

基于判决门限的 RFID 防碰撞 Q 值算法

任守纲 杨帆 王浩云 熊迎军 徐焕良

(南京农业大学信息科学技术学院 南京 210095)

摘要 标签的识别速度是 RFID 技术高强度、大规模应用的关键。为了提高射频标签的识别速度,提出了一种基于判决门限的防碰撞算法——QA-DTCI,详细阐述了算法的思想、运算流程和门限阈值的确定方法。在 QA-DTCI 算法中,读写器增加了两个计数器来分别计算碰撞和空闲时隙的个数,单独处理空闲时隙和碰撞时隙。当检测到碰撞时隙时,碰撞计数器自增;当检测到空闲时隙时,空闲计数器自增;同时对空闲计数器与碰撞计数器进行差值运算并与预设定的门限阈值比较,从而动态调整 Q 值。仿真结果表明,与 QA 算法相比,QA-DTCI 算法在不损耗系统吞吐率的情况下,识别时延最大缩短了 4%、识别速度提高了 10%。

关键词 RFID,防碰撞算法,EPC-C1G2,Q 值,判决门限

中图分类号 TP301.6,TP309

文献标识码 A

DOI 10.11896/j.issn.1002-137X.2014.08.034

Decision Threshold-based Q Algorithm for RFID Anti-collision

REN Shou-gang YANG Fan WANG Hao-yun XIONG Ying-jun XU Huan-liang

(School of Information Sciences and Technology, Nanjing Agricultural University, Nanjing 210095, China)

Abstract One of the hottest research topics in RFID technology is the speed of tag identification, which is the key to wide and large-scale application. Thereby this may increase the system identification of the time overhead. To increase the speed of tag identification, the article proposed the decision threshold based anti-collision algorithm—QA-DTCI algorithm, and elaborated the concept of the algorithm, the process of operation and the determination method of threshold parameters. In this algorithm, every reader is added with two counters to calculate the number of collisions and the idle slots respectively. When a collision slot is detected, the collision counter will increase, while when an idle slot is detected, the idle counter will increase as well. Comparing the subtraction of collision counter and idle counter against the setting threshold, it's can adjust the Q dynamically. The simulation results prove that the proposed algorithm can not only shorten the identificaton delay by maximum 4% but also increase the identificaton speed by 10%, without affecting the the system throughput.

Keywords RFID, Anti-collision algorithm, EPC-C1G2, Q value, Decision threshold

1 引言

射频识别(RFID)是一种非接触式的自动识别技术,广泛应用在物流、跟踪、零售等领域中。在 RFID 系统中,标签共享相同带宽,当多个标签同时发送各自数据时,从标签反向散射的消息很容易相互干扰和破坏,这就是 RFID 技术的标签碰撞问题^[1]。由于标签碰撞的问题关系到 RFID 技术大规模应用的成败,因此标签的防碰撞算法已成为该领域研究的热点之一。

标签防碰撞算法主要解决在读写器有效通信范围内,多个标签同时与读写器进行通信的问题。现有的防碰撞算法主要分为两类:基于时隙的 ALOHA 协议随机性算法^[1]和基于二叉树搜索的确定性算法^[1]。这两类算法都属于时分多路,

它们分别存在各自的优缺点:二叉树搜索算法虽然标签的识别率可达到 100%,但是算法比较复杂,收敛速度较慢,识别时间相对较长,如二叉树搜索算法^[1](BSA, Binary Searching Algorithm)、动态二叉树搜索算法^[1](DBSA, Dynamic Binary Searching Algorithm)、自适应二叉树搜索算法(ABSAs, Adaptive Binary Searching Algorithm)^[2-4]、查询树算法^[5](QT, Query Tree Algorithm);ALOHA 协议算法的复杂度及对标签的要求低,识别标签速度快,受系统中标签数目影响小,实施成本低,如动态时隙 ALOHA 算法^[7,9-10](DFSA, Dynamic Framed Slotted ALOHA Algorithm)、标签估计 ALOHA 算法^[8](TEFA, Tag Estimated Framed ALOHA Algorithm)、分组时隙 ALOHA 算法^[6](GFSA, Group Framed Slotted ALOHA Algorithm)等。

到稿日期:2013-10-09 返修日期:2014-02-05 本文受国家科技支撑计划项目(2011BAK21B05),中央高校基本科研业务专项资金项目(Y0201100080),江苏省科技支撑计划重大项目(BE2011398),江苏省工业和信息产业转型升级专项引导项目(2011CSZ1)资助。

任守纲(1977—),男,博士,副教授,主要研究方向为物联网技术应用、人工智能,E-mail:rensng@njau.edu.cn;杨帆(1988—),男,硕士生,主要研究方向为 RFID 防碰撞算法;王浩云(1981—),男,博士,讲师,主要研究方向为物联网技术;熊迎军(1984—),男,博士,讲师,主要研究方向为无线传感器技术;徐焕良(1963—),男,博士,教授,主要研究方向为物联网关键技术。

EPCglobal 提出的 EPC-C1G2 标准所采用的防碰撞算法叫 Q 算法^[10-14] (QA, Q Algorithm)。读写器通过 Query、QueryAdjust 和 QueryRep 等命令查询每个时隙标签的应答状况并调整 Q 值,使系统工作在 2Q 的动态帧长下。QA 采用单一参数 c 调整 Q 值,由于碰撞时隙和空闲时隙时间消耗的不同,单一参数并不能单独考虑碰撞时隙和空闲时隙两种情况;同时计算机中所有的数据均以二进制形式表示,浮点数也不例外,故当标签数量较多时,读写器的运算负担加重,识别时间的开销增加,另外多次的二进制近似计算也会造成误差,降低识别效率。针对上述不足,本文提出了碰撞空闲差值门限的 Q 算法(QA-DTCI, Q Algorithm with D-Value Threshold for Collision and Idle)。文中详细阐述了该算法的思想、算法运算流程和判决门限参数的确定方法。仿真结果显示,该算法能够缩短识别时延、提高识别速度。

2 基本 EPC-C1G2 防碰撞算法

EPC Gen2 标准采用了 3 组基本的操作命令来管理标签群,分别为选择操作、盘存操作和访问操作。由于盘存操作所需要的时间会因为选择算法的不同而有很大的差异,同时在 RFID 的标签识别过程中,读取速度是非常关键的性能评价指标,因此系统的性能在很大程度上取决于盘存操作的算法设计。

盘存操作的命令集合包括 Query、QueryAdjust、QueryRep、ACK 和 NAK 指令。Query 指令用于开启新的盘存周期,指令中含时隙计数参数 Q,用于控制标签在该盘存周期随机选择一个时隙作为应答时隙。参与识别的标签应在 $[0, 2Q-1]$ 范围内产生一个随机数,并将其载入自身时隙计数器(Slot Counter, SC)中。只有 SC 的值为 0 的标签才会向读写器发送 RN16(16 位的随机数)作为应答信息。

QueryRep 指令将使所有标签持有的随机数自减,若有标签计数至 0,则该标签将返回一个 RN16。当有多个标签时隙计数器计数至 0 时,就会发生碰撞,当无标签计数至 0 时,就会发生空闲。

QueryAdjust 指令在不改变盘存操作其他参数的前提下,更改参数 Q 的值(Q, Q+1, Q-1)。接收到该命令的所有标签将重新在 $[0, 2Q-1]$ 间产生一个随机数。若产生的是 0,则该标签将返回一个 RN16。

对于标签向读写器的回复,可能会出现 3 种结果:成功回复、碰撞回复和空回复,如图 1 所示。

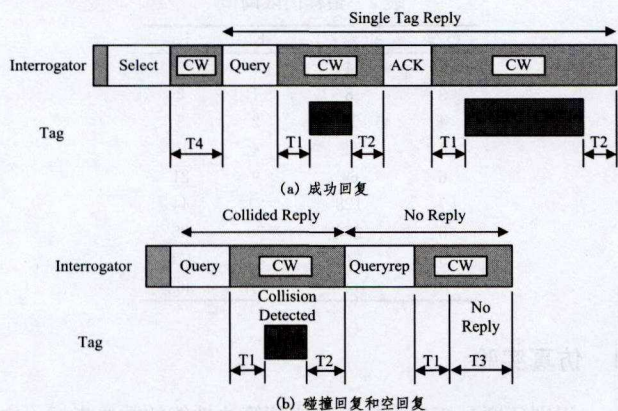


图 1

若仅有一个标签发送自己的 RN16,当前回复为成功回复;若多个标签同时发送自己的 RN16,当前回复为碰撞回复;若没有标签发送自己的 RN16,则当前回复为空回复。由于 3 种回复消耗的时间不同,成功回复消耗的时间最长,碰撞回复次之,空回复消耗的时间最少,同时不同算法产生的 3 种回复的数量不同,因此系统读取速率会依算法选择的不同而有所差异。

3 QA-DTCI 算法

在标签数量变化很大的情况下, Q 算法能否实现系统高效率,主要取决于参数 c 的取值情况。c 太大会造成帧长度频繁调整,增加功耗和识别时间; c 太小又会增加最优帧的选择时间,降低识别效率。在理想通信状态下,任一帧发生碰撞和空闲的概率是不同的,同时碰撞时隙和空闲时隙消耗的时间也不同,仅仅采用单一的参数 c 调整无法使系统工作在最优状态,导致识别标签速率迅速下降,系统效率也大大降低。

3.1 QA-DTCI 算法

在 QA-DTCI 算法中,读写器增加了两个计数器:空闲时隙计数器(ISC, Idle Slot Count)和碰撞时隙计数器(CSC, Collision Slot Count),分别用于统计帧中空闲和碰撞时隙的个数。同时将 Q 算法中单一浮点调整 Q 值的参数 c 转变为两个整形的门限阈值参数 K_c 和 K_i 。当帧中 CSC 与 ISC 的差值大于门限阈值 K_c 时, Q 自增,当帧中 ISC 与 CSC 的差值大于门限阈值 K_i 时, Q 自减,公式如下:

$$CSC - ISC > K_c \quad Q = Q + 1 \quad (1)$$

$$ISC - CSC > K_i \quad Q = Q - 1 \quad (2)$$

QA-DTCI 算法要求空闲时隙的个数不小于碰撞时隙的个数,因此门限阈值 $K_c < K_i$ 。同时随着 Q 值的增大或减小,帧长也随之改变,门限阈值随着 Q 值的变化发生递增或递减。

3.2 QA-DTCI 算法工作流程

QA-DTCI 算法工作流程如图 2 所示,主要分为 3 步:时隙统计、门限阈值比较和 Q 值调整。

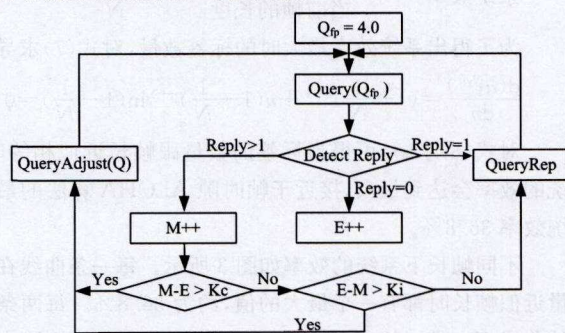


图 2 QA-DTCI 算法工作流程

具体的算法步骤如下:

步骤 1 读写器向通过 Select 命令选择的标签群发送 Query 指令,开启一个新的 Inventor 周期。Query 指令中含有时隙计数参数 Q,收到读写器发送 Query 指令的标签从 $[0, 2Q-1]$ 范围内产生一个随机数存入自身的 SC 中。同时初始化读写器的碰撞计数器和空闲计数器。

步骤 2 如果标签 SC 的值为 0,则该标签向读写器发送自身 RN16,否则保持沉默。

步骤3 标签响应会出现3种情况:

Case 1 仅有一个标签响应。读写器发送 QueryRep 指令,返回步骤2继续识别剩余标签。

Case 2 无标签响应。读写器的空时隙计数器自增1,即 $ISC=ISC+1$ 。同时 CSC 和 ISC 的差值与门限值 K_i 比较。若 $ISC-CSC > K_i$,读写器发送 QueryRep 命令调整 Q 值为 $Q=Q-1$;否则读写器发送 QueryRep 指令,返回步骤2继续执行。

Case 3 多个标签响应。读写器的碰撞时隙计数器自增1,即 $CSC=CSC+1$ 。同时 CSC 和 ISC 的差值与门限值 K_c 比较。若有 $CSC-ISC > K_c$,读写器发送 QueryRep 命令调整 Q 值为 $Q=Q+1$;否则读写器发送 QueryRep 指令,返回步骤2继续执行。

步骤4 重复前面的相似步骤,直到所有的标签识别完毕,退出程序。

3.3 QA-DTCI 算法中门限阈值的确定

门限阈值 K_c 和 K_i 是 QA-DTCI 算法最重要的两个参数,其决定着当前的 Q 值是否为最佳值。为求取门限阈值 K_c 和 K_i ,现假设在某一时刻待识别的标签数量为 n ,帧长 $N=2^Q$,则某个时隙中出现 r 个标签的概率服从二项分布,记为:

$$B_{(n,N)}(r) = C_n \left(\frac{1}{N}\right)^r \left(1 - \frac{1}{N}\right)^{n-r} \quad (3)$$

因此,在一个盘存周期内空时隙的个数、成功时隙的个数、碰撞时隙的个数分别为:

$$E = a_0^{N,n} = N \cdot B_{(n,N)}(0) = N \cdot \left(1 - \frac{1}{N}\right)^n \quad (4)$$

$$S = a_1^{N,n} = N \cdot B_{(n,N)}(1) = N \cdot n \left(\frac{1}{N}\right) \left(1 - \frac{1}{N}\right)^{n-1} \quad (5)$$

$$M = a_2^{N,n} = N - E - S \quad (6)$$

式中, $a_i^{N,n}$ 表示在帧长为 N 、待识别标签数量为 n 的情况下,时隙存在 i 个标签的时隙数量。那么,可以计算出系统效率:

$$\text{系统效率} = \frac{\text{成功识别标签的时隙数}}{\text{当前帧的长度}} = \frac{a_1^{N,n}}{N} \quad (7)$$

为了得出系统效率最大时的标签数量,对式(7)求导:

$$\frac{d(a_1^{N,n})}{dn} = \left(1 - \frac{1}{N}\right)^{n-1} + n \left(1 - \frac{1}{N}\right)^{n-1} \ln\left(1 - \frac{1}{N}\right) = 0 \quad (8)$$

对式(8)求解,可得当标签的数量跟帧长近似相等时,系统的效率会达到最大,接近于帧时隙 ALOHA 算法的最大系统效率 36.8%。

不同帧长下系统的效率如图3所示。每一条曲线在标签量近似帧长时都有一个最大的值,约为 36.8%。每两条相邻的曲线都有唯一的交点,交点左边的曲线呈递减趋势,右边的曲线呈递增趋势,即说明当标签数量小于交点的值时,左边曲线的系统效率高于右边曲线的系统效率,反之右边曲线的系统效率高于左边曲线的系统效率。因此,本文将该交点定义为极值点(EP, Extreme Point),每条曲线($Q=1$ 和 $Q=15$ 除外)都有两个 EP,分别为左极值点(LEP, Left Extreme Point)和右极值点(REP, Right Extreme Point)。则在当前 Q 值下,最佳标签的数量为 $[\text{LEP}, \text{REP}]$ 。例如,当 $Q=6$ 时,LEP 和 REP 的值分别为 45 和 88,即最佳标签的数量在 $[45, 89]$ 范围内。

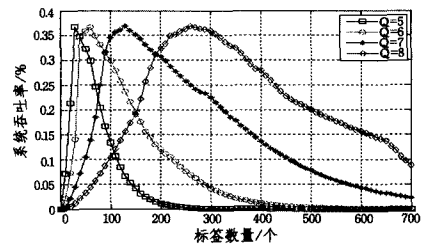


图3 Q值和系统效率的关系

对于一个给定数量的 Q 值,通过式(7)可以求得 LEP 和 REP 的值,从而可以得到当前 Q 值下最优标签的数量范围。结果如表1所列。

表1 不同 Q 值的最佳标签范围

Q 值	N(帧长)	LEP	REP
2	4	3	5
3	8	6	11
4	16	12	22
5	32	23	44
6	64	45	88
7	128	89	180
8	256	181	355
9	512	356	700
10	1024	701	1420

若标签的数量为当前 Q 值的 LEP 值,通过式(2)和式(4)可知 E 的值要高于 M,则定义 K_i 为:

$$\begin{aligned} \lfloor K_i \rfloor &= E - M = 2 \cdot E - N + S \\ &= 2 \cdot N \cdot \left(1 - \frac{1}{N}\right)^{\text{LEP}} - N + \text{LEP} \cdot \left(1 - \frac{1}{N}\right)^{\text{LEP}-1} \end{aligned} \quad (9)$$

若标签的数量为当前 Q 值的 REP 值,通过式(2)和式(4)可知 M 的值高于 E,则定义 K_c 为:

$$\begin{aligned} \lfloor K_c \rfloor &= M - E = N - S - 2 \cdot E \\ &= N - 2 \cdot N \cdot \left(1 - \frac{1}{N}\right)^{\text{REP}} - \text{REP} \cdot \left(1 - \frac{1}{N}\right)^{\text{REP}-1} \end{aligned} \quad (10)$$

利用式(9)和式(10),可以分别计算出 K_c 和 K_i 的值。比如,当 $Q=7$ 时,通过表1可以得到 $N=128$ 、 $\text{LEP}=90$ 、 $\text{REP}=177$,通过式(9)和式(10)计算得到 $K_c=20$ 、 $K_i=43$ 。不同 Q 值下门限阈值 K_c 和 K_i 的取值如表2所列。当 $Q=1$ 即 $N=2$ 时,由于帧长较小,通过完全识别两个时隙就可以得到未读取的标签,因此不需要门限值。

表2 最佳门限阈值

Q 值	N(帧长)	K_c	K_i
2	4	0	1
3	8	1	2
4	16	2	3
5	32	4	10
6	64	9	21
7	128	21	44
8	256	39	85
9	512	72	176
10	1024	157	362

4 仿真实验

识别速度是衡量多标签防碰撞算法性能的重要指标。本文采用蒙特卡洛方法,仿真了不同标签数量、不同初始 Q 值

下的 QA-DTCI 算法的识别速度, 并与 Low Bound、Schoute、QA 等防碰撞算法的识别速度进行了比较。所有结果是独立做 100 次试验得到的平均值。

4.1 仿真条件

在 Gen-C1G2 标准中, 定义了读写器到标签发数据 0 时的基准时间间隔, 称作 T_{ari} , 而其他的所有命令的时间都是基于 T_{ari} 来规定的, 如表 3 所列。因为标准中 T_{ari} 值有几种形式, 现我们定义 $T_{ari}=12.5\mu s$, 则相应的传输速率为 80kbps。

表 3 EPC 协议标准通信时间

Interrogator-To-Tag Preamble	112.5 μs
Inerrogator-to-Tag Frame Sync	62.5 μs
Tag-to Interrogator Preamble	112.5 μs
Query Command	52.5 μs
QueryRep Command	137.5 μs
ACK Command	400 μs
RN16	212.5 μs
PC+EPC+CRC16	912.5 μs
T1	62.5 μs
T2	62.5 μs
T3	62.5 μs
T4	112.5 μs

通过图 1 和表 3, 可以分别计算出成功、空闲、碰撞情况下的消耗时间分别为: 1912.5 μs 、262.5 μs 、475 μs 。

4.2 仿真结果

4.2.1 标签识别速度

将 QA-DTCI 算法与几种经典的 ALOHA 算法、QA 算法的标签读取速度进行对比。同时利用 EPC 标准中的通信时间, 得出了标签量从 100 递增到 1000 的过程中碰撞时隙和空闲时隙消耗时间的变化情况。图 4 显示了碰撞时隙消耗的时间。QA-DTCI 算法碰撞时隙消耗的时间比 QA 算法明显减少, 且随着初始 Q 值的增大, 时间随之减少。当标签量为 1000 时, QA 碰撞时隙消耗的时间为 413ms, QA-DTCI 算法 ($Q=4$) 碰撞时隙消耗的时间为 374ms, 减少了 9.4%。图 5 显示了空闲时隙消耗的时间。QA-DTCI 算法并没有以大量空闲时隙的时间消耗为代价来减少碰撞时隙的时间消耗, 其空闲时隙消耗的时间只比 QA 算法略多一些。当标签量为 1000 时, QA 空闲时隙消耗的时间为 270 μs , QA-DTCI 算法 ($Q=4$) 空闲时隙消耗的时间为 283 μs 。当增大初始 Q 时, 空闲时隙消耗的时间也随之增多。图 6 显示了不同算法的标签读取速度。LowBound 算法和 Schoute 算法的读取速度分别约为 250 s^{-1} 和 260 s^{-1} , QA 算法的读取速度约为 290 s^{-1} , 而 QA-DTCI 算法的读取速度最快, 约为 320 s^{-1} , 比 Low-Bound 算法、Schoute 算法、QA 算法分别提高了 28%、23%、10%。

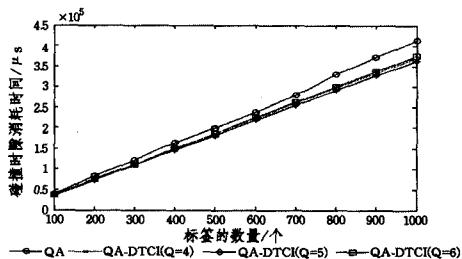


图 4 QA-DTCI 算法与 QA 的碰撞时隙时间消耗

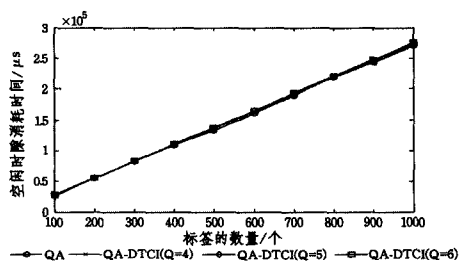


图 5 QA-DTCI 算法与 QA 的空闲时隙时间消耗

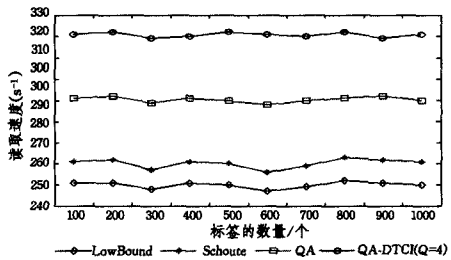


图 6 QA-DTCI 与各种算法读取速度

4.2.2 识别时延与系统效率

在常规标签规模下, 将 QA-DTCI 算法识别时延(通过平均时隙总数来衡量)和系统吞吐量与几种经典的 ALOHA 算法、QA 算法及经验算法 ($c=0.8/Q$) 进行对比。图 5 显示了几种不同识别算法的系统识别时延。LowBound 算法和 Schoute 算法随标签数量增加性能急剧下降, 且需要较多的时隙, 而 QA-DTCI 算法所用的时隙最少。当标签数为 1000 时, QA-DTCI 算法只需要 2889 个时隙, 识别时延比 Low-Bound 算法 (3339)、Schoute 算法 (3199)、QA 算法 (3012) 分别降低了 13.8%、9.7%、4%。图 6 显示了几种不同识别算法的系统吞吐量。从整体的性能来看, LowBound 算法和 Schoute 算法随着标签数量的增加, 系统吞吐量急剧恶化。当标签数量为 1000 时, 系统吞吐量还不到 20%。经验算法能够使系统吞吐量维持在 34%, 而 QA-DTCI 算法没有明显提高系统吞吐量, 只是基本稳定在 34.8% 左右。

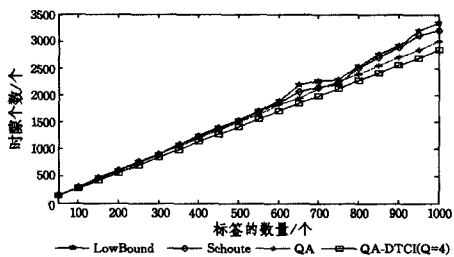


图 7 QA-DTCI 算法与其它算法识别时延的比较

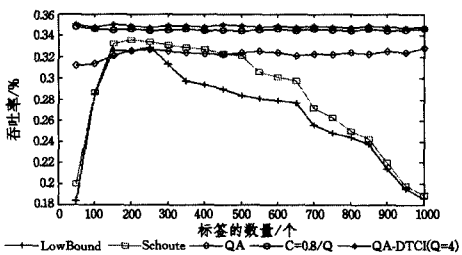


图 8 QA-DTCI 算法与其它算法系统效率的比较

结束语 本文提出了一种基于判决门限的防碰撞算法 QA-DTCI。该算法针对 QA 算法中使用单一调整参数调整 Q

(下转第 163 页)

效性,与其他方法的比较也显示该方法具有处理数据量小、不需要先验知识以及不需要其他路由器信息的优点。

参 考 文 献

- [1] Rekhter Y, Li T, Hares S. A Border Gateway Protocol 4 (BGP-4)[EB/OL]. RFC4271. 2006
- [2] 黎松, 诸葛建伟, 李星. BGP 安全研究[J]. 软件学报, 2013, 24(1):121-138
- [3] Brown M A. Pakistan hijacks YouTube. Renesys Blog [EB/OL]. <http://www.renesys.com/blog/2008/02/pakistan-hijacks-youtube-1.shtml>, 2008
- [4] Prefix hijacking by Michael Lindsay via Internap[EB/OL]. <http://mailman.nanog.org/pipermail/nanog/2011-August/039379.html>. 2011.08
- [5] Hiran R, Carlsson N, Gill P. Characterizing Large-scale Routing Anomalies A Case Study of the China Telecom Incident[C]// Passive and Active Measurement. 2012;229-238
- [6] Hu X, Mao Z M. Accurate real-time identification of IP hijacking [C]// Proc. 07th Security and Privacy. Berkeley, CA, 2007;3-17
- [7] Kruegel C, Mutz D, Robertson W, et al. Topology-based detection of anomalous BGP messages[C]// Proc. 6th Symp. Recent Advances in Intrusion Detection(RAID). 2007;17-35
- [8] Hong S C, Hong J W K, Ju H. IP prefix hijacking detection using the collection of AS Characteristics[C]// Proc. 17th Network Operations and management symposium. Taipei, China, 2011;1-7
- [9] Li J, Ehrenkrantz T, Elliott P. Buddyguard: a buddy system for fast and reliable detection of IP prefix anomalies[C]// Proc. 20th IEEE International Conference(ICNP). 2012;1-10
- [10] Zhao X, Pei D, Wang L, et al. An analysis of BGP multiple origin AS(MOAS) conflicts[C]// Proc. of the SIGCOMM Internet Measurement Workshop, 2001. San Francisco: ACM, 2001;31-35
- [11] <http://www.gossamer-threads.com/lists/nanog/users/144024> [OL]. 2011
- [12] Route Views Project Page[OL]. <http://www.routeviews.org>. 2005

(上接第 157 页)

值的不足,通过对读写器增加两个不同的计数器,统计碰撞和空闲的时隙个数,差值运算后比较门限阈值,从而动态调整 Q 值的大小。该算法避免了读写器计算浮点的运算,同时将碰撞空闲时隙单独考虑,提高了系统效率和识别速度。仿真结果显示,与 EPC-CIG2 的 QA 算法相比,QA-DTCI 算法在不损耗系统吞吐率的情况下,提高了标签的读取速度,缩短了识别时延。在标签量为 1000、初始 Q 为 4 时,QA-DTCI 算法比 QA 算法在标签的读取速度上提升了 10%,在识别时延上降低了 4%,使系统的吞吐率维持在 34.8%左右。后续工作是优化 K_i 和 K_c 门限阈值,在保证当前碰撞时隙消耗的时间不变的情况下,继续提高系统的吞吐率,同时降低空闲时隙消耗的时间。

参 考 文 献

- [1] Finkenzeller K. RFID Handbook: Radio-frequency Identification Fundamentals and Applications (Second Edition)[M]. England: John Wiley and Sons, 2003
- [2] Jihoon M, Wonjun L, Srivastava J. Adaptive binary splitting for efficient RFID tag anti-collision[J]. IEEE Communications Letters, 2006, 10(3):144-146
- [3] Lai Yuan-cheng, Lin C-C. A pair-resolution blocking algorithm on adaptive binary splitting for RFID tag identification[J]. IEEE Communications Letters, 2008, 12(6):432-434
- [4] Jihoon M, Wonjun L. Adaptive binary splitting: a RFID tag collision arbitration protocol for tag identification[C]//2nd International Conference on Broadband Networks. Boston, United States, 2005, 1:347-355
- [5] Law C, Lee K, Kai-Yeung S. Efficient memoryless protocol for tag identification [C] // Proceedings of the 4th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications. Boston, USA: ACM, 2000;75-84
- [6] Hwang T-W, Lee B-G, Kim Y-S. Improved anti-collision scheme for high speed identification in RFID system[C]//First International Conference on Innovative Computing, Information and Control. Beijing, China, 2006, 2:449-452
- [7] Schoute F C. Dynamic frame length aloha [J]. IEEE Transactions on Communications, 1983, 31(4):565-568
- [8] Cha J R, Kim J H. Novel anti-collision algorithms for fast object identification in RFID system[C]// Proceedings of the 11th International Conference on Parallel and Distributed Systems. Washington D. C. , USA: IEEE, 2005;63-67
- [9] Lee S R, Joo S D, Lee C W. An enhanced dynamic framed ALOHA algorithm for RFID tag identification[C]// Proceedings of the 2nd Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services. Washington D. C. , USA: IEEE, 2005;166-174
- [10] Chen W T. An accurate tag estimate method for improving the performance of an RFID anticollision algorithm based on dynamic frame length ALOHA[J]. IEEE Transactions on Automation Science and Engineering, 2009, 6(1):9-15
- [11] EPCglobal. EPC radio-frequency identity protocols class-1 generation-2 UHF RFID protocol for communications at 860MHz-960MHz(version 1.2.0)[S]. 2008
- [12] Lee D, Kim K, Lee Won-jun. Q+ Algorithm: An Enhanced RFID Tag Collision Arbitration Algorithm[J]. Ubiquitous Intelligence and Computing: Lecture Notes in Computer Science, 2007, 4611/2007:23-32
- [13] Maguire Y, Pappu R. An optimal Q-algorithm for the ISO 18000-6C RFID protocol[J]. IEEE Trans. Automation Science and Engineering, 2009, 6(1):16-24
- [14] Fan X, Song I, Chang K. Gen2-based hybrid tag anti-collision Q algorithm using Chebyshev's inequality for passive RFID systems[C]//IEEE 19th International Symposium on Personal, Indoor, and Mobile Radio Communications, 2008;1-5