

一种事件驱动型无线传感器网络的分层路由算法^①

沈永增, 陈宣扬, 贾莲莲

(浙江工业大学 计算机应用技术系, 杭州 310023)

摘要: 为了降低无线传感器网络在突发事件监测系统中的能量消耗, 延长网络生存周期, 结合突发事件监测系统的特点, 在 TEEN 协议的基础上进行改进, 设计实现了一种基于事件驱动的分层路由算法—TEEN-PE 算法。该算法采用先分区再建簇的方式解决了 TEEN 协议中最优簇首个数难以确定、簇首分布不均匀等问题。同时, 在簇形成阶段和簇间通信阶段, 综合考虑节点剩余能量以及节点到基站的距离因素, 以提高网络在时间上和空间上的通信平衡。该算法能有效降低节点能耗、延长网络生存周期。最后通过 NS-2 进行仿真, 仿真结果显示, TEEN-PE 算法在节点存活率以及能耗上要明显优于 TEEN 协议。

关键词: 无线传感器网络; 突发事件监测; TEEN-PE; TEEN; 剩余能量; 距离

Hierarchical Routing Algorithm in Event-Driven Wireless Sensor Networks

SHEN Yong-Zeng, CHEN Xuan-Yang, JIA Lian-Lian

(Department of Computer Application Technology, Zhejiang University of Technology, Hangzhou 31100, China)

Abstract: Based on researches on TEEN protocol and combined the characteristic of monitoring emergencies system, an event-driven hierarchical routing algorithm (TEEN-PE) has been proposed to reduce the energy consumption in the monitoring system of emergencies and prolong the lifetime of the wireless sensor networks (WSN). TEEN-PE has solved the problems of the difficulty in determining the initial number of the optimal clusters and the asymmetrical distribution of cluster in TEEN protocol. And to improve the communication balance of the networks on times and space, the algorithm is taken into accounts the factor of the residual energy of the nodes and the distance of the node to the base station during the phase of cluster formation and communicates among clusters. The algorithm can effectively reduce energy consumption and prolong the lifetime of the networks. It is proved by simulated in NS-2, and the simulation results show that TEEN-PE has an advantage over TEEN protocol on nodes survival and energy consumption.

Key words: wireless sensor networks; monitoring emergencies; TEEN-PE; TEEN; residual energy; distance

无线传感器网络是当前国际上备受关注的、由多学科高度交叉形成的新兴前沿研究热点领域。它通过各类集成化的微型传感器协作的实现实时感知、监测和采集环境信息, 并通过随机自组织无线通信网络以多跳中继方式将感知信息传送到用户终端^[1-3]。这种网络可以广泛地应用于国防军事、环境监测、医疗卫生、反恐抗灾等领域。

与计算机网络不同, 无线传感器网络中的节点一般由电池供电、难以补充, 而在无线路由通信过程中所消耗的能量要远远大于计算和传感过程中所消耗的

能量^[4]。同时, 无线传感器网络的应用环境千差万别, 没有一个路由机制适合所有的应用。因此, 开发与具体应用相适应的高效节能路由协议是十分必要的。

一般来讲, 无线传感器网络按照应用类型可分为两类, 一类是时间触发型, 这种网络适用于周期性数据监测应用, 无线节点以恒定间隔周期性开启传感器和收发器对环境进行感知, 并将数据传递给基站。另一类是事件驱动型, 无线传感器节点的收发器大部分时间处于休眠状态, 只有当突发事件发生时才开启收发器, 传送感知数据到基站。这类应用系统对网络的

① 收稿时间:2010-11-20;收到修改稿时间:2011-01-08

实时性、可靠性以及可扩展性要求较高,适合建立层次结构的拓扑模型。目前许多研究者提出了大量层次路由协议,典型的有 LEACH、PEGAGIS、TEEN 等,但这些典型的层次路由中,除了 TEEN 协议外,都未考虑事件驱动型应用系统的特点。因此本文针对事件驱动型无线传感器网络系统的特点,在 TEEN 协议的基础上提出了一种改进型的无线传感器网络分层路由算法。

1 事件驱动型系统特点

事件驱动型系统是无线传感器网络的一个重要应用,如化工厂的可燃气体泄露、建筑物内烟雾超标等。这类应用具有以下特点:

1) 网络中有唯一的基站(Base Station, BS)和数量众多的传感器节点,其中基站部署在监测区域以外的固定位置。传感器节点部署在监测现场,节点数目有具体的应用规模决定,节点部署完后一般不再移动。

2) 基站拥有持续的能量供给,能够使用足够大的无线发射功率,可直接与传感器节点取得通信。而传感器节点能量相同且不可补充,且无线发射功率有限、可调。

3) 传感器节点不需要周期性的向基站发送采样数据,仅在突发事件发生的情况下才发送数据,发送数据量比较小,但要求及时发现和快速响应。

4) 传感器节点的无线收发模块大部分时间处于休眠状态,但是一旦监测到危险情况,将存在多个节点被唤醒,同时发送数据。

5) 监测区域可根据实际情况分成多个子区域,各传感器节点根据部署的位置以及区域设置编号和所属区域号。

其中,前两个特点是当前无线传感器网络的默认假设。而后面三个特点是事件驱动型系统所独有的。其中第5点至关重要,因为对于事件监测型系统来说,确定事件发生的区域是整个监测系统的目的所在,如果不知道节点所属的区域,整个监测系统就没有任何意义。

2 相关算法

2.1 LEACH 算法

LEACH 协议^[5]是第一个提出数据聚合的层次路由协议。通过周期性等概率的随机选举簇首的方式来

平衡网络各节点的能量消耗。每轮选举的方法是:各节点产生一个[0,1]之间的随机数,如果该数小于 $T(N)$,则该节点为簇首。 $T(N)$ 的计算公式如下:

$$T(n) = \begin{cases} \frac{p}{1-p(r \bmod 1/p)} & n \in G \\ 0 & \text{其他} \end{cases} \quad (1)$$

式中, p 为网络中簇首数与总节点数的百分比; r 为当前的选举轮数。 G 为最近 $1/p$ 轮中不是簇首的节点集。

这样,在 $1/p$ 个周期里,每个节点至少有一次成为簇首。成为簇首的节点在无线信道中广播这一消息,其余节点选择加入信号最强的簇首。簇内节点通过一跳通信将数据传送给簇首,簇首也通过一跳通信将聚合后的数据传送给 sink 节点。该协议采用随机选举簇首的方式避免簇首过多消耗能量,提高网络生存时间。但协议层次化的目的在于数据聚合,而且仍采用一跳通信,虽然传输延时小,但是要求节点具有较大功率的通信能力。网络扩展性差,不适合大规模监测系统;即使在小规模系统中,离 sink 节点较远的节点由于采用大功率通信,也会导致生存时间较短。

2.2 TEEN 算法

TEEN 协议^[6]是首个为事件驱动型系统设计的路由协议,它具有实时性,可以对突发事件做出快速反应。TEEN 采用与 LEACH 协议相同的成簇方式。成簇完成后, Sink 节点通过簇头向全网节点通告两个门限值(分别为硬门限和软门限)来过滤数据发送。在簇的稳定阶段,节点通过传感器不断地感知其周围环境,当节点第一次监测到数据超过硬门限时,节点打开收发器向簇头上报数据,并将当前监测数据保存为监测值(Sensed Value, SV)。此后节点再次进行数据传送时要满足两个条件:①当前节点监测到的数据比硬门限大;②当前的监测值与 SV 之差的绝对值不小于软门限,节点才向簇首上报数据,并将当前监测数据保存为 SV。在簇重构过程中,如果新一回合的簇首确定,该簇首可重新设定和发布以上两个参数。且簇首通过 TDMA 方法实现数据调度,避免冲突碰撞。

TEEN 协议利用软、硬门限大大减少了数据传输量和传送次数。用户可通过设置不同的软阈值来平衡监测准确性与系统节能性两项指标。虽然 TEEN 协议比 LEACH 更节能,但存在以下不足:

1) 簇首分布不平衡:协议选择簇首的方式具有较大的随机性,可能会造成簇首分布的不平衡和簇内成

员节点数量不均衡。

2) 节点状态不可知: 如果开始某个节点感知的数据达不到硬门限, 且若以后也未达到, 则用户将无法得到任何数据, 也无法知道这个节点是否失效。

3) 能量消耗不均衡: 根据协议, 离 Sink 节点越近的簇首节点不仅要转发自己的监测数据, 同时又要作为比自己远的节点的中继节点, 而 TEEN 协议在选举簇首时没有考虑到节点剩余能量状态, 这将会导致离 Sink 节点越近的节点过早死亡, 从而导致网络的失效。

3 TEEN-PE算法

如前所述, TEEN 算法是一种专门针对事件驱动型应用提出的分层路由协议, 有其自身的优缺点。本文针对事件驱动型系统的特点, 在 TEEN 协议的基础上进行改进, 提出了一种基于事件驱动的分层路由算法—TEEN-PE 算法。

该算法主要分为网络系统初始化、簇的建立、数据传输三个阶段。

3.1 系统初始化

系统初始化阶段, 基站广播一个 Inquire 消息, 所有节点接收到这个消息后, 向基站回馈一个 Hello 消息。Hello 消息的格式如图 1 所示, 其中 TYPE 为消息类型, ID 为节点的标识号, ID_area 为节点所属的分区号, Energy 为节点剩余能量。

TYPE	ID	ID_area	Energy
------	----	---------	--------

图 1 Hello消息格式

基站在获取 Hello 消息后, 经过进一步处理, 可获知监控区域中的分区数 Z, 以及各分区中的节点数目 N。根据文献[7], 最优簇首个数可以根据如下公式计算得到:

$$K_{opt} = \frac{\sqrt{N}}{\sqrt{2\pi}} \sqrt{\frac{\epsilon_{fmp}}{\epsilon_{tmp}}} \frac{M}{d_{toBS}^2} \quad (2)$$

其中, N 为网络中的节点数目, M 为节点通信范围, d_{toBS} 为节点到基站的距离, ε_{fmp} 和 ε_{tmp} 分别为在 m₂ 和 m₄ 空间范围内发生 1bit 数据消耗的放大器能量, 大小为固定值。对于一个固定网络来说, R 为固定值, 因此整个网络范围内 d_{toBS}、ε_{fmp}、ε_{tmp} 的值是固定的, 如果 N 已知, 那么 K_{opt} 的值就可计算

得到。在系统初始化阶段, 基站通过获取 Hello 消息并处理后, 就可用公式(2)计算得到每个分区的最优簇首数目, 同时根据公式(3)计算出各分区中最优簇首数与节点数的百分比。

$$P_{opt}(i) = K_{opt}(i) / N(i) \quad i \in Z \quad (3)$$

式中, K_{opt}(i), N(i), P_{opt}(i)分别为分区 i 中最优簇首数目, 节点总数以及最优簇首比例。

基站在处理得到各个子区域的最优簇首比例后, 向各分区广播 Init 消息。Init 消息格式如图 2 所示, 其中, TYPE 为消息类型, ID_area 为分区号, N_area 为最优簇首个数, P_area 为最优簇首比例。

TYPE	ID_area	N_area	P_area
------	---------	--------	--------

图 2 Init 消息格式

子分区中的各节点根据 Init 消息包中的 ID_area 来选取需要的数据。由于无线传感器网络中, 节点可以随时加入或者退出网络, 所以最优簇首的数目也具有动态变化性。这就需要在每轮重新选举簇首节点之前, 由簇首重新确定该分区中的最优簇首数目和比例。

在系统初始化阶段, 通过将整个监控区域进行分区划分, 确定每个分区的最优簇首数目和比例, 解决了 TEEN 算法中最优簇首数难以确定的问题。同时在各分区内选取各自的簇首, 也有效的解决了簇首分部不均匀的问题。

3.2 簇的建立

TEEN-PE 算法在簇首的选举上考虑节点的剩余能量状态, 以实现能量的均衡使用。

3.2.1 参数的设定

结合节点剩余能量因素, 将公式(1)修改如下:

$$T(i, j) = \begin{cases} \frac{p_{opt}(i)}{1-p_{opt}(i)[r(i) \bmod 1/p_{opt}(i)]} * \frac{E_{j_cure}}{E_{j_init}} & i \in Z, j \in G_i \\ 0 & \text{其他} \end{cases} \quad (4)$$

其中, E_{j_cure} 为节点 j 当前的剩余能量; E_{j_init} 为节点 j 初始能量。这样, 在 1/p_{opt}(i)个周期里, 子分区 i 中能量越多的未当选过簇头的节点成为簇首的可能性越大。同时为了使子分区中簇首的数目保持在最优值, 防止产生过多的簇首, 算法中引入权值 W(n), 公式如下所示。

$$W(j) = \omega_1 * (1 - \frac{E_{j_cure}}{E_{j_init}}) + \omega_2 * (\frac{D_{2BS}(j)}{D_{max}}) \quad j \in Gi \quad (5)$$

其中, E_{j_cure} 为节点 j 当前的剩余能量, E_{j_init} 为节点 j 初始能量。 $D_{2BS}(j)$ 为节点 j 到基站(BS)的距离, D_{max} 为网络中节点到基站的最远距离。 $\omega_1 > 0$, $\omega_2 > 0$ 为其对应的参数值。根据 $W(j)$ 设置定时器。当节点能量越大、距离基站越近, $W(n)$ 的值越小, 定时时间也越短, 即获得越早的广播簇首权。

3.2.2 簇的建立

各节点按公式(5), 计算自己的权值, 根据权值的大小产生一个定时器, 定时时间和权值大小成正比。若节点 j 所在的分区 i 中簇首数目未达到最优个数, 当定时时间到达后, 该节点产生一个 0 到 1 之间的随机数, 若当前轮, 这个值小于设定的阈值 $T(i,j)$, 则这个节点成为簇首, 并在网络中广播 ADV_CH 消息, 告知其它节点自己是分区 i 的簇首节点, 否则退出选举成为普通节点。其中, ADV_CH 消息主要包括 ID 号、分区号、剩余能量以及表明消息类型的头部。该分区中未当选簇首的普通节点根据自己与簇首之间的距离选择加入哪个簇, 并发送 JOIN_REQ 消息告知簇首节点它要加入该簇。JOIN_REQ 包含节点 ID、分区号、节点剩余能量以及表明消息类型的头部。整个算法流程如图 3 所示。

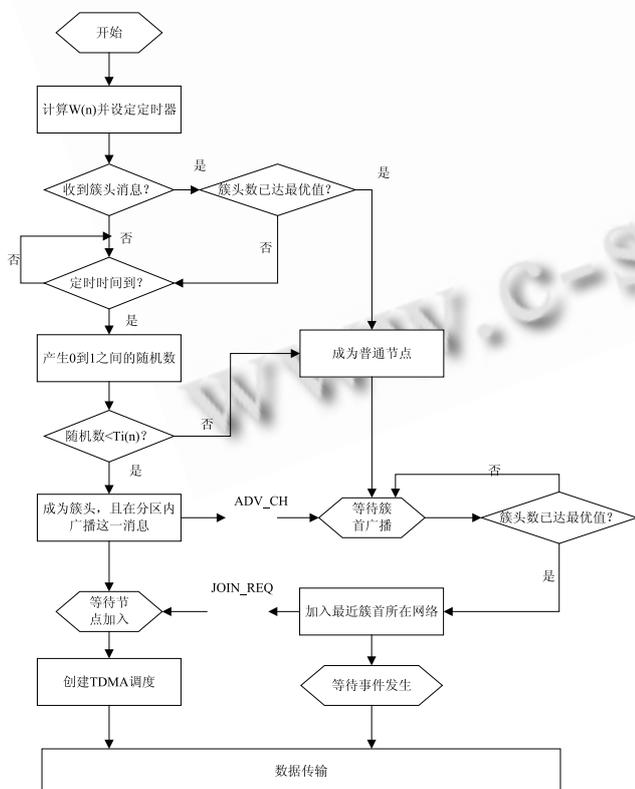


图 3 TEEN-PE 算法成簇

簇建立后由簇首向各成员节点发送时间同步报文, 以实现簇内的时间同步。并且发送一个控制帧, 控制帧的格式如图 4:

帧类型	硬阈值	软阈值	时间片	计数时间	簇更新时间
FT	HT	ST	TS	CT	TN

图 4 控制帧格式

其中, TS(Time Slice)为簇头分配的 TDMA 时间片。CT(Count Time)计数事件: 规定当普通节点距最后一次通信超过 CT 时间而没发送数据, 此时, 强制该节点往簇首发送一次数据, 即向簇首报告当前节点存活。TN(New Time): 下轮簇首的更新时间。

与 TEEN 算法不同, 各簇首节点除了维持一张成员表外, 还维持一张邻接簇首表。当收到其它簇首的 ADV_CH 消息后, 更新自己的邻接簇首表, 同时回传 ADV_CH 消息, 以实现邻接簇首表的相互更新。

3.3 数据传输

数据传输为两种: 簇内普通节点与簇首之间的簇内传输以及簇首与基站间的簇间传输。簇内节点采用 TEEN 原本的策略使用单跳方式与簇首实现数据传输: 簇首负责确定和维护一个 TDMA 时间表, 簇内普通节点在相应时隙内把待发送的数据传给簇首。而大部分簇首与基站之间的距离比较远, 根据文献[8], 为了降低能耗, 采用多跳的方式建立传输路径进行数据传输。其过程如下:

- ① 簇首节点 i 在其邻接簇首表中选取出比它本身到 BS 近, 且剩余能量是大于簇首能量维持下限(Head Energy Keep Threshold, HEKT)的簇首节点集 K 。
- ② 从簇首节点集 K 中选取出距离 BS 最近的簇首作为节点 i 的下一跳节点。
- ③ 其下一跳簇首节点继续重复①, ②操作, 直到找不到这样的邻接簇首为止。然后由该簇首节点与 BS 直接单跳通信。

由此, 建立起一条逐渐接近 BS 的传输路径。此方法同时考虑了节点到 BS 的距离和中继簇首节点的剩余能量, 能够在网络生命周期和能量使用上取得平衡。

4 TEEN-PE算法仿真分析

本文采用 NS-2 网络模拟器对 TEEN-PE 和 TEEN 算法在网络平均能耗和节点存活数目两个方面进行比较, 以此来检验 TEEN-PE 算法的性能。仿真主要参数

设置如下:

仿真模拟系统有 100 个节点组成, 节点分布在 $(x=0,y=0)$ 和 $(x=100,y=100)$ 的 $100*100m^2$ 的正方形区域内, 节点部署后不移动, 基站(BS)位于坐标 $(x=150, y=50)$ 处; 节点初始能量为 2J。对于 TEEN-PE 算法, 将正方形区域分成 5 个子分区。各节点根据设置的区域号分配到 5 个子分区中, 在各分区中, 节点随机部署, 部署后不再调整位置。对于 TEEN 算法, 节点随机的部署到正方形区域中, 部署后不再调整位置。仿真时间为 2500s。在整个仿真过程中, 对每个传感器节点所消耗的能力以及节点存活数目进行记录。

图 5 比较了两种算法的节点平均能耗:

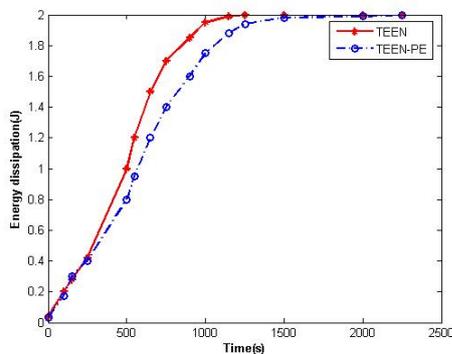


图 5 节点平均能耗对比图

从图中可以看出, 当网络运行到 1000s 时, TEEN 算法的节点平均能耗为 1.8J 左右, 而 TEEN-PE 算法的节点平均能耗小于 1.6J, 同时, TEEN 算法在网络运行到 1250s 时, 节点几乎用完了所有能量, 而 TEEN-PE 算法在运行到 2000s 左右才用完所有能量。所以, 在网络的整个运行时间内, TEEN-PE 算法节点的平均能耗要明显少于 TEEN 算法, 即 TEEN-PE 算法较 TEEN 算法有更优的节能效果。

图 6 比较了两种算法在网络运行过程中的节点存活率。

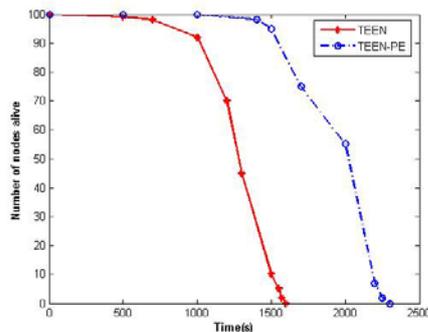


图 6 节点存活数

从图中可以看出, 在相同的运行时间内, 采用 TEEN-PE 算法存活的节点数目要多于采用 TEEN 算法。TEEN 算法中, 在网络运行超过 1500s 时, 大部分节点都已死亡, 而采用 TEEN-PE 算法, 运行到 2250s 左右, 大部分节点才死亡。所以, 采用 TEEN-PE 算法的节点存活率要明显高于 TEEN 算法。

5 结束语

本文针对事件监测型应用系统的特点, 在 TEEN 协议的基础上进行改进, 提出了一种基于事件驱动的分层路由算法—TEEN-PE。该算法通过先分区再建簇的方式解决了 TEEN 协议中最优簇首个数难以确定以及簇首分布不均匀问题。同时该算法在成簇和簇间通信阶段, 综合考虑节点剩余能量以及到基站的距离因素, 因此能有效平衡网络中各节点的能量消耗, 延长网络的生命周期。仿真结果表明, TEEN-PE 算法在节点能量消耗及节点存活率上要明显优于 TEEN 协议。

参考文献

- 1 Akyildiz IF, Weilian S, Sankarasubramaniam Y, Cayirci E. A survey on sensor. Communications Magazine. IEEE, 2002, 40(8):102-114.
- 2 任丰原, 黄海宁, 林闯. 无线传感器网络. 软件学报, 2003, 14(7):1282-1291.
- 3 Tseng YC, Pan MS, Tsai YY. Wireless sensor networks for emergency navigation. Computer, 2006, 39(7):52-62.
- 4 Younis O, Fahmy S. Distributed clustering in ad-hoc sensor networks: a hybrid, energy-efficient approach. INFOCOM 2004. 23th Annual Joint Conference of the IEEE Computer and Communications Societies. West Lafayette, 2004.
- 5 Heinzelman WR, Chandrakasan A, Balakrishnan H. Energy-Efficient communication protocol for wireless microsensor networks. Proc. of the Hawaii Int'l Conf. on System Sciences. San Francisco: IEEE Computer Society, 2000. 3005-3014.
- 6 Manjeshwar A, Dharma P. TEEN: A routing protocol for enhanced efficiency in wireless sensor networks. 1st International Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing. 2001.
- 7 Heinzelman WB, Chandrakasan AP, Balakrishnan H. An application-specific protocol architecture for wireless microsensor networks. IEEE Trans. on Wireless Communications, 2002, 1(4):660-670.
- 8 孙利民, 叶驰, 廖勇. 传感器网络的路由机制. 计算机科学, 2004, 31(3):54-57.