

# 一种改进的自适应多叉树防碰撞算法

王 汉 武 于 涛

(湖南大学信息科学与工程学院 长沙 410082)

**摘 要** 针对传统自适应多叉树防碰撞算法在标签识别过程中存在的空闲时隙过多、阅读器与电子标签间的通信负载量过大等不足,提出了一种改进的自适应多叉树防碰撞算法(Improved Adaptive division Collision Tree algorithm, IACT)。该算法通过计算碰撞因子决定采用二叉树或四叉树。当采用二叉树时,若阅读器检测到碰撞位只有一位,则无需再次发送命令即可以直接识别出标签;采用四叉树时,阅读器首先发送一命令,要求标签返回最高两个碰撞位对应的编码,然后根据编码得到碰撞信息。在标签中加入计数器,使用最高两个碰撞位和计数器值作为查询命令,响应的电子标签将序列号的后缀信息发送给阅读器处理。算法性能分析和实验仿真表明,IACT 算法能有效减少系统总时隙,降低了通信负载开销,提高了标签识别效率。

**关键词** 标签识别算法,防碰撞,自适应多叉树,碰撞位

中图分类号 TN92 文献标识码 A DOI 10.11896/j.issn.1002-137X.2018.11.008

## Enhanced Adaptive Division Anti-collision Algorithm

WANG Han-wu YU Tao

(College of Information Science and Engineering, Hunan University, Changsha 410082, China)

**Abstract** Aiming at the disadvantages of the traditional adaptive division collision tree algorithm in the process of tag identification such as many idle time slots, large communication overhead between the reader and tags, this paper presented an improved adaptive division collision tree algorithm (IACT). The algorithm determines the adoption of the binary-tree or quad-tree by calculating collision factor. As for the binary-tree, if the collision-bit is only one, tags can be identified directly without sending commands again. As for the quad-tree, reader first sends a command for tags to return the corresponding coding of the highest two collision bits, and obtains collision information through encoding. The reader takes the value of counter and the highest two collision bits as the parameter of query command by using the counter, and tags only send the postfix of ID to the reader. The performance analysis and simulation results show that IACT algorithm can effectively reduce the total timeslot consumption and the communication load, and improve the recognition efficiency as well.

**Keywords** Tag identification algorithm, Anti-collision, Adaptive division collision tree, Collision bits

## 1 概述

无线射频识别(Radio Frequency Identification, RFID)是一种运用无线通讯信号识别目标并通过非接触方式读写相关目标信息的通信技术,具有适应性强、不需要人为操作、可以同时多个目标进行识别等技术优点<sup>[1]</sup>。RFID 系统主要由阅读器、电子标签和计算机处理系统组成<sup>[2]</sup>,阅读器和电子标签之间通过传送无线通讯信号来进行数据通信。当多个电子标签同时应答同一个阅读器时,各信号会相互干扰,使得阅读器不能正确读取电子标签传送的数据信息,从而产生“标签碰撞”问题<sup>[3]</sup>。

到目前为止,标签防碰撞算法主要分为基于 ALOHA 的概率型算法<sup>[4-5]</sup>和基于树的确定性算法<sup>[6]</sup>。基于 ALOHA 的算法操作简单,并且对新到达的标签具有较好的适应性,但是该算法的吞吐率较低,容易出现“饥饿现象”。

## 2 基于树的确定性算法

基于树的确定性算法的基本思想是,当处于工作范围内的电子标签收到阅读器的查询命令后,从最高位依次与其进行匹配,匹配成功的标签才响应并将数据信息发送给阅读器。基于树的算法主要包括基本二进制树算法、动态二进制树算法、后退式二进制树算法、查询树算法和四叉树算法等。基本

收稿日期:2017-09-15 返修日期:2017-12-13 本文受国家自然科学基金面上项目:WiMAX 网络中信道时变特性明确及信道传输优化方案的研究(61472129)资助。

王汉武(1980—),男,副教授,主要研究方向为高速无线网络、资源管理等,E-mail:whanwu1@hnu.edu.cn(通信作者);于涛(1989—),女,硕士生,主要研究方向为通信网络。

二进制树算法是基于树分裂无记忆的基本查询算法<sup>[7-9]</sup>。阅读器发送长度和标签等同的初始序列号作为询问指令,只有小于或等于该指令的标签才响应。为减少冗余,在二进制基础上提出一种改进的防碰撞算法——动态二进制算法<sup>[10]</sup>,该算法只需将最高碰撞位之前的序列号  $q_1 q_2 \cdots q_{u-1}$  作为下一次发送的查询指令。“后退式”<sup>[11-12]</sup>算法提出在成功识别出标签后,阅读器无需回到初始状态,而是回退至前一次发生碰撞的位置,减少了阅读器的查询次数。

自适应多叉树防碰撞算法是将二叉树和四叉树结合形成的新型防碰撞算法<sup>[13]</sup>。针对目前算法存在的通信开销大、识别效率低等问题,文中提出了一种改进的自适应多叉树防碰撞算法(Improved Adaptive division Collision Tree, IACT)。该算法通过碰撞因子动态选择防碰撞算法多叉树,为避免出现空闲时隙,对四叉树进行前缀先验;在标签中加入储存机制,存储碰撞的最高位信息和状态标志位 flag。当 flag 为 0 时,标签处于待识别状态;当 flag 为 1 时,标签处于屏蔽状态,不响应阅读器的查询命令,可以有效减少阅读器和标签的重复命令,从而减少通信开销。

假设系统中有  $m$  个待识别标签,每个标签序列号的长度为  $k$ ,其中发生碰撞位的位数为  $k_c$ ,由文献[13]可知碰撞因子为  $\mu = k_c/k$ 。假设没有碰撞发生的概率为  $p = (\frac{1}{2})^{m-1}$ ,则可以推导出碰撞因子的取值为  $\mu = 1 - (\frac{1}{2})^{m-1}$ 。其中,没有碰撞发生的概率为:

$$P(\frac{0}{m}) = (\frac{1}{2})^{m-1} \quad (1)$$

由式(1)可以得到碰撞因子:

$$\mu = 1 - (\frac{1}{2})^{m-1} \quad (2)$$

以上公式表明碰撞因子与标签数量有密切关系,待识别标签数量  $m$  越大,碰撞因子  $\mu$  越小,反之碰撞因子  $\mu$  越大。

系统中用  $B$  表示节点的分支数,  $L$  表示搜索深度,则如果  $L$  为 1,即搜索深度在第一层时,标签识别的概率为  $P(1) = (1 - \frac{1}{B})^{m-1}$ ;搜索深度为  $b$  时,标签被识别的概率为  $P(b) = P(1)[1 - P(1)]^{b-1}$ 。因此,搜索深度的平均值为:

$$E(b) = \sum_{b=0}^{\infty} [1 - P(1)]^b = \frac{1}{1 - [1 - P(1)]} = \frac{1}{P(1)} = \frac{1}{(1 - \frac{1}{B})^{m-1}} \quad (3)$$

从而得到平均总时隙为:

$$T = E(b)B = \frac{B}{(1 - \frac{1}{B})^{m-1}} \quad (4)$$

对于二叉树,所得平均总时隙为:

$$T_2 = \frac{2}{(1 - \frac{1}{2})^{m-1}} \quad (5)$$

对于四叉树,所得平均总时隙为:

$$T_4 = \frac{4}{(1 - \frac{1}{4})^{m-1}} \quad (6)$$

从式(5)和式(6)可以得到,当  $m < 3$  时,  $T_2 < T_4$ ; 当  $m \geq 3$  时,  $T_2 > T_4$ 。  $m = 3$  是二叉树和四叉树的临界点,从而得到碰撞因子为  $\mu = 0.75$ 。

### 3 改进的多叉树防碰撞算法

#### 3.1 算法改进机制

为降低通信开销,IACT算法在标签中加入计数数值,模型如图 1 所示。在识别过程中,阅读器从栈顶取出指令  $(r_u r_{u+1}, c)$  发送给标签  $(r_u r_{u+1}$  表示最高两个碰撞位,  $c$  表示计数器值),只有与  $r_u r_{u+1}$  和  $c$  完全匹配的标签才响应,响应的标签将最高碰撞位之后的序列号  $r_{u+2} r_{u+3} \cdots r_{k-1} r_k$  发送给阅读器,同时其内部计数器加 1。如果阅读器同时接收到多个标签信息,则会发生碰撞,检测到碰撞位后,将标签的前缀序列号  $r_1 r_2 \cdots r_{u-2} r_{u-1}$ 、最高的两个碰撞位  $r_u r_{u+1}$  和计数器值  $c$  压入工作栈中。如果只有一个标签响应,阅读器将接收到标签传送的数据信息加上工作栈中的前缀序列号即可得到完整的 ID 码,从而成功识别出该标签。

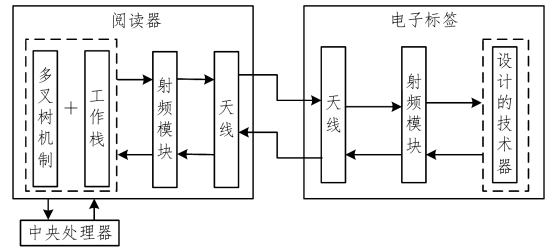


图 1 改进算法的 RFID 系统模型图

Fig. 1 RFID system model of improved algorithm

阅读器检测到碰撞后,通过曼彻斯特编码解码接收到的数据,计算出碰撞因子,然后根据碰撞因子选择二叉树或四叉树。如果碰撞因子  $\mu < 0.75$ ,则选择二叉树防碰撞算法,阅读器确定最高碰撞位  $(x)$ ,然后将以最高碰撞位为参数的 REQUEST( $x$ )查询命令发送给标签。如果检测出碰撞位只有一位,则可以断定该碰撞位是 0 或 1,然后将接收到的数据加上堆栈中的前缀信息得到完整的序列号,从而正确识别出标签。

如果计算出碰撞因子  $\mu \geq 0.75$ ,则采用四叉树防碰撞算法,阅读器编码后得到最高的两个碰撞位。为了避免出现空闲时隙,阅读器首先发送一个命令给当前的碰撞标签,标签根据序列号对应的最高两个碰撞位返回相应的信息。表 1 列出标签中的最高两个碰撞位及需要发送的对应编码。

表 1 标签碰撞位替换编码

Table 1 Replaced encoding for label collision bits	
标签中对应的最高两个碰撞位	标签需发送的对应编码
00	0001
01	0010
10	0100
11	1000

#### 3.2 算法的运行机制

IACT 算法的识别过程如下:

1) 阅读器向工作范围内的标签发送初始指令 REQUEST (Initial)命令。

2)与阅读器发送指令匹配成功的标签响应,响应标签的内部计数器加1,然后将序列号的后缀信息发送给阅读器。

3)阅读器检测是否有标签应答,如果没有标签应答,识别过程结束;如果只有一个标签应答,则识别出该标签,转步骤4);如果有多个标签同时应答,阅读器端采用曼彻斯特编码进行解码,如果碰撞位只有一位,可以断定该位是0或1,从而可以正确识别出该标签。如果有多位碰撞位,计算出碰撞因子 $\mu$ ,当 $\mu < 0.75$ 时,采用二叉树防碰撞算法,将最高碰撞位设置为0和1两个子集并将其压入堆栈中。如果 $\mu \geq 0.75$ ,则采用四叉树防碰撞算法,阅读器从工作栈中取出计数器值 $c$ 发送给当前的碰撞标签,并根据标签传送的编码得到碰撞位的具体信息,将其压入工作栈中。

- 4)阅读器从栈顶获取查询指令发送给标签,转步骤2)。
- 5)循环执行。
- 6)直到堆栈中的值为空,所有标签被识别,算法结束。

## 4 算法性能分析

### 4.1 时间复杂度

时间复杂度即为查询次数,是指识别过程中标签所需要的总时隙。由上文分析可知,IACT算法的总时隙为二叉树和四叉树两部分之和。

假设阅读器范围内有 $n$ 个待识别的电子标签,“后退式”搜索算法的总时隙为 $2n-1$ 。在IACT算法中,当只有一位碰撞位时,无需发送查询命令即可以直接识别,假设在识别过程中出现 $m$ 次发生一位碰撞位的情况,则整个算法减少了 $2m$ 次查询次数,总时隙为:

$$T'(n) = 2n - 1 - 2m \quad (7)$$

假设搜索深度为 $k$ ,子节点的平均标签数量为3,由自适应多叉树防碰撞算法的思想可得到,当搜索深度大于 $k$ 时,采用二叉树;当搜索深度小于或等于 $k$ 时,采用四叉树算法。

$$k = \lfloor \log_3 \frac{n}{3} \rfloor \quad (8)$$

其中, $\lfloor \cdot \rfloor$ 代表截尾取整运算。

搜索深度从0到 $k$ 层时,总时隙为:

$$T_1(n) = \sum_{i=0}^k 4^i = \frac{n-3}{9} \quad (9)$$

搜索深度大于 $k$ 时,总时隙为:

$$T_2(n) = \frac{5n}{3} - 2m, 0 \leq m \leq \frac{n}{2} \quad (10)$$

由式(9)和式(10)得到 IACT 算法的总时隙为:

$$T_{\text{IACT}}(n) = \frac{16n-3}{9} - 2m, 0 \leq m \leq \frac{n}{2} \quad (11)$$

### 4.2 吞吐量

吞吐量是指可读时隙与总时隙之比。对于待识别的 $n$ 个标签,可读时隙即为 $n$ ,因此 IACT 算法的吞吐量为:

$$S_{\text{IACT}} = \frac{n}{T_{\text{IACT}}(n)} \quad (12)$$

### 4.3 通信开销

整个系统的通信开销包括标签开销和阅读器开销,它是指识别过程中阅读器和标签所需要传送指令的总位数。

在 IACT 算法中,阅读器发送命令的位数取决于采用的

叉树,如果采用二叉树则只需发送一位碰撞位,若采用四叉树则阅读器需要发送两位碰撞位,而响应标签只需发送后缀序列号。假设标签的序列号为 $k$ ,则标签发送的平均长度为:

$$L = \frac{1}{k} + 2 \frac{1}{k} + 3 \frac{1}{k} + \dots + k \frac{1}{k} = \frac{k+1}{2} \quad (13)$$

IACT 算法中包括二叉树和四叉树,由式(9)和式(10)可以得到 IACT 算法的通信开销为:

$$Q_{\text{IACT}} = \frac{n-3}{9} (2 + \frac{k+1}{2}) + (\frac{5n}{3} - 2m) (1 + \frac{k+1}{2}) \quad (14)$$

其中, $0 \leq m \leq \frac{n}{2}$ 。

## 5 实验仿真

本文在 MATLAB 环境下对提出的 IACT 算法进行仿真分析与对比,不计控制、校验冗余等开销。按照 EPCglobal 编码标准,仿真标签数量从 10 增加到 1600,随机生成 96 位电子标签 ID 码,在理想的信道下进行仿真。图 2、图 3、图 4 分别表示的是 IACT 算法与自适应多叉树防碰撞算法(ACT 算法)<sup>[13]</sup>、自调整混合树防碰撞算法(AHT 算法)<sup>[14]</sup>、NAM 算法<sup>[15]</sup>在时间复杂度、吞吐量和通信开销方面的比较。

图 2 为在考虑不同数量标签的情况下,IACT 算法与其他 3 种算法的时间复杂度比较。从图中可以看出,所提出的 IACT 算法的时间复杂度明显低于其他 3 种算法。当标签数为 200 时,IACT 算法的时间复杂度约为 260,AHT 算法的时间复杂度约为 360,NAM 算法的时间复杂度约为 370;当标签数达到 1500 时,IACT 算法的时间复杂度约为 1900,AHT 算法的时间复杂度为 2700。从实验结果可以得到,IACT 算法在时间复杂度方面有所改善,且随着标签数量的增加,效果更好。

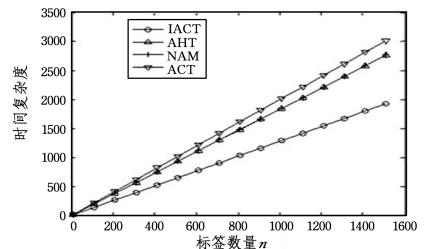


图 2 时间复杂度的比较

Fig. 2 Comparison of time complexity

图 3 显示了 IACT 算法与 AHT 算法、NAM 算法和 ACT 算法的吞吐量对比。随着标签数量的增加,IACT 算法的吞吐量逐渐趋近于 0.78,而 AHT 算法和 NAM 算法最高只有 0.58,且趋近于 0.55,ACT 算法的吞吐量只有 0.51。实验结果表明,IACT 算法相对其他算法具有更高的吞吐量。

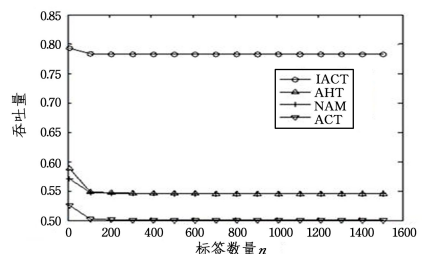


图 3 吞吐量的比较

Fig. 3 Comparison of throughput

图 4 显示了 IACT 算法与其他 3 种算法的通信开销比较。可以看出,IACT 算法的通信开销明显低于 ACT, AHT, NAM 算法。当标签数量达到 1600 时,IACT 算法的通信开销只有  $10^5$ ,而 NAM 算法的通信开销为  $1.6 \times 10^5$ , AHT 算法达到  $2.8 \times 10^5$ 。随着标签数量和序列号长度的增加,IACT 算法的通信开销降低得逐渐明显。

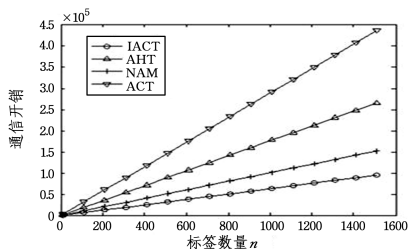


图 4 通信开销的比较

Fig. 4 Comparison of communication cost

**结束语** 本文针对现有自适应多叉树算法存在的缺陷与不足,提出了一种改进型自适应多叉树防碰撞算法(IACT)。该改进算法通过在标签中加入计数器功能机制,缩短了阅读器和标签发送查询指令的长度。当发生碰撞时,先计算出碰撞因子,然后据此决定采用二叉树或者四叉树,如果  $\mu < 0.75$ ,则采用二叉树防碰撞算法,反之则采用四叉树防碰撞算法。采用二叉树时,如果碰撞位只有一位,则可以直接识别出标签,从而减少查询次数;当采用四叉树时,为了避免出现空闲时隙,发送碰撞后,阅读器首先发送计数器值给碰撞标签,要求其返回最高两个碰撞位对应的编码,然后将得到的编码数据转换成相应的二进制即可以得到具体的碰撞位信息。性能分析和实验仿真表明,IACT 算法在时间运行复杂度、吞吐效率和系统通信开销等方面都有提高,并且随着标签数量和序列号的增加,效果更加明显。

## 参考文献

- [1] SUN Q Y, ZHANG H J, MO L F. Dual-reader wireless protocols for dense active RFID identification[J]. International Journal of Communication Systems, 2011, 24(11): 1431-1444.
- [2] LIU X, ZHANG S G, XIAO B, et al. Flexible and Time Efficient Tag Scanning with Handheld Readers[J]. IEEE Transactions on Mobile Computing, 2016, 15(4): 1-13.
- [3] YANG X, WU H F, ZENG Y, et al. Capture-aware estimation for the number of RFID tags with lower complexity[J]. IEEE Communications Letters, 2013, 17(10): 1873-1876.
- [4] ZHANG S G, LIU X, WANG J X, et al. Accurate Range-Free Localization for Anisotropic Wireless Sensor Networks[J]. ACM Transactions on Sensor Networks, 2015, 11(3): 1-28.
- [5] SHENG Z G, HONG D F, WEN G J. An Effective Frame Breaking Policy for Dynamic Framed Slotted Aloha in RFID[J]. IEEE Communications Letters, 2016, 20(4): 692-695.
- [6] LANDALUCE H, PERALLOS A, ZUAZOLA I J G. A Fast RFID Identification Protocol with Low Tag Complexity[J]. IEEE Communications Letters, 2013, 17(9): 1704-1706.
- [7] ZHU S Q, JIN X F, JIN L B. A Octree-based Grouping Recoding RFID Anti-collision Algorithm[C] // 2015 6th IEEE International Conference on Software Engineering and Service Science (ICSESS). 2015: 758-761.
- [8] LIU X, XIAO B, ZHU F, et al. Let's work together: Fast Tag Identification by Interference Elimination for Multiple RFID Readers[C] // Proc. of the IEEE ICNP. Singapore, 2016: 1-10.
- [9] JIA X, FENG Q, YU L. Stability analysis of an efficient anti-collision protocol for RFID tag identification[J]. IEEE Transactions on Communications, 2012, 60(8): 2285-2294.
- [10] BU K, XU M J, LIU X, et al. Toward Fast and Deterministic Clone Detection for Large Anonymous RFID System[C] // Proc. of the 11th IEEE International Conference on Mobile Ad Hoc and Sensor Systems. Philadelphia, PA, USA, 2014: 416-424.
- [11] TAN W J, ZHANG S S, TIAN Z H. Research on RFID anti-collision algorithm based on regressive-style binary search[J]. Computer Application and Software, 2012, 29(5): 191-194. (in Chinese)
- [12] 樊文静, 张姗姗, 田智慧. 基于后退式二进制搜索的 RFID 防碰撞算法的研究[J]. 计算机应用与软件, 2012, 29(5): 191-194.
- [13] BU K, LIU X, LUO J Q, et al. Unreconciled Collisions Uncover Cloning Attacks in Anonymous RFID Systems[J]. IEEE Transactions on Information Forensics and Security, 2013, 8(3): 429-439.
- [14] DING Z G, ZHU X Y, GUO L, et al. An adaptive anti-collision algorithm based on multi-tree search[J]. Acta Automatica Sinica, 2010, 36(2): 237-241. (in Chinese)
- [15] 丁治国, 朱学永, 郭立, 等. 自适应多叉树防碰撞算法研究[J]. 自动化学报, 2010, 36(2): 237-241.
- [16] SONG J H, GUO Y J, HAN L S. An adjustable hybrid tree anti-collision algorithm for RFID multi-tag identification[J]. Acta Electronica Sinica, 2014, 42(4): 685-689. (in Chinese)
- [17] 宋建华, 郭亚军, 韩兰胜. 自调整混合树 RFID 多标签防碰撞算法[J]. 电子学报, 2014, 42(4): 685-689.
- [18] REN S J, HAO Y S, XU B H, et al. A new anti-collision algorithm based on adjustable multi-tree[J]. Computer Measurement and Control, 2015, 23(12): 4180-4183. (in Chinese)
- [19] 任少杰, 郝永生, 许博浩, 等. 一种新的自调整多叉树防碰撞算法[J]. 计算机测量与控制, 2015, 23(12): 4180-4183.