法，随着  $n$  值的增大，BLBO 算法较 BS 算法和 DBS 算法的优势越明显。

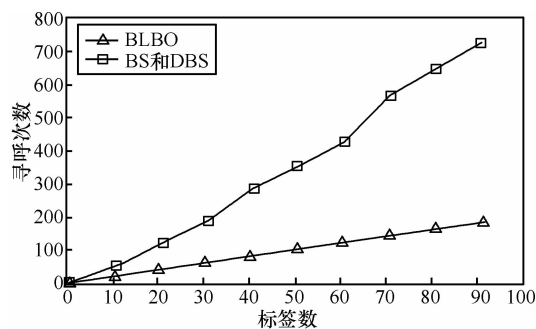


图 5 BLBO 算法与 BS 和 DBS 算法的阅读器寻呼次数比较

由此得到下面的结论：识别  $n$  个电子标签，在阅读器的寻呼次数方面 BLBO 算法少于 BS 算法，而 BS 算法的寻呼次数与 DBS 算法相当。

在数据传输速率为 100kbit/s，标签 ID 的长度  $k$  取值为 64bit 的条件下，碰撞发生的比特数  $x$  的取值分别为  $\text{Integ}(\text{lb}n)$ 、 $\frac{\text{Integ}(\text{lb}n) + k}{2}$ 、 $k$  时得到这 3 种

算法的传输时延随标签个数  $n$  的变化关系如图 6 所示。从图中可以看到无论  $x$  取区间  $[\text{Integ}(\text{lb}n), k]$  中的何值，即使取最大值  $k$ ，BLBO 算法的时延都要小于 BS 算法和 DBS 算法。由此得到下面的结论：在传输时延方面，BLBO 算法小于 DBS 算法，而 BS 算法时延最大。

文献[15]中给出了 2 种防碰撞算法的标签能耗，下面将本算法与这 2 种算法进行比较。假设  $\frac{C}{C_{load}} \frac{V_{drop}}{V_{dd}}$  值为 1，发生冲突的比特数  $x$  的取值分别

为  $\text{Integ}(\text{lb}n)$ 、 $\frac{\text{Integ}(\text{lb}n) + k}{2}$ 、 $k$  时得到 BLBO 算

法和 BS 算法以及 DBS 算法的标签能耗随标签个数  $n$  的变化关系的比较如图 7 所示。从图 7 中可以看出随着  $n$  值的增大，图 7 (a) 和图 7 (b) 中 BLBO 算法的标签能耗小于二叉树 BT 算法和寻呼

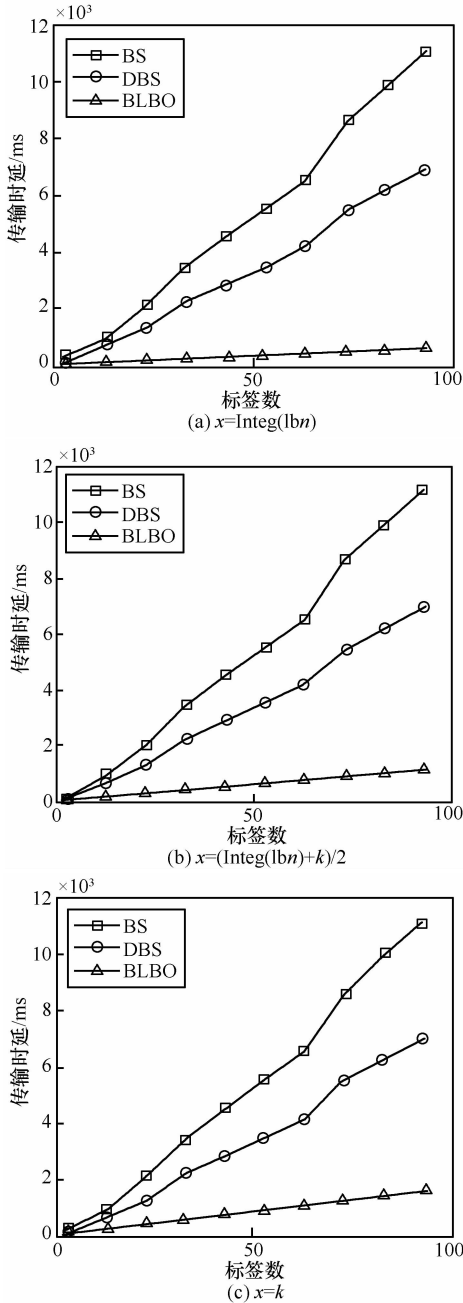


图 6 BLBO 算法和 BS、DBS 算法的时延比较

树 QT 算法。当  $x$  的值增大到  $k$  时，图 7(c) 中 BLBO 算法的标签能耗随着  $n$  值的增大逐渐趋近于 QT 算法，但仍优越于 BT 算法，一般情况下，发生冲突的比特数是小于  $k$  的， $x$  取  $k$  是极限值。

标签能耗的大小主要取决于阅读器的寻呼次数以及标签接收和发送的比特长度，由于 BS 算法和 DBS 算法阅读器的寻呼次数和标签每次接收和发送的比特长度都大于 BLBO 算法，又由于 BS 算法和 DBS 算法的寻呼次数都是一样的，而 DBS 算法中标签每次接收和发送的比特长度小于 BS 算

法，因此由文献[15]可以得到下面的结论：识别  $n$  个电子标签的标签能耗，BLBO 算法小于动态二叉搜索树算法，二叉搜索树算法能耗最高。

由此得到下面结论：识别  $n$  个电子标签，在标签能耗方面，BLBO 算法的能耗最小，BT 算法能耗高于 QT 算法，BS 算法能耗高于 DBS 算法。

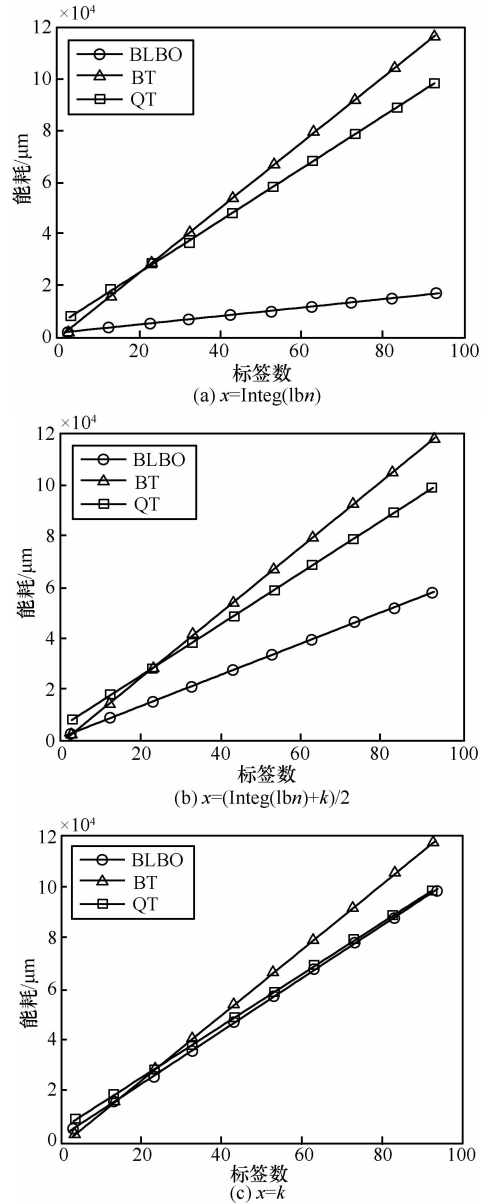
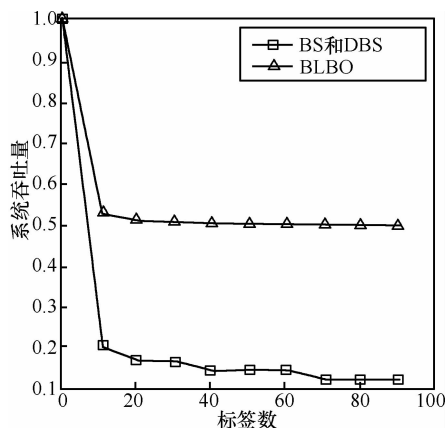


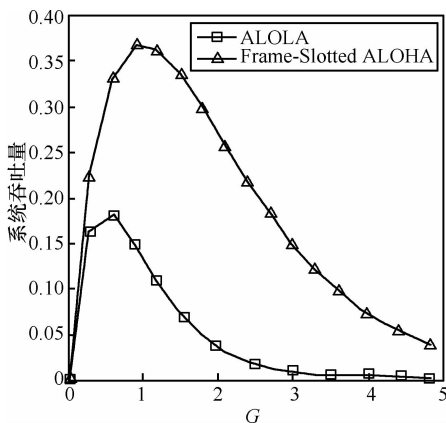
图 7 BLBO 算法与 BT、QT 算法的标签能耗比较

图 8(a) 为 BLBO 算法和 BS 算法以及 DBS 算法的吞吐量比较，从图中可以看出 BLBO 算法的吞吐量明显优越于 BS 算法和 DBS 算法，随着  $n$  的增大，BLBO 算法的吞吐量趋近于 50%。图 8(b) 为 ALOHA 算法和 Frame-Slotted ALOHA 算法的吞吐量的比较，从图中可以看到在  $G = 0.5$  时，ALOHA 算法的

吞吐量才得到最大值，且最大值仅为 18.4%，而 Frame-Slotted ALOHA 算法的吞吐量在  $G = 1$  时取得最大值，且最大值仅为 36.8%。



(a) BLBO 算法和 BS、DBS 算法吞吐量比较



(b) ALOHA 算法和 Frame-Slotted ALOHA 算法吞吐量比较

图 8 BLBO 算法和其他几种算法吞吐量比较

### 5 结束语

本文对目前存在的几种防碰撞算法进行了分析和比较，得出了它们各自的优缺点。在此基础上提出了“锁位”的概念，对发生碰撞的比特进行锁定，在锁定的比特上再进行二叉防碰撞处理，二叉防碰撞处理过程采用后退策略，即每识别一个标签后，返回到上一个发生碰撞的节点，通过对锁位后退防碰撞算法的性能进行分析和仿真，结果表明，锁位后退算法 BLBO 在阅读器的寻呼次数、传输时延、标签能耗以及系统吞吐量方面，都明显优于二叉搜索树算法等二叉树算法。

### 参考文献:

[1] FINKENZELLER K. RFID-Handbook Fundamentals and Applica-

tions in Contactless Smart Cards and Identification (2nd Edition)[M]. New York: Wiley and Sons, 2003.

[2] MIT Auto-ID Center. Draft protocol specification for a 900MHz class 0 radio frequency identification tag[EB/OL]. <http://auto-id.mit.edu>, 2003.

[3] 王必胜, 张其善. 可并行识别的超高频 RFID 系统防碰撞性能研究[J]. 通信学报, 2009, 30(6): 108-113.

WANG B S, ZHANG Q S. Study of anti-collision performance in parallelizable identification UHF RFID system[J]. Journal on Communications, 2009, 30(6): 108-113.

[4] KIM S S, KIM Y H, LEE S J, *et al.* An improved anti collision algorithm using parity bit in RFID system[A]. The 7th IEEE International Symposium on Network Computing and Applications[C]. 2008. 224-227.

[5] BONUCCELLI M A, LONETTI F, MARTELLI F. Tree slotted aloha: a new protocol for tag identification in RFID networks[A]. Proc of IEEE Int Symposium on a World of Wireless, Mobile and Multimedia Networks[C]. 2006. 603-608.

[6] CHA J R, KIM J H. Dynamic framed slotted ALOHA algorithms using fast tag estimation method for RFID system[A]. Consumer Communications and Networking Conference, CCNC 2006, 3rd IEEE[C]. 2006. 768-772.

[7] PENG Q S, ZHANG M, WU W M. Variant enhanced dynamic framed slotted ALOHA algorithm for fast object identification in RFID system[A]. Anti-Counterfeiting, Security, Identification, 2007 IEEE International Workshop[C]. 2007. 88-91.

[8] FINKENZELLER K. RFID Handbook: Fundamentals and Applications in Contactless Smart Cards and Identification, 2<sup>nd</sup> ed[M]. John Wiley & Sons Ltd, 2003.

[9] YU S S, ZHAN Y J, WANG Z P, *et al.* Anti-collision algorithm based on jumping and dynamic searching and its analysis[J]. Computer Engineering, 2005, 31:19-20.

[10] WANG T P. Enhanced binary search with cut-through operation for anti-collision in RFID systems[J]. IEEE Communications Letters, 2006, 10(4):236-238.

[11] CAPETANAKIS J. Tree algorithm for packet broadcast channels[J]. IEEE Trans Inform Theory, 1979, 25: 505-515.

[12] MEHLBORN K. Dynamic Binary Search[M]. Springer Berlin/Heideberg, 1977. 323-336.

[13] WANG C C, DERRYBERRY J, SLEATOR D D.  $O(\log \log n)$  competitive dynamic binary search trees[A]. Proceedings of the Seventeenth Annual ACM-SIAM Symposium on Discrete Algorithm[C].



2006. 374-383.

- [14] LAW C, LEE K, SIU K Y. Efficient memoryless protocol for tag identification[A]. Proceedings of the 4th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications[C]. Boston, Massachusetts, 2000.75-84.
- [15] ZHOU F, CHEN C, JIN D, *et al.* Evaluating and optimizing power consumption of anti-collision protocols for applications in RFID systems[A]. Proceedings of the International Symposium on Low Power Electronics and Design[C]. 2004.357-362.
- [16] COVER T M, THOMAS J A. Elements of Information Theory (2nd Edition)[M]. John Wiley & Sons, 2001.



钱志鸿（1957-），男，吉林长春人，吉林大学教授、博士生导师，主要研究方向为基于蓝牙、RFID 等短距离无线通信技术的无线个域网与无线传感器网络。



胡正超（1984-），男，江苏徐州人，吉林大学硕士生，主要研究方向为 RFID 技术与应用。

作者简介：



王雪（1984-），女，吉林靖宇人，吉林大学博士生，主要研究方向为 RFID 防碰撞机制。



李奕男（1978-），男，吉林长春人，吉林大学博士生，主要研究方向为 ad hoc 的安全机制。

（上接第 48 页）

Conference on Advances in Cryptology[C]. California, USA, 2001. 213-229.

- [16] 胡磊, 冯登国, 文铁华. 一类 Koblitz 椭圆曲线的快速点乘[J]. 软件学报, 2003, 14(11): 1907-1910.

HU L, FENG D G, WEN T H. Fast multiplication on a family of Koblitz elliptic curves[J]. Journal of Software, 2003, 14(11): 1907-1910.



高克宁（1964-），女，辽宁沈阳人，东北大学教授，主要研究方向为计算机网络及信息集成。

作者简介：



柳秀梅（1976-），女，辽宁本溪人，东北大学博士生，主要研究方向为信息安全。



常桂然（1946-），男，河北曲周人，东北大学教授、博士生导师，主要研究方向为计算机系统结构及计算机安全等。