

基于位编码单元的双时隙防碰撞算法

张学军^{1,2}, 马军飞³, 鲁友²

- (1. 南京邮电大学 宽带无线通信与传感网技术教育部重点实验室, 江苏 南京 210003;
2. 南京邮电大学 电子科学与工程学院, 江苏 南京 210003;
3. 华为技术有限公司南京研究所, 江苏 南京 210008)

摘要:在 RFID 系统中,一个阅读器在同时有多个标签响应时就会发生标签碰撞。文中通过在标签内设置一个编码单元,提出一种基于标签位编码单元(BCU)的双时隙 RFID 防碰撞算法。该算法中,采用位编码单元对碰撞位信息进行重新编码,使阅读器不需要遍历查询就能一次性识别标签的碰撞位信息,从而降低了查询的次数,减少了二进制树算法延时,使得标签能够快速识别。算法性能分析和仿真结果显示,文中所提出的算法在存在大量标签的环境下识别效率有较大的提高。

关键词:射频识别技术;防碰撞算法;Manchester 编码;位编码单元

中图分类号:TN92

文献标识码:A

文章编号:1673-629X(2014)09-0093-05

doi:10.3969/j.issn.1673-629X.2014.09.021

A Bi-slot Anti-collision Algorithm Based on Bits Coding Unit

ZHANG Xue-jun^{1,2}, MA Jun-fei³, LU You²

- (1. Key Laboratory of Broadband Wireless Communication and Sensor Network Technology of Ministry of Education, Nanjing University of Posts and Telecommunications, Nanjing 210003, China;
2. School of Electronic Science and Engineering, Nanjing University of Posts and Telecommunications, Nanjing 210003, China;
3. Nanjing Institute of Huawei Technologies Co., Ltd., Nanjing 210008, China)

Abstract: In RFID system, tags collisions occur while a reader attempts to identify multiple tags at the same time. Propose a bi-slot RFID anti-collision algorithm based on bits coding unit by setting a Bits Coding Unit (BCU) in every tags. In this algorithm, the bit encoding unit is used to recode the collision bit information, so that the reader which does not need to traverse queries can identify the collision bit information of tags once, thereby reducing the number of queries and the identification delay, making the tags can be identified quickly. Analysis of performance and the results of simulation show that the proposed algorithm improves the identification efficiency under the condition of a lot of tags.

Key words: radio frequency identification; anti-collision algorithm; Manchester code; bits coding units

0 引言

射频识别(Radio Frequency Identification, RFID)^[1]技术,作为快速、实时、准确采集与处理信息的高新技术和信息标准化的基础,其在物流、跟踪、定位等领域已得到广泛的应用,被公认为是 21 世纪十大重要技术之一。其中,用于解决阅读器作用范围内的多个标签识别问题的防碰撞算法是该领域研究的热点之一。

典型的 RFID 系统主要包括电子标签(Tag)(也称应答器)、阅读器(Reader)和中间件及应用软件等三个部分。标签贴附在被识别的物体上,阅读器根据所采用的结构、技术将具有读或写/读的功能,阅读器利用天线所构建的特定频率的电磁波为系统提供通信所需要的全部或部分能量,电子标签利用电磁场的能量驱动标签内部接收电路,然后标签将内部的数据信息输出,此时阅读器根据系统所规定的通信协议接收信息,

收稿日期:2013-11-10

修回日期:2014-02-16

网络出版时间:2014-07-17

基金项目:国家自然科学基金资助项目(60973140,61001077,61170276);南京邮电大学自然科学基金项目(NY211076)

作者简介:张学军(1969-),男,江苏南通人,教授,硕导,博士,研究方向为无线射频识别技术、通信网络的性能分析、流量控制、QoS 理论与技术等;马军飞(1988-),男,硕士,研究方向为无线射频识别技术。

网络出版地址: <http://www.cnki.net/kcms/detail/61.1450.TP.20140717.1229.028.html>

由中间件根据内部设定的指令处理信息数据,并将所需要的信息传送给阅读器,从而实现无接触、无人工干预的智能通信。

在 RFID 系统中,当多个标签同时响应阅读器时会引起标签碰撞,降低了识别的效率。为了解决这个问题,防碰撞算法使得阅读器阅读范围内的标签能够按照一定的规则响应阅读器,从而大大提高识别的效率。

文中提出了一种基于标签位编码单元的双时隙 RFID 防碰撞算法 (Bi-Slot anti-collision algorithm based on Bits Code Unit, BS-BCU)。该算法应用位编码单元(Bit Code Unit)对标签识别码进行编码,针对编码后的标签识别码能够一次性识别,从而减少标签与阅读器之间的通信次数,增加系统的吞吐率。文中通过数学分析,准确地描述了 BS-BCU 算法识别标签所需的时隙数。仿真结果表明,在大量标签存在的情况下,BS-BCU 算法具有较高的识别速度和识别效率。

1 相关工作

RFID 常见的防碰撞算法主要分为两大类:基于 ALOHA 的概率型防碰撞算法^[2-5]和基于树的确定型防碰撞算法^[6-8]。

1.1 ALOHA 算法

ALOHA 算法^[9]是一种信号随机接入的方法,采用电子标签控制方式。当标签进入阅读器的作用范围内,就自动向阅读器发送自身的序列号,随即与阅读器开始通信。在一个电子标签发送数据的过程当中,如果其他的电子标签也在发送数据,那么发送的信号重叠引起碰撞。

阅读器一旦检测到碰撞产生,就会发送命令让其中一个电子标签暂停发送数据,随机等待一段时间以后再重新发送数据。由于每个数据帧的发送时间只是重复发送时间的一小部分,以致在两个数据帧之间产生相当长的间歇,该算法可能导致某一个标签一段时间内无法识别,即存在“标签饥饿”的现象。

1.2 二进制搜索算法

二进制搜索算法(Binary Search algorithm, BS)^[10],即电子标签的序列号可以构成一棵完全二叉树,在阅读器作用范围内同步向阅读器发送信号的标签的序列号也构成一棵二叉树,阅读器根据信号冲突的情况反复对完全二叉树的分枝进行筛剪,最终找出这棵二叉树。在寻找的过程当中逐一确定了作用区域内响应的电子标签,同时也完成了它们与阅读器之间的信息交换。

与 ALOHA 法相比,二进制搜索法识别率较高,随着识别区域内标签数量的增加,效率会保持并逼近在

50%,并且该算法不存在错误判断的问题,但算法实现的时延长。实现二进制搜索算法的前提是系统必须能够对碰撞按位进行识别,所以必须选择相对复杂的信道编码(一般采用 Manchester 编码)。

1.3 时隙二进制树算法

时隙二进制树算法(Slotted Binary Tree algorithm, SBT)^[11],即当阅读器传输查询信息给标签时,通过随机数的选择(0 或 1)将响应的标签分为两组,即当阅读器在第 i 个时隙发送查询命令,则选择随机数为 0 的标签在 $i+1$ 个时隙响应,如果有标签识别或者该时隙为空闲时隙,则选择随机数为 1 的标签在第 $i+2$ 个时隙响应。如果 $i+1$ 时隙中发生碰撞,则标签再随机地选择随机数进行分组,直到有标签识别或者该时隙空闲。时隙二进制树算法通过随机数的选择避免了二进制树搜索算法中对标签标识码的依赖,在一定程度上减少了空闲时隙的发生,从而提高了系统的识别效率。

2 基于位编码单元的双时隙防碰撞算法

2.1 位编码单元(BCU)

在 BS-BCU 算法中,标签内部设有 1 个位编码单元(BCU)^[12],可以将标签的 ID 信息经过编码后发送给阅读器。当标签的识别码为 00、01、10、11 时,位编码单元编码后发送的信息为 1、01、001、0001。当阅读器接收到标签编码后的信息时就能够根据碰撞位的信息识别当前范围内标签的信息。例如,当标签 ID 信息为 01 和 10 时,标签经过编码单元发送 01 和 001,此时阅读器接收到的信息为 0x1。当标签信息为 10 和 11 时,分别发送 001 和 0001,阅读器接收到的信息为 00x1。通过利用编码单元,在阅读范围内的标签的任意两位信息都能够识别,即不存在碰撞查询,从而提高识别效率。

2.2 BS-BCU 算法步骤

BS-BCU 算法是结合标签识别码信息的特征,通过检测标签响应的信息检测出碰撞位发生的位置,利用标签位编码单元对碰撞位进行重新编码。如果发生碰撞就让标签通过内部的位编码单元将发生碰撞位的信息进行编码,然后发送给阅读器。在 BS-BCU 算法根据碰撞位的个数决定分配时隙的个数时,如果碰撞位的个数大于 2,第 1 位碰撞位为 0 的标签在第 1 个时隙响应,第 1 位碰撞位为 1 的标签在第 2 个时隙响应,从而提高识别标签的速度。根据标签碰撞位的个数分为不同的状态。

(1) 只有 1 位碰撞位,阅读器可以直接识别两个标签。

(2) 有 2 位发生碰撞,第 1 位碰撞位分别置 0 和

1,并将识别的信息压入栈中。在这种状态下,碰撞位的信息不需要经过BCU进行编码。

(3)其他情况,第1位碰撞位为0的标签在时隙1中响应阅读器,第1位碰撞位为1的标签在第2个时隙响应阅读器。后面紧接着的2位碰撞位经过BCU编码单元编码后发送给阅读器。

BS-BCU算法步骤为:

步骤1:阅读器发送查询前缀 $prefix = \varepsilon$;

步骤2:与查询前缀相匹配的标签响应,如果发生碰撞,跳转到步骤3;如果只有一个标签响应识别该标签,跳转到步骤4执行;

步骤3:如果碰撞位的个数大于2,则阅读器发送Query命令,第1位碰撞位为0的标签在第1个时隙响应,碰撞位为1的标签在第2个时隙响应,标签通过BCU编码单元只响应第2、3位碰撞信息,阅读器经过

解码后将收到的信息进栈;如果碰撞位个数等于2,则将第1位碰撞位信息置1和0分别进栈;如果碰撞位只有1个,则同时识别两个标签,跳转到步骤4执行;

步骤4:栈中的内容不为空,将信息出栈,阅读器根据栈中的信息构造前缀prefix查询命令查询,跳转到步骤2执行;如果栈为空,则识别过程结束。

图1利用伪代码描述了BS-BCU算法的执行过程,代码第1~24行是阅读器识别标签的操作过程,首先判断是否有标签响应,然后根据碰撞位个数信息发送不同的命令给标签,同时将信息进栈,构造下一次查询前缀。代码第25~37行为标签根据收到不同的查询命令执行的操作,如果碰撞位的个数大于3,就将第2、3位碰撞位信息经过编码单元编码后发送给阅读器,否则直接发送标签识别码信息给阅读器。

BS-BCU protocol: Reader operation	BS-BCU protocol: Tag operation
1. send Query command with prefix	24. Receive Query command
2. function identifyProcess:	25. if tag ID match prefix
3. Build prefix and send with Query command	26. if collision bits>3
4. if Tag response	27. if first bit is 0
5. if only one tag response	28. coding ID and response in first slot
6. identify tag	29. else if first bit is 1
7. else if detect only one bit collision	30. coding ID and response in second slot
8. identify two tags	31. end if
9. else if two bit collision	32. else
10. set first bit is 0 and 1 push into stack	33. send ID to reader
11. else	34. end if
12. if Response in the first time slot	35. end if
13. set first collision bit is 0	
14. else if Response in the second time slot	
15. set first collision bit is 1	
16. end if	
17. analysis the coding information, and push into stack	
18. end if	
19. if stack no empty	
20. function identifyProcess	
21. end if	
22. end if	
23. end function	

图1 BS-BCU算法伪代码

2.3 算法举例

假设有8个待识别标签的ID分别为a:010011,b:001001,c:010100,d:010010,e:101101,f:111011,g:101011,h:111001。当阅读器发送查询前缀 ε 查询命令后,所有的标签响应阅读器,此时阅读器收到的信息为xxxxxx,阅读器根据收到的信息发送查询命令,使第1位碰撞位为“0”的标签在第1个时隙响应,第2、3位碰撞位经过标签内部的编码单元重新编码后发送信息。在该例子中标签a:010011,b:001001,c:010110,d:010010在第1个时隙中响应,且其后2位分别是10,01,10,10,则标签分别发送001,01,001,001,此时阅读器接收到的信息为0x1,则阅读器经过分析可以判断出在阅读器范围内存在信息为01和10的标签。由于这些标签在第1个时隙中响应,则在前面加上“0”后将其压栈;在第2个时隙中,第1位碰撞位为

“1”的标签按照上述规则发送信息,即为01,0001,01,0001,阅读器接收到的信息为0x01,阅读器经分析解码并在前端加1后得到101和111,将其压栈。阅读器以001作为查询前缀,由于与001相匹配的标签只有标签b,则阅读器能够识别该标签。阅读器以010作为查询前缀时,标签a,c,d响应,阅读器收到的信息为010xxx,碰撞位有3位,在第一个时隙中标签a,d响应阅读器,阅读器收到信息为00x1,经过解码后阅读器可以判定信息为10和11,故标签a,d同时被识别;在第2个时隙中只有标签c响应,标签c被识别;阅读器以101查询时,标签e,g响应,阅读器收到的信息为xx1,同样由于只有两位信息发生碰撞,第一位碰撞位置0和1并将信息保存在栈中,阅读器以1010查询时,识别标签g;阅读器以1011查询时识别标签e;同理阅读器可以识别标签f,h。识别过程如表1所示。

表 1 算法举例识别过程

序号	阅读器 查询命令	阅读器收 到的信息	碰撞位 个数	时隙	阅读器接收信息 (BCU 编码信息)	响应标签	碰撞信息	堆栈分 组码状态
1	ε	xxxxxx	6	1	0x1	a,b,c,d	碰撞	{001,010}
				2	0x01	e,f,g,h	碰撞	{101,111,001,010}
2	001	001001	0	Φ	Φ	b	识别 b	{101,111,010}
3	010	010xxx	3	1	00x1	a,d	识别 a,d	{101,111}
				2	001	c	识别 c	{101,111}
4	101	101xx1	2	Φ	Φ	e,g	碰撞	{1010,1011,111}
5	1010	101011	0	Φ	Φ	e	识别 e	{1011,111}
6	1011	101101	0	Φ	Φ	g	识别 g	{111}
7	111	1110x1	1	Φ	Φ	f,h	识别 f,h	Φ

3 算法分析

在 QT 算法中,标签识别可以被描述成一棵二叉树,因此所有的中间节点都是碰撞节点,所有的叶节点都是空闲节点或识别节点。假设要识别 k 个标签, L 表示查询树的深度,则 QT 算法需要的时隙总数^[13-15]为

$$T_{QT}(k) = 2C_{QT}(k) + 1 \quad (1)$$

令 $I_{QT}(k)$, $R_{QT}(k)$ 和 $C_{QT}(k)$ 分别表示空闲时隙、识别时隙和碰撞时隙,在 k 个标签存在的情况下,第 L 层所产生的碰撞时隙为

$$C_{QT}(k) = \sum_{l=0}^L C_{QT}(l) \quad (2)$$

令 $p(L) = 1 - 2^{-L}$ 表示第 L 层中的任意一个时隙标签都没有发送信息的概率,则第 L 层存在空闲时隙的期望为

$$I_{QT}(k) = 2^L p(L)^k \quad (3)$$

只有一个标签响应的时隙的个数为

$$R_{QT}(k) = kp(L)^{k-1} \quad (4)$$

因此,

$$C_{QT}(k) = 2^k - I_{QT}(k) - R_{QT}(k) = 2^k \{1 - p(L)^k - kp(L)^{k-1}\} \quad (5)$$

由式(1)、式(2)、式(5)可以得出,对于任意的 k , 存在

$$T_{QT}(k) = 1 + 2 \sum_{k=0}^{\infty} \{2^k - p(L)^{n-1} (2^k + k - 1)\} \quad (6)$$

其中, $p(L) = 1 - 2^{-L}$ 。

碰撞时隙的个数为

$$C_{QT}(k) = \frac{1}{2} (T_{QT}(k) - 1) \quad (7)$$

空闲时隙的个数为

$$Z_{QT}(k) = \frac{1}{2} T_{QT}(k) - k + \frac{1}{2} \quad (8)$$

由于在 BS-BCU 算法中利用双时隙的特点将碰撞位为 0 或 1 的标签分别在两个时隙中响应,同时利用 BCU 编码单元使得阅读器能够同时识别 2 位碰撞位信息,阅读器可以继续进行搜索查询,不需要对碰撞节点重复查询,使得查询树的深度减少为原来的 1/3。由于 BS-BCU 算法中 BCU 编码单元的特殊性质,在一个时隙中识别的标签的平均数目约为 2.5 个,考虑空闲时隙存在的情况下,该算法所需的总时隙的个数约为

$$T'_{BS-BCU}(k) = T_{QT}(k) \Big|_{L=L/3} = \frac{1}{2^3} (Z_{QT}(k) + C_{QT}(k)) + \frac{3}{5} k \quad (9)$$

由于阅读器在识别过程中能够根据编码约束进行判决,没有空闲时隙的产生,所以要识别 k 个标签所需要的时隙为

$$T_{BS-BCU}(k) = T'_{BS-BCU}(k) - Z'_{BS-BCU}(k) \approx 1.38k \quad (10)$$

其中, $Z'_{BS-BCU}(k) = \frac{1}{2} T'_{BS-BCU}(k) - k + \frac{1}{2}$ 。

4 仿真分析

Matlab 是 MathWorks 公司发布的主要用于数值分析、交互式程序设计,并具有强大的可视化功能的软件。下面利用 Matlab 软件对 QT 算法、DBS 算法和 BS-BCU 算法在识别相同数目标签的情况下所需要的时隙总数进行仿真,如图 2 所示。从图中可以看出,在阅读器范围内存在 100 个标签时,DBS 算法大约需要 630 个时隙,QT 算法需要约 300 个时隙,BS-BCU 利用特殊的编码方式进行编码,避免了识别过程中的空闲时隙,在识别 100 个标签时仅需要约 130 个时隙,系统

的识别速度大大提高了。

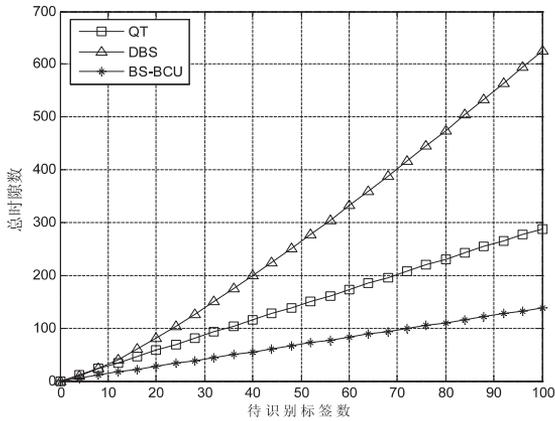


图2 QT算法、DBS算法、BS-BCU算法总时隙比较

由于RFID系统中标签多用无源的,所以标签能够接收到的能量是有限的,因此通信量复杂度是衡量RFID防碰撞算法性能的一个关键指标,因为它决定了系统的功耗^[16]。假设标签识别码信息的长度ID为64位(ISO(ISO-18000-6B)),仿真图如图3所示。可以看出QT算法的通信量随着标签数目的增加迅速增加,BS-BCU算法虽然利用BCU使系统变得稍微复杂,但它的系统通信量约是DBS算法的1/4,QT算法的1/2。

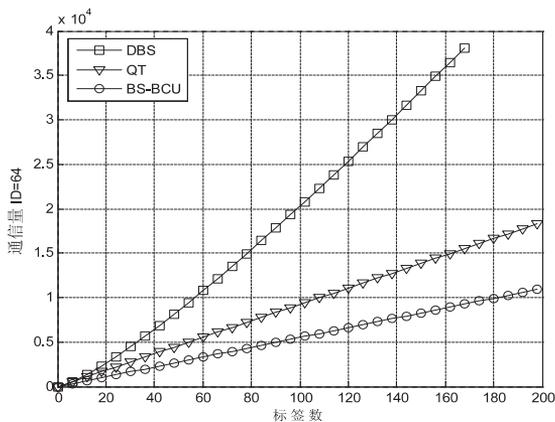


图3 QT算法、DBS算法、BS-BCU算法系统通信量比较

系统吞吐率是考察算法性能的重要指标,定义为:成功识别标签时隙的个数与总消耗时隙的个数比。QT算法、DBS算法以及BS-BCU算法的系统吞吐率比较如图4所示。从图中可以看出BT算法随着标签数目的增加吞吐率呈逐渐下降的趋势,这是由于标签数目过多,则查询树深度就会随之增加,导致系统性能下降;DBS算法保持性能比较稳定,达到0.35左右;BS-BCU算法系统吞吐率可以达到0.72左右,且性能比较稳定。

5 结束语

文中提出的BS-BCU算法中,利用标签中的BCU

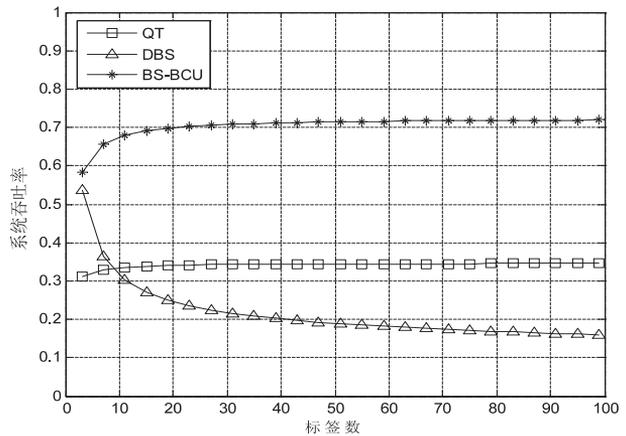


图4 QT算法、DBS算法、BS-BCU算法系统吞吐率比较

算法系统吞吐率比较

对碰撞位进行重新编码,并根据碰撞位信息将发生碰撞的标签在两个时隙中响应,使阅读器能够准确地识别最多3位碰撞位信息,从而能够快速向前搜索标签信息,使得所用的总时隙数目大大减少,并且在标签ID长度一定的情况下,随着标签数目的增加,BS-BCU算法识别效率保持稳定。

参考文献:

- [1] Finkenzeller K. 射频识别(RFID)技术[M]. 陈大才,译. 第3版. 北京:电子工业出版社,2005.
- [2] 张学军,王绪海,蔡文琦. 基于分组码的改进型防碰撞算法研究[J]. 计算机应用研究,2012,29(11):4265-4268.
- [3] 张学军,马军飞,陈彦君. 增强型Q参数混合防碰撞算法[J]. 计算机技术与发展,2013,23(8):47-51.
- [4] 单剑锋,谢建兵,庄琴清. 基于分组的动态帧时隙ALOHA防碰撞算法研究[J]. 计算机技术与发展,2011,21(11):39-41.
- [5] Shakiba M,Zavvari A,Sundararajan E. Fitted dynamic framed slotted ALOHA anti-collision algorithm in RFID systems [C]//Proc of 5th international conference on IT and multimedia. Malaysia; [s. n.],2011:1-6.
- [6] Hush D R,Wood C. Analysis of tree algorithms for RFID arbitration[C]//Proc of IEEE international symposium on information theory. Cambridge, MA:IEEE,1998.
- [7] 王春华,刘迟时,徐浩,等. 一种改进的基于二叉树的防碰撞算法[J]. 湖南大学学报(自然科学版),2013,40(8):97-101.
- [8] 王雪,钱志鸿,胡正超,等. 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.
- [9] Chen Ying,Zhang Fuhong. Study on anti-collision Q algorithm for UHF RFID[C]//Proc of international conference on communications and mobile computing. [s. l.]:[s. n.],2010:168-170.
- [10] Lai Yuancheng,Lin Chih-Chung. A blocking RFID anti-collision protocol for quick tag identification[C]//Proc of interna-

到一定程度的减少。

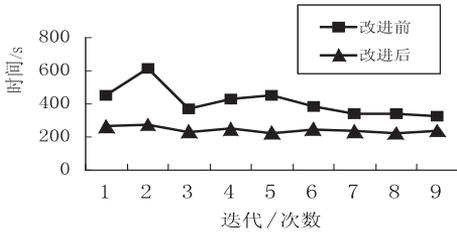


图 4 每次迭代对比图

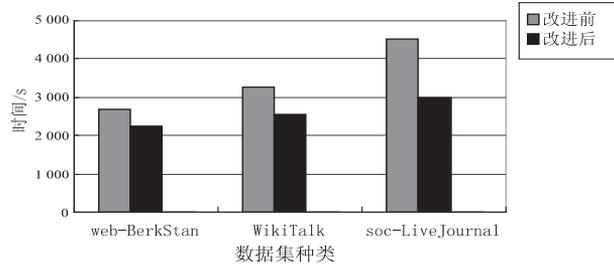


图 5 三组数据 8 次迭代时间对比图

4 结束语

文中首先总结了 Hadoop 处理迭代应用存在的问题,并叙述了目前的解决方案。然后在 HaLoop 的基础上提出了 Map 端存储策略,通过保证数据的本地性以及 Map 端的扩展,有效地减少了每次迭代运行 MapReduce 的次数或步骤,从而减少了运行时间。最后通过实验验证了改进后的方案比之前的方法在性能上有一定的提高,达到了预期的结果。

参考文献:

[1] 覃雄派,王会举,杜小勇,等. 大数据分析—RDBMS 与 MapReduce 的竞争与共生[J]. 软件学报,2012,23(1):32-45.

[2] Dean J, Ghemawat S. MapReduce:simplified data processing on large clusters[J]. Communications of the ACM,2008,51(1):107-113.

[3] White T. Hadoop 权威指南[M]. 曾大聃,周傲英,译. 北京:清华大学出版社,2010.

[4] 江务学,张璟,王志明,等. MapReduce 并行编程架构模型研究[J]. 微电子学与计算机,2010,28(6):168-170.

[5] 于戈,谷峪,鲍玉斌,等. 云计算环境下的大规模图数据处理技术[J]. 计算机学报,2011,34(10):1753-1767.

[6] 饶君,张仁波,东昱晓,等. 基于 MapReduce 的大规模图挖掘并行计算模型[J]. 应用科技,2012,39(3):56-60.

[7] Bu Yingyi,Howe B,Balazinska M,et al. HaLoop:efficient iterative data processing on large clusters[J]. Proceedings of the VLDB Endowment,2010,3(1-2):285-296.

[8] Elnikety E,Elsayed T,Ramadan H E. iHadoop:asynchronous iterations for MapReduce [C]//Proc of IEEE third international conference on cloud computing technology and science. Athens:IEEE,2011:81-90.

[9] Zhang Yanfeng, Gao Qixin,Gao Lixin,et al. iMapReduce:a distributed computing framework for iterative computation[J]. Journal of Grid Computing,2012,10(1):47-68.

[10] Malewicz G,Austern M H,Bik A J C,et al. Pregel:a system for large-scale graph processing [C]//Proceedings of the 2010 ACM SIGMOD international conference on management of data. [s.l.]:ACM,2010:135-146.

[11] 任年海. 一个有效的并行模型—BSP 并行模型[J]. 计算机与现代化,2006(3):34-36.

[12] Ghemawat S,Gobioff H,Leung S T. The Google file system [J]. ACM SIGOPS Operating Systems Review,2003,37(5):29-43.

[13] Kamvar S,Haveliwala T,Golub G. Adaptive methods for the computation of PageRank[J]. Linear Algebra and Its Applications,2004,386:51-65.

[14] 李远方,邓世昆,闻玉彪,等. Hadoop-MapReduce 下的 PageRank 矩阵分块算法[J]. 计算机技术与发展,2011,21(8):6-9.

[15] 王晓军,孙惠. 基于 MapReduce 的多路连接优化方法研究[J]. 计算机技术与发展,2013,23(6):59-62.

(上接第 97 页)

tional conference on wireless and optical communications networks. Cairo:IEEE,2009:1-6.

[11] Kim Y H,Kim S S,Lee S J,et al. Improved 4-ary query tree algorithm for anti-collision in RFID system [C]//Proc of international conference on advanced information networking and applications. Bradford:IEEE,2009:699-704.

[12] Kim Y H,Kim S S,Ahn K. A rapid tag identification method with two slots in RFID systems [C]//Proc of eighth IEEE international symposium on network computing and applications. Cambridge:IEEE,2009:292-295.

[13] Myung J,Lee W J,Srivastava J. Adaptive binary splitting for

efficient RFID tag anti-collision [J]. IEEE Communications Letters,2006,10(3):144-146.

[14] Myung J,Lee W. Adaptive binary splitting:a RFID tag collision arbitration protocol for tag identification [J]. Mobile Networks and Applications,2006,5(11):711-722.

[15] Kim S S,Kim Y H,Ahn K. An enhanced slotted binary tree algorithm with intelligent separation in RFID systems [C]//Proc of IEEE symposium on computers and communications. Sousse:IEEE,2009:237-242.

[16] 张学军,王娟,王锁萍. 基于标签识别码分组的连续识别防碰撞算法研究[J]. 电子与信息学报,2011,33(5):1159-1165.

基于位编码单元的双时隙防碰撞算法

作者: 张学军, 马军飞, 鲁友, ZHANG Xue-jun, MA Jun-fei, LU You

作者单位: 张学军, ZHANG Xue-jun(南京邮电大学 宽带无线通信与传感网技术教育部重点实验室, 江苏 南京 210003; 南京邮电大学 电子科学与工程学院, 江苏 南京 210003), 马军飞, MA Jun-fei(华为技术有限公司南京研究所, 江苏 南京, 210008), 鲁友, LU You(南京邮电大学 电子科学与工程学院, 江苏 南京, 210003)

刊名: 计算机技术与发展 

英文刊名: Computer Technology and Development

年, 卷(期): 2014(9)

本文链接: http://d.wanfangdata.com.cn/Periodical_wjfz201409021.aspx