

一种基于树的无线传感器网络数据收集方法

李宏 于宏毅 刘阿娜

(信息工程大学信息工程学院 郑州 450002)

摘要: 该文提出一种基于树的无线传感器网络数据收集方法, 它将查询请求的传递与数据回传结构的建立有机结合起来, 通过使用一种洪泛避免的方法传递查询请求, 建立起一棵以 sink 节点为根的、包含最少中间节点的查询转发树, 作为数据回传结构。理论分析和仿真试验表明, 该方法不会降低无线传感器网络的连通性, 可以有效节省能量。

关键词: 无线传感器网络; 数据收集; 数据聚合; 路由

中图分类号: TP393

文献标识码: A

文章编号: 1009-5896(2007)07-1633-05

A Tree Based Data Collection Scheme for Wireless Sensor Network

Li Hong Yu Hong-yi Liu A-na

(Department of Communication Engineering, Information Engineering University, Zhengzhou 450002, China)

Abstract: Combining the delivery of query messages with the setting up of data transmitting structure, a novel distributed data collection scheme for wireless sensor network—TBDCS (Tree Based Data Collection Scheme) is proposed in this paper. Using a flooding avoidance method, TBDCS sets up a tree with minimum intermediate nodes, which are also data aggregators when sensor nodes send data back. Theoretical analysis proves that TBDCS changes neither the network connectivity nor the shortest path's length between the sink and any other sensor nodes. Simulations show it significantly reduces the traffic and achieves longer system lifetime.

Key words: Wireless sensor network, Data collection, Data aggregation, Routing

1 引言

无线传感器网络由于具有部署灵活及信息获取精度高、可靠性强、经济性好等特点, 在军事侦察、环境监测、智能建筑、交通运输等领域具有广阔的应用前景, 是目前信息技术领域的一个热点研究方向^[1]。无线传感器网络中的数据收集需要通过节点间的有机协作完成, 由于传感器节点由电池供电, 仅具有有限的计算、通信及能量等资源, 所以数据收集需要解决的一个重要问题是如何节省能量, 以延长无线传感器网络的生命周期。考虑到在无线传感器网络中, 通信的代价比计算的代价高几个量级^[2], 所以数据收集过程中常采用数据聚合技术^[3,4], 尽量减少传感器网络中的总传输次数。

数据聚合技术, 即节点利用本地处理能力, 先对采集到的或接收到的其它传感器节点发送的数据进行网内处理, 消除冗余信息, 然后再传输处理后的数据^[5]。由于多个传感器节点对同一事件采集到的数据是相同或相近的, 并且数据回传存在一定的重复路径, 所以使用数据聚合技术可以有效节省传感器网络的能量。选择何种结构回传采集的数据, 才能使总的分组回传次数最少呢? Estrin等人提出^[6], 在一次数据回传中, 最优的数据聚合结构为包含所有源传感器节点及

sink节点的最小Steiner树, 当网络拓扑任意, sink节点的位置任意时, 求解上述问题为一个NP问题。文献[5]中给出了3种典型的构造数据聚合树的近似优化算法, 其主要思想分别为: (1)CNS(Center at Nearest Source)算法: 与某次数据采集相关的所有源传感器节点, 首先都把采集到的信息传送到离sink节点最近的传感器节点, 由该传感器节点完成数据聚合并最终传递至sink节点; (2)SPT(Shortest Path Tree)算法: 所有源传感器节点均沿自己至sink节点的最短路径独立传输数据分组, 在各个路径交叉点进行数据聚合, 形成一棵以sink节点为根的数据聚合树; (3)GIT(Greedy Incremental Tree)算法: sink节点首先建立至与之距离最近的传感器节点的一条最短路径, 然后依次将下一个最近的传感器节点加入当前树。但是上述3种算法在实际使用时, 当源传感器节点数量巨大且分布范围广时, CNS算法的节能效果远远小于GIT和SPT算法, 不再适用^[6]; GIT算法的性能虽然较好, 但实现时需要全局信息, 难于分布式实现; SPT算法虽然比较容易进行分布式实现, 但是由于各个传感器节点独立选择数据回传路径, 路径交叉点是随机形成的, 所以可进一步优化以提高数据聚合的概率。同时, 现有的无线传感器网络路由协议在建立路由时, 大多数在选路时没有考虑对数据聚合的支持, 如文献[3, 6]等, 有的还使用洪泛转发建路请求消息^[3], 这将导致广播风暴^[7], 极大消耗传感器网络的能量。由于一般情况下无线传感器网络中的数据收集包括查询请求的传

2005-10-19 收到, 2006-04-10 改回

国家自然科学基金(60472064)和河南省自然科学基金(0511010700)资助课题

递及采集数据的回传两个过程,前者将查询请求从一个sink节点传送到所有传感器节点,通信模式为一到多,后者将多个源传感器节点采集的信息回传至sink节点,通信模式为多到一,两者路径正好相反,因此本文把数据采集过程中的路由选择和数据聚合结构的建立有机结合起来,提出了一种新的无线传感器网路数据收集方法TBDCS(Tree Based Data Collection Scheme)。TBDCS在查询请求的传递过程中,选择能量充足且邻接点多的最少数目查询转发点,构成一棵以sink节点为根的查询转发树,并把这棵树作为数据回传结构。由于选择尽可能少的中间节点进行查询转发,不仅可以避免洪泛带来的广播风暴,有效减少查询请求的转发次数,而且沿所形成的树回传数据时,具有很高的数据聚合概率。

理论分析和仿真试验表明:本文设计的TBDCS数据采集方法,充分利用了节点的本地信息,在查询请求的传递过程中选择尽可能少的中间转发节点建立数据回传结构,可以有效节省无线传感器网络的能量,并且不降低网络的连通性。

2 网络模型

设 $G=(V, E)$ 表示一个连通的无线传感器网络, V 为网络中所有节点(这里暂不考虑节点的休眠)的集合, E 为所有能够直接通信的节点之间的连线。TBDCS 研究的问题是:在网络中仅有一个 sink 节点的情况下,当 sink 节点发起一次查询请求时,多个随机分布的源传感器节点如何以多跳方式向 sink 节点回传采集到的数据。假设每个传感器节点均已到达 sink 节点的最小跳数,一跳邻节点及其能量等信息,且相邻传感器节点间交互邻节点信息,从而使每个节点均了解自己的两跳邻节点信息。中间节点进行数据聚合时,使用简单的数据聚合函数,如重复分组压缩、求最大值及最小值等,将多个输入分组聚合为一个输出分组。

3 TBDCS 原理

3.1 转发节点集合的定义

设 x 为传感器网络中的某个节点, x 距离 sink 节点的最小跳数为 $\text{hop}(x)$, 令

$$\left. \begin{aligned} A^{(1)}(x) &= \{i \mid i \text{ 是 } x \text{ 的一跳邻节点且 } \text{hop}(i) = \text{hop}(x) + 1\} \\ A^{(2)}(x) &= \{i \mid i \text{ 是 } x \text{ 的两跳邻节点且 } \text{hop}(i) = \text{hop}(x) + 2\} \end{aligned} \right\} \quad (1)$$

定义转发节点集合 $\text{FNS}(x)$ 为集合 $A^{(1)}(x)$ 的一个最小子集,使得

$$A^{(2)}(x) = \bigcup_{y \in \text{FNS}(x)} N(y) \quad (2)$$

其中 $N(y) = \{z \mid y \in A^{(1)}(x), e(y, z) \in E, \text{hop}(z) = \text{hop}(x) + 2\}$ 。

图 1 为 FNS 的示例,节点 0 为 sink 节点, sink 节点的一跳邻节点集合 $A^{(1)} = \{1, 2, 3, 4, 5, 6, 7\}$, 两跳邻节点集合 $A^{(2)} = \{8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 20\}$, 转发节点集合 $\text{FNS} = \{1, 2, 4, 6, 7\}$ 。节点 6 的一跳邻节点集合 $A^{(1)} = \{17, 18, 20\}$, 两跳邻节点集合 $A^{(2)} = \{21, 22, 33, 32, 31,$

$30\}$, 转发节点集合 $\text{FNS} = \{18, 20\}$ 。

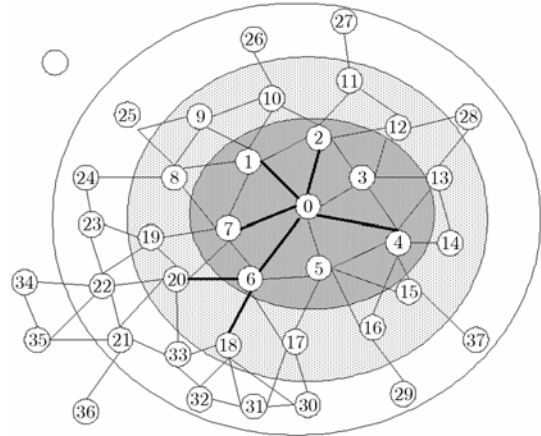


图 1 FNS 示例

3.2 求解节点 FNS 的算法

该问题可以归结为求解顶点覆盖问题,是一个已知的 NP 问题[8]。这里用贪婪算法求解该问题,依次在未被挑选的 $A^{(1)}$ 集合节点中,选择一个其一跳邻节点覆盖 $A^{(2)}$ 中节点最多的节点。同时,考虑到无线传感器网络中的节点能量有限, FNS 中节点的选择还要考虑能量的均衡,选择当前剩余能量大于某个门限值(设为 $E_{\text{threshold}}$)的节点作为 FNS 中的节点。设 E_x 表示节点 x 的当前剩余能量,某节点 x 求其 FNS 的算法 Find-FNS 如下所示。

算法 Find-FNS ($x, A^{(1)}, A^{(2)}$)

输入

x : x 为无线传感器网络中的某个节点,其距离 sink 的跳数为 $\text{hop}(x)$

$A^{(1)}$: 节点 x 的一跳邻节点集合,其中节点距离 sink 的跳数为 $\text{hop}(x) + 1$

$A^{(2)}$: 节点 x 的两跳邻节点集合,其中节点距离 sink 的跳数为 $\text{hop}(x) + 2$

输出 节点 x 的 FNS

Begin

FNS = {};

While $A^{(2)}$ 非空{

从集合 $A^{(1)}$ 中挑选节点 y ;

If $|N(y) \cap A^{(2)}|$ 为最大值且 $E_x \geq E_{\text{threshold}}$

Then

$A^{(2)} = A^{(2)} - N(y)$;

FNS = FNS + { y };

$A^{(1)} = A^{(1)} - \{y\}$;

EndIf

} End While

Return FNS;

End

需要说明的是:集合{从 $A^{(1)}$ 中每个节点出发,可以到达的,至 sink 节点的跳数为 $\text{hop}(x) + 2$ 的节点}与集合 $A^{(2)}$ 相等。因为节点 x 至 sink 节点的跳数为 $\text{hop}(x)$,所以节点 x 的一跳邻

节点至sink节点的跳数小于等于 $\text{hop}(x)+1$,从而节点 x 的两跳邻节点至sink节点的跳数小于等于 $\text{hop}(x)+2$,只有通过跳数为 $\text{hop}(x)+1$ 的点,才能到达跳数为 $\text{hop}(x)+2$ 的点,所以本算法是正确的。

3.3 查询请求的转发及 FNS 树的建立过程

在 TBDCS 中,当用户欲通过无线传感器网络获取信息时,sink 节点将广播查询请求 QREQ(Query REQuest)消息。sink 节点在发送前先调用 Find-FNS 算法计算自己的 FNS,生成 FNS 节点列表,其中较早选出的、具有较多一跳邻节点的节点排列在较晚选出的、具有较少一跳邻节点的节点前边,然后广播带有该 FNS 节点列表的 QREQ 消息。当 sink 节点的某个一跳邻节点收到 QREQ 消息后,将会执行下文所示的 FNS_Forward 算法。如果它是 sink 节点的 FNS 节点,则计算自己的 FNS 集合,并发送带有新 FNS 节点列表的 QREQ 消息。如此循环,将得到离 sink 节点远近不同的一系列 FNS 集合,构成一棵以 sink 节点为根的 FNS 树。

算法 FNS_Forward (ID_X, QREQ)

输入

ID_X: 节点 X 的标识符

QREQ: 查询请求分组

输出 QREQ 分组是否成功转发

Begin

If QREQ 为重复查询请求 Then

Return FALSE;

记录至 sink 节点的上一跳为 QREQ 分组的发送节点;

If ID_X 不在 QREQ 的 FNS 列表中 Then

Return FALSE;

Else

Call Find-FNS(x);

用自己的 FNS 列表,替换 QREQ 中的 FNS

列表;

根据 ID_X 在 QREQ 的 FNS 列表中的位置,

延迟一个成正比的时间;

广播新生成的 QREQ 消息;

EndIf

Return TRUE;

End

需要说明的是:如果节点 x 收到不同上一跳节点转发的同一查询分组,则视为重复包,这里选择最早到达的 QREQ 消息发送节点为自己的上一跳 FNS 节点。广播新生成的 QREQ 消息前等待一段时间是为了避免传输冲突,使得一跳邻节点数目多的节点较早发送 QREQ 消息。另外,由 QREQ 消息的转发过程可知,每个传感器节点均知道自己在 FNS 树中的父节点,FNS 树中的每个节点也均知道自己在 FNS 树中的子节点。

3.4 数据的回传过程

当源传感器节点采集到与查询相关的数据时,如果不是 FNS 节点,则向自己至 sink 节点路径上的上一跳节点,即 FNS 树中的某个节点回传数据,否则先等待并选择合适的时机进行数据聚合。数据聚合时机的选择应使尽可能多的相关数据进行聚合,同时也要考虑时延的要求,需要在数据聚合和时延之间进行折衷[9, 10]。由于 FNS 树中的每个节点均知道自己有哪些子 FNS 节点,所以可以选择当节点收到自己所有子 FNS 节点的数据后执行相应的聚合操作,同时为了防止部分子 FNS 节点不参与该次数据采集或转发,可以通过分析已经接收到数据包的时间优先级要求、或周期性监测应用中的历史信息等,决定执行聚合操作前的最大等待时间,超时则结束等待,对已经收到的数据分组执行应用所要求的数据聚合处理,然后将聚合后的数据发送至自己在 FNS 树中的上一跳节点,如此循环,最后将聚合后的数据回传至 sink 节点。

3.5 FNS 树的维护

采用软状态的方法维护 FNS 树,每个 FNS 节点均有一定的有效期,超时自动失效,这种方法有助于均衡地消耗网络中节点的能量。如果在周期性数据采集中使用 TBDCS,则需要 sink 节点周期性发送 QREQ 消息,重建 FNS 树。QREQ 消息的刷新周期选择需要考虑应用对时延的要求、均衡消耗节点的能量等因素。另外,由于每个节点可能收到一个或多个节点转发的 QREQ 消息,所以节点可以保存多个上一跳节点的信息,在当前上一跳节点失效时进行聚合树的局部修复。

4 对 TBDCS 的分析

4.1 理论分析

定理 1 设 $d(x)$ 表示节点 x 与 sink 节点之间的最小跳数, $dF_{d(x)}(x)$ 表示仅使用 FNS 树中的节点作为转发节点时,节点 x 与 sink 节点之间的最小跳数,则 $d(x) = dF_{d(x)}(x)$ 。

证明 设 $S_FNS(i)$ 表示离 sink 节点 i 跳的所有转发节点集合,则:

(1) 当 $d(x)=0$ 时,节点 x 为 sink 节点, $dF_0(x)=0$,则 $d(x) = dF_0(x)$;

(2) 当 $d(x)=1$ 时,节点 x 为 sink 节点的一跳邻节点, $dF_1(x)=1$,则 $d(x) = dF_1(x)$;

(3) 假设当 $d(x)=k$ 时, $d(x) = dF_k(x)$ 成立, k 为大于 1 的整数,当 $d(x)=k+1$ 时:

根据 Find-FNS 算法及上述假设, $\exists h \in S_FNS(k)$, 使 x 是 h 的一跳邻节点,则 $dF_{k+1}(x) = dF_k(h)+1$; 又由假设得, $dF_k(h)=k$, 则 $dF_{k+1}(x) = dF_k(h)+1 = k+1 = d(x)$, 所以, $dF_{k+1}(x) = d(x)$ 结论成立。

由(1),(2)及(3)可知, $d(x) = dF_{d(x)}(x)$ 。

证毕

定理 2 引入 FNS 集合,不会影响 sink 节点与其它传

传感器节点的连通性。

证明 假设 x 为任意节点, sink节点与 x 之间可达, 则一定存在一条最短路径, 设其长度为 $d(x)$, 由定理 1 得, $dF_{d(x)}=d(x)$, 即也存在一条仅使用FNS树中的节点作为转发节点的路径, 其长度为 $d(x)$ 。

证毕

4.2 能耗分析

本文使用网络中总的分组传输次数计算能量开销[3], 假设无线传感器网络中的节点总数为 N 。

(1) 发送 QREQ 消息所消耗的能量

在无线传感器网络中传递 QREQ 消息时, 如果使用洪泛, 则 QREQ 消息的总发送次数为 N , 如果使用 FNS 树, 根据算法 Find-FNS 可知, FNS 节点一般为邻接点较多的节点, 因此仅需通过少量节点就可以将 QREQ 消息转发至无线传感器网络中的节点。又根据算法 FNS_Forward, 一个节点如果被两个以上节点选做 FNS, 该节点只转发一次, 因此使用 FNS 可大大减少查询请求的总发送次数。

(2) 数据回传所消耗的能量

由定理 1 可知在 TBDCS 中, 每个源传感器节点实际均沿自己至 sink 节点的最小跳数路径回传采集的数据, 根据算法 Find-FNS, 仅有部分节点被选作数据转发点, 因而增加了各个源节点数据回传时最短路径交叉的概率, 所以在回传时的总聚合点数小于等于 SPT 算法的总聚合点数, 因为后者的聚合点是随机形成的。

4.3 空间开销分析

每个节点均需要存储自己一跳及两跳邻节点的信息, 可以对此进行优化, 仅存储比自己离 sink 节点远的一跳及两跳邻节点的信息。在无线通信距离一定的情况下, 无线传感器网络中节点的密度越高, 节点需要的存储空间越大, 同时由 Find-FNS 算法得到的 FNS 集合中结点数也越多, 使 QREQ 消息中 FNS 列表的开销也越大。由于节点发送的每个分组一般有最大长度的限制, 所以 FNS 列表的长度也有限制, 即要求传感器网络节点的密度在一定的范围内。

4.4 仿真结果

本文在RS(Random Sources)模型^[4]下, 比较了TBDCS与典型数据聚合算法GIT, SPT及不采用数据聚合、以地址为中心方法AC(Address Centric)^[4]在数据回传时的性能。仿真基于 10 个归一化的单位场景, 每个场景中随机分布 99 个传感器节点, sink节点位于(0.45, 0.45)。传感器节点的无线通信半径 R 为 0.15~0.45, 源传感器节点个数 k 为 3~15。仿真在理想的通信环境下进行, 没有考虑传输冲突及分组丢失。仿真首先为上述 4 种算法建立数据回传路径, 然后进行了两组实验。第一组实验随机选取 9 个源传感器节点, 研究这些节点沿不同算法建立的路径回传数据时, 网络中总的数据传输次数随无线通信半径变化的情况; 第二组实验使无线通信半径为 0.3, 研究沿不同算法建立的路径回传数据时, 网络

中总的数据传输次数随源节点数目变化的情况。每组试验在不同场景下各运行了 100 次, 去除个别不可达节点, 取 10 个场景中总数据回传次数的平均值作为仿真结果, 衡量传感器网络的能量消耗。仿真结果如图 2 和图 3。

图 2 比较了当传感器节点的无线通信半径变化时, 不同算法在数据回传时的总传输次数, 反映网络中总的能量消耗。当源传感器节点离 sink 的距离较远时, GIT, SPT 和 TBDCS 比 AC 能够节省 34~45%的能量开销。图 3 显示当源传感器节点的数目增加时, 虽然总的传输开销增加, 但是当采用数据聚合技术时, 节省的能量也大大增加。两图中 TBDCS 的平均传输次数都是大于 GIT 且小于 SPT。

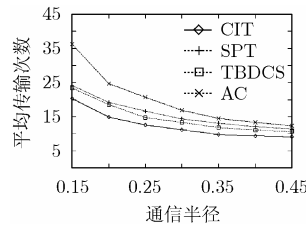


图 2 平均传输次数与通信半径的关系

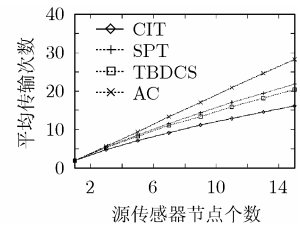


图 3 平均传输次数与源传感器节点个数的关系

需要说明的是, 虽然 GIT 算法的性能较好, 但是由于单个传感器节点能量及存储空间受限, 无法获得分布式实现 GIT 算法所需的全局信息, 如哪个节点离当前 GIT 树最近等, 而 TBDCS 是一种分布式方法, 仅需要传感器节点局部获取其一跳及两跳邻接点的信息。TBDCS 不仅比 SPT 的性能好, 而且由于每个传感器节点均知道自己至 sink 节点的上一跳 FNS 节点, FNS 树中的每个节点也均知道自己有哪些子 FNS 节点, 因而采用 2.4 节中的方法后, TBDCS 的回传时延也可以进行优化。

5 结束语

通过查询请求消息的传递, TBDCS 建立了一棵以 Sink 节点为根的、包含最少中间转发节点的 FNS 树, 作为数据回传结构, 避免了查询分组在网络中的洪泛, 同时增加了数据回传时数据聚合的概率, 有效节省了无线传感器网络的能量。TBDCS 由于仅利用传感器节点的局部信息, 所以容易分布式实现。另外, 由于每个数据聚合节点均了解自己所有的子节点, 所以可以为合理选择数据聚合时机提供支持。TBDCS 的不足之处是 FNS 节点的选择没有考虑数据的相关性, 当数据相关性较高且源节点的分布比较集中时, 不一定能保证选择最佳聚合路径, 另外, FNS 节点选择时使用的剩余能量门限值, 没有考虑如何根据网络的运行状况动态进行自适应调整, 今后都需要进一步深入研究。

参考文献

- [1] Akyildiz I, Su W, Sankarasubramaniam Y, and Cayirci E. Wireless sensor networks: A survey. *Computer Networks*, 2002, 38(4): 393-422.
- [2] Pottie G and Kaiser W. *Wireless sensor networks*.

- Communications of the ACM*, 2000, 5, 43(5): 51-58.
- [3] Intanagonwiwat C, Govindan R, and Estrin D. Directed diffusion: A scalable and robust communication paradigm for sensor networks. ACM/IEEE International Conference on Mobile Computing and Networks, Boston, 2000: 56-67.
- [4] Heidemann J, Silva F, Intanagonwiwat C, Govindan R, Estrin D, and Ganesan D. Building efficient wireless sensor networks with low-level naming. 18th ACM Symposium on Operating Systems Principles, New York, 2001: 146-159
- [5] Krishnamachari B, Estrin D, and Wicker S. Modeling data-centric routing in wireless sensor networks. USC Computer Engineering Technical Report CENG, 2002: 02-14.
- [6] Braginsky D and Estrin D. Rumour routing algorithm for sensor networks. 1th ACM International Workshop on Wireless Sensor Networks and Applications, Atlanta, 2002: 22-31.
- [7] Tseng Y C, Ni S Y, Chen Y S, and Sheu J P. The broadcast storm problem in a mobile ad hoc network. *Wireless Networks*, 2002, 8(2-3): 153-167.
- [8] Sheu Jang-Ping, Hung Pei-Kai, and Hsu Chih-Shun. Scheduling of broadcasts in multihop wireless networks. The handbook of ad hoc wireless networks, Boca Raton, FL, CRC Press, Inc., 2003.
- [9] Solis I and Obraczka K. The impact of timing in data aggregation for sensor networks. IEEE International Conference on Communications (ICC), Paris, 2004, 6: 3640-3645.
- [10] Yuan W, Krishnamurthy S V, and Tripathi S K. Synchronization of multiple levels of data fusion in wireless sensor networks. IEEE GLOBECOM, San Francisco, 2003, 1: 221-225.
- 李宏: 女, 1970年生, 博士生, 主要研究方向为 Ad hoc 网络、无线传感器网络的路由协议等。
- 于宏毅: 男, 1961年生, 教授, 博士生导师, 主要研究方向为信号处理、移动通信、Ad hoc 网络、无线传感器网络等。
- 刘阿娜: 女, 1976年生, 博士生, 主要研究方向为无线传感器网络的数据链路层协议。