

基于最大似然估计的 RFID 帧时隙估算方法^①

杨 灵, 蔡旭灿, 邹 娟

仲恺农业工程学院 信息科学与技术学院, 广州 510225

摘要: 对动态帧时隙 ALOHA 算法(Dynamic Framed Slotted Aloha, DFSA)中的效率最优化问题进行了分析, 提出了一种基于最大似然估计的帧时隙估算方法. 该方法根据上一帧获取的碰撞时隙和空闲时隙, 采用最大似然估计对帧时隙进行估算, 同时采用线性反馈移位寄存器(LFSR)来实现哈希函数, 使得标签能够均匀选择不同的时隙, 减少估算误差, 降低误差率. 通过仿真实验证明, 该方法的误差随着标签时隙比的增加而明显降低, 满足误差率小于 15% 的实际应用要求, 当标签时隙比大于 1.2 时, 比 Vogt 算法和 UPE 算法估算的误差更小, 误差率低于 5%.

关键词: 最大似然估计; 帧时隙; ALOHA 算法; 估算

中图分类号: TN914

文献标志码: A

文章编号: 1000-5471(2017)05-0010-07

射频识别(RFID)系统最主要的目的是确保阅读器和标签之间的数据完整和准确通信, 现实中主要有 2 种破坏数据通信的原因: ①外部信号干扰. 对于外部信号干扰, 可以在传输数据中加上奇偶校验位来解决; ②信道冲突, 即为不同的通信分配了同一个信道, 从而不同的通信信号之间产生了信号冲突. 对于信道冲突, 业内最主要的解决方法是动态帧时隙 ALOHA 算法(DFSA). 动态帧时隙 ALOHA 算法^[1-2]的主要思路是阅读器根据当前的标签数动态改变下一帧的时隙, 关键是通过保存的上一帧碰撞时隙和空闲时隙来快速、准确地估算出当前标签数目, 再通过调整帧时隙来减少碰撞时隙, 提高算法效率.

当前, DFSA 算法中的帧时隙估算方法主要有 Vogt, Quan, UPE 共 3 种算法. ①Vogt 算法^[3-4]. Vogt 算法主要采用遍历搜索算法对所有符合碰撞时隙、空时隙、独占时隙 3 种时隙的数目进行遍历, 搜索出一种最优算法, 该算法估算准确, 但计算复杂度太高. 同时, Vogt 算法还根据独占时隙、空时隙之间的关系做了一个线性组合, 再依据这个组合估算出标签数目. 这个算法计算复杂度较低, 可以较方便地进行预测, 但由于估算误差也与标签数目成正比关系, 因此当标签数目较大时, 估算误差偏高. ②UPE 算法^[5-6]. UPE 算法也是依据独占时隙、空时隙之间的线性关系进行估算, 该算法计算复杂度低, 操作方便, 但随着标签数目的增加, 估算误差率也呈线性增加. ③Quan 算法^[7-8]. Quan 算法是依据独占时隙、空时隙之间的统计关系进行估算, 该算法估算准确, 但计算复杂度较高, 随着标签数量的增加, 误差率也成比例增加. 本文根据独占时隙、碰撞时隙、空时隙之间的统计关系提出了一种基于最大似然估计的帧时隙估算方法, 有效地解决了帧时隙估算问题.

1 基于最大似然估计的 RFID 帧时隙估算方法

1.1 动态帧时隙 ALOHA 算法效率分析

S: 标签通信的有效吞吐率, 即标签在 T_0 时段内成功完成通信的平均次数.

① 收稿日期: 2016-09-13

基金项目: 广东省自然科学基金项目(S2012010010329); 广东省工业攻关项目(2015A020209173, 2012B010100034); 广州市产学研协同创新重大专项(201704020030).

作者简介: 杨 灵(1980-), 男, 江西崇仁人, 硕士, 副研究员, 主要从事射频识别及网络安全研究.

G : T_0 时段内标签平均到达次数.

P_e : 标签完成通信的概率.

由于标签到达的数目是按照泊松分布,那么在 t 秒到达 n 个标签的概率为^[9]

$$P(n) = \frac{(\lambda t)^n e^{-\lambda t}}{n!} \quad (1)$$

其中 λ 为平均每秒到达的标签数目,则输入负载为 $G = \lambda T_0$.

同时,在一个时隙内没有其他标签信息到达的概率为 $P_e = e^{-G}$

因此,标签通信的有效吞吐率为

$$S = GP_e = Ge^{-G} \quad (2)$$

由上式可知,当 $G = 1$ 时,即 $n = l$,标签通信的吞吐率为最大;当 $G < 1$ 时,即 $n < l$,标签空闲时隙就会增加,造成时间浪费;而当 $G > 1$ 时,即 $n > l$,标签碰撞时隙就会增加,从而造成重复发送的次数增加,导致时间浪费.因此,如何合理地设置 l 就是算法效率的关键,而 l 的设置必须先准确估算标签数量.

1.2 哈希函数

为了减少碰撞时隙,提高算法效率,须使得标签可以线性选择时隙进行通信.本文利用哈希函数来线性选择通信时隙,即 $H(key) \in [0, l-1]$. 选择的哈希函数须满足 3 个条件:① 不重复性,即对于 2 个不同的 k_1, k_2 ,其 $H(k_1), H(k_2)$ 值相等的概率约等于 0;② 单向性,即由 $H(k_1)$ 的值反推 k_1 的值的可能性基本为 0;③ 实现简单.系统中需用到大量的哈希函数,为减少算法复杂度,该哈希函数实现必须简单.本文采用 LFSR 来实现哈希函数.

线性反馈移位寄存器^[10](LFSR)由一系列移位寄存器和少量异或元件组成,用来产生二进制序列,可用于产生伪随机数.线性反馈移位寄存器由随机数种子来初始化,每一个时钟,则寄存器的值就向右移动一位,其产生的值进行异或运算后将结果放到寄存器单元的最左边,同时输出一个位.这时,如果种子没有被泄露,则 LFSR 就可产生加密的伪随机数.而且, LFSR 产生的伪随机数具有很好的统计特性,方便应用到统计算法中.

LFSR 的结构可以用生成多项式表示为

$$G(x) = g_m x^m + g_{m-1} x^{m-1} + \cdots + g_1 x + g_0 \quad (3)$$

其结构如图 1 所示.

在系统中,使用 LFSR 产生的伪随机数也存在一定缺陷:① 重复性.由于 LFSR 生成的随机数是以 $2^l - 1$ 为周期,即 $2^l - 1$ 步后随机数将重复,因此若连续获得 2^l 位的数据,就可以破解.② LFSR 随机数是由数学多项式推导而来,具有线性特性,导致数据不能直接用于加密.但是,由于本文在使用过程中并不是直接使用 LFSR 随机数,而是将其与真正的随机数进行异或运算,并在寄存器中进行循环移位,可以有效避开 LFSR 随机数的缺点.

1.3 估算方法

本估算方法的主要思想是根据获取的上一帧的空闲时隙数、冲突时隙数、正常时隙数及目前的帧长,采用最大似然估计法(maximum likelihood estimate method,简称 ML 算法),估算出目前的标签数目,同时采用 LFSR 来实现哈希函数,使得标签能够均匀选择不同的时隙,减少估算误差,降低误差率.

X_i, Y_i, Z_i 分别表示在第 i 时隙空闲、独占、冲突情况,即当第 i 时隙为空闲时, $X_i = 1$; 当第 i 时隙为独占时, $Y_i = 1$; 当第 i 时隙为冲突时, $Z_i = 1$.

M_0, M_1, M_2 分别记录上一帧中空闲、独占、冲突时隙的数量,即 $M_0 = \sum_{i=1}^l X_i, M_1 = \sum_{i=1}^l Y_i, M_2 = \sum_{i=1}^l Z_i$.

m_0, m_1, m_2 分别表示上一帧中空闲、独占、冲突时隙的观察值.

n : 标签的个数.

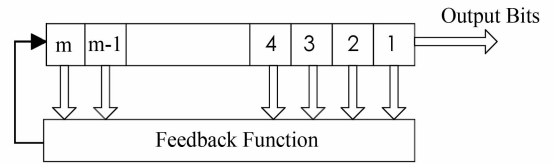


图 1 线性反馈移位寄存器结构图

n' : 标签数量 n 的预估值.

n_0, n_1, n_2 分别表示空闲、独占、冲突标签的数目.

l : 时隙的个数, 即帧长.

λ : 载入因子, 其中 $\lambda = n/l$.

m : 试验次数.

P : 第 i 个时隙为空闲时隙的概率.

$B[k]$: 阅读器用于保存时隙占用情况的矩阵. 其中, $B[k] = 1, k \in [0, l-1]$ 表示第 k 个时隙被占用, $B[k] = 0$ 表示第 k 个时隙未被占用.

根据概率论, 标签的到达是服从泊松分布的, 由于泊松分布可由二项分布的极限得到, 则二项频率函数为

$$P(k) = \frac{m!}{k!(m-k)!} p^k (1-p)^{m-k} \quad (4)$$

假设 $mp = \lambda$, 则

$$P(k) = \frac{m!}{k!(m-k)!} \left(\frac{\lambda}{m}\right)^k \left(1 - \frac{\lambda}{m}\right)^{m-k} \quad (5)$$

$$P(k) = \frac{\lambda^k}{k!} \frac{1}{m^k} \frac{m!}{(m-k)!} \left(1 - \frac{\lambda}{m}\right)^m \left(1 - \frac{\lambda}{m}\right)^{-k} \quad (6)$$

当 $n \rightarrow \infty$, 则

$$\frac{\lambda}{n} \rightarrow 0 \quad (7)$$

$$\frac{m!}{(m-k)!m^k} \rightarrow 1 \quad (8)$$

$$\left(1 - \frac{\lambda}{m}\right)^m \rightarrow e^{-\lambda} \quad (9)$$

$$\left(1 - \frac{\lambda}{m}\right)^{-k} \rightarrow 1 \quad (10)$$

将式(7),(8),(9),(10) 带入式(6), 可得参数为 λ 的泊松频率函数

$$P(X = k) = \frac{\lambda^k}{k!} e^{-\lambda}, k = 0, 1, 2, \dots \quad (11)$$

其中: $e^\lambda = \sum_{k=0}^{\infty} \left(\frac{\lambda^k}{k!}\right)$

假设第 i 时隙为空闲时隙, 则

$$P[X_i = 1] = \left(1 - \frac{1}{l}\right)^n \quad (12)$$

根据式(7),(9) 可得

$$P[X_i = 1] \approx e^{-\lambda} \quad (13)$$

因此

$$E[N_0] = \sum_{i=1}^l P[X_i = 1] \approx le^{-\lambda} \quad (14)$$

同理可得

$$P_r[Y_j = 1] = n \frac{1}{l} \left(1 - \frac{1}{l}\right)^{n-1} \approx \lambda e^{-\lambda} \quad (15)$$

$$E[N_1] = \sum_{i=1}^l P[Y_i = 1] \approx \lambda e^{-\lambda} \quad (16)$$

由于 $X_i + Y_i + V_i = 1$, 那么

$$E[N_c] = \sum_{j=1}^l p[V_i = 1] \approx l - le^{-\lambda} - \lambda e^{-\lambda} \quad (17)$$

将式(14) 带入可得

$$m_0 = E[M_0] = le^{-n_0/l} \quad (18)$$

$$m_1 = E[M_1] = l(n_1/l)e^{-n_1/l} \quad (19)$$

$$m_2 = E[M_2] = l - (l + n_2)e^{-n_2/l} \quad (20)$$

由式(18)可得

$$n_0 = -l * \ln(m_0/l) \quad (21)$$

另外, 由于式(19),(20)是超越方程, 其计算结果根据参数取值结果($m_1 \times e$ 与 l 的大小)分为无解、1个解和2个解3种情况, 所以式(19),(20)无法保证一定能估算出结果。

1.4 估算范围

当标签数 $n \gg l$ 时, 帧中的所有时隙将以高概率遭遇冲突, 导致 $n_0 = n_1 = 0$, 而 $n_2 = l$, 在这种情况下, 对 n 的估算就失去意义。而当标签数 $n \ll l$ 时, 帧中的所有槽将以低概率遭遇冲突, 导致 $n_2 = n_1 = 0$, 而 $n_0 = l$, 在这种情况下, 对 n 的估算也失去意义, 也就是说 n_0, n_1, n_2 的估算是有一定范围的。因为, 如果标签数固定, 随着帧的大小增长(载入因素下降), 估算成功的概率 θ 将增加。换言之, 如果帧的大小固定, 估算成功的标签数具有一定范围。

假设 n 个标签每个随机地从 l 个时隙中选一个时隙并传送入槽, 其中 $n, l \rightarrow \infty$, 载入因子 $\lambda = n/l$ 。那么空闲时隙数 n_0 接近参数为泊松随机变量 $\lambda_0 = le^{-\lambda}$, 而独槽数 n_1 接近参数为泊松随机变量 $\lambda_1 = l\lambda e^{-\lambda}$ 。

因此, 阅读器取得有限个 n_0 的失败概率是 $N_0 = 0$ 的概率, 其中 N_0 是参数 λ_0 的泊松变量。

$$P_r[N_0 = 0] = e^{\lambda_0}$$

当 $\lambda_0 \leq -\log(1 - \theta)$ 时, 失败概率小于 $1 - \theta$, 如果设置 $\theta = 0.99$, 相应的设置 $\lambda_0 \leq 5$, 也就是说如果 $\lambda_0 = 5$, 则失败概率约为0.007。

当估算冲突时隙时, 估算过程仅当时失败, 即没有空闲时隙或独占时隙。其概率公式为

$$P_r[N_0 = 0, N_1 = 0] \approx e^{-\lambda_0 - \lambda_1}$$

再次假定 $\theta = 0.99$, 本文发现只要载入因素保证 $\lambda_0 + \lambda_1 \leq 5$, 那么冲突估算失败的概率小于0.007。图2比较了 $\theta = 0.99$ 时, 空闲时隙估算和冲突时隙估算的估算范围, 其中X轴为时隙数, Y轴为最大标签数。从图2可知, 冲突时隙估算的范围大于空闲时隙估算, 而且随着时隙数增加, 估算范围也随着增加。

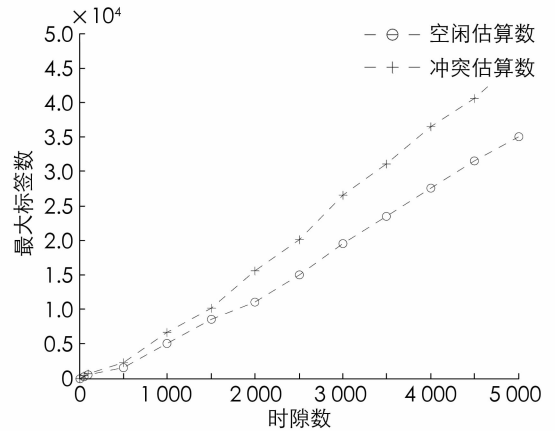


图2 空闲时隙和冲突时隙估算范围

2 算法分析及实验仿真

2.1 算法分析

2.1.1 帧长

本文使用哈希函数的散列特性来线性选择时隙, 在时隙 $t-1$ 响应的概率是 $1/t$, 则 n 个标签在时隙 $t-1$ 响应的概率是 n/t , 为确保效率最高, 在矩阵中“0”, “1”应均匀分布在时隙中, 即帧长大约为 n , 因此本协议帧长复杂度为 $O(n)$ 。

2.1.2 存储空间

在本文中, 由于每个标签都保存一个哈希函数和一个哈希值用于线性选择时隙, 那么阅读器须建立一个矩阵 $B[k]$ 用于保存时隙的使用情况, “0”为未使用、“1”为已使用, 其最小为 n , 因此本协议空间复杂度为 $O(n)$ 。

2.1.3 计算时间

在本文中, 由于 $B[k]$ 存储了时隙占用情况, 为了更充分掌握时隙的占用情况, 需对矩阵 $B[k]$ 进行全面扫描, 这个操作是计算时间要求最多的。由于需对矩阵 $B[k]$ 进行全面扫描, 那么设矩阵扫描时间为 n , 即时间复杂度为 $O(n)$ 。

表1列出了4种常用RFID帧时隙估算算法分析情况。

表 1 4 种帧时隙估算算法分析情况表

	帧长	空间复杂度	时间复杂度
Vogt	$O(n)$	$O(n\log n)$	$O(n^2)$
UPE	$O(n)/p$	$O(n)$	$O(n\log n)$
Quan	$O(n)$	$O(n)$	$O(n^2)$
本方法	$O(n)$	$O(n)$	$O(n)$

从表 1 可以看到, 本协议与 Vogt 协议比较, 帧长相同, 其空间复杂度和时间复杂度更优; 与 UPE 协议比较, 空间复杂度相同, 时间复杂度更优; 与 Quan 协议比较, 帧长和空间复杂度相同, 时间复杂度更优.

2.2 误差率

在最大似然估计中, 为了减少碰撞时隙, 提高算法效率, 本文将标签 ID 作为输入数据, 并利用 hash 函数的线性特性来选择通信时隙, 即 $H(key) \in [0, l-1]$. 由于标签的 ID 值都不相同, 根据 hash 函数的特性, 其输出值必不相同, 这样可以比较合理地选择时隙, 尽量减少冲突. 但当 $n > l$ 时, 必将产生部分冲突, 这将对估算产生一定的误差. 现将估算中产生的误差分析如下:

由式(18),(19) 可得

$$E(M_0^2) - E(M_1^2) = le^{-n/l}(1 - (1 + n/l)e^{-n/l})$$

(22)

则

$$Std(n'/n) = \frac{\sqrt{m}(e^{-n/l} - n/l - 1)^{1/2}}{n}$$

(23)

从公式(23) 可以看出, 误差率与标签数成正比, 与帧长成反比. 如果误差率不变, 则标签数与帧长成正比, 即如果标签数不变, 要保证一定的误差率, 需适当增加帧长. 在第一次估算时, 由于标签数目不确定, 帧长的初始值一般比较小, 产生大范围碰撞的概率较高, 当产生大范围碰撞后, 本文将快速、准确地估算出当前标签数目, 并根据动态帧时隙 ALOHA 算法特性合理地设置帧长. 表 2 列出了当误差率为 5% 时, n 与 l 的关系.

表 2 n 与 l 的关系表

n	500	5 000	10 000	50 000
l	346	1 892	3 418	13 816

本文使用多个哈希函数求平均值的方式来尽量减少误差, 其误差计算公式为^[11]

$$\sigma_n^2 = \sigma_x^2 + Q(\log_2 n) + o(1)$$

(23)

其中 $\sigma = 1.121\ 3$.

本文在阅读器上保存 k 个矩阵, 用于保存时隙使用情况, 其中 k 个最低位记为 $W_1, W_2, \cdots, W_k$, 同时在标签上保存 k 个哈希函数, 则

$$\bar{W} = (W_1 + W_2 + \cdots + W_k)/k$$

(24)

$$\bar{W} \text{ 的期望值: } E(\bar{W}) \approx \log_2(\varphi n)$$

(25)

$$\bar{W} \text{ 的偏差: } \sigma(\bar{W}) \approx \sigma_\infty / \sqrt{m}$$

(26)

由于

$$n' = 1.292\ 8 \times 2 \sum_{iR_i/m}$$

(27)

则误差率为 $Err = \frac{|n' - n|}{n}$, 从而得到误差率与 hash 函数的关系. 表 3 列出误差率与 hash 函数的对照关系.

表 3 误差率与 hash 函数对照情况

M	8	16	32	64	128	256	512
$Err/\%$	31.9	19.7	13.9	10.1	6.68	4.72	3.43

一般来说,为满足实际需求,须要求估算误差率小于实际值的 15%^[12],动态帧时隙 ALOHA 算法就可以合理地设置帧时隙 l ,即哈希函数数目只需要达到 32 个就能满足误差率小于 15%的要求。

2.3 仿真结果

本文采用 Matlab 软件进行仿真测试,仿真范围为 $10 \times 10 \text{ m}^2$,仿真时间为 60 s,均值采用 $500 \mu\text{s}$,数据请求符合指数分布,仿真环境为自由空间传输路径损耗,不考虑衰减。RFID 阅读器的读写距离为 1.7 m,工作频率为 915 MHz,移动速率不超过 0.5 m/s,阅读器最初分布随机,RFID 标签均匀分布,所有仿真结果采用计算平均值的方式,每个结果重复 100 次。其计算公式为

$$Err = \frac{|n' - n|}{n} \quad (28)$$

$$Aerr = \frac{1}{100} \sum \frac{|n' - n|}{n} \quad (29)$$

$Aerr/\%$ 实验结果如图 3 所示。

从图 3 可知,当时隙数目 l 很小时,估算误差率很大,超过 30%,即当 $n \gg l$ 时,帧中的所有时隙将以高概率遭遇冲突,估算的误差很大,但随着时隙数目 l 增加,即载入因子 λ 逐渐减少,估算误差率依次减少,当超过 256 后,误差率小于 5%。估算误差率与标签数目大小关系不大,只与时隙数目 l 成正比,当 l 一定时,随着标签数目的增加,估算误差率略有增加。同时由图 3 可知,随着标签数目 n 的增加,载入因子 λ 逐渐增加,估算误差率也随着增加,但只要时隙数 l 超过 256 后,估算误差率都小于 5%。

从图 4 可知, Vogt 算法在 λ 较小时,估算误差率比较小,这是由于 Vogt 算法是采用符合碰撞时隙、空时隙、独占时隙 3 种时隙的数目进行遍历,搜索出最优算法,故 Vogt 估算在 λ 较小时比较准确。但 Vogt 算法的误差率与 λ 成正比,即随着 λ 增加,估算误差率上升,当 $\lambda=2$ 时,误差率大约为 15%,基本达到动态帧时隙 ALOHA 算法要求的边界;当 $\lambda=4$ 时,误差率大约为 40%,误差过大。UPE 算法的误差率也与 λ 成正比,但其估算误差率比较大,基本在 40%以上,无法完成标签数目准确估算。ML 算法误差率与 λ 成反比,当 $\lambda > 0.3$ 时,ML 算法误差率基本低于 15%,满足动态帧时隙 ALOHA 算法要求。当 $\lambda > 2$ 左右时,ML 算法误差率基本维持在 5%以下,同时当 $\lambda > 1.2$ 时,ML 算法误差率比 Vogt 算法和 UPE 算法的误差率要低。这是由于 ML 算法采用最大似然算法进行估算,当标签数目在估算范围内,其估算精确度较高。本文采用 LFSR 来生成随机数,实现哈希函数,使得标签能够均匀选择不同的时隙,有效地减少估算误差。

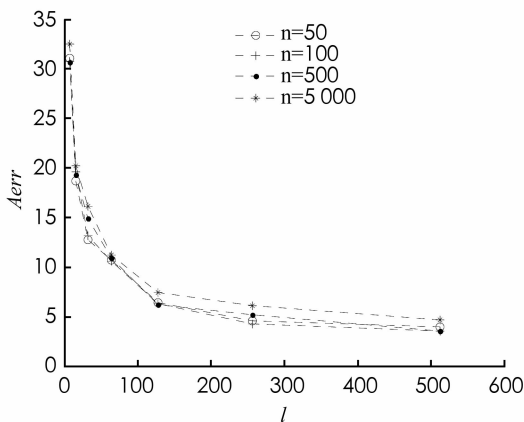


图 3 $Aerr$ 实验结果

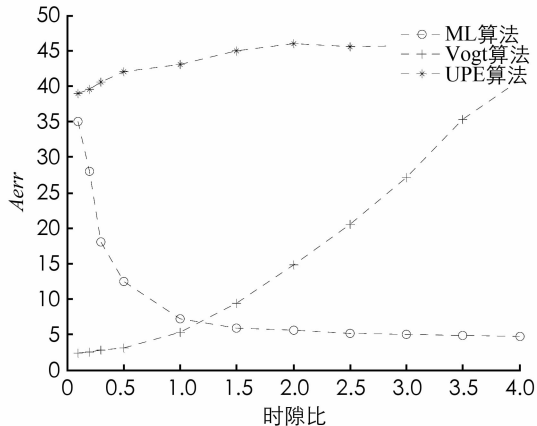


图 4 3 种误差算法误差比较

3 结 语

本文提出了一种基于最大似然估计的标签数目估算方法。通过 Matlab 仿真实验,当标签数目在估算范围内,ML 算法的估算精确度较高。由于本文采用的是数理统计方法估算标签,因此标签数量必须在一定范围内才能确保标签估算成功。同时,本文推导出了估算失败概率与时隙数的关系,并得出通过冲突时隙估算的范围大于空闲时隙估算范围。

参考文献:

- [1] SHAHZAD M, LIU A X. Fast and Accurate Estimation of RFID Tags [J]. IEEE/ACM Transactions on Networking, 2015, 23(1): 241—254.
- [2] WANG Y, WU H, ZENG Y. Capture-Aware Estimation for Large-Scale RFID Tags Identification [J]. Signal Processing Letters IEEE, 2015, 22(9): 1274—1277.
- [3] ZHENG Y, LI M. Probabilistic Estimating Tree for Large-Scale RFID Estimation [J]. Mobile Computing IEEE Transactions on, 2012, 11(6): 1763—1774.
- [4] QUAN C, MO H S, LEE D H, et al. Method for Estimating Number of Tags in Slotted Aloha-Based RFID System; US 7675403 B2 [P]. 2010.
- [5] MYUNG J, LEE W, SRIVASTAVA J, et al. Tag-Splitting: Adaptive Collision Arbitration Protocols for RFID Tag Identification [J]. IEEE Transactions on Parallel & Distributed Systems, 2007, 18(6): 763—775.
- [6] VOGT H. Efficient Object Identification with Passive RFID Tags [J]. Lecture Notes in Computer Science, 2012, 24(14): 98—113.
- [7] BARBARA M, FILIPO R. Identification of Nonlinear Aeroelastic and Aeroservoelastic Systems Using High-Order Iterative Technique [C]//In: 47th AIAA Guidance, Navigation, and Control Conference. Newport: American Institute of Aeronautics and Astronautics Press, 2006.
- [8] MYUNG J, LEE W. Adaptive Splitting Protocols for RFID Tag Collision Arbitration [C]//ACM International Symposium on Mobile Ad Hoc NETWORKING and Computing, MOBIHOC 2006. Florence: DBLP, 2006.
- [9] DENG D J, LIN C C, HUANG T H, et al. On Number of Tags Estimation in RFID Systems [J]. IEEE Systems Journal, 2015(99): 1—8.
- [10] ZHAO J, WANG F, LI D, et al. Energy-Balanced Algorithm for RFID Estimation [J]. International Journal of Electronics, 2016, 103(10): 1617—1625.
- [11] JIANG W, ZHU Y, LI B. AFR: Accurate and Fast RFID Estimation [C]//Global Communications Conference (GLOBECOM). Singapore: 2013 IEEE, 2013.
- [12] ŠOLIĆ P, RADIĆ J, ROŽIĆ N. Energy Efficient Tag Estimation Method for ALOHA-Based RFID Systems [J]. IEEE Sensors Journal, 2014, 14(10): 3637—3647.

A Method for Estimation Number of RFID Frame Time Slot Based on Maximum Likelihood Estimation

YANG Ling, CAI Xu-can, ZOU Juan

Department of Information Science and Technology, Zhongkai University of Agriculture and Engineering, Guangzhou 510225, China

Abstract: A problem has been analyzed in this paper on efficiency optimization based on dynamic frame slotted ALOHA algorithm, and a method been presented on estimating RFID frame time slot based on Maximum Likelihood Estimation(MLE). According to collision time slot and idle time slot obtained by previous frame, the method uses MLE to estimate frame time slot and LFSR to gain the hash function at the same time, making tags choose time slot evenly, reduce the error of estimation, and decrease the error rate. Simulation test results show; the error of MLE will obviously decrease with the increase of the ratio of time slot of tags, and it satisfies actual requirements that the error rate is less than 15%. When the ratio of time slot is more than 1.2, the error of MLE is less than the Vogt algorithm and UPE algorithm, and the error of MLE is less than 5%.

Key words: Maximum likelihood Estimation; frame time slot; ALOHA algorithm; Estimation