

LRST: 低冗余搜索树防碰撞算法

黄琼¹, 凌江涛¹, 张敏², 阳小龙²

(1. 重庆邮电大学 移动通信技术重点实验室, 重庆 400065; 2. 北京科技大学 先进网络技术与新业务研究所, 北京 100083)

摘要: 针对 RFID 标签防碰撞树型算法在识别过程中因询问命令过多、过长而产生大量冗余数据导致通信开销过大的问题, 在后退式动态搜索树算法的基础上提出一种低冗余搜索树防碰撞算法 (LRST): 为减少询问次数, 提出了“一问两答”询问方式, 即碰撞标签根据最高碰撞位比特分别在第一个时隙或第二个时隙响应; 为减小询问命令的长度, 用计数器替代标签中的前缀匹配电路, 使算法不再需要前缀作为询问命令的标识参数; 此外, 提出的预测识别和标签屏蔽机制规避了不必要的询问。理论分析和仿真结果表明, 通信开销大大降低。

关键词: RFID; 防碰撞; 搜索树; 低冗余

中图分类号: TP393

文献标识码: A

文章编号: 1000-436X(2014)06-0110-07

LRST: searching tree anti-collision algorithm with low-redundancy

HUANG Qiong¹, LING Jiang-tao¹, ZHANG Min², YANG Xiao-long²

(1. Key Laboratory of Mobile Communication Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China;

2. Institute of Advanced Network Technology and New Services, University of Science and Technology Beijing, Beijing 100083, China)

Abstract: During the RFID tag identification process, the tree-based anti-collision algorithms usually incur large amount of redundant data due to an excess of long query commands, which increases the communication overhead. To resolve this problem, a searching tree anti-collision algorithm with low-redundancy on the basis of regressive-style dynamic searching tree algorithm was proposed. In order to reduce the number of queries, a novel query mode was developed, i.e., single query with duo responses. Depending on the most significant collided bit, the collided tags respond in the first or second slot separately. In order to reduce the length of query command, the prefix matching circuit in tag was replaced with a counter, which eliminated the prefix as the parameter of query command. The predictive identification and blocking technique were also introduced to avoid unnecessary queries. Theoretical analysis and simulation results show that the communication overhead is greatly reduced.

Key words: RFID; anti-collision; searching tree; low-redundancy

1 引言

射频识别(RFID, radio frequency identification)系统中, 多个标签同时响应阅读器请求时会发生碰撞。为了正确识别标签, 必须进行标签防碰撞处理。目前, 考虑到实现的复杂度和成本等因素, 其中时分多址(TDMA)是解决 RFID 系统碰撞问题的常用方法之一。基于 TDMA 的防碰撞算法主要有两大

类^[1,2]: 一类是基于 ALOHA 协议的算法; 另一类是基于树型的算法。虽然基于 ALOHA 协议的算法对标签硬件要求较低, 但它们的吞吐率低 (不超过 36.8%), 并会出现“标签饥饿”问题 (即某个标签可能一直与其他标签碰撞而无法被识别)。然而树型算法不仅吞吐率高, 且不存在“标签饥饿”问题。目前, 树型算法主要有分裂树、查询树、搜索树、逐位识别树等, 其中搜索树算法因采用了曼彻

收稿日期: 2013-05-06; 修回日期: 2013-12-06

基金项目: 国家重点基础研究发展计划 (“973”计划) 基金资助项目(2012CB315905); 国家自然科学基金资助项目(61171111, 61102063); 长江学者和创新团队发展计划基金资助项目(IRT1299); 重庆市科委重点实验室专项经费基金资助项目

Foundation Items: The National Basic Research Program of China (973 Program) (2012CB315905); The National Natural Science Foundation of China (61171111, 61102063); Changjiang Scholars and Innovative Research Team in University (IRT1299); The Special Fund of Chongqing Key Laboratory

斯特编码^[3], 多个标签发生碰撞时, 可通过译码结果确定每个碰撞比特的准确位置。因此, 阅读器可根据最高位碰撞比特的位置(最高碰撞位), 设置询问条件, 使得每次询问都会有标签响应, 避免了因空时隙带来的时间开销^[4]。

二进制搜索树(BST, binary searching tree)算法^[3]是基本的搜索树算法。阅读器发送以参考 ID (参考 ID 与标签 ID 序列等长, 初值为全比特 ‘1’) 为标识参数的询问命令, ID 序列小于或等于参考 ID 的标签响应询问命令。若仅一个标签响应, 则识别该标签; 若不止一个标签响应, 则阅读器通过 ID 碰撞信号的曼彻斯特译码结果确定最高碰撞位, 然后将参考 ID 的相应位置比特置 ‘0’, 再次发送带有新参考 ID 的询问命令。上述过程不断重复, 直到仅一个标签响应为止。被识别的标签进入静默状态, 不再响应阅读器的询问命令。该算法虽可解决标签碰撞问题, 但通信开销大。其因在于: 阅读器总以完整的参考 ID 序列作为询问命令的标识参数, 标签也总以完整的 ID 序列作为响应。为此, 在动态二进制搜索树(DBST, dynamic binary searching tree)算法^[5]的识别过程中, 阅读器仅将接收的 ID 序列最高碰撞位之前的高位部分(称为前缀)作为询问命令的标识参数。若标签 ID 序列的高位部分与标识参数相同(即前缀匹配), 则标签将 ID 序列的其余低位部分作为响应。DBST 算法采用“不定长 ID 序列”方式缓解了 BST 算法通信开销大的问题, 但吞吐率仍不高。其因在于: 阅读器识别一个标签后, 下次的询问命令仍以初始参考 ID 作为标识参数, 而没有记录以往询问过程的路径, 利用历史询问信息减少不必要的询问。为此, 后退式动态二进制搜索树(RDBST, regressive-style dynamic binary searching tree)算法^[6]在阅读器中设置一个“栈”, 每次碰撞都将前缀压入“栈”中。阅读器识别一个标签后, 抛出“栈”顶存放的前缀作为下次询问命令的标识参数。RDBST 算法采用“后退式”询问策略将询问次数由 $\lg(n!)+n$ (设 n 为标签数) 降为 $2n-1$ 。

目前的搜索树算法采用“后退式”询问策略大大减少了询问次数, 使吞吐率远高于其他算法; 采用“不定长 ID 序列”方式使阅读器询问和标签响应平均发送的比特数减半, 降低了通信开销, 但相比基于 ALOHA 协议的算法, 通信开销仍很大。为此, 本文致力于减少询问过程中产生的冗余数据,

以此降低通信开销。研究发现当阅读器检测到碰撞后, 会将碰撞标签分为 2 组, 对这 2 组标签分别进行询问, 而且询问命令将已识别的前缀作为标识参数, 不但增加了通信开销, 而且每次询问还要求所有未识别的标签都要进行前缀匹配, 而标签在响应时仅需获得最高碰撞位。

2 低冗余搜索树防碰撞算法

基于上文对搜索树防碰撞算法的分析, 本算法从阅读器询问次数、询问命令长度、响应时隙数 3 方面进行改进, 进一步减少询问过程中产生的冗余数据。

1) “一问两答”询问方式: 减少阅读器询问次数

在目前的搜索树算法中, 当阅读器检测到碰撞并发送询问命令后, 只有 ID 序列最高碰撞位比特为 ‘0’ 的碰撞标签响应, ID 序列最高碰撞位比特为 ‘1’ 的碰撞标签仍需阅读器再次发送一个询问命令进行询问, 这里将其称为“一问一答”询问方式。对于所有碰撞标签, ID 序列最高碰撞位之前的高位序列是相同的, 只是最高碰撞位比特分别是 ‘0’ 或 ‘1’。本文利用这一特点提出一种“一问两答”询问方式。当阅读器检测到碰撞并发送一个询问命令后, ID 序列最高碰撞位比特为 ‘0’ 的碰撞标签首先在第一个时隙响应, ID 序列最高碰撞位比特为 ‘1’ 的碰撞标签等待一个时隙后在第二个时隙响应(如图 1 所示)。这种“一问两答”询问方式不仅可将碰撞标签分为 2 组, 而且这 2 组标签的响应也仅需一次询问。

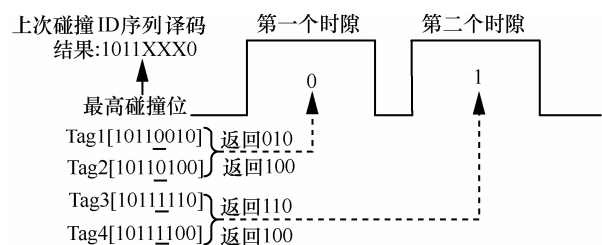


图 1 碰撞标签分别在 2 个时隙响应

2) 计数器“触发开关”: 减少询问命令中标识参数的长度

在目前的搜索树算法中, 标签接收到询问命令后, 根据其 ID 序列与询问命令中的前缀是否匹配而确定是否响应, 标签在响应时将 ID 序列最高碰撞位之后的低位部分发送给阅读器。因此, 标签中的前缀匹配只充当了标签响应的“触发开关”, 响应标签只需获得最高碰撞位。为减少询问命令中标识参

数的长度，本算法将前缀与最高碰撞位压入“栈”中，只将最高碰撞位作为询问命令的标识参数，用计数器替代前缀匹配电路作为标签响应的“触发开关”。计数器初始值为 0，计数器为 0 的标签才能响应询问命令，计数器不为 0 的标签处于等待状态。

计数器也为标签跟踪前缀在“栈”中的深度，使响应标签的 ID 与出“栈”的前缀相匹配，从而使阅读器接收到标签发回的 ID 序列最高碰撞位之后的低位部分后，加上出“栈”的前缀可组成一个完整的 ID 序列。识别过程中，当阅读器检测到碰撞并发送询问命令后，碰撞标签分别在对应时隙响应。第一个时隙若发生碰撞，将前缀与最高碰撞位压入“栈”中，并发送计数器命令 (counter)，所有标签的计数器加 1 (除等待第二个时隙的标签)。第二个时隙若发生碰撞，将新的最高碰撞位作为询问命令的标识参数继续询问 (询问命令为 query)。第二个时隙若成功识别标签，抛出“栈”顶存放的前缀与最高碰撞位，将最高碰撞位作为询问命令的标识参数进行后退式询问。注意，后退式询问时发送的询问命令为 re-query，所有标签的计数器先减 1，计数器减 1 后为 0 的标签才能在对应时隙响应。因此，计数器的变化规律为：在第一个时隙，若有前缀入“栈”，计数器加 1；在第二个时隙，若有前缀出“栈”，计数器减 1。

3) 预测识别：减少标签响应时隙数

阅读器对接收到的 ID 信号经曼彻斯特译码后可逐位识别出碰撞比特。若没有碰撞比特，则直接识别标签；若仅有一个碰撞比特，也可直接预测识别 2 个标签，可节省这 2 个标签响应所需的时隙。因为仅有一个碰撞比特时，只有 2 个标签发生碰撞，这 2 个标签的 ID 序列该位比特分别为 ‘0’ 和 ‘1’。文献[7]中的算法，先将 ID 序列的所有比特进行异或运算，将运算值加在 ID 序列的最高位组成一个新 ID 序列。当 ID 碰撞信号的译码结果只有 2 个碰撞比特时，阅读器可通过 ID 序列中所有未碰撞比特的异或值得到这 2 个碰撞比特。但这 2 个碰撞比特是由 2 个标签造成的，该算法中不会出现只有一个碰撞比特的情况。所以，只能在仅有 2 个碰撞比特的情况下直接识别 2 个标签。因此，该算法增加了复杂度，效果却与本文算法相同。

4) 标签屏蔽机制：避免对已被识别的标签再次进行识别

在移动阅读器应用中，阅读器扫描完某一区

域的标签后，会扫描另一个区域的标签。若这 2 个区域的重叠部分存在标签，它仍会扫描已被识别的标签 (如图 2 所示)。为避免再次询问已被识别的标签，本文引入一种标签屏蔽机制：标签预留一个存储区，存储已识别该标签的读者的序列号 (RID)。阅读器扫描标签时，首先发送以该阅读器 RID 为参数的初始化命令 (initial)。收到该命令后，标签将命令中的 RID 与存储的 RID 进行比较。若相同，表明它已被该阅读器识别，则进入静默状态不再响应；若不同，表明它未被该阅读器识别，则用接收的 RID 替换原存储的 RID，在初始化计数器后进行响应。通过该机制可屏蔽已被识别的标签，减少对已被识别的标签再次识别而带来的通信开销。

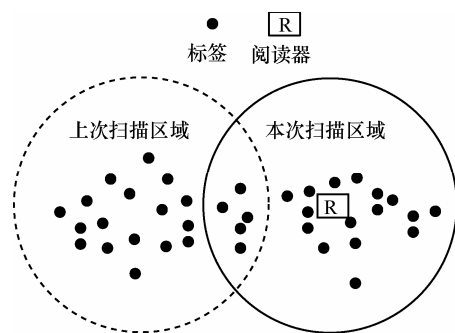


图 2 移动阅读器应用场景

为防止在识别过程中到达的标签不能被识别，本算法在一次扫描后进行二次扫描。因为搜索树算法的询问次数与标签数有关，识别过程中的新到标签又是少量的，所以会瞬间完成第二次扫描。二次扫描后，若没有标签响应，便停止扫描；若仍有标签响应，识别完标签后，再次扫描，直到不再有标签响应。

3 低冗余搜索树算法具体实现

图 3 为低冗余搜索树算法的工作流程，该流程分为 3 个小流程：阅读器发送 initial(RID)初始化命令，上次被成功识别的标签进入静默状态，未被识别的标签初始化计数器并响应；阅读器开始迭代询问过程，直到“栈”为空；当“栈”为空时，再次扫描，直到无标签响应，结束识别。

算法主要步骤如下。

1) 阅读器发送 initial(RID)初始化命令，若标签存储的 RID 与命令中的 RID 相同，则标签进入静默状态；若不相同，则标签将计数器初始化为 0，并发送完整 ID 序列作为响应。

点(图 4 中阴影圆形), 不必再进行询问即可识别 2 个标签(图 4 中阴影方形)。因此, 询问总数再减去只有一个碰撞比特的碰撞节点数。若只有一个碰撞比特的碰撞节点数为 $m(m \leq \text{floor}(n/2))$, floor 为向下取整), 则本算法的询问总数为

$$Q' = 1 + C(n) - m = n - m \quad (2)$$

当标签群的 ID 序列连续分布时, 每两个标签的 ID 序列之间只有一个比特不同, 每两个标签可通过预测识别, 则 m 达到最大值 $\text{floor}(n/2)$, 此时, 询问总数最少

$$Q' = n - \text{floor}(n/2) = \text{ceil}(n/2) \quad (3)$$

其中, ceil 为向上取整。由式(1)和式(2)可知, 与当前算法相比, 本算法至少可减少一半的询问次数。

3) 询问命令中标识参数的长度

当前的算法采用前缀作为询问命令的标识参数, 若标签的 ID 序列为 k 位, 则询问命令中标识参数的平均长度为

$$L = 1 \frac{1}{k} + 2 \frac{1}{k} \dots k \frac{1}{k} = \frac{k+1}{2} \quad (4)$$

本算法询问命令的标识参数为最高碰撞位。因此, 表示最高碰撞位所需的比特数即为标识参数的长度, 则询问命令中标识参数的长度为

$$L' = \text{lb}k \quad (5)$$

从式(4)和式(5)可看出 L' 明显小于 L , 且随着 k 的增加, 它们的差距也随之增大。

4) 通信开销

为公平对比, 不考虑在移动阅读器应用中, 标签屏蔽机制带来的明显收益。假设除标识参数外询问命令的其他字段为 s bit, 标签响应时附加的前导码等信息为 t bit。因标签在响应时只发送 ID 序列最高碰撞位之后的低位部分, 所以每个响应标签平均发送的 ID 序列为 L bit(式(4)), 则返回式动态搜索树算法识别 n 个标签所需的通信开销为

$$\begin{aligned} S &= Q[(L+s) + (L+t)] \\ &= (2n-1)\left(\frac{k+1}{2} + s + \frac{k+1}{2} + t\right) \\ &= (2n-1)(k+1+s+t) \end{aligned} \quad (6)$$

本算法所需的通信开销为

$$\begin{aligned} S' &= Q'(L'+s) + 2Q'(L+t) \\ &= (n-m)\left[\text{lb}(k) + s\right] + 2(n-m)\left(\frac{k+1}{2} + t\right) \\ &= (n-m)\left[\text{lb}(k) + s + 2t + k + 1\right] \end{aligned}$$

$$= (2n-2m)\left[\frac{\text{lb}(k) + k + 1 + s}{2} + t\right] \quad (7)$$

对比式(6)和式(7), 可看出 S' 小于 S , 且随 n 、 k 的增加, 它们的差距也随之增大。因此本算法明显降低了识别标签所需的通信开销。

5 算法仿真与分析

本文从询问次数、询问命令长度、吞吐率与通信开销 4 个方面, 将 LRST 算法与 RDBST 算法^[6]和 FST 算法^[7]进行性能比较。为简单起见, 仿真中不考虑控制、前后缀和校验冗余等带来的通信开销, 而主要将询问命令中的标识参数与标签发送的 ID 序列作为识别标签所需的通信开销, 并定义吞吐率为标签总数与识别标签所需的总时隙数之比。因识别过程与标签群的 ID 分布有关, 分别对 16 位 ID 随机分布、连续分布 2 种情况进行仿真。

从图 5 和图 6 可看出, 标签群的 ID 无论是随机分布还是连续分布, LRST 算法的询问次数都少于其他 2 种算法, 且仅为 FST 算法的一半。这表明

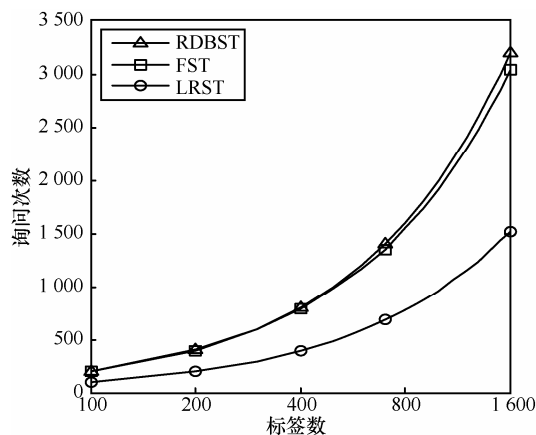


图 5 询问次数比较(ID 随机分布)

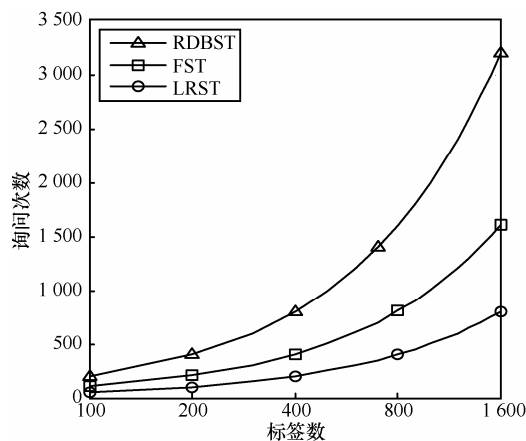


图 6 询问次数比较(ID 连续分布)

本算法提出的“一问两答”询问方式可使整个询问过程的询问次数减少一半。

图 7 为前缀、最高碰撞位分别作为询问命令标识参数的情况下, 标识参数长度的变化曲线。LRST 算法采用计数器作为标签响应的“触发开关”, 询问命令的标识参数为最高碰撞位。从图 7 可看出, 当 ID 序列的长度翻倍时, 询问命令的标识参数仅需增加一个比特; 而其他 2 种算法采用前缀匹配电路作为标签响应的“触发开关”, 则需将前缀作为询问命令的标识参数, 当 ID 序列的长度翻倍时, 询问命令的平均标识参数也随之翻倍。当 ID 序列为 16 bit 时, LRST 算法采用最高碰撞位作为询问命令的标识参数仅需 4 bit, 而其他 2 种算法采用前缀作为标识参数平均需 9 bit。这表明采用最高碰撞位替代前缀作为询问命令的标识参数可大大减小询问命令的长度。

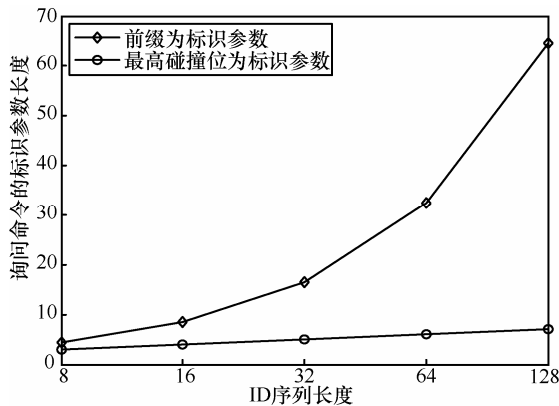


图 7 询问命令中标识参数的长度比较

从图 8 可看出, RDBST 算法的吞吐率(u)接近 50%, 符合理论公式: $u=n/(2n-1)$ 。FST 算法与 LRST 算法的吞吐率明显高于 RDBST 算法的吞吐率, 由吞吐率定义可知, 预测识别明显减少了时隙数。从图 9 可看出, RDBST 算法的吞吐率仍为 50%, LRST 算法与 FST 算法的吞吐率接近 1。这说明预测识别可利用标签群 ID 的分布特性, 提高吞吐率。从图 8 和图 9 可看出, LRST 算法与 FST 算法的吞吐率曲线基本重合。这表明 2 种预测识别在减少时隙数方面基本相同, 因此, LRST 算法中的简单预测识别可替代 FST 算法中的复杂预测识别。

图 10 和图 11 为系统识别标签所需的通信开销, 通信开销可反映时延与功耗^[8,9]。从图 10 可看出 LRST 算法的通信开销明显低于其他 2 种算法。LRST 算法的通信开销比 RDBST 算法降低了约 42%, 比 FST 算法降低了约 38%。标签群的 ID 连

续分布时, LRST 算法达到最优。从图 11 可看出, LRST 算法的通信开销比 RDBST 算法降低了约 69%, 比 FST 算法降低了约 39%。这表明标签群的 ID 无论是随机分布还是连续分布, LRST 算法都明显降低了通信开销。

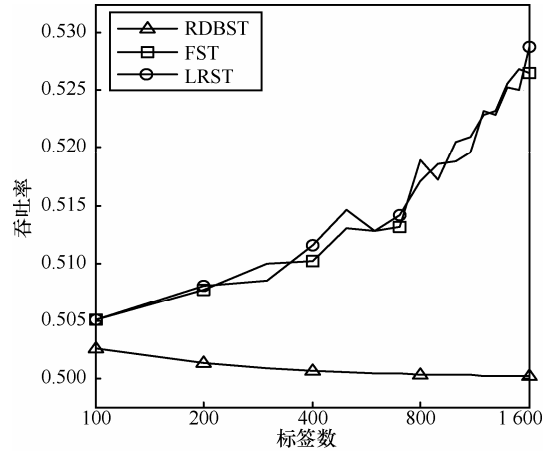


图 8 吞吐率比较(ID 随机分布)

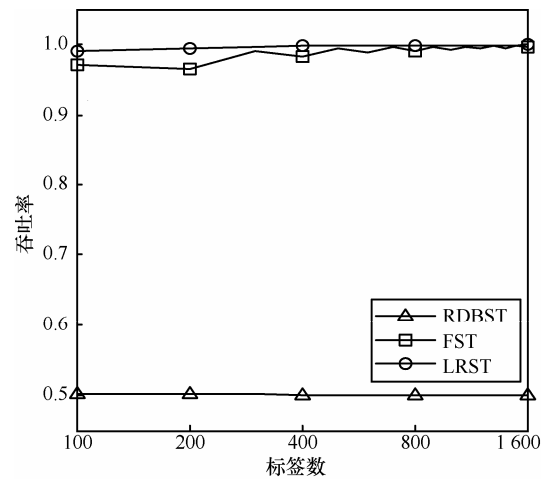


图 9 吞吐率比较(ID 连续分布)

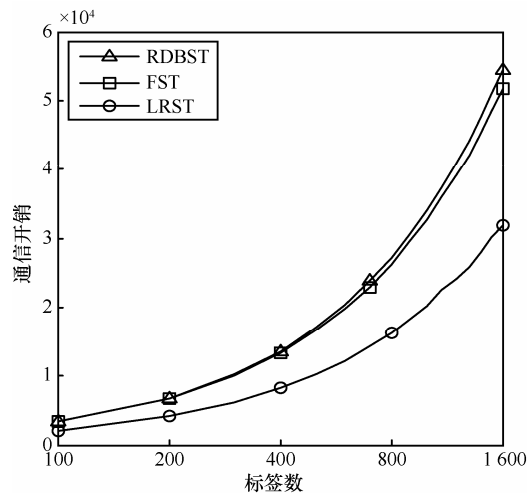


图 10 通信开销比较 (ID 随机分布)

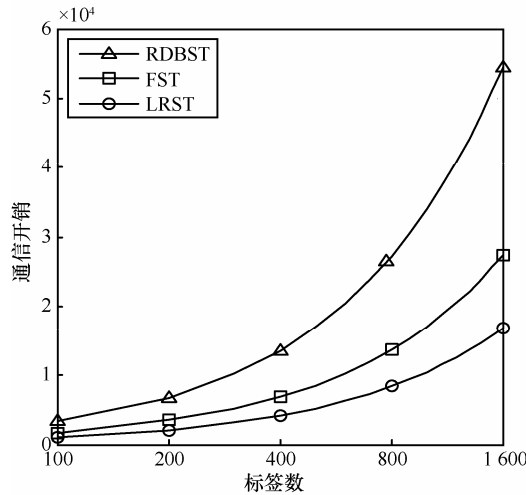


图 11 通信开销比较 (ID 连续分布)

由仿真可看出，当标签群的 ID 连续分布时，LRST 算法表现出更好的优越性。在很多应用中，标签群的 ID 是连续分布的，例如仓库管理、生产线等。有些算法加入预处理机制^[10]或者锁定发生碰撞的比特后再执行搜索树算法^[11]，每次询问后将 ID 序列中未识别的部分组成一个新 ID，以达到减少传输比特的目的。它只能在识别少量标签时，节省某些比特。在识别大量同类商品的标签时，节省的仍是 ID 序列高位部分的比特。因此，这种机制增加了标签复杂度，却得不到很好的效果。

6 结束语

本文研究了 RFID 系统中的标签防碰撞算法，对搜索树算法及其改进算法进行了分析，并在此基础上提出了一种低冗余搜索树防碰撞算法。该算法能大幅度减少阅读器的询问次数和询问命令中标识参数的长度，从而降低了系统的通信开销，减小了时延，降低了功耗。当接收到的 ID 序列中只有一个碰撞比特时，可直接识别 2 个标签，进一步减少了时隙总数，提高了吞吐率。仿真结果表明，标签群的 ID 无论随机分布还是连续分布，本算法都提高了吞吐率，且明显降低了通信开销。因此，本算法能够迅速有效地识别标签。

参考文献:

[1] LIU H C. The approaches in solving passive RFID tag collision problems[J]. Radio Frequency Identification Fundamentals and Applications, 2010, 3:49-56.
 [2] KLAIR D K, CHIN K W, RAAD R. A survey and tutorial of RFID anti-collision protocols[J]. IEEE Communications Surveys & Tutorials,

2010, 12(3):400-421.
 [3] FINKENZELLER K. RFID-Handbook Fundamentals and Applications in Contactless Smart Cards and Identification (2nd Edition)[M]. HoboRen: Jahn: Wiley and Sons Inc, 2003.
 [4] SHIH B Y, CHEN C W, CHEN C Y. Merged search algorithms for radio frequency identification anticollision[J]. Mathematical Problems in Engineering, 2012, 2012:1-20.
 [5] YU S S, ZHAN Y J, WANG Z P. Anti-collision algorithm based on jumping and dynamic searching and its analysis[J]. Computer Engineering, 2005, 31:19-20.
 [6] HU J G, LIN K, WANG D M. A novel anti-collision algorithm for RFID system[A]. Program for the IEEE International Conference on RFID-Technology and Applications[C]. Guangzhou, China, 2010. 302-307.
 [7] KIM S S, KIM Y H, LEE S J. An improved anti collision algorithm using parity bit in RFID system[A]. Korea Seventh IEEE International Symposium on Network Computing and Applications[C]. 2008. 224 - 227.
 [8] JIA X L, FENG Q Y, YU L H. Stability analysis of an efficient anti-collision protocol for RFID tag identification[J]. IEEE Transactions on Communications, 2012, 60(8):2285-2294.
 [9] DANG F, YANG X L, LONG K P. Spray and forward: efficient routing based on the Markov location prediction model for DTN[J]. Science China Information Sciences, 2012, 55(2):433-440.
 [10] 周艳聪, 孙晓晨, 顾军华. 一种改进二进制防碰撞算法研究[J]. 计算机应用研究, 2012, 29(1):256-262.
 ZHOU Y C, SUN X C, GU J H. Research on improve binary anti-collision algorithm[J]. Application Research of Computers, 2012, 29(1):256-262.
 [11] 王雪, 钱志鸿, 胡正超. 基于二叉树的 RFID 防碰撞算法的研究[J]. 通信学报, 2010, 31(6):49-57.
 WANG X, QIAN Z H, HU Z C. Research on RFID anti-collision algorithms based on binary tree[J]. Journal on Communications, 2010, 31(6):49-57.

作者简介:



黄琼 (1971-), 女, 四川西昌人, 硕士, 重庆邮电大学教授, 主要研究方向为个人通信、光互联网技术。

凌江涛 (1987-), 男, 山东潍坊人, 重庆邮电大学硕士生, 主要研究方向为 RFID 系统防碰撞算法。

张敏 (1972-), 女, 重庆人, 北京科技大学副教授, 主要研究方向为互联网理论与技术、IP 网络安全等。

阳小龙 (1970-), 男, 四川邻水人, 博士, 北京科技大学教授、博士生导师, 主要研究方向为互联网理论与技术、网络安全、光交换与光互联。