

新的基于分组处理的射频识别标签防碰撞算法

刘迟时*, 王春华, 伏奎

(湖南大学 信息科学与工程学院, 长沙 410082)

(* 通信作者电子邮箱 iamliucs@sina.com)

摘要:针对现有几种基于二叉树的防碰撞算法识别时延较长、数据传输量大的情况,提出一种新颖的防碰撞算法。算法采用分组策略,读写器按顺序依次识别每个分组,减少了每次应答标签的数量和碰撞发生的概率;此外,把标签ID分为两段识别,第一段为前7位,剩余部分为第二段,标签ID分段处理能减少冗余数据的传输。仿真结果表明,该算法相比其他几种算法,查询次数少,数据传输量仅为动态二叉树搜索(DBS)算法的1/6,识别效率有较大幅度提高。

关键词:射频识别;防碰撞;二叉树;ALOHA;标签识别

中图分类号: TP391.45 **文献标志码:** A

New group-based processing tag anti-collision algorithm for RFID system

LIU Chishi*, WANG Chunhua, FU Kui

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

Abstract: Concerning long length of identification time and large amounts of data transmission in some binary search schemes, a new anti-collision algorithm was proposed. The algorithm adopted grouping strategy, and all tags in each subset were identified by reader in order. The grouping strategy could reduce the quantity of responding tags in each query and the probability of collision. On the other hand, tag ID was divided into two segments, the first seven bits were the first part and the remaining bits were the second part. Therefore, the transmission of redundant data could be reduced. The simulation results show that, compared with several other algorithms, the proposed algorithm has fewer searching times, and the data transmission is only one-sixth of Dynamic Binary Search Tree (DBS) algorithm. Thus, the identification efficiency of the proposed algorithm is significantly improved.

Key words: Radio Frequency Identification (RFID); anti-collision; binary tree; ALOHA; tag identification

0 引言

近年来,射频识别(Radio Frequency Identification, RFID)逐渐成为发展最快的热门技术之一,广泛应用于电子门禁、交通管理、供应链管理、商业自动化等领域^[1]。RFID是一种非接触的自动识别技术,由读写器和拥有唯一身份识别码(Identification, ID)的标签组成。当读写器的识别区域内有多个标签同时传送数据时,产生信道争用,信号相互干扰,读写器不能正确识别标签,这就是标签碰撞问题^[2]。标签碰撞将严重影响RFID系统标签识别效率,并且额外增加标签和读写器的能量消耗。因此,高效的防碰撞算法对RFID系统至关重要。

目前,RFID标签防碰撞算法主要分为两类^[3-4]:一类是基于二叉树的确定性算法,另一类是基于ALOHA的随机性算法。基于ALOHA的算法的思想是:标签随机选择一个时隙传送数据来减小碰撞发生的概率。基于ALOHA算法一般包括纯ALOHA算法、帧时隙ALOHA(Frame Slotted ALOHA, FSA)算法^[5]、动态帧时隙ALOHA(Dynamic Frame Slotted ALOHA, DFSA)算法^[6]和一些其他的改进算法^[7]。基于二叉树的算法思想是:根据标签ID把标签分为两个子集,直到分解成只含单个标签的子集,读写器就能识别此标签。基于二

叉树的算法包括查询树(Query Tree, QT)算法^[8]、二叉树搜索(Binary Search Tree, BS)算法^[9]、动态二叉树搜索(Dynamic Binary Search Tree, DBS)算法^[10]、后退式索引的二叉树搜索算法^[11],以及文献[12]提出的一种基于标签信息分组的防碰撞算法和一些其他的改进算法^[13-14]。基于ALOHA的算法相对简单,容易实现,在标签数少的情况下能快速识别完标签。但是,随着标签数量的增加,基于ALOHA的算法性能急剧下降,并且还存在着“标签饥饿”的问题^[15]。基于树的算法不存在“标签饥饿”,但是识别时延长。

本文提出一种新颖的基于分组处理的防碰撞(Group-based Processing Anti-collision, GPA)算法。GPA算法对标签和标签ID均进行分段处理,减少了标签碰撞和通信数据量,解决了二叉树算法中每次从根节点识别时延长的时间。仿真结果表明GPA算法传输的数据量小、查询次数少,具有较高的识别速率。

1 GPA 算法

1.1 算法思想

所有标签进行分组,每组中包含较少数量的标签。算法开始时,标签产生一个0~t范围内的随机数并存入存储器TR,所有TR中数值相等的标签被分入一组。读写器根据所

收稿日期:2012-08-09;修回日期:2012-09-23。

作者简介: 刘迟时(1988-),男,湖南湘潭人,硕士研究生,主要研究方向:RFID; 王春华(1963-),男,湖南永州人,教授,博士,主要研究方向:RFID系统、混沌保密通信; 伏奎(1987-),男,湖南岳阳人,硕士研究生,主要研究方向:射频接收机。

生成随机数数值的大小依次识别每个分组中的标签,分组值小的先被识别。TR 为 0 的分组最先被读写器选中并被识别。当一个分组中的标签全部被识别完后,比当前分组值大 1 的标签将被选中并等待应答。重复以上步骤,读写器便可一一识别所有分组中的标签。

标签 ID 划分成两段,读写器先识别第一段,再识别第二段,每次查询标签只发送其中一段的数据。假设标签 ID 长度为 L 位,第一段长度为 K 位,那么第二段长度为 $L - K$ 位,标签分段处理如图 1 所示。

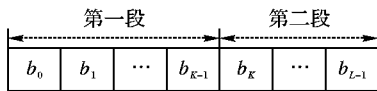


图1 标签ID分段处理

1.2 相关命令

1) REQUEST(*parameter*, *p*)。 *parameter* 为查询前缀, *p* 为查询前缀位置标识位。当查询命令前缀位于第一段时, $p = 0$; 当查询命令前缀位于第二段时, $p = 1$ 。标签接收到读写器发送的 REQUEST 命令后,活动的标签根据 *p* 的值去匹配查询前缀并决定应答其中一段 ID。

2) Rand-generate(*t*)。标签接收到此命令后,产生 $0 \sim t$ 的随机数,并存储于 TR 中。

3) Select(*parameter*)。 *parameter* 为标签 ID, ID 与 *parameter* 相同的标签将被选中。

4) Read-data。读写器读出被选中标签的数据。

5) Unselect(*parameter*)。 *parameter* 为标签的 ID。已成功读写的标签将被设置进入“休眠”状态。除非将处于“休眠”状态的标签移出读写区域,然后再移进,否则将不再应答读写器的任何命令。

6) Group-Activate(*parameter*)。标签分组选择命令, *parameter* 为下一个被选中分组的组号。TR 中与 *parameter* 相等的分组被选中,被选中分组的标签把标识位 *g* 设为 1,等待下一次查询并应答。算法开始时或一个组中的标签全部被成功识别后,读写器发送此命令。

1.3 算法代码

GPA 算法的伪代码如下所示,算法 1 和算法 2 分别表示读写器和标签的操作。读写器有一个堆栈 *Q* 用来存储发送的查询命令;每个标签都有一个存储器 TR,用来存储生成的随机数,也即分组号;每个标签都有一个表示该分组是否被选中的标识位 *g*, *g* 初始为 0,当 $g = 1$ 时,表示该分组被选中。

算法开始时读写器进行初始化操作,发送 Rand-generate(*t*) 命令给标签,所有标签产生 $0 \sim t$ 的随机数作为自身的分组号。标签分组完后,读写器选择第 0 分组进行识别,如算法 1 中 2)~4) 行。读写器接收到标签发送的数据,检测碰撞位并产生一个新的查询命令。如果标签返回的第一段数据有碰撞位,则 $p = 0$,继续查询标签 ID 的第一段;否则 *p* 的值设为 1,开始识别标签 ID 的第二段。如果接收到的数据无碰撞并且 $p = 1$,则成功识别一个标签,如算法 1 中 16)~20) 行所示。

相应地,应答的标签接收到 Request 命令后,检测 *p* 的值,如果 $p = 0$,则发送 ID 的第一段数据给读写器,如果 $p = 1$,则发送 ID 的第二段数据,如算法 2 中 11)~15) 行所示。

读写器成功识别一个标签后,如果堆栈 *Q* 不为空,则进行两次出栈操作,返回至倒数第二次发送的查询命令,如算法 1 中 21)~25) 行所示,这样进行两次出栈操作能减少查询过

程中产生的空时隙。当 *Q* 为空且无标签应答或只有一个标签应答则表示这个分组中的标签全部被成功识别。读写器识别完一个组中的标签后,发送 Group-Activate 命令,选中下一个分组,被选中的分组等待应答读写器下一轮的查询。

Algorithm 1: 读写器伪代码。

```

1) Initialize;
2) Rand-generate(t);
3) Group-Activate(0);
4) Request(q, 0);
5) Push(q, 0); //查询命令(q, 0) 入栈
6) while(Not all tags are identified) do
7)   Receive tag response and detect a collision;
8)   if (CN > 0) then //CN 为碰撞位数;
9)     Get the highest collision bit Dz;
10)    Request(q, z);
11)    Push(q, z); //命令(q, z) 入栈
12)   else if (CN == 0) then
13)     if (P == 0) then
14)       Request(q, 1);
15)       Push(q, 1); //命令(q, 1) 入栈
16)     else if (P == 1) then
17)       Select(EPC);
18)       Read_data(EPC);
19)       Unselect(EPC);
20)     end if
21)     if (Q == NULL) then
22)       Pop(Q); //返回至上一次发送的命令
23)     if (Q != NULL) then
24)       Pop(Q); //返回至上一次发送的命令
25)     end if
26)     else if (Q == NULL)
27)       Group-Activate(tr++); //选择下一个分组
28)     end if
29)   end if
30) end while
31) end while

```

Algorithm 2: 标签伪代码。

```

1) Initialize;
2) switch(order)
3)   case( Rand-generate(t) )
4)     TC = Rand(t); //产生  $0 \sim t$  的随机数
5)   case( Group-Activate(tc) )
6)     if (tc == TC) then
7)       g = true;
8)     end if
9)   case( Request(q, p) )
10)    if (g == true && ID matches q) then
11)      if (p == 0) then
12)        Response the bits in first segment;
13)      else if (p == 1)
14)        Response the bits in second segment;
15)      end if
16)    end if
17)   case( Select(parameter) )
18)     The tag whose EPC is the same as parameter is selected by reader;
19)   case( Unselect(parameter) )
20)     The tag whose EPC is the same as parameter will be in dumb mode;

```

2 算法分析与仿真

2.1 算法分析

假若读写器范围内的标签数为 *N*, 标签产生随机数的范

围为 $[0, t)$, 即标签分组数为 t 。设 $T(i)$ 、 $S(i)$ 分别为识别第 i 个分组中所有标签所消耗的时隙数和所传输的总数据($0 \leq i < t$), 那么总查询次数为

$$T(N) = T(0) + T(1) + \dots + T(t-2) + T(t-1) = \sum_{i=0}^{t-1} T(i) \quad (1)$$

总的传输数据为

$$S(N) = S(0) + S(1) + \dots + S(t-2) + S(t-1) = \sum_{i=0}^{t-1} S(i) \quad (2)$$

若标签数量足够大, 标签 ID 均匀分布。标签产生的随机数服从均匀分布, 即每个标签产生 $[0, t)$ 区间任意一个数的概率为 $1/t$, 那么每个分组中的标签数的期望值为 N/t 。近似有 $T(0) = T(1) = \dots = T(t-2) = T(t-1)$, $T(N)$ 可以表示为

$$T(N) = tT(0) = \dots = tT(i) = \dots = tT(t-1) \quad (3)$$

如果每个分组所消耗的时隙数最小、数据传输量最少, 则算法整体性能能达到最佳。下面分别求解标签 ID 分段第一段的最优长度 K 和标签的最优分组数 t , 使识别单个分组中标签的效率最高。

2.2 分段长度 K 和分组数 t 的最优解

假定读写器范围内的标签数 N 的值足够小, 所有标签被划分入一个组; 分组中标签数量 N 的值由2逐渐增大到10; 标签 ID 第一段的位数为 K , K 的值由2增加到10, 另外 K 再取15作补充。随着分组中标签数和标签 ID 第一段长度的变化, 观察识别该组所有标签所需的查询次数和识别每个标签所传输的平均数据量。在 VC++ 6.0 环境下, 用 C 语言编写仿真程序, 标签 ID 为 96 位, 运行 3000 次取平均值。

图2表示在不同的标签数和不同的第一段长度 K 的情况下, 识别所有标签所需的查询次数。图2中, 当标签数量一定时, 总的查询次数几乎不随第一段的长度变化而变化。

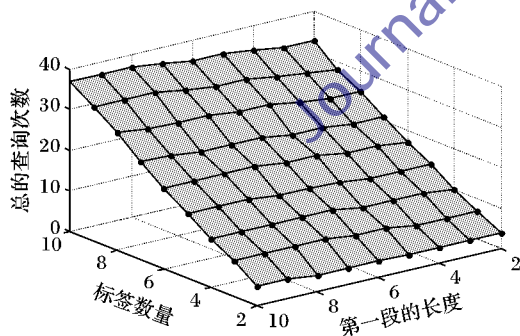


图2 标签数量与第一段长度变化下的查询次数

图3表示在不同的标签数和不同的第一段长度 K 的情况下, 识别每个标签所传输的平均数据量。从图3中可以看出, 标签数量一定的情况下, 第一段的长度存在一个值得使得平均每识别一个标签所传输的数据最少。本文中第一段长度 K 的值取为7, 当 $K=7$, 无论分组中标签数目如何变化, 几乎都可以使平均传输的数据量最少。

当第一段长度为7位时, 从图3中可以看出, 随着分组中标签数量的减少, 平均每识别一个标签所传输的数据逐渐减少, 因此分组中的标签数量越少越好。但如果平均每个分组中的标签数量太少, 则会引起空查询时隙数的急剧增加。综合考虑, 本文中每个分组中标签数量定为3。如果读写器范围内标

签总数为 N , 那么总的分组数为 $N/3$, 命令 Rand-generate(t) 中参数 t 的值即为 $N/3$ 。

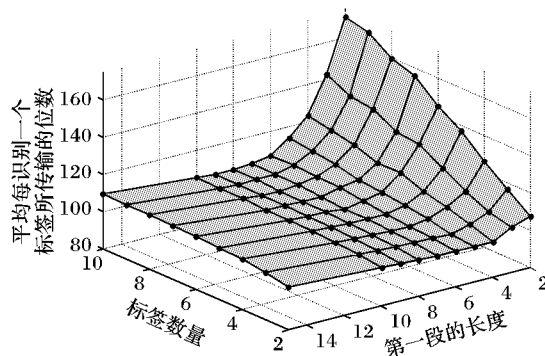


图3 标签数量与第一段长度变化下识别每个标签平均传输的数据

2.3 仿真结果

标签 ID 长度为 96 位, 标签数量从 50 增加到 1000; 为确保数据的精确性, 仿真程序连续运行 50 次取平均值。把本文提出的算法(GPA)与二叉树搜索算法(BS)、动态二叉树搜索算法(DBS)、动态帧时隙 ALOHA 算法(DFSA)以及文献[12]所提出的算法(在图表中标记为算法S)进行比较。

算法的时间复杂度比较如图4所示, 从中可以看出, BS 算法和 DBS 算法识别 N 个标签所需的查询次数最多, 它们有相同的查询次数; DFSA 和文献[12]提出的算法所需的查询次数次之, 相对较少; GPA 算法所需的查询次数随着标签数量的增加线性增加, 当标签数量为1000时, GPA 所需的查询次数约为3000。在这几种算法中, GPA 算法所需的查询次数最少。

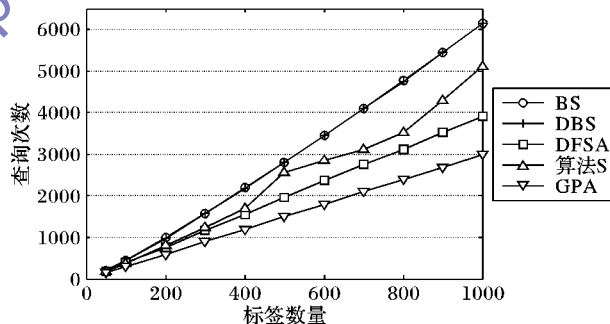


图4 几种算法查询次数随标签数量变化情况

由于 GPA 算法把标签 ID 进行了分段处理, 只有在识别完第一段之后标签才发送 ID 的第二段。所以, 在识别过程中, 极大地减少了标签传送的数据量。算法的通信复杂度仿真结果如图5所示, 可以看出 GPA 算法明显优于其他算法, 而且随着标签数量的增加, 优势更加明显。当标签数量为1000时, GPA 的通信数据量仅为 DBS 算法的 $1/6$, 为 DFSA 算法和文献[12]的算法的 $1/5$ 。

4 结语

本文提出了一种基于标签分组、标签 ID 分段识别处理的防碰撞算法 GPA。为使算法整体性能最佳, 求解了标签 ID 分段识别的最优分段长度和一定标签数量下的最优标签分组。对标签进行分组减少了碰撞数量, 减小了每次从根节点开始识别的识别时延, 大大缩短了标签的识别时间; 标签 ID 分段识别, 减少了冗余数据传输量。本文算法具有很高的读取效率, 适用于标签密集度高、标签信息数据位长的 RFID 应用场合。

(下转第 599 页)

较高的定位精度以及较高的鲁棒性。本文的后续研究工作将围绕以下内容展开:选择合适的高维数据相似性度量算法,选择合适的重采样算法,考虑算法的简洁易实现。

致谢:特别感谢德国图宾根大学发布的免费实验数据。

参考文献:

- [1] NI L M, LIU Y H, LAU Y C, *et al.* LANDMARC: indoor location sensing using active RFID [J]. *Wireless Networks*, 2004, 10(6): 701–710.
- [2] DELLAERT F, FOX D, BURGARD W, *et al.* Monte Carlo localization for mobile robots [C]// *Proceedings of the International Conference on Robotics and Automation*. Piscataway: IEEE, 1999: 1322–1328.
- [3] CHOI J S, LEE H, ENGELS D W, *et al.* Passive UHF RFID-based localization using detection of tag interference on smart shelf [J]. *IEEE Transactions on Systems, Man and Cybernetics: Part C: Applications and Reviews*, 2012, 42(2): 268–275.
- [4] BEKKALI A, SANSON H, MATSUMOTO M. RFID indoor positioning based on probabilistic RFID map and Kalman filtering [C]// *WiMOB 2007: Proceedings of the Third IEEE International Conference on Wireless and Mobile Computing, Networking and Communications*. Piscataway: IEEE, 2007:21.
- [5] VORST P, ZELL A. A comparison of similarity measures for localization with passive RFID fingerprints [C]// *ISR/ROBOTIK 2010: 41st International Symposium on Robotics and 2010 6th German Conference on Robotics*. Piscataway: IEEE, 2010:354–361.
- [6] VORST P, KOCH A, ZELL A. Efficient self-adjusting, similarity-based location fingerprinting with passive UHF RFID [C] // *2011 IEEE International Conference on RFID Technologies and Applications*. Piscataway: IEEE, 2011:160–168.
- [7] 张健翀, 黄以华. 基于射频识别 (RFID) 技术室内定位系统研

究[D]. 广州: 中山大学, 2010.

- [8] 小宝. 美西杰斐逊医疗中心计划用 RFID 定位医疗设备[EB/OL]. (2007–03–01) [2012–05–22]. http://www.rfidinfo.com.cn/Info/html/n4440_1.htm.
- [9] 刘瑜. 希思罗机场使用 RFID 定位地下设施[EB/OL]. (2009–10–20) [2012–05–26]. <http://www.rfidinfo.com.cn/Info/n151031.html>.
- [10] JOHO D, PLAGEMANN C, BURGARD W. Modeling RFID signal strength and tag detection for localization and mapping [C]// *ICRA 2009: Proceedings of the IEEE International Conference on Robotics and Automation*. Piscataway: IEEE, 2009:3160–3165.
- [11] BURGARD W, FOX D, THRUN S. *Probabilistic robotics* [M]. Cambridge: MIT Press, 2005.
- [12] FU X, JIA Y, DU J, *et al.* A new resampling algorithm for generic particle filters [C]// *2010 American Control Conference*. Piscataway: IEEE, 2010: 6846–6851.
- [13] BEADLE E R, DJURIC P M. A fast-weighted Bayesian bootstrap filter for nonlinear model state estimation [J]. *IEEE Transactions on Aerospace and Electronic Systems*, 1997, 33(1): 338–343.
- [14] BOLIC M, DJURIC P M, HONG S. Resampling algorithms for particle filters: a computational complexity perspective [J]. *EURASIP Journal on Advances in Signal Processing*, 2004, 2004(15): 2267–2277.
- [15] GORDON N J, SALMOND D J, SMITH A F M. Novel approach to nonlinear/non-Gaussian Bayesian state estimation [J]. *IEE Proceedings F: Radar and Signal Processing*, 1993, 140(2): 107–113.
- [16] HAKALA I, IHALAINEN J, KIVEL I, *et al.* Evaluation of environmental wireless sensor network — case foxhouse [J]. *International Journal on Advances in Networks and Services*, 2010, 3(1/2): 22–32.

(上接第 594 页)

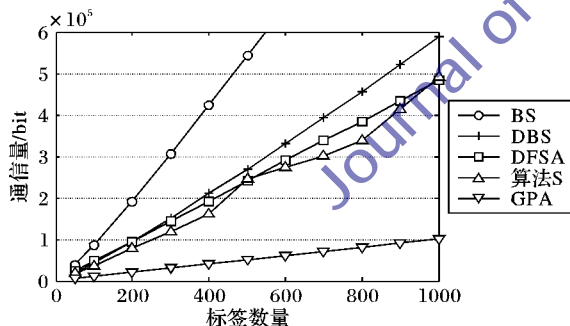


图5 通信数据量

参考文献:

- [1] KLAIR D K, CHIN K-W, RAAD R. A survey and tutorial of RFID anti-collision protocols [J]. *IEEE Communications Surveys and Tutorials*, 2010, 12(3): 400–421.
- [2] ENGELS D W, SARMA S E. The reader collision problem [C]// *Proceedings of IEEE International Conference on Systems, Man and Cybernetics*. Piscataway: IEEE, 2002, 3: 6–9.
- [3] 丁治国, 郭立, 朱学永, 等. 基于二叉树分解的自适应防碰撞算法[J]. *电子信息学报*, 2009, 31(6): 1395–1398.
- [4] 李萌, 钱志鸿, 张旭, 等. 基于时隙预测的 RFID 防碰撞 ALOHA 算法[J]. *通信学报*, 2011, 32(12): 43–50.
- [5] ZHEN B, KOBAYASHI M, SHIMIZU M. Framed ALOHA for multiple RFID objects identification [J]. *IEICE Transactions on Communications*, 2005, E88-B(3): 991–999.
- [6] LEE S-R, JOO S-D, LEE C-W. An enhanced dynamic framed slot-

ted ALOHA algorithm for RFID tag identification [C]// *MobiQuitous 2005: The Second Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services*. Piscataway: IEEE, 2005: 166–172.

- [7] 陈章, 廖明宏. 快速 RFID 防冲突算法[J]. *计算机应用*, 2010, 30(S1): 18–20.
- [8] LAW C, LEE K, SIU K-Y. Efficient memoryless protocol for tag identification [C]// *Proceedings of the 4th International Workshop on Discrete Algorithms and Methods for Mobile Computing and Communications*. New York: ACM, 2000: 75–84.
- [9] FINKENZELLER K. 射频识别技术[M]. 3 版, 吴晓峰, 陈大才, 译. 北京: 电子工业出版社, 2001.
- [10] 李兴鹤, 胡咏梅, 王华莲, 等. 基于动态二进制的二叉树搜索结构 RFID 反碰撞算法[J]. *山东科学*, 2006, 19(2): 51–55.
- [11] 杜海涛, 徐昆良, 王威廉. 基于返回式二进制树形搜索的反碰撞算法[J]. *云南大学学报: 自然科学版*, 2006, 28(S1): 133–136.
- [12] 李波. 基于标签信息分组的射频识别防碰撞算法[J]. *华南理工大学学报: 自然科学版*, 2011, 39(5): 32–37.
- [13] 何晓桃, 郑文丰. RFID 中基于二分叠加的二进制防碰撞算法[J]. *华南师范大学学报: 自然科学版*, 2011, 3: 61–64.
- [14] YANG C-N, HE J-Y. An effective 16-bit random number aided query tree algorithm for RFID tag anti-collision[J]. *IEEE Communication Letters*, 2011, 15(5): 539–541.
- [15] 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]. *电子信息学报*, 2011, 33(5): 1159–1165.