

# 一种基于语义的无线传感器网络路由算法

张婷婷, 周鸣争

(安徽工程科技学院 计算机科学与工程系, 安徽 芜湖 241000)

**摘要:**在无线传感器网络中,为了监测多个事件,监测区域可能需要布置多种不同类别的传感器节点以监测不同的事件属性,通过对每个传感器节点感知属性信息的融合处理,来判别事件是否发生。如何快速地将生成的事件路由到 sink 节点是个重要的研究课题。文中将传统的路由算法与分布式事件存储和语义思想相结合,提出了一种基于语义的无线传感器网络路由算法。在该算法中,一个传感器节点只需要关心自己通信范围内的邻居节点,不需要知道整个网络的状况,算法具有冗余数据少,节能性好,网络生存周期比较理想,实现简单等特点。该算法也可以为无线传感器网络的时空查询算法的研究提供一定基础。

**关键词:**无线传感器网络;事件;语义;路由;事件存储节点

中图分类号:TP391

文献标识码:A

文章编号:1673-629X(2007)12-0103-04

## A Routing Algorithm in Wireless Sensor Networks Based on Semantic Idea

ZHANG Ting-ting, ZHOU Ming-zheng

(Dept. of Computer Science and Engineering, Anhui University of Science and Technology, Wuhu 241000, China)

**Abstract:** In wireless sensor network, for inspecting many kinds of incident, inspecting area would be disposed sensor nodes of different sorts in order to inspect different observational attributes. Judge whether the incident occurs through fusion and processing of the attribute information every sensor node inspects. How to route the incident information created to sink node quickly is an important research problem. In this paper, presents a routing algorithm based on semantic idea though combining traditional routing algorithm with distributed incident storage and semantic idea. In the algorithm, every sensor node only concerns those nodes in its own communicating range instead of knowing all the nodes in WSN. The features of this algorithm are less redundant data, lower query consumption, longer life time and easier to implement. The algorithm can offer some useful reference for the research of spatio-temporal query algorithm.

**Key words:** wireless sensor networks; incident; semantic; routing; incident storage node

### 0 引言

近年来,无线传感器网络(Wireless Sensor Networks, WSN)技术得到了飞速发展。传感器网络是由一组传感器以 Ad Hoc 方式构成的有线或无线网络,其目的是协作地感知、采集和处理网络覆盖的地理区域中感知对象的信息,并发布给观察者<sup>[1]</sup>。观察者感兴趣的是特定的时间特定的区域某事件的发生情况,而不是传感器本身或者大量无关的观察数据<sup>[2]</sup>。比如,用户希望监测节点路由这样的信息:今天 15:30 区域

A 发生霜冻。但是,用户绝不希望监测节点路由这样的信息:ID 为 30 的传感器监测的温度是 20℃。因此,怎样高效路由用户关心的事件信息是人们面临的一个非常紧迫的挑战。

目前已提出的无线传感器网络路由算法主要分为两大类:一类为平面路由算法,如 SPIN 算法<sup>[3]</sup>和 Directed Diffusion 算法<sup>[4]</sup>;另一类为层次路由算法,如 LEACH 算法<sup>[5]</sup>和 TEEN 算法<sup>[6]</sup>。这两类路由算法各有各的优缺点,但是都没有引入语义的思想。

文中将传统路由算法与分布式事件数据存储及语义思想相结合,提出一种基于语义的无线传感器网络路由算法。

### 1 事件的存储

#### 1.1 相关定义

(1) 观察属性。

收稿日期:2007-03-15

基金项目:国家自然科学基金(60574028);安徽省自然科学基金重点项目(2007kj036zd)

作者简介:张婷婷(1983-),女,安徽芜湖人,硕士研究生,研究方向为无线传感器与计算机网络;周鸣争,教授,研究方向为信息融合、无线传感器网络、智能控制与计算机应用。

观察属性是指传感器监测的某种指标,例如,温度、压强、湿度等,记为  $A_i$ 。一个传感器节点可以监测的观察属性由它自身的传感器模块决定,因此在设计时就已经被指定,用户不能更改。

文中只考虑一个传感器节点负责监测一个观察属性情况。

### (2) 属性值。

属性值是某个传感器节点在特定时刻监测到自己观察属性的具体值,例如,一观察属性为温度的传感器节点在  $T$  时刻的属性值为  $15^{\circ}\text{C}$ ,记为  $AV_i$ 。文中,一个传感器节点只能有一种类型的属性值。

### (3) 事件类型。

事件类型是指观察者感兴趣的事情。例如,海啸、霜冻、台风、暴雨等,记为  $ET_i$ 。每种事件类型,都由特定的若干观察属性构成,可以由  $A_i$  表示,即  $ET_i$  可表示为  $(A_1, A_2, \dots, A_k)$ ,其中  $(A_1, A_2, \dots, A_k)$  为  $ET_i$  的观察属性。每个传感器节点监测一个观察属性,并且知道自己负责监测的观察属性属于哪个事件类型,即每个传感器节点对应一个要监测的事件类型,这在设计时就被指定,用户不能更改。

### (4) 事件。

事件是事件类型在特定时间特定地点的一个实例,记为  $E_i$ 。

### (5) 阈值。

设事件类型  $ET_i$  由若干观察属性  $A_1, A_2, \dots, A_k$  构成,当某一时刻属性值  $AV_1, AV_2, \dots, AV_k$  均达到各自事先设定好的一个值时,就可以认为此时事件  $E_i$  发生了。这里,称这些事先设定好的值为阈值,记为  $TV_i$ 。阈值是判断事件是否发生的重要条件,可以根据具体情况决定是在设计时设定好还是允许用户自己设定。

### (6) 事件存储节点(Node\_e)。

WSN 中需要路由的是事件的发生信息,而传感器节点只能监测观察属性,不能直接监测事件。因此为了捕获事件类型为  $ET_i$  的事件  $E_i$ ,需要从  $ET_i$  的观察节点  $A_{i1}, A_{i2}, \dots, A_{ik}$  中选择一个节点作为  $ET_i$  的事件存储节点,负责事件类型为  $ET_i$  的事件  $E_i$  的各属性值的融合处理与存储。

## 1.2 节点的数据结构模型

### (1) 监测节点。

每个监测节点都维护三张表:邻居列表(List\_neighbors)、路由表(List\_routing)和数据库表(List\_db)。邻居列表记录自己通信范围内的邻居节点,用于构建整个逻辑覆盖网络;路由表用于记录自己到对应事件存储节点的路径;数据库表记录监测节点标识(0)、各自资源关键字 RK 以及一些出厂设计标识,包

括观察属性  $A_i$ 、事件类型  $ET_i$  和阈值  $TV_i$ 。

### (2) 事件存储节点(Node\_e)。

事件存储节点由普通监测节点产生,主要维护四张表:邻居列表(List\_neighbors)记录自己通信范围内的邻居节点;数据库表(List\_db)记录事件存储节点标识(1)、各自的资源关键字 RK 以及一些出厂设计标识,包括观察属性  $A_i$ 、事件类型  $ET_i$  和阈值  $TV_i$ ;事件存储表(List\_es)记录已经合成的事件;sink 标志表(List\_flag\_s)标识自己是否为 sink 节点的事件存储节点,如果是则标识为 1,不是则标识为 0;sink 路由表(List\_routing\_s)记录该事件存储节点到 sink 节点的路径,目的是再有相同的事件存储节点要路由事件信息时不必重复查询就可以直接路由。当 List\_flag\_s 为 1 时,在事件产生阶段, List\_routing\_s 就记录下自己到 sink 节点的路径,当 List\_flag\_s 为 0 时,在语义路由阶段, List\_routing\_s 才能记录下自己到 sink 节点的路径。

### (3) sink 节点。

网络中的 sink 节点维护两张表:路由表(List\_routing)和数据库表(List\_db)。路由表用于记录自己到对应事件存储节点的路径;数据库表和监测节点类似,记录 sink 节点标识(2)、各自资源关键字 RK 以及一些出厂设计标识,包括观察属性  $A_i$ 、事件类型  $ET_i$  和阈值  $TV_i$ 。

## 1.3 事件的产生

事件存储节点根据监测节点和 sink 节点发送来的监测数据包(packet)合成事件。

监测数据包格式见表 1。

表 1 监测数据包

Flag	$ET_i$	Area	$T$	$AT_i$	$AV_i$
------	--------	------	-----	--------	--------

Flag:节点类型标识。0 表示监测节点;1 表示事件存储节点;2 表示 sink 节点。

$ET_i$ :监测的事件类型,也是数据包发送到目的事件存储节点的依据。

$AT_i$ :监测的观察属性。

$AV_i$ :实际监测到的  $AT_i$  值。

Area:节点所在的监测区域。

$T$ :监测的时刻。

监测数据包的发送条件为:当监测节点监测的数据达到自己的阈值时( $AV_i = TV_i$ ),就向事件存储节点发送监测数据包。

数据包根据自己的  $ET_i$  传送到对应的事件存储节点,同时将传送路径记录到自己的路由表中。事件存储节点在  $T$  时刻收到若干事件类型为  $ET_i$  的监测数据

包,丢弃其中 Flag = 1 的数据包,检查其中有无 Flag = 2 的数据包,如果有则标识 List\_flag\_s 为 1,没有则标识 List\_flag\_s 为 0,然后根据合成算法判断事件  $E_i$  是否发生,合成算法伪代码如下:

```
While(not equal(packetx, NULL))
{Search((packetx.Area, Areai) && (packetx.T, Tj))
Insert(List1[Areai][Tj], packetx.ATi) //将 Area 与 T 项相同的数据包中的观察属性项 ATi 合并到二维表 List1 中的同一行,此时 AT 项可能会有重复
While(not equal(List1[Areai][Tj], NULL))
{Search((List1[Areai][Tj], AT1) && (List1[Areai][Tj], AT2) && ... && (List1[Areai][Tj], ATk))
Insert(List2[Areai][Tx], Tj) && (List2[Areai][Tj], Areai) //若表 List1 中 Area 行 T 列中包括 AT1, AT2, ..., ATk, 则某时某地事件类型为 ETi(AT1, AT2, ..., ATk) 的事件发生,在表 List2 中记录此次事件发生的区域和时刻
```

1.4 事件的存储方式

已合成的事件被就地存放到事件存储节点,采用以数据为中心的中心映射存储策略<sup>[2]</sup>,即如果观察节点均匀分布在观察区域内,指定距离网络中心最近的节点作为事件存储节点,可使传输事件所消耗的能量最少。

设监测区域的面积为  $(M \times M)m$ ,如果此区域需要监测的事件类型为  $N$  种,则将区域  $(M \times M)m$  在逻辑上均匀划分为  $N$  块,每块的面积为  $(M \times M/N)m$ ,每逻辑块中寻找离区域中心最近的节点作为一种事件类型的事件存储节点,总共需要  $N$  个这样的事件存储节点,每个节点对应一种事件类型。

事件存储节点根据 List2 生成事件存储表,事件存储表的格式如表 2 所示。

表中第一列为事件存储节点标识 1;第二列为已经合成的事件所属的事件类型,对于每个事件存储节点来说都是唯一的;表中第三列记录事件产生的位置;第四列记录一特定位置事件发生的时刻。

表 2 事件存储表

1	$E_i$	Area I	$T_1$
			$T_2$
			.....
		Area II	$T_1$
			$T_2$
			.....
.....	.....		
	.....		
	.....		

2 基于事件的语义路由算法

2.1 语义表示

语义路由就是利用路由关键字与节点的映射关

系,将路由请求转发给一部分被认为最有可能满足请求的节点<sup>[7]</sup>。在文中,要建立路由关键字与 sink 节点索引位置的对应关系,就可以通过路由关键字将路由数据包发送到 sink 节点,完成网络中的一次事件路由过程。

确定合适的资源关键字(路由关键字 ROUK)对事件路由起着至关重要的作用。对于 1.4 中的存储方式,每个监测节点资源关键字 RK 包括 4 项(关键字 1——0(Flag),关键字 2—— $ET_i$ ,关键字 3——Area,关键字 4—— $T$ ),存储在各自的数据库表中,每个事件存储节点资源关键字 RK 包括 2 项(关键字 1——1(Flag),关键字 2—— $ET_i$ ),存储在各自的数据库表中,sink 节点资源关键字 RK 包括 4 项(关键字 1——2(Flag),关键字 2—— $ET_i$ ,关键字 3——Area,关键字 4—— $T$ ),存储在各自的数据库表中。路由关键字 ROUK 采用与资源关键字 RK 相同的语义表示形式  $(2, ET_i, Area, T)$ 。这样,路由关键字就与索引位置(节点位置)产生了对应关系。

2.2 路由算法

事件存储节点通过发送路由数据包转发事件信息到 sink 节点。路由数据包(packet\_rou)格式见表 3。

表 3 路由数据包

ROUK	Visited-Neighbors	Life-Time
------	-------------------	-----------

ROUK 为路由关键字  $(2, ET_i, Area, T)$ ,其中 2 表示查询目的地为 sink 节点。Visited-Neighbors 集合为路由数据包要转发的下一跳节点的集合,指导路由数据包到 sink 节点的一次路由。Life-Time 为路由数据包的存活时间。

设一区域中网络要监测的事件类型有  $n$  种。事件类型 1 语义表示为  $ET_1$ ,被映射到观察区域 1,事件类型 2 语义表示为  $ET_2$ ,被映射到观察区域 2,.....,事件类型  $i$  语义表示为  $ET_i$ ,被映射到观察区域  $i$ ,.....事件类型  $n$  语义表示为  $ET_n$ ,被映射到观察区域  $n$ 。若某一时刻要向 sink 节点报告  $T_x$  时刻区域  $A_x$  事件类型为  $ET_x$  的事件的发生情况,则路由关键字 ROUK 为  $(2, ET_x, A_x, T_x)$ ,路由算法描述如下:

(1) 整个网络逻辑上均匀分为  $n$  块区域,计算出每块区域中距离网络中心最近的顶点坐标  $\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$ ,分别作为事件类型为  $ET_1, ET_2, \dots, ET_n$  的事件存储节点。

(2) 需要路由事件信息的事件存储节点 Node\_e\_x 首先查询自己的 List\_flag\_s\_x,当 List\_flag\_s\_x 为 1 时,则直接将 List\_routing\_s\_x 的内容添加到 Visited-Neighbors 中,packet\_rou 按照 Visited-Neighbors 路由

到 sink 节点。

(3) 如果  $List\_flag\_s\_x$  为 0, 则  $Node\_e\_x$  查找  $List\_neighbors\_x$  中的邻居节点, 找到与自己  $List\_db\_x$  中  $RK\_ETi\_x$  不同的节点 A, 作为  $packet\_rou$  第一次发送的目的地, 将 A 节点保存到  $Visited\_Neighbors$  中, 若存在多个满足条件的节点, 则随机选择一个。

(4) 按照  $List\_routing\_A$  中的路径找到对应的事件存储节点  $Node\_e\_A$ , 同时将路径添加到  $Visited\_Neighbors$  中。查看  $Node\_e\_A$  的  $List\_flag\_s\_A$ , 若值为 1, 则将  $List\_routing\_s\_A$  添加到  $Visited\_Neighbors$  中, 同时将  $packet\_rou$  按照  $List\_routing\_s\_A$  路由到 sink 节点, 最后将  $Visited\_Neighbors$  的内容保存到  $List\_routing\_s\_x$  中。下次若  $Node\_e\_x$  还要路由事件信息, 则可直接按照  $List\_routing\_s\_x$  找到 sink 节点, 不必重复查找。

(5) 如果  $List\_flag\_s\_A$  的值为 0, 则清除  $Visited\_Neighbors$ 。在  $List\_neighbors\_x$  中的邻居节点里, 找到与自己  $RK\_ETi\_x$  和  $RK\_ETi\_A$  不同的节点 B, 作为  $packet\_rou$  第一次发送的目的地, 将 B 节点保存到  $Visited\_Neighbors$  中, 若存在多个满足条件的节点, 则随机选择一个。重复过程(4)。

(6) 若  $List\_neighbors\_x$  中所有的邻居节点都不满足条件, 则  $Node\_e\_x$  随机查找自己两跳范围内  $RK\_ETi$  不同的节点 C, 重复过程(4)。

(7) 查询过程中如果  $Life\_Time$  减至 NULL 则返回查询失败。

### 3 试验检验

试验时, 假定数据长度  $k = 1000\text{bit}$ 。当监测区域为  $50\text{m} \times 50\text{m}$ , 节点数为  $50 \sim 800$  时, 实验结果如图 1、图 2 所示。结果表明, 和传统的无线传感器网络路由算法相比, 基于语义的无线传感器网络路由算法的节能性大大优化, 网络生存时间更理想。

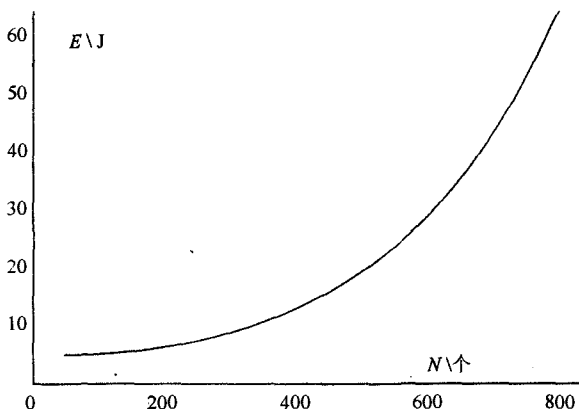


图 1 Flooding 算法的能量消耗曲线

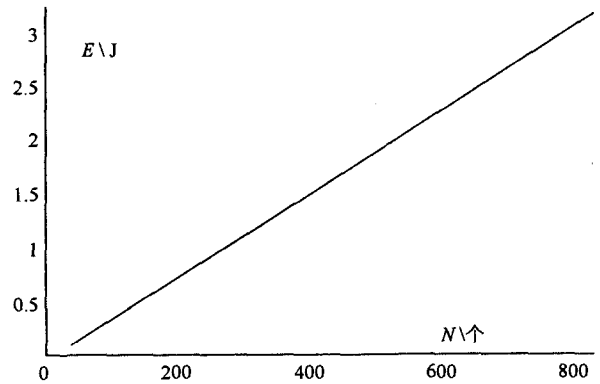


图 2 文中算法的能量消耗曲线

### 4 结论

提出一种基于语义的无线传感器网络路由算法, 这种路由策略在传统的 WSN 路由算法的基础上引入了分布式事件存储和语义的思想, 在需要监测多种事件类型且每种事件类型包括多个观察属性的无线传感器网络环境下特别适用。这种策略具有节能性能好、数据冗余小、网络生存时间长等优点, 且可以为 WSN 的时空查询算法的研究提供一定基础。但是该算法是在假设节点非移动的理想情况下提出的, 当网络中节点开始死亡或者节点大规模移动时事件信息的路由就会遇到一定困难, 这是今后要研究的重点问题。

#### 参考文献:

- [1] Li J Z, Li J B, Shi S F. Concepts, issues and advance of sensor networks and data management of sensor networks[J]. Journal of Software, 2003, 14(10): 1717-1727.
- [2] 郭龙江, 李建中, 李贵林. 无线传感器网络环境下时-空查询处理方法[J]. 软件学报, 2006, 7(4): 794-805.
- [3] Kulik J, Heinzelman W R, Balakrishnan H. Negotiation based protocols for disseminating information in wireless sensor networks[J]. Wireless Networks, 2002, 8(2-3): 169-185.
- [4] Intanagonwivat C, Govindan R, Estrin D, et al. Directed diffusion for wireless sensor networking [J]. IEEE/ACM Trans. on Networking, 2003, 11(1): 2-16.
- [5] Heinzelman W, Chandrakasan A, Balakrishnan H. Energy-efficient communication protocol for wireless microsensor networks[C]// In: Proc. of the 33rd Annual Hawaii Int'l Conf. on System Sciences. Maui: IEEE Computer Society, 2000: 3005-3014.
- [6] Manjeshwar A, Agrawal D P. TEEN: A protocol for enhanced efficiency in wireless sensor networks[J]. In: Int'l Proc. of the 15th Parallel and Distributed Processing Symp. San Francisco: IEEE Computer Society, 2001: 2009-2015.
- [7] 叶春, 葛燧和. 基于语义路由的 P2P 信息检索[J]. 计算机仿真, 2004, 21(10): 143-145.