

一种无线传感器网络改进 GAF 算法的设计

孔德川¹, 王建平^{1,2}, 陈 伟², 牛立元¹

(1 河南科技学院 信息工程学院 河南 新乡 453003; 2 武汉理工大学 信息工程学院 湖北 武汉 430070)

摘 要: 提出一种改进的 GAF 算法, 将 GAF 的正四边形网格改变为正六边形方式, 加上分层群组控管机制, 来达到更好的省电效果. 基于 MATLAB 进行了 GAF 及改进 GAF 算法的性能分析, 分别进行了高速及低速环境下两个算法的性能分析, 结果显示, 提出的 GAF 改进算法在网络生存时间、平均数据传输延迟、数据包发送率等方面均优于 GAF 算法.

关键词: 无线传感器网络; 连通支配集; 改进 GAF; 网络生存时间; 数据包传输率; 传输延迟

中图分类号: TP212.9 TP391.9

文献标识码: A

文章编号: 1000-7180(2014)09-0147-06

A Design on Improved GAF Algorithm by Wireless Sensor Network

KONG De-chuan¹, WANG Jian-ping^{1,2}, CHEN Wei², NIU Li-yuan¹

(1 School of Information and Engineering, Henan Institute of Science and Technology, Xinxiang 453003, China;

2 School of Information and Engineering, Wuhan University of Technology, Wuhan 430070, China)

Abstract: An improved GAF which changes the perfect square network into regular hexagon as well as hierarchical group control mechanisms achieves a better purpose of saving electricity. The analysis of GAF and improved GAF algorithm on the basis of MATLAB in high-speed environment and low-speed environment shows that the improved GAF precedes GAF in network lifetime, average data transmission delay and packet delivery rate.

Key words: wireless sensor network; connected dominating set; improved GAF; network lifetime; packet delivery rate; data transmission delay

1 引言

无线传感器网络(Wireless Sensor Network, WSN)是由多个节点随机散布构成的感测网络. 在 WSN 网络中, 每个节点主要包含了感测、计算、通信以及供电^[1]等四个单元. 感测单元用来实现感测环境的变化并产生相关数据. 计算单元用来分析节点所测得的数据, 通汇聚节点(sink). 供电单元则提供各单元运行所需的电能支持. 然而, 由于每个节点所携带的电量有限, 如何有效利用电量是当前 WSN 网络面临的重大挑战.

研究证明, 在 WSN 网络的多数时间段内, 传感器节点都处于等待状态, 而处在等待状态时, 其电量消耗量和处于接收或是发送状态时的电量消耗是相

近的^[2]. 为此, 当前很多基于连通支配集的节能算法研究被提出.

LEACH 是由 Wendi 等^[3]提出的基于连通支配集的层次路由协议. 该协议通过将整个负载平均分配到 WSN 网络中的每个传感器节点, 从而实现 WSN 网络整体能耗的最小化. 然而在 LEACH 协议中, 没有给出簇头结点的分布方式, 因此, 可能会产生远离汇聚节点的簇头节点过早失效, 这就会导致整个传感器网络产生覆盖盲区, 这对于大型的随机传感器网络部署来说是无效的. LEACH 算法的效率相对较高, 但是构建簇的代价较大, 这将导致网络的公平性被破坏.

TMPO^[4]是 Bao 等人提出的基于 CDS 的节能算法, 该算法中, 每个传感器节点仅需要获知其相邻

收稿日期: 2013-11-12; 修回日期: 2014-01-07

基金项目: 国家自然科学基金项目(31371525); 河南省信息技术教育研究重点项目(ITE12037)

的邻居节点信息,通过推举的簇头结点来构建 MIS,通过形成的 MIS 来构造 CDS. 该算法采用节点的剩余电量来标记其优先级. 为了延长节点的生存时间,该算法中要求不断的推举更换簇头节点,这将使得 WSN 网络的拓扑处于不断的变化中.

GAF^[5]是 YAxu 等人提出的基于地理位置的节能算法,GAF 将整个网络划分成许多相等的虚拟正方形网格,并且控制每个虚拟网格中只有一个节点是处于活动模式,使其来担任传递数据包(packet forwarding)的工作,而其他的节点则进入休眠模式,并且周期性的醒来查验自己是否需要转换状态到活动模式,但是,GAF 只向上下左右四个邻居单元转发数据的正方形结构并不是最有效的节能结构^[6].

本文提出了一种基于改进 GAF 的无线传感器网络节能算法. 并通过 MATLAB 仿真软件实现了对改进 GAF 算法的性能分析.

2 GAF 算法

2.1 GAF 的虚拟网格

GAF 假设每个节点通过 GPS 或是其他的定位系统可以知道节点目前所在的位置^[7],并且使用自己的区域信息以及虚拟网格的概念来决定哪些节点为等价的节点. 如图 1 所示,每个节点的距离为相等的单位,而传输范围为两个单位的距离. 假定现在有一对传输介于节点 1 与节点 4 之间,在这对传输中节点 2 与节点 3 都是可以担任传递数据包的工作,因此可以将它们视为等价节点;以另一个例子来说,假设介于节点 1 与节点 5 之间存在一对传输,就只有通过节点 3 的转发才可完成,因此在这个传输中,并没有任何与节点 3 等价的节点存在.

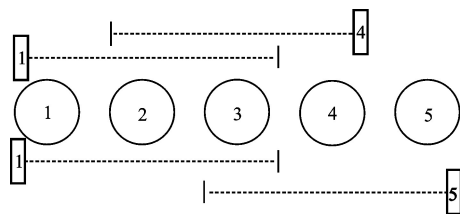


图 1 节点等价问题

GAF 将这个问题以虚拟网格的方式解决,并且为了保持连通性,GAF 要求任何两个邻近的虚拟网格必须能够彼此通信. 如图 2 所示,虚拟网格 A 可以与虚拟网格 B 通信,虚拟网格 B 可以与虚拟网格 C 通信,也就是说节点 1 可以与节点 2,3,4 通信,而节点 2,3,4 的数据可以直接通信至节点 5. 因此,节

点 2,3,4 是属于等价节点,它们其中的两个可以进入休眠模式而不影响整体网络的连接性.

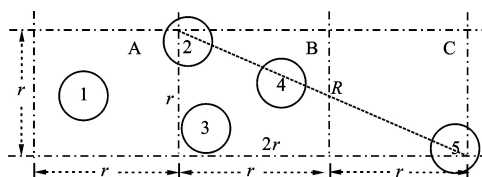


图 2 GAF 的虚拟网格

根据如图 2 的关系可以,可以重新计算每个虚拟网格的边长. 假设无线传输的信号范围是 R ,而虚拟网格的边长是 r ,可以得到以下的公式:

$$r^2 + (2r)^2 \leq R^2 \quad (1)$$

$$r \leq \frac{R}{\sqrt{5}} \quad (2)$$

2.2 GAF 的状态转移

GAF 协议有活动、发现、休眠三种状态. 在系统初始时,所有节点都处于发现状态,每个节点都开启通信单元,并且通过交换同一个虚拟簇中各节点的发现消息来更新彼此的信息. 发现信息包含了四种节点的信息:节点序号(Node ID, NID)、网格序号(Grid ID, GID)、估计的节点活动时间(Estimated Node Active Time, T_a),以及节点状态(Node State, NS).

GAF 设计了三个定时器分别在不同的状态下使用,分别为 T_a 、 T_d 、 T_s . T_a 表示节点处于活动状态的持续时间, T_d 表示节点处于发现状态的持续时间,而表示节点处于休眠状态的持续时间. 节点进入发现状态时, T_d 定时器就开始倒数,当 T_d 倒数至 0 却还未收到来自其他节点的发现信息时,节点便进入活动状态,并且设定自己的 T_a 定时器,随后广播出一个发现信息来抑制同个虚拟网格中的其他节点进入活动状态;否则,如果在 T_d 倒数未至 0 时,便收到来自其他节点的发现信息时,节点在关闭通信单元进入休眠状态,并且设定自己的 T_s 定时器.

进入活动状态的节点必须周期性的广播它的发现信息,这可以避免其他进入发现状态的节点成为多余的活动节点. 而进入休眠状态的节点,在 T_s 定时器倒数完毕后,便自动将自己通信单元开启并且进入发现状态.

3 基于改进 GAF 的节能路由协议的设计

3.1 簇的构建模型

在 GAF 协议中采用的是正方形的网格,本文提出的算法采用正六边形的网格. 理论上无线信号

传输呈现的是一个球面的空间区域,为此,设计的网格平面模型越接近一个平面圆越好,设计的正六边形则更切近圆的区域.本文提出的算法采用7个正六边形来构建一个簇,如图3所示.假设每个传感器节点都处于正六边形的中心位置.

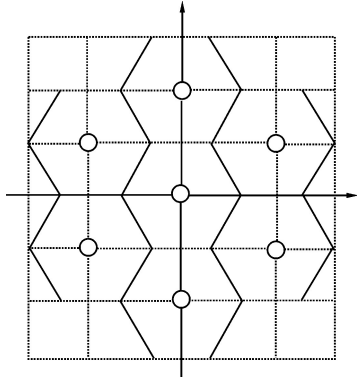


图3 簇的构建模式

为了有效减少处于活动状态的节点,使得更多的节点能够进入休眠状态,在设计上要求任何三格虚拟网格必须能够彼此通信.为了保持连通性,必须重新计算每个虚拟网格的最大边长.假设无线传输的信号范围是 R ,而虚拟网格的边长是 r ,如图4列出了无线传输信号的范围和虚拟网格的边长之间的关系.

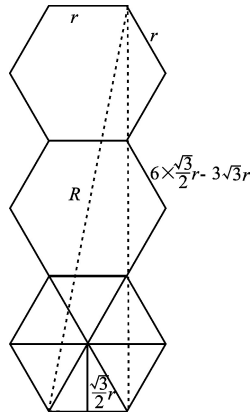


图4 虚拟网格与传输距离的约束关系

从图4可以看到,传感器的最大的传输距离 R 和虚拟网格边长 r 满足勾股定理,可以得到以下的公式:

$$r^2 + (3\sqrt{3}r)^2 \leq R^2 \quad (3)$$

$$\text{即: } r \leq \frac{R}{\sqrt{28}} \quad (4)$$

这样,就可以由设定的传感器传输范围 R 来规划整个传感器节点的部署范围和密度.假设整个部署区域的面积为 A ,则整个区域内部署的传感器数量为

$$n = \frac{A}{\pi r^2} = \frac{28A}{\pi R^2} \quad (5)$$

3.2 改进 GAF 算法

本文提出改进的 GAF 算法,算法的状态包括休眠、发现和激活三个状态.图5为本算法的状态转移图.

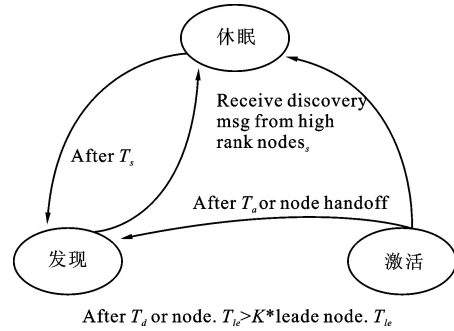


图5 改进 GAF 算法的状态转移图

3.2.1 发现状态算法

在算法的发现状态过程中,在 T_d 归 0 前,若节点的优先权值为 0,且收到来自其隶属群组节点的信标,或是节点的优先权值不为 0,且收到来自其两方隶属群组节点的信标,代表其隶属的群组原本已有簇头节点.

在簇都有头节点的情况下,若簇中某个节点由休眠状态转为发现状态,在该节点的估计活动时间 T_a 值(大于 30 s)大于簇头节点的 k 倍时,表示当前簇头节点的电量远低于该活动节点,则执行强制接管方式.设定该节点的活动时间 $T_a = T_a$,广播信标,并且将状态转换成活动状态,以该节点来接管这个群组,虽然刚强迫接管的一开始群组会有两个簇头节点,原有的簇头节点会听到新的簇头节点的信标信息,但原有簇头节点等级将因为其拥有较短 T_a 值而转入休眠状态;否则,上述两条件没有成立,则将该节点的 T_d 值设定为常值 C ,并且把状态转移至休眠状态来进行省电.改进 GAF 的发现状态算法如下:

```

1  int i=1;
   //初始化节点数量;
2  int k=10;
3  node[i].state= 'discovery';
   //节点置为发现状态
4  node[i].radio='ON';
   //开启射频单元
5  node[i].location=GetLocation(node[i].area);
   //定位节点位置
6  wait(node[i].slot);
   //监听节点时槽
7  node[i].information=GetInfo(node[i].location);
   //从节点位置获取节点信息
8  if(node[i].T_d!=0)
9  node[i].T_d--;
   //节点的发现持续时间自减

```

```

10  if(node[i]. beacon. status='received')
    //节点时槽接收状态
11  if(node[i]. Tea>node[head]. Tea
    &&node[i]. Tea>30 s);
12  node[i]. Ta=Tea
    //将估计最大时间置为节点 i 的活动时间
13  Broadcast(node[i]. beacon);
    //广播节点 i 的时槽
14  node[i]. state='active';
    //节点置为活动状态
15  if(node. Ts=g)
16  node[i]. state='sleep';
    //节点状态修改为休眠模式
17  node. Ta=The
    //设置节点的活动时间为 The
18  if(node. Ta<30 s)
19  vnode. Ta=Tle;
    //设置节点的活动时间为 Tle
20  if (node[i]. beacon. status='nrcy')
    //节点时槽没有收到
21  i++;
    //节点计数器增加
22  else
23  broadcast(node[i+1]. beacon);
    //广播节点 i+1 的时槽
24  node[i+1]. state='active';
    //置节点 i+1 的状态为激活
25  seek(node[i+1]);

```

3.2.2 改进 GAF 算法的活动状态算法

节点进入活动状态后,每工作一段时槽便倒数 T_a 定时器,并且判断节点是否发生切换、收到同一个群组其他活动节点的信标或是节点电量耗尽.当 T_a 倒数至 0 或是节点发生切换,则将该节点 T_d 计时器选择一个由 0 到某常数的随机值,并且切换状态至发现状态并执行发现状态的程序.如果收到同一个群组其他节点的信标,并且自己的 T_{le} 小于该节点的 T_{le} ,则代表有其他节点将要接管这个群组,将 T_s 定时器设定为 C 后,将状态转换至休眠状态.改进 GAF 的活动状态算法如下所示.

```

1  node[i]. state='active';
    //节点处于活动状态
2  wait(node[i]. Ts);
3  node[i]. Ta--;
4  if(node[i]. Ta=0)
5  node[i]. Td=random(0, p);
6  node[i]. state='discovery';
    //节点置为发现状态
7  elseif(node[i]. beacon. status='received')
8  if(node[i]. Tle< node[i+1]. Tle)
9  node[i]. Ts=C;
10 node[i]. state='sleep';
    //节点置为休眠状态
11 else
12 node[i]. Ts>τi;
    //节点 i 的 Ts 超时;
13 node[i]. state='die';
    //节点失效;
14 seek(node[i+1]);

```

3.2.3 休眠状态算法

节点进入休眠状态后就倒数 T_s 定时器,直到 T_s 定时器归 0 后便重新设定节点的 T_d 定时器,并且将该节点的状态转换成发现状态.改进 GAF 的休眠状态算法.

```

1  node[i]. state='sleep';
    //节点初始状态为休眠
2  node[i]. radio='OFF';
    //关闭射频单元
3  wait(node[i]. Ts);
4  node[i]. Ts--;
    //时槽置自减
5  if (node[i]. Ts=0)
6  node[i]. Td=random(0, p);
7  node[i]. state='discovery';
    //节点 i 置为发现状态
8  i++;
9  seek(node[i+1]);

```

3.3 改进 GAF 算法的性能分析

以 GAF 而言,我们考虑有 n 个节点完全平均地散布在 A 单位面积的区域上,假定每个节点的传输范围为 R ,可知一个 GAF 虚拟网格的最大边长为 $\frac{R}{\sqrt{5}}$,令虚拟网格数量为 m ,则 $m = \frac{A}{\left(\frac{R}{\sqrt{5}}\right)^2}$,因为我们

的假设是所有节点完全平均地散布,因此,每一个网格平均最多拥有 $\frac{n}{m}$ 个节点,即为 $\frac{n \times R^2}{5 \times A}$ 个节点.在最佳的理想情况下(假设不考虑 GAF 控制数据包所产生的负担),单一网格最多应只有一个活动节点,其他同在一个网格的节点将会进入休眠状态,由上述公式我们可以证明 GAF 最多可以延长其网络生存时间为 $\frac{n \times R^2}{5 \times A}$ 倍.由该公式我们不难看出,当节点数量 n 越大,GAF 可以有更长的网络生存时间.

接下来以同样的假设来考虑改进 GAF 算法,当 n 个节点完全平均地散布在 A 单位面积的区域上,每个节点的传输范围为 R ,根据 2.1 节,改进 GAF 算法的正六边形虚拟网格最大边长为 $\frac{R}{\sqrt{28}}$,单

一网格最大面积为 $\frac{3 \sqrt{3} \times R^2}{56}$,因此,一个簇的最大

面积为 $\frac{57 \sqrt{3} \times R^2}{56}$.令群组数量为 g ,根据改进 GAF 算法算法定义,在一个簇中分别的七个群组,个别只需要一个活动节点即可达到连通性,所以我们可以推算出每个平均群组面积为 $\frac{57 \sqrt{3} \times R^2}{392}$,根

据上述说明可推得 $g = \frac{A}{\frac{57 \sqrt{3} \times R^2}{392}}$,因为我们的假

设是所有节点完全平均地散布,因此,每一个群组平均最多拥 $\frac{n}{g}$ 个节点,即为 $\frac{57\sqrt{3}\times n\times R^2}{392\times A}$ 个节点.同理,在最佳的理想情况下(假设不考虑改进GAF算法控制数据包所产生的负担),由上述公式我们可以证明改进GAF算法最多可以延长其网络生存时间为 $\frac{57\sqrt{3}\times n\times R^2}{392\times A}$ 倍.

将GAF以及改进GAF算法最多可延长的网络生存时间倍率相比,GAF比改进GAF算法为 $\frac{n\times R^2}{5\times A}:\frac{57\sqrt{3}\times n\times R^2}{392\times A}$,约为1:1.25,经由上述简单的推导,理论上证明在网络生存时间上,改进GAF算法优于GAF约25%.

4 实验

为了对比该协议的性能,基于MATLAB进行了实验仿真,对比了GAF和本文提出的改进GAF算法的性能.仿真采用的相关参数如表1所示.仿真时,设定节点在 $1\,500\text{ m}\times 300\text{ m}$ 的范围中进行随意方向的运动,定义了行人和车辆两种节点移动速度.分别仿真在不同的停留时间以及移动速度情况下这两种算法的性能情况.设定传感器节点传输数据包的大小为512 B,数据产生率则为20 p/s.信号传输范围为250 m,设定节点发送数据的耗能为1.6 W、接收数据的耗能为1.2 W、等待信号(listening)耗能为1.0 W.节点进入休眠状态时的耗能为0.025 W.不考虑源节点和目标节点的电量,为此只观察构建的网络骨干节点的电量消耗情况.设定骨干节点的传输电量初始值为500 焦耳(Joules).由于GAF以及改进GAF算法都通过GPS得到位置信息,因此,假设GPS的电量耗能为0.033 W.当节点进入休眠状态并且关闭通信单元时,GPS部分并不会被关闭,以确保可以正确的更新位置信息.

表1 仿真相关参数

参数设置	参数取值
网络范围	$1\,500\text{ m}\times 300\text{ m}$
射频传输距离	30 m
骨干节点数量	50
节点移动速度	低速:0~1 m/s 高速:20 m/s
传输数据包的大小	512 B
数据包传输速率	20 p/s
节点初始电量	500 J
发送功耗	1.6 W
接收功耗	1.2 W
监听功耗	1.0 W
休眠功耗	0.025 W
GPS功耗	0.033 W

4.1 网络生存时间

无线传感器网络中,网络的生存时间通常用剩余节点存活率来衡量.在整个WSN中,随着运行时间的增加,剩余存活节点的数量就会减少,在某个时段内,剩余存活节点比例越高代表该算法的网络生存时间越长.

图6所示为两个算法分别在低速和高速移动时的网络生存时间对比值.通过对比可以看到,在高速环境下GAF和改进GAF算法的网络生存时间都比低速时差,因为在高速情况下,节点唤醒检查活动节点的频率较高,多浪费了额外的电量.改进GAF算法的接管机制使得节点能够较平均被利用,因此能够延长有效的网络生存时间.仿真结果显示,无论是在任何速度下或是停留时间的长短,改进GAF算法在网络生存时间都优于GAF.

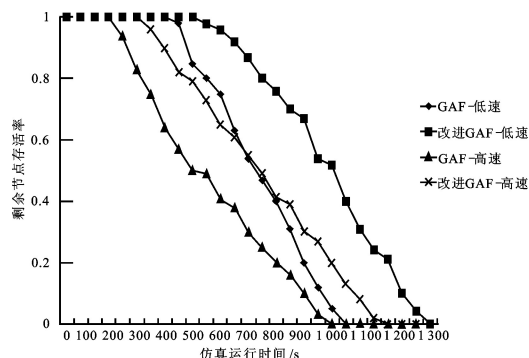


图6 网络生存时间对比

4.2 数据包发送率

图7所示为在高速和低速两种移动速度下GAF算法和改进GAF算法的数据包发送率对比.可以看到两种算法在低速移动时的数据包发送率要高一些.因为在高速移动环境下,不断切换导致的发送出错率较高.由于GAF会将一些节点强制进入休眠,使得当节点移出网格时会出现网格无人接管的问题,导致连通性被破坏而降低数据包发送率.改进GAF算法采用正六边形的网格,其进入休眠状态的节点比GAF多,但是基于平均接管的方式使得构建的CDS稳定性比GAF强,因此改进GAF的数据包传输率高.

4.3 平均数据传输延迟

图8为GAF算法和改进GAF算法的平均数据传输延迟的仿真对比结果.可以看到,在低速移动时的平均数据传输延迟要比高速移动时的平均数据传输延迟要小.在低速的情况下,GAF休眠的节点少于改进GAF算法,因此数据传输延迟高于改进GAF算法.在高速情况下GAF算法发生链路中断

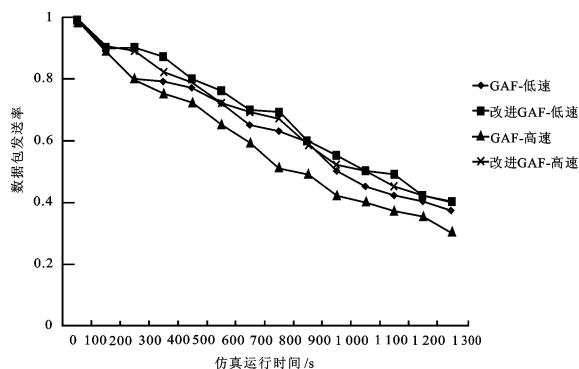


图7 数据包发送率对比

的次数被改进 GAF 算法要多. 为此, 改进 GAF 算法的平均延迟优于 GAF.

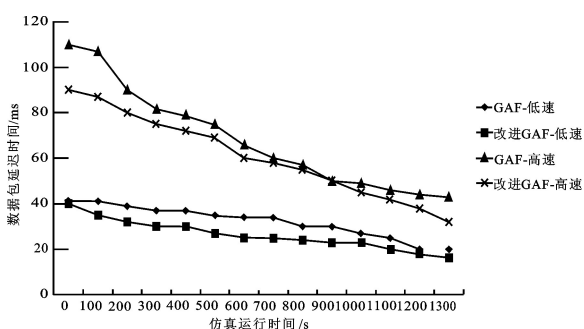


图8 平均数据传输延迟对比

5 结束语

本文提出一种改进的 GAF 算法, 仿真实验表明本文提出的改进 GAF 算法的网络生存时间、数据包平均传达率、平均的数据传输延迟比 GAF 协议要高.

参考文献:

- [1] G J Pottie, W J Kaister. Wireless integrated network sensors[J]. Communications of the ACM, 2000, 43(5):51-58.
- [2] Akyildiz I F, Weilian S, Sankarasubramanian Y, et al. A Survey on Sensor Networks[J]. IEEE Communication Magazine, 2002, 48(8):1-14.
- [3] Heinzelman WR, Chandrakasan A, Balakrishnan H. Energy-Efficient communication protocols for wireless microsensor network[C]// Proc of the Hawaaiian Int'l Conf on Systems Science. Hawaii:IEEE, 2000.
- [4] Bao L, Garcia-Luna-Aceves JJ. Topology management in ad hoc networks[C]//Proc of the MobiHoc. [s. l.]: IEEE, 2003.
- [5] Y Xu, J Heidemann, D Estrin. Geography-in form ed energy conservation for Ad Hoc Routing[C]// Proceedings of the 7th Annual Internati onal Conference on Mobile Computing and Networking. [s. l.]: IEEE, 2001.
- [6] 刘曙, 刘林华, 陶军. 一种基于蜂窝结构的改进 GAF 算法[J]. 计算机技术与发展, 2009, 39(1):39-43.
- [7] P Baul, V N Padmanabhan. User Location and Tracking in An In-building Radio Network[R]. Technical Report MSR-TR-99-12, Microsoft Research, Feb. 1999.

作者简介:

孔德川 男, (1977-), 硕士, 讲师. 研究方向无线传感器网络、嵌入式系统.

(上接第 146 页)

- [3] 郭文平, 刘政林, 陈毅成, 等. 高吞吐率、低能耗的 SHA₁ 加密算法的硬件实现[J]. 微电子学与计算机, 2008, 25(5):76-79.
- [4] Ling Bai, Shuoguo Li. VLSI implementation of high-speed SHA-256[C]// IEEE 8th International Conference on ASIC. Changsha, Hunan: IEEE, 2009: 131-134.
- [5] Eun-Gu Jung, Daewan Han. Low area and high speed SHA-1 implementation[C]// International Soc Design Conference. Jeju: IEEE, 2011:365-367.
- [6] 丁冬平, 高献伟. SM3 算法的 FPGA 设计与实现[J]. 微型机与应用, 2012, 31(5):26-27.
- [7] 王晓燕, 杨先文. 基于 FPGA 的 SM3 算法优化设计与实现[J]. 计算机工程, 2011, 37(19):1-4.

- [8] 刘宗斌, 马原, 荆继武, 等. SM3 哈希算法的硬件实现与研究[J]. 信息安全, 2011(9):191-193.

作者简介:



张倩 女, (1988-), 硕士研究生. 研究方向为杂凑密码芯片设计.

李树国 男, (1963-), 博士生导师. 研究方向为信息安全算法及其大规模集成电路设计与实现.