

一种时间高效的易于实现的多标签射频识别技术

苏健^{1,3}, 杨晓娇², 韩雨³

(1. 南京信息工程大学计算机与软件学院, 江苏南京 210044; 2. 重庆交通大学信息技术中心, 重庆 400074;
3. 电子科技大学通信与信息工程学院, 四川成都 611731)

摘要: 多标签碰撞问题严重影响射频识别 RFID 系统的识别效率. 在研究子帧观测机制的基础上, 针对常规动态帧时隙 Aloha 多标签防碰撞算法存在的复杂度高、时间效率低等问题, 提出了一种基于子帧的动态帧时隙 Aloha 算法, 其在识别过程中采取设定的子帧观测, 运用空闲与碰撞时隙数的关系估计剩余标签数, 再依据预估的结果优化设置新的帧长, 显著提升了大容量多标签 RFID 系统的识别效率. 该算法的运算复杂度低、计算量小, 易于在常规 RFID 读写器中实现, 工程应用前景广阔. 仿真结果表明: 同传统的 Aloha 类防碰撞算法相比, 提出的算法具有复杂度低、稳定好、识别效率高等优势.

关键词: 射频识别; 防碰撞; 复杂度; 时间效率

中图分类号: TP399

文献标识码: A

文章编号: 0372-2112 (2018)04-0903-08

电子学报 URL: <http://www.ejournal.org.cn>

DOI: 10.3969/j.issn.0372-2112.2018.04.019

A Time-Efficient and Easy-to-Implement RFID Technology for Multiple Tags

SU Jian^{1,3}, YANG Xiao-jiao², HAN Yu³

(1. School of Computer and Software, Nanjing University of Information Science and Technology, Nanjing, Jiangsu 210044, China;

2. Information Technology Centre, Chongqing Jiaotong University, Chongqing 400074, China;

3. School of Communication and Information Engineering, University of Electronic Science and Technology of China, Chengdu, Sichuan 611731, China)

Abstract: Multiple tags collision problem severely degrade the identification efficiency of an RFID system. Based on the research of sub-frame observation mechanism, a sub-frame based dynamic framed slotted Aloha (SUBF-DFSA) algorithm is presented to tackle the high computation complexity and low time efficiency existing in the conventional DFSA algorithm. The SUBF-DFSA algorithm makes effective use of the idle and collision statistics during a sub-frame to estimate the tag backlog, determine the optimal frame size for the next identification round, so that it can dramatically improve the identification efficiency of an RFID system with large volume of tags. It is easier to implement into the conventional RFID reader because of the low computation overhead. Simulation results are supplemented to demonstrate the advantages of the proposed algorithm in achieving low computation complexity, good stability, and high identification efficiency compared to the traditional Aloha-based algorithms.

Key words: radio frequency identification; anti-collision; complexity; time efficiency

1 引言

射频识别 (Radio Frequency Identification, RFID) 是一种用于目标自动识别的无线通信技术, 现已广泛的应用于物流管理、供应链管理、室内定位、工业自动化等

领域. 在大多数工业应用中, 无源超高频 (UHF) RFID^[1] 由于其通信距离长、识别速度快、存储容量大、成本低等优点, 受到更多的青睐. RFID 系统的一个典型应用是将大量标签贴在高密集度的产品上, 然后快速而有效的识别它们. 从而实现目标产品的追踪与识别. 由于多个

标签共享同一个信道与读写器进行通信,从而导致了多标签碰撞问题. UHF RFID 系统读写器覆盖范围广,系统内覆盖的标签数目很多,同时对标签的读写要求很高,这样使得 UHF RFID 系统的标签碰撞问题更为突出. 因此,高效、稳定、复杂度低的防碰撞算法是 UHF RFID 系统研究的一个核心问题,也是 UHF RFID 产业化的一个先决条件.

技术上,多标签防碰撞算法可以分为 Aloha 类算法^[2-4]和树形算法^[5,6]. 树形算法的核心在于碰撞比特识别和追踪技术. 然而,在 UHF RFID 系统中,由于读写器端的接收信号存在频率偏移,导致读写器不能有效的检测到具体的碰撞位,因此使得树形算法难以应用到 UHF RFID 系统中,诸如 EPCglobal C1 Gen2^[7]或 ISO/IEC 18000-6B^[8]. 例如,在 EPCglobal C1 Gen2 系统中,不同标签的调制信号的符号速率最大偏移可以达到 22%^[9],这就意味着不同标签的响应最大间隔时间为 24 μ s. 类似地,在 ISO/IEC 18000-6B 标准中,不同标签的符号速率最大偏差达到 15%.

Aloha 类算法无需检测碰撞的具体位置,从而可以很好地适用于 UHF RFID 系统. 特别地,动态帧时隙 Aloha (DFSA)是目前应用最广泛的一种 Aloha 算法,该算法的最大优势在于:将时间分成动态可调整的时隙,多个时隙为一帧,标签在一帧中随机选择一个时隙来响应读写器操作,在每轮识别中,读写器根据标签剩余数来动态的调整帧的大小,从而完成对标签的识别. 由于 DFSA 算法对标签数的估计都是基于上一帧的统计结果,一旦出现估计错误就会对最终的性能造成较大的影响. 为了提高标签数估计的准确性,大多数现有算法采用最优化和循环迭代搜索,这些都要求较高的计算负载或存储空间,无法满足移动读写器. 近来,研究者们提出了一些先进的 DFSA^[10-12]类算法,可以应用到计算能力受限的读写器. 文献[10]提出了一种基于分组动态帧时隙的 RFID 防碰撞算法 (Grouped Dynamic Frame Slotted Aloha, GDFSA). 该算法首先预估待识别的标签数,然后采用分组以及动态帧时隙策略对标签进行识别. GDFSA 算法的标签数估计方法需要不断的迭代搜索,这无疑会增加计算开销. 文献[11]提出了一种可行的易于实现的防碰撞算法 (Feasible Easy-to-Implement Anti-Collision, FEIA). 该算法需要在每个时隙都进行一次标签数估计和帧长的设置,这同样会增加读写器的计算成本,特别是计算能力严格受限的移动读写器. 此外,FEIA 算法受限于一帧中初始时隙为空闲状态的应用场景. 为了降低读写器的能耗,作者在文献[12]中提出了一种提高型线性组合模型 (Improved Linearized Combinatorial Model, ILCM),ILCM 引入适度的浮点运算 (Floating Point Operations, FLOP) 成本,可以作为标

签数估计方法应用于防碰撞算法中. ILCM 降低了标签数估计所产生的能量消耗,但是它的性能随着标签数的增加而恶化.

为了降低 DFSA 算法的计算复杂度和确保其识别性能的可靠性,本文针对 EPCglobal C1 Gen2 标准提出了一种基于子帧的动态帧时隙 Aloha (SUB-Frame based Dynamic Frame Slotted Aloha, SUBF-DFSA) 算法. 该算法基于对当前子帧 (完整帧的前 m 个时隙) 的观测来确定下一帧的帧长. 在子帧中利用空闲时隙和碰撞时隙的关系来提高标签估计的有效性. 除此之外,考虑到不同时隙状态的持续时间的差异,传统的系统吞吐率指标^[2,3,10-12] (标签数与所需的总时隙数之间的比率) 不能有效地从识别时间的角度来衡量算法的性能. 因此,本文在算法的设计中考虑了时间效率. 仿真结果显示 SUBF-DFSA 在时间效率方面优于现有算法. 测试结果进一步表明:本文所提出的 SUBF-DFSA 算法具有很好的识别性能,超过了 Impinj 公司设计的 R2000 防碰撞算法.

2 算法描述

2.1 帧长的调整策略

大多数现存 DFSA 算法解决碰撞和调整帧长都是基于上一个完整帧的观测结果. 一旦上一帧的帧长不合适,性能就会受到较大的影响. EPCglobal C1 Gen2 标准指定了一种基于帧内调整机制的 Q 算法. Q 算法的主要优势在于便于实现,但是标准中并未详细给出具体的调整策略.

SUBF-DFSA 算法利用子帧 (子帧的大小为 F_{sub}) 来确定下一帧的帧长. 当读完 F_{sub} 个时隙后,读写器统计子帧中空闲时隙和碰撞时隙出现的次数. 如果两者的比值超过一定的阈值范围,那么读写器就根据子帧中统计的结果来调整帧长. 假定系统内存在 n 个待识别标签,帧长为 F ,那么系统吞吐率 U 可以计算为^[3]:

$$U = \left(\frac{n}{F}\right) \cdot \left(1 - \frac{1}{F}\right)^{n-1} \quad (1)$$

当 $F = n$ 时,系统可以实现最大吞吐率 0.368. 由于 EPCglobal C1 Gen2 标准规定,帧长的设定必须为 $2^{Q[7]}$ (其中 Q 为 0 到 15 之间的整数),所以帧长 F 有时不能严格等于 n . 为了实现稳定的吞吐率,读写器需要针对估计出的标签数合适的调整帧长. 如图 1 所示,对于不同大小的帧 F ,都对应一组 n 值,这组 n 值在该 F 下可以实现最高的吞吐率.

为了避免性能恶化,我们针对不同的标签数范围推导出了最佳的帧长. 给定一个 n ,令 $F_L (F_L = 2^Q)$ 下的吞吐率与 $F_H (F_H = 2^{Q+1})$ 下的吞吐率相等, F_L 和 F_H 为相邻的两个帧长,我们有

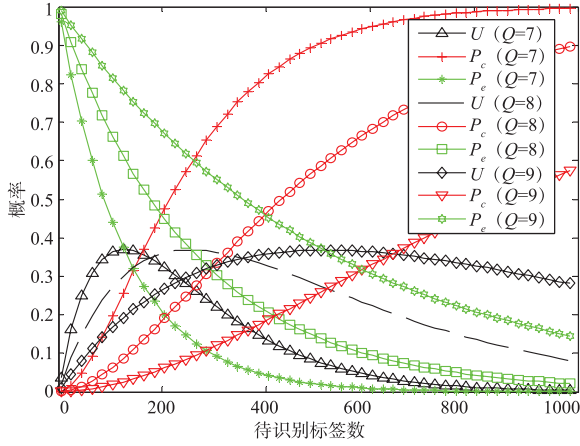


图1 不同 F 下的吞吐率,空闲概率(P_e)和碰撞概率(P_c)的关系

$$\left(\frac{n}{F_L}\right) \cdot \left(1 - \frac{1}{F_L}\right)^{n-1} = \left(\frac{n}{F_H}\right) \cdot \left(1 - \frac{1}{F_H}\right)^{n-1} \quad (2)$$

其中 n 是决定是否需要调整帧长的关键值,通过式(2), n 可以表示为:

$$n = \left\lfloor \frac{1 + \ln\left(\frac{F_H}{F_L}\right)}{\ln\left(\frac{F_L \cdot F_H - 1}{F_H \cdot F_L - 1}\right)} \right\rfloor \quad (3)$$

其中 $\lfloor * \rfloor$ 表示向下取整. 例如, $F_H = 8, F_L = 4$ 时, $n = 5$. 它表示帧长 $F = 4$ 的标签数上限为 5. 同样的,可以算出帧长 $F = 4$ 的标签数下限为 4. 那就是说对于标签数范围 4 到 5 而言,最佳帧长为 4. 值得注意的是,作者在文献[13]中提出了另外一种 Q 值选择方法,该方法可以得出相同的结果. 但是本文的方法更为简单、直观且无需多次迭代.

2.2 标签数估计方法

图 1 同样反映了吞吐率,空闲概率以及碰撞概率之间的关系. 从图中可以观察到,吞吐率为凸函数,空闲概率(P_e)和碰撞概率(P_c)分别为递减函数和递增函数. 那我们可以定义如下的估计方法. 当合理的调整帧长时,最优 DFSA 算法可以实现最大的吞吐率 0.368. 我们给定一个较高的平均吞吐率 U , 确保 $U \geq 0.35$, 然后计算出满足不同 F_{cur} 和 n_{est} 关系下的 P_e 和 P_c 的关系,结果如表 1 所示.

表 1 不同 n_{est} 和 F_{cur} 下, P_e 和 P_c 的关系

n_{est} 和 F_{cur} 的关系	空闲概率和碰撞概率应满足的关系
$n_{est} = F_{cur}/4$ ($C_f = -2$)	$15.1P_c < P_e \leq 63.8P_c$
$n_{est} = F_{cur}/2$ ($C_f = -1$)	$3.2P_c < P_e \leq 15.1P_c$
$n_{est} = F_{cur}$ ($C_f = 0$)	$0.6P_c < P_e \leq 3.2P_c$
$n_{est} = 2F_{cur}$ ($C_f = 1$)	$0.08P_c < P_e \leq 0.6P_c$
$n_{est} = 4F_{cur}$ ($C_f = 2$)	$0 \leq P_e \leq 0.08P_c$

其中 F_{cur} 表示当前帧的帧长度, n_{est} 表示本轮估计出的待识别标签数. 不同帧长 F_{cur} 和待识别标签数 n_{est} 关系下, 空闲概率 P_e 和碰撞概率 P_c 的关系推导如下:

S1: 定义 $C_f = \log_2(n_{est}/F_{cur})$, 初始化 $Q = Q_{ini} = 4$ (参考 EPCglobal C1 Gen2 标准), $i = 1$;

S2: 判断当前 Q 值是否超过设定的最大值, 若是则跳转到步骤 S4, 否则进入步骤 S3;

S3: 令 $Q_f = Q + C_f$, 令 $Q_{next} = Q_f$ 并根据表 1 得到 n_{est} 对应的最小值 n_1 和最大值 n_2 , 计算空闲概率和碰撞概率: $P_e(n_1)/P_c(n_1) = b_i, P_e(n_2)/P_c(n_2) = a_i$, 其中, $F = 2^Q, P_e = (1 - 1/F)^n, P_c = 1 - P_e - U$, 空闲概率与碰撞概率的关系为: $a_i * P_c \leq P_e \leq b_i * P_c$, 然后 $Q++$, $i++$, 并返回 S2;

S4: 对所有的 a_i 及 b_i 分别求平均值, 得到空闲概率 P_e 和碰撞概率 P_c 的约束关系, 为了保证约束的连续性, 对任意两个相邻的约束取平均值.

考虑到一个完整帧的前 m 个时隙(子帧 $F_{sub} = m$), 其空闲概率和碰撞概率关系的比值满足 P_e/P_c , 因此, 表 1 适用于不同大小的子帧. 通过表 1 中的空闲概率和碰撞概率的关系, 读写器可以在子帧结束后计算空闲概率和碰撞概率的关系从而判断当前的帧长是否合适.

2.3 算法流程

EPCglobal C1 Gen2 标准规定, 帧长 $F = 2^Q$ (其中 Q 为 0 到 15 之间的整数), 所以子帧 F_{sub} 的设置通常为 $F/2, F/4, F/8, F/16$ 等. 由于在识别过程中, 帧长 F 是动态调整的, 因此 F_{sub} 也是动态变化的, 表 2 列出了不同完整帧 F 下的子帧 F_{sub} 的取值.

表 2 不同帧长 F 下的子帧 F_{sub} 设置

F	8	16	32	64	128	256	512	>512
F_{sub}	F	$F/2$	$F/4$	$F/8$	$F/8$	$F/16$	$F/32$	$F/64$

结合标签数估计, 帧长调整策略, 本文提出的 SUBF-DFSA 算法的流程如图 2 所示.

由图 2 可以看出, 本文提出的算法融合了标签数估计, 最佳帧长设置和子帧识别的原则, 克服了现存算法在标签数量增长情况下吞吐率波动及估计过程中计算复杂度高等缺陷, 具有高吞吐率、低复杂度和稳定好的优势. 仿真、实验环节将验证本文算法的有效性.

3 仿真实验

我们衡量了 SUBF-DFSA 算法的时间效率, 并同现有的算法进行了对比分析, 包括: 最大后验概率估计算法 (Maximum A Posteriori Estimation, MAP)^[3], GDFSA^[10], FEIA^[11] 以及 ILCM^[12]. 所有的仿真实验均在 MATLAB 2010 平台上进行. 为了确保实验结果的可靠性, 采用了蒙特卡洛仿真方法, 标签数在 20 到 1000 之

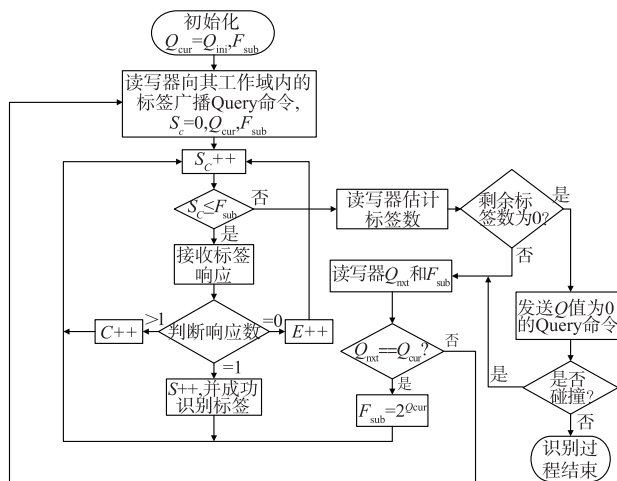


图2 SUBF-DFSA算法流程图

间变化,步长为 20,每次实验的仿真次数为 500 次. 根据 EPCglobal C1 Gen2 标准的规定^[7],时间效率 T_{effi} 可以定义为:

$$T_{\text{effi}} = \frac{S \cdot T_{\text{succ}}}{T_{\text{slots}} + T_{\text{FLOP}}} \quad (4)$$

$$T_{\text{slots}} = S \cdot T_{\text{succ}} + E \cdot T_{\text{idle}} + C \cdot T_{\text{coll}} \quad (5)$$

其中 T_{slots} 为发送和接收数据所需的时间, T_{FLOP} 为浮点运算(FLOP)所花费的时间. 其中 S, E, C 分别为识别过程中的成功时隙数,空闲时隙数和碰撞时隙数. $T_{\text{succ}}, T_{\text{idle}}$ 和 T_{coll} 分别为单个成功时隙,空闲时隙和碰撞时隙的持续时间.

$$T_{\text{idle}} = T_{\text{cmd}} + T_1 + T_3 \quad (6)$$

$$T_{\text{succ}} = T_{\text{cmd}} + 2(T_1 + T_2) + T_{\text{RN16}} + T_{\text{ACK}} + T_{\text{PC+EPC+CRC}} \quad (7)$$

$$T_{\text{coll}} = T_{\text{cmd}} + T_1 + T_{\text{RN16}} + T_2 \quad (8)$$

上述公式中 T_{cmd} 为读写器在识别过程中所发送命令的持续时间,其具体命令主要包括 Query, QueryAdj 和 QueryRep 等^[7].

$$T_{\text{FLOP}} = \frac{\sum_{i=1}^{T_{\text{sw}}} N_{\text{FLOP}}^i}{R_{\text{FLOPS}}} \quad (9)$$

式(9)中 T_{sw} 定义为识别轮数,由读写器在整个识别过程中统计得到. 考虑到便携式 RFID 读写器的处理器通常为单核嵌入式处理器,例如:ARM AT91SAM7S256,同智能手机或平板电脑的多核处理器相比,计算能力非常受限. 因此,我们将 FLOP 列为防碰撞算法的设计中需要考虑的另一个重要指标. FLOP 可以很好的衡量算法的复杂度以及可行性. 显而易见,如果一个算法的 FLOP 很高,那么该算法就不适用于计算能力受限的读写器. N_{FLOP}^i 定义为第 i 轮识别过程中读写器所需的 FLOP. R_{FLOPS} 表示读写器的浮点运算能力. 在我们的仿真中, R_{FLOPS} 设置为 6 MFLOPS (10^6 FLOP/s).

为了获得识别单个标签所需要的平均时间 ($T_{\text{slots}} + T_{\text{FLOP}})/n$,我们需要计算整个识别过程中每一步所花费的时间,包括,数据接收、发送、命令占用时间(式(6)~(8)),以及 FLOP 所需的时间(式(9)). 表 3 列出了仿真中所采用的时间参数,其设置均参考 EPCglobal C1 Gen2 标准.

表 3 仿真参数设置

参数	取值	参数	取值
读写器到标签 data-0	1Tari	RTcal	37.5 μs
读写器到标签 data-1	2Tari	TRcal	50 μs
读写器到标签数据率	80kbps	T1	62.5 μs
标签到读写器数据率	160kbps	T2	62.5 μs
Tpri	6.25 μs	T3	50 μs
Tari	12.5 μs	RN16	16bits
DR	8	EPC	128bits
Query	22 bits	Ack	18bits
QueryAdj	9bits	QueryRep	4bits

五种算法在不同的初始帧长 F_{ini} 下的时间效率如图 3 所示. 从图 3(a)(b)(c)可以看出 MAP, ILCM 以及 GDFSA 算法的时间效率受初始帧长影响较大. 随着标签数量增大,而初始帧长很小时,这些算法均不能调整合适的帧长来适应未识别标签,从而导致性能恶化. 换言之,这三种算法的稳定性较差以至于不能适应标签数在大范围内变化. 作为对比,FEIA 算法的时间效率基本独立于初始帧长,那即是说 FEIA 算法可以在识别过程中有效的调整帧长从而适应当前的剩余标签数. 然而,该算法需要在每个时隙对剩余标签数进行估计且必须依赖于已读时隙的统计结果,所以当已读的时隙数不够大时,估计结果会变得不准确从而影响性能. 因此,在标签数较少时,FEIA 的性能维持在较低的水平. 从图 3(e)可以看出, SUBF-DFSA 算法的时间效率几乎收敛于 0.7125. 在 SUBF-DFSA 算法中,读写器可以通过子帧的观测结果来判定当前帧长是否适合,从而做出相应的调整来实现稳定的性能.

为了更好的比较算法性能的稳定性,我们定义如下稳定函数:

$$C_v = (S_{TD}/M_{\text{ave}}) \quad (10)$$

其中 S_{TD} 和 M_{ave} 分别表示变量(时间效率或平均识别时间)的标准差和均值. C_v 反映了性能的稳定性. 特别地, C_v 越小,表示该性能的波动性越小,则说明稳定性越好. 表 4 列出了标签数在 20 到 1000 之间以步长 20 变化时,不同算法分别以 16, 32, 64, 128 和 256 进行识别时,所能达到的平均时间效率 (T_{effi}), 单个标签的平均识别时间 (T_{ave}), 时间效率的稳定性 (C_{effi}), 平均识别时

间的稳定性 (C_{ave}). 其中每一列括号内的百分比表示对 应性能的提高百分比, (*) 表示以此为参考值.

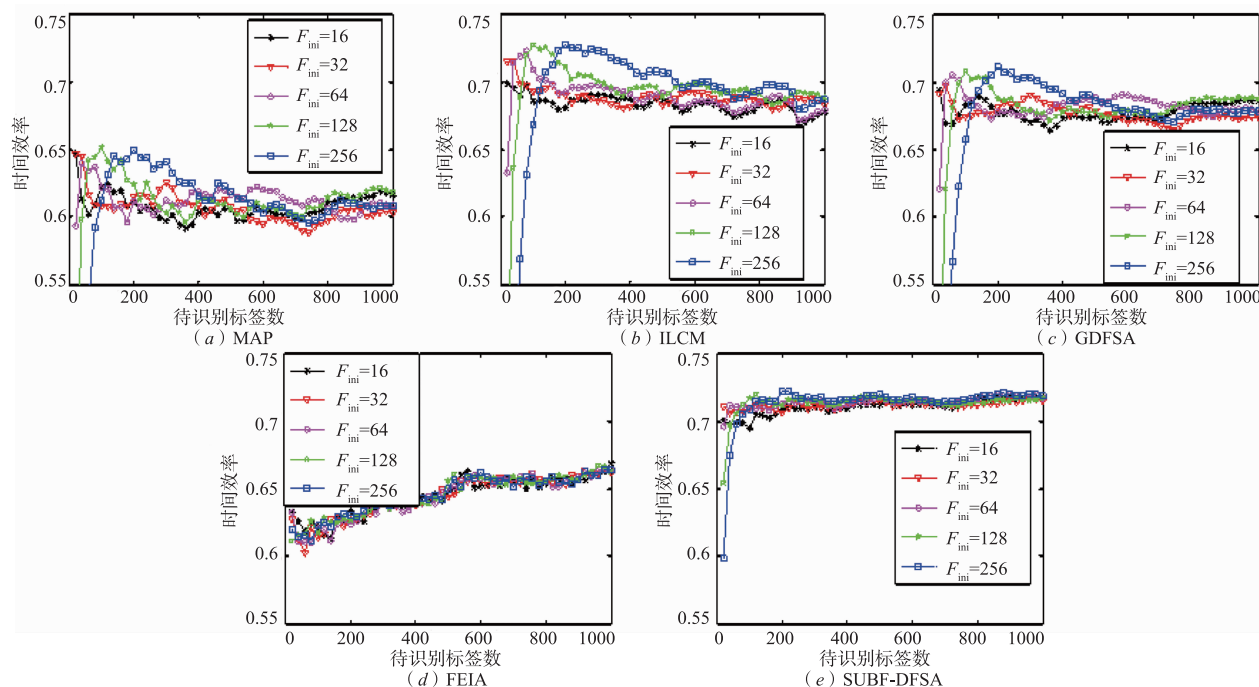


图3 初始帧长对时间效率的影响

表 4 不同初始帧长下的平均性能对比

算法	T_{eff}	C_{eff}	T_{ave}	C_{ave}
MAP	0.6086 (*)	4.52% (10.9%)	2.5347 (*)	7.56% (17.8%)
FEIA	0.6446 (5.92%)	2.33% (54.0%)	2.3865 (5.85%)	2.36% (74.4%)
GDFSFA	0.6769 (11.2%)	4.88% (3.75%)	2.2812 (10.0%)	8.97% (2.50%)
ILCM	0.6877 (13.0%)	5.07% (*)	2.2457 (11.4%)	9.20% (*)
SUBF-DFSFA	0.7125 (17.1%)	1.35% (73.4%)	2.1584 (14.9%)	1.52% (83.5%)

从表 4 中可以看出 SUBF-DFSFA 算法在时间效率, 平均识别时间, 以及稳定性上都具有最佳的性能.

4 测试结果

如前文所述, SUBF-DFSFA 算法是基于 EPCglobal C1 Gen2 标准设计, 且完全兼容现有的读写器和标签. 为了验证 SUBF-DFSFA 算法在实际 RFID 平台上的有效性, 我们在真实的 RFID 硬件平台上对本文提出的 SUBF-DFSFA 算法进行了性能测试.

在本次测试中, 读写器和标签都是基于 UHF RFID 国际标准 EPCglobal C1 Gen2. 读写器的核心部件采用

Atmel 公司生产的 AT91SAM7S256 微处理器和 Impinj 公司生产的 Indy R2000 芯片^[14]. 其中 AT91SAM7S256 是一款基于 ARM7TDMI 内核的 32 比特 RISC 微处理器, 单周期存取频率为 30MHz, 片上带有 256KB 的 Flash 存储单元和 64KB 的 SRAM 存储单元. Indy R2000 是 Impinj 公司生产的适用于 UHF 频段的 RFID 读写器芯片, 具有性能高、功耗低、接收灵敏度高等优点, 能够满足远距离读取和多标签识别的需求. Indy R2000 读写器芯片集成了射频和基带模块, 能够从兼容的 RFID 标签接收数据, 该芯片现已被工业界和企业界广泛使用. 为了便于描述, 后文将 AT91SAM7S256 微处理器和 Indy R2000 芯片为核心模块组成的读写器统称为 R2000 读写器, 将 Impinj 公司自己设计的防碰撞算法称为 R2000 防碰撞算法^[15]. 防碰撞算法测试环境如图 4 所示.

本节测试所用的参考算法为 Impinj 公司设计的 R2000 防碰撞算法和 EPCglobal C1 Gen2 标准规定的 Q-算法. 根据数据编码、调制方式和速率的不同, 测试实验分为两组. 表 5 所示为每组测试读写器与标签通信的链路参数.

为了更直观的对比防碰撞算法的实际性能, 并考虑到真实的应用场景, 本节采用了以下三个性能测试指标.

标签平均盘存个数 相同的盘存时间内成功盘存到的标签数的期望, 定义如下:

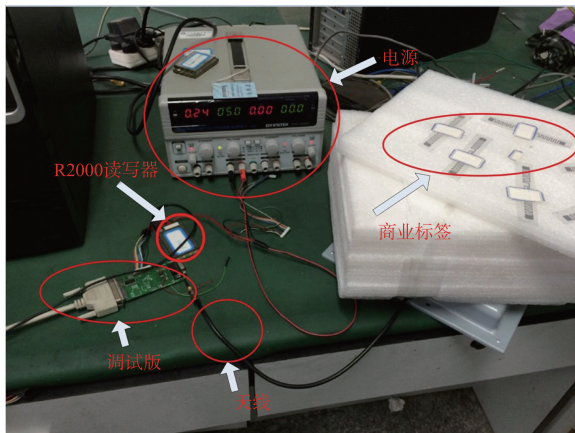


图4 防碰撞算法测试环境

表5 读写器与标签通信链路参数设置

测试场景	1	2
调制方式	PR-ASK	DSB-ASK
前向编码	PIE	PIE
Tari (μs)	25	6.25
RTcal (μs)	62.5	15.63
DR	64/3	8
反向编码	Miller-4	FM0
反向链路频率 (kHz)	250	400
反向速率 (kbps)	62.5	400

$$N_{inv} = \frac{\sum_{i=1}^{cycle} N_i}{cycle} \quad (11)$$

其中 N_i 表示第 i 次盘存周期中成功盘存到的标签数, $cycle$ 表示最大盘存周期, 本章测试中 $cycle$ 等于 60. 该指标直观的反映了防碰撞算法的盘存性能, 即盘存率越高, 盘存性能越好.

盘存波动率 该指标的定义参照式(10), 其具体定义如下:

$$f_{inv} = \frac{S_{inv}}{M} \quad (12)$$

其中 S_{inv} 表示多轮盘存后, 成功盘存到的标签数的标准差, M 表示成功盘存到的标签数的均值. 该指标反映了防碰撞算法的稳定性, 盘存波动率越小, 则说明该算法越稳定.

盘存误差 该指标定义如下:

$$e_{inv} = \frac{|E(n) - n|}{E(n)} \quad (13)$$

其中 $E(n)$ 表示期望盘存到的标签数, n 表示实际盘存

到的标签数. 平均盘存误差越低, 就说明防碰撞算法的性能越好.

图5所示为读写器在测试场景1中, 分别采用SUBF-DFSA算法、R2000算法和Q-算法盘存标签的测试结果. 从图5(a)中可以观测到, 在相同的时间内SUBF-DFSA算法可以识别到更多的标签数. 随着盘存时间的增加, SUBF-DFSA算法的性能优势越来越明显. 特别地, 当盘存时间设置为400ms时, SUBF-DFSA算法成功盘存到的标签数为106, 而R2000算法和Q-算法成功盘存到的标签数分别为94和67. 从图5(b)可以观测到, 随着盘存时间的增加, 各个算法的盘存波动率都会降低. 那是因为, 在总的标签数不变的情况下, 当盘存时间增加到一定值的时候, 各个算法能够成功盘存到的标签数会逐渐趋于一个常数. 从图5(b)中还可以看出, 相较于R2000算法和Q-算法, SUBF-DFSA算法在盘存波动率方面也存在绝对的优势. SUBF-DFSA算法的平均盘存波动率为3.73%, R2000算法和Q-算法的平均盘存波动率分别为6.96%和5.67%. 图5(c)比较了测试场景1中, 各个算法在不同的盘存时间下的盘存误差. 从图5(c)中可以发现, 当盘存时间较短时, 各个算法的盘存误差较大, 导致很多标签被漏读, 随着盘存时间的增加, 盘存误差越来越小, 从而很好的控制了标签漏读几率. 从图5(c)中可以观测到, 在三种算法中, SUBF-DFSA算法的盘存误差曲线收敛最快, 特别地, 当盘存时间为500ms时, SUBF-DFSA算法几乎可以无漏识别全部标签.

图6所示为测试场景2下, 读写器分别采用SUBF-DFSA算法、R2000算法和Q-算法盘存标签的测试结果. 从图6(a)中可以发现, 所有的算法在反向编码采用FM0编码时, 性能都急剧恶化, 在相同时间内成功盘存到的标签数要明显少于采用Miller编码的时候. 那是因为采用Miller编码时, 抗干扰能力强, 读写器端更容易恢复标签信号, 从而能保证读写器与标签之间有更好的通信质量. 同图6(a)所示的结果类似, 在FM0编码情况下, 所有算法的盘存波动率都变大了. 特别地, SUBF-DFSA算法的平均盘存波动率达到了70.4%, 要高于R2000算法和Q-算法. 图6(c)所示为各个算法在场景2下的盘存误差. 由于通信质量变差, 各个算法在场景2下的盘存误差都变得很高, 明显高于场景1下的盘存误差. 然而, 在相同的条件下, SUBF-DFSA算法还是维持着自身的优势, 其盘存误差明显小于R2000算法和Q-算法. 综合图5和图6的测试结果, 我们可以得出结论, SUBF-DFSA算法的性能要优于Impinj公司设计的R2000防碰撞算法, 同时SUBF-DFSA算法的性能远远优于EPCglobal C1 Gen2标准制定的Q-算法.

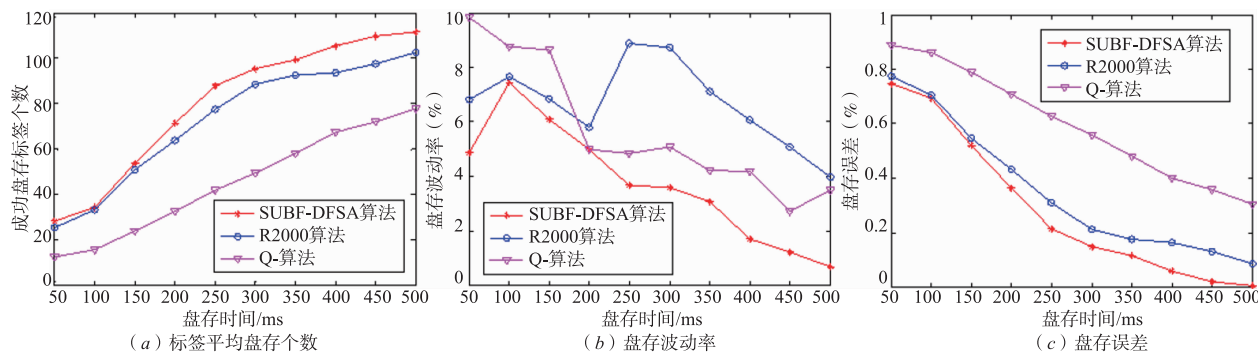


图5 场景1下的测试结果

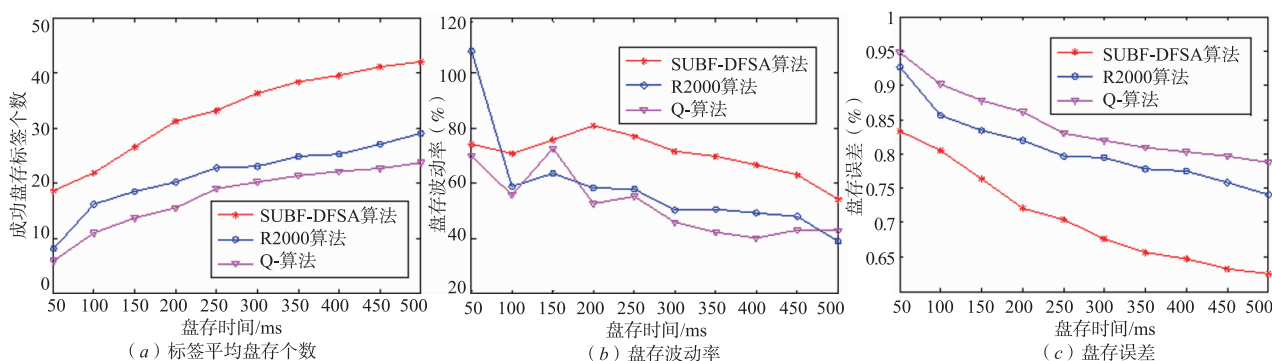


图6 场景2下的测试结果

5 结论

本文基于 UHF RFID 国际标准 EPCglobal C1 Gen2 提出了一种时间高效的动态帧时隙 Aloha 防碰撞算法 SUBF-DFSA. 该算法在传统 DFSA 算法的基础上引入了一种子帧观测机制, 并通过预设的配置表在子帧中估计标签数并最优化帧长设置, 从而降低了系统的复杂度. 仿真结果表明, SUBF-DFSA 算法在吞吐率、时间效率和稳定性方面都具有较大的优势. 通过复杂度分析和比较, SUBF-DFSA 算法也具有低复杂度的特点. 测试结果进一步验证了本文所提出的 SUBF-DFSA 算法的有效性. 因此, 综合结果显示, 本文提出的算法有效的提高了 RFID 系统的识别性能, 具有广阔的应用价值.

参考文献

- [1] LEHTO A, NUMMELA J, UKKONEN L. Passive UHF RFID in paper industry: Challenges, benefits and the application environment[J]. IEEE Transactions on Automation Science and Engineering, 2009, 6(1): 66-79.
- [2] 苏健, 韩雨, 骆忠强, 谢良波, 文光俊. 超高频 RFID 系统中一种可行的时间最优防碰撞算法[J]. 电子学报, 2015, 43(8): 1651-1655.
- SU Jian, HAN Yu, LUO Zhong-qiang, XIE Liang-bo, WEN

Guang-jun. A feasible time-optimal anti-collision algorithm for UHF RFID systems[J]. Acta Electronica Sinica, 2015, 43(8): 1651-1655. (in Chinese)

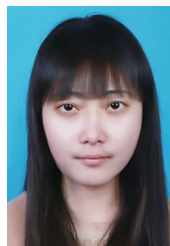
- [3] CHEN W T. An accurate tag estimate method for improving the performance of an RFID anticollision algorithm based on dynamic frame length ALOHA[J]. IEEE Transactions on Automation Science and Engineering, 2009, 6(1): 9-15.
- [4] 付钰, 钱志鸿, 孟婕, 王雪. 基于连续时隙预测的帧时隙 Aloha 防碰撞算法[J]. 电子学报, 2016, 44(9): 2081-2086.
- FU Yu, QIAN Zhi-hong, MENG Jie, WANG Xue. FSA anti-collision algorithm based on continuous slot prediction[J]. Acta Electronica Sinica, 2016, 44(9): 2081-2086. (in Chinese)
- [5] JIA X, FENG Q, YU L. Stability analysis of an efficient anti-collision protocol for RFID tag identification[J]. IEEE Transactions on Communications, 2012, 6(8): 2285-2294.
- [6] 胡应梦, 张小红. 基于信息位编码的自适应搜索 RFID 防碰撞算法研究[J]. 电子学报, 2016, 44(8): 1791-1798.
- HU Ying-meng, ZHANG Xiao-hong. Research of an adaptive searing anti-collision algorithm for RFID based on information-bit encoding[J]. Acta Electronica Sinica, 2016, 44(8): 1791-1798. (in Chinese)

- [7] EPCglobal INC. EPC Radio-Frequency Identify Protocols Class-1 Generation-2 UHF RFID Protocol for Communications at 860MHz – 960MHz[S]. 2013.
- [8] ISO/IEC 18000-6. Information Technology Automatic Identification and Data Capture Techniques — Radio Frequency Identification for Item Management Air Interface — Part 6: Parameters for Air Interface Communications at 860 – 960MHz[S]. 2005.
- [9] ANGERER C, LANGWIESER R, RUPP M. RFID reader receivers for physical layer collision recovery [J]. IEEE Transactions on Communications, 2010, 58 (12): 3526 – 3537.
- [10] 庞宇, 彭琦, 林金朝, 等. 基于分组动态帧时隙的射频识别防碰撞算法[J]. 物理学报, 2013, 62(14): 148401-1 – 148401-8.
PANG Yu, PENG Qi, LIN Jin-chao, et al. Reducing tag collision in radio frequency identification systems by using a grouped dynamic frame slotted Aloha algorithm[J]. Acta Physica Sinica, 2013, 62(14): 148401-1 – 148401-8. (in Chinese)
- [11] CHEN W T. A feasible and easy-to-implement anticollision algorithm for the EPCglobal UHF Class-1 Generation-2 RFID protocol[J]. IEEE Transactions on Automation Science and Engineering, 2014, 11(2): 485 – 491.
- [12] SOLIC P, RADIC J, ROZIC N. Energy efficient tag estimation method for ALOHA-based RFID systems [J]. IEEE Sensors Journal, 2014, 14(10): 3637 – 3647.
- [13] ALONSO J, DELGADO V, LOPEZ E, et al. Multiframe maximum-likelihood tag estimation for RFID anticollision protocols[J]. IEEE Transactions on Industry Informatics, 2011, 7(3): 487 – 496.
- [14] Impinj Inc. Indy R2000 Reader Chip Datasheet[OL]. <http://support.impinj.com/hc/en-us/articles/202755828>, 2015-07-30.
- [15] Impinj Inc. Indy R2000 Firmware Datasheet[OL]. <https://support.impinj.com/hc/en-us/articles/202755738>, 2011.

作者简介



苏 健 男, 1986 年 3 月出生, 湖北荆州人. 2008 年, 2012 年和 2016 年分别在汉口学院、华中师范大学和电子科技大学获工学学士、工学硕士和工学博士学位. 现为南京信息工程大学教师, 主要从事物联网技术、射频识别技术和无线网络等方面的研究工作.
Email: sj890718@gmail.com



杨晓娇 女, 1988 年 4 月出生于重庆长寿. 2013 年毕业于电子科技大学获得工学硕士学位. 现工作于重庆交通大学, 主要研究方向为射频识别技术.

韩 雨 男, 1991 年 10 月出生于河南商丘. 现为电子科技大学通信与信息工程学院博士研究生, 主要研究方向为射频集成电路与系统、无线射频识别技术等.