

大规模无线传感器网络的路由协议研究

马祖长^{1,2} 孙怡宁¹

¹(中国科学院合肥智能机械研究所,合肥 230031)

²(中国科学技术大学精密仪器系,合肥 230026)

E-mail: zcma@sohu.com

摘要 对于大规模无线传感器网络的路由协议进行了研究,提出了基于最小跳数的路由协议。详细介绍了最小跳数场的建立过程,从理论上证明了每个节点只会收到一个导致最小代价变小的信息包,分析了节点随机延时对跳数场建立的影响和相应的处理措施。该协议可以在大规模网络内使用,节点只要记忆自己的转发节点集,就可以沿着最短路径向网关发送数据。分析和仿真实验都证实了设计的正确性。

关键词 无线传感器网络 路由 最小跳数

文章编号 1002-8331-(2004)11-0165-03 文献标识码 A 中图分类号 TN92

Research on Routing Protocol of a Large Wireless Sensors Network

Ma Zuchang^{1,2} Sun Yining¹

¹(Inst. of Intelligent Machines, Chinese Academy of Sciences, Hefei 230031)

²(Dept. of Precision Machinery and Instrumentation, University of Science and Technology of China, Hefei 230026)

Abstract: This paper researches on routing protocol in a large wireless sensors network and proposes a routing protocol based on minimum hop. It introduces the process of setting up minimum hop field in detail and proves that only one message diminishes the node's minimum hop, analyzes the impact of random delay on setup of minimum hop field and gives some measures. This protocol can be used in a large sensors network and ensure that data is forwarded along lowest path and message number is least. Simulations and analysis show the correctness of the design.

Keywords: wireless sensors network, routing, minimum hop

1 引言

无线传感器网络是监视远程环境的有力工具,随着微处理器和无线通信技术的发展,其重要性越来越突出。无线传感器网络具有很多独特的优点和应用领域,引起了人们的广泛关注^[1]。传感器节点的通信距离有限,一般在10~100米范围内。节点只能与其射频覆盖范围内的邻居直接通信,如果希望访问射频覆盖范围外的节点,必须采用多跳路由实现。传感器节点通常由电池供电,高效使用有限的电池资源,尽量延长节点的生命周期是任何路由协议的首要考虑因素。同时由于传感器网络节点的计算能力、存储空间都十分有限,所设计的路由协议应该简单有效,使得节点的运算开销尽量的小。

无线传感器网络的路由协议是一个非常活跃的研究领域,在这方面国外学者提出了很多的解决方案。大致可以分为两类:平面路由协议和分层的路由协议^[2]。平面路由协议需要维持路由表,在大规模网络中,这样的路由表维持需要消耗大量的存储空间,同时发送信息中所包含的路由信息也会引起通信负担的加重。采用分层的网络可能在一定程度上缓解这个问题^[3],但是,分层网络中群头节点是瓶颈,如果群内节点数量众多,群头节点的电池很快就会耗尽,导致网络需要重新组织。上述协议都不是针对大范围无线传感器网络设计的^[3]。

文献[3]提出了一种基于代价场的路由协议,可以克服上述

不足。但是采用该算法进行信息路由,会在网络内引起很多冗余的信息。所以作者提出了基于最小跳数的无线传感器网络自组织与路由协议。

采用基于最小跳数的路由协议,可以保证任何节点发出的信息都沿着最优(所经过的中间节点最少)的路径向网关发送,并且在网络内所引起的信息包数量是最少的。所有节点只要记忆自己的转发节点集和最小跳数,就可以实现信息路由。该协议可以在任何规模的网络内使用,并且对于移动节点也可以很简单地实现信息转发。该协议实现简单,可以在任何规模的网络内应用。

为了验证算法的可靠性,作者采用OMNET++作为仿真工具,进行了一系列仿真实验,证明了该算法的正确性。

2 OMNET++简介^[4]

OMNET++是布达佩斯技术大学电信学院开发的面向对象的分布事件仿真工具,OMNET++是Objective Modular Network TestBed in C++的英文缩写。它可用于通信协议、计算机网络多处理器和分布式系统、管理系统等的建模和仿真。

一个OMNET++模型(Model)由多层嵌套的模块(Module)构成,模块的嵌套深度没有限制,这样用户可以将实际系统的逻辑结构影射到模型结构。模块之间通过消息(Message)传送

进行通信,消息可以直接向目的模块发送,也可以通过通道(Gate)和连接(Connection)发送。

模块包括简单模块和复合模块。其中简单模块由用户编程实现,它们反映了系统的算法。在仿真执行过程中,采用多线程技术,这些简单模块并行执行。在特定的网络中,简单模块和复合模块都是模块类型的实例。在描述模型时,用户定义模块类型,并使用它们定义更复杂的模块类型,最后,用户产生系统模块,作为预定义模块类型的实例。网络中所有的模块都是系统模块的子模块。

OMNET++的主要特色是它的模型拓扑描述语言 NED,采用它可以完成一个网络模型描述。网络描述包括下列组件:输入申明、信道定义、系统模块定义、简单模块和复合模块定义。使用 NED 描述网络,产生 NED 文件,该文件不能被 C++ 编译器使用,需要首先采用 OMNET++ 提供的编译工具 NEDC 将 NED 文件编译成 .cpp 文件。最后,使用 C++ 编译器将这些文件与用户自己设计的简单模块程序连接成可执行程序。

OMNET++ 具有编程简单、易于掌握等特点,只要用户具有一定的 C++ 语言知识就可以使用。使用它的 NED 语言可以非常方便地构建一个网络模型,它还提供 Plove 作为输出参数的观察工具。目前,在无线传感器网络的理论研究中,OMNET++ 被广泛采用,很多网上讨论组建议为了对大家的成果有一个统一的比较手段,在算法设计和分析时,都采用 OMNET++ 作为仿真工具。

3 最小跳数场的建立过程

在无线传感器网络内,网关的跳数为 0。其它节点发出的信息包沿着最短路径到达网关所经过的中间节点数量+1,就是该节点的最小跳数。例如:网关通信范围内的邻居节点跳数为 1,通过一个中间节点与网关通信的节点跳数为 2。

可以采用经典的扩散算法(Flooding)建立跳数场。最初,每个节点的跳数被设置为无穷大,例如 1000(可以认为是通常的网络中最大跳数永远不可能达到的数量)。网关广播一个跳数场建立信息,说明自己的跳数为 0,这个信息被它的一跳邻居节点接收到,这些节点就将自己的跳数设置为 $0+1=1$,将网关作为自己的转发节点,并产生一个新的跳数为 1、包括自己地址的信息包,向自己的邻居广播。

当节点 N 从节点 M 收到比自己当前最小跳数小的广告信息时, N 的最小跳数就设置为 L_M+1 ,其中, L_M 是节点 M 的跳数,并将 M 加入自己的转发节点集,同时广播一个包含自己最小跳数和地址的广告。如果接收到的信息所表示的跳数与自己的最小跳数相同,就将该信息包发送节点加入自己的转发节点集,但是不发出新的广告信息包。如果接收到的信息所表示的跳数大于自己当前最小跳数,就丢弃这个信息包,不做任何处理。上面的过程可以用图 1 来表示。这样的过程一直持续下去,最后网络中的每个节点都获得了自己到网关的最小跳数和转发节点集。

采用这种方法建立跳数场,从理论上来说,不论节点距离网关多远,只收到一次导致自己跳数变小的信息。这个命题的证明如下:

命题:采用 Flooding 算法建立跳数场,每个节点只会收到一次导致自己跳数变小的信息。

证明:所有广告信息包的长度是一样的,则理论上每个节

点转发信息包所花费的时间是一样的,那么信息包每经过一个节点所花费的时间是相同的,记作 T 。节点收到跳数为 K 的广告信息的时间就是 $K*T$ 。这样,越晚到达的广告信息表示的跳数也越大,只有第一个接收到的信息中的跳数是最小跳数。

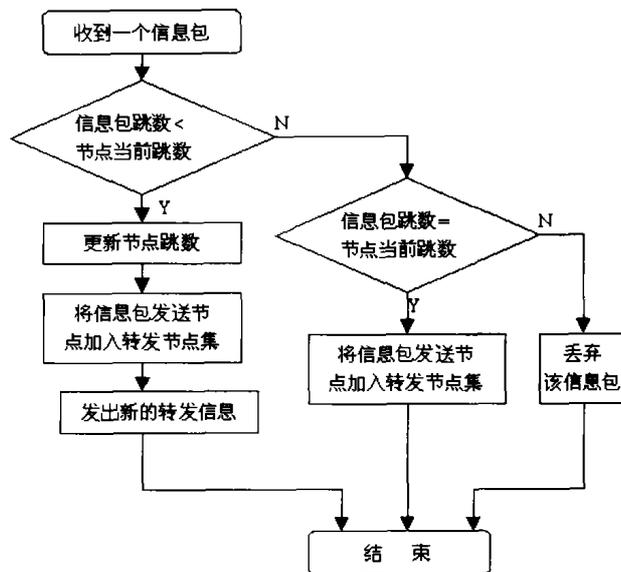


图 1 最小跳数场建立过程中节点的处理逻辑

但是在实际应用中,不能直接使用这样的算法。因为上面的算法没有考虑信息包在无线传输中可能出现的延时,延时主要来自于发送、传播、处理和因为信道冲突所造成的延时。

如果每个信息包发送的时间为 0,则延时不会对跳数场的建立产生任何影响,但是实际上,每个信息包的传输需要一定的时间。例如,一个长度为 30 字节的信息包,传输速率为 9600bps,则传输时间为 $(30*8)/9600=0.025s$ 。如果延时与信息包传输时间同一个数量级或更大,则一些节点可能会接收到多个导致自己跳数变小的信息包,从而导致网络内传送了多余的信息包。

当然,延时要对算法造成影响,必须是随机的。如果每个信息包的延时时间为恒定的,不会对算法造成任何影响。其证明与上面命题的证明基本一样,只是将上面的 T 用 $(T+T_{delay})$ 来代替。

作者采用仿真实验研究随机延时对跳数场建立的影响,仿真条件是在 $150*150$ 米区域内随机分布 500 个节点。在实验中,信息包长度为 30 字节,网络传输速率设置为 9600bps。延时时间是在某一时间段内均匀分布的,作者观察了分别使用下列 10 个随机延时时间段,跳数场建立过程中在网络内引起的信息包总量: 0, 0ms~5ms, 0ms~10ms, 0ms~15ms, 0ms~20ms, 0ms~25ms, 0ms~30ms, 0ms~35ms, 0ms~40ms, 0ms~45ms, 0ms~50ms, 实验结果如图 2 所示。其中横坐标为最大延时时间(ms),纵坐标为网络内发送的信息包总量。从图中可以看出,随着随机延时的增加,网络内用于建立跳数场的信息包数量也随之增加。

为了保证每个节点只会收到一个导致自己跳数变小的信息包,减少跳数场建立过程中需要的信息包总量,节点收到使得自己跳数变小的信息包的时候,不是立刻发送自己的跳数信息,而是等待一个时间 T_{wait} 。在此期间节点接收可能到达的新的最小跳数信息,以保证自己发出的信息包中的跳数就是自己真实的最小跳数。

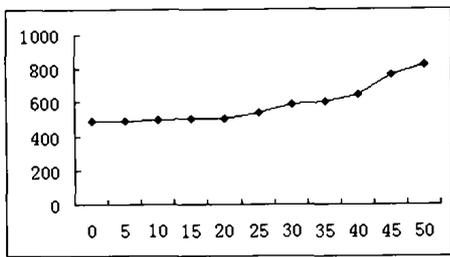


图2 随机延时大小—信息包数量

这个等待时间 T_{wait} 应该根据网络内可能出现的随机延时时间的大小来决定。作者做了一个仿真实验,节点广告信息包发送的随机延时时间在 0 到 45ms 之间随机分布,分别观察了节点采用不同的等待 T_{wait} 建立跳数场所需要的信息包总量,结果如图 3 所示。随着节点延时的加大,信息包总量逐渐减小,只要延时 T_{wait} 达到一定的大小,网络内广播的信息包数量就接近于最少数量。

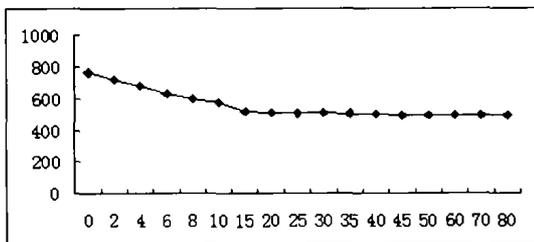


图3 节点延时时间—信息包数量

4 基于最小跳数的路由协议

基于最小跳数路由协议的基本思想是每个节点记忆自己的一跳邻居内,最小跳数比自己小 1 的节点。当有信息需要发送的时候,可以从符合条件的节点集中指定一个作为转发节点。采用图 4 的例子来说明,假设节点 S 最小跳数为 10,它的转发节点集包括 A、B、C、D 四个跳数为 9 的节点。节点 C 的转发节点集包括 E、F、G 三个跳数为 8 的节点。当节点 A 有信息要向网关发送时,它从转发节点集中任意指定一个节点,比如 C 来进行信息转发。同样 C 可以选择 E、F、G 中的任意一个作为自己的转发节点。这样,可以保证信息按照最短的路径传送,不会产生多余的信息。

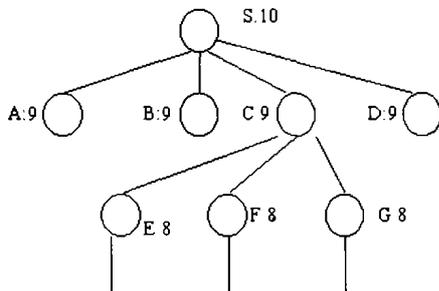


图4 基于最小跳数的路由协议举例

采用最小跳数路由协议进行信息转发,在信息包中只要指定转发节点,不需要包含其它任何的路由信息。节点需要维持的路由信息仅仅是自己的转发节点集,在正常的网络中不会超过 10 个,这对于有 256(128)个字节 RAM 的单片机来说是可以接受的。

跳数场建立之后,就可以用于信息路由。当一个节点有信

息需要向网关传送的时候,它就从自己的转发节点集中选取一个节点进行信息转发。它所有的一跳邻居都收到了这个信息,但是只有信息包中指定的节点才转发这个信息。信息包以这样的方式沿着最短路径向网关逐跳发送。最短的路径也就意味着参与信息包转发的节点数量最小,用于转发的信息包数量最少,这对于无线传感器网络来说,是至关重要的。

无线传感器网络的一个基本特征是节点具有一定的移动性,即在同一时刻,大部分节点处于静止状态,小部分节点在移动。节点位置移动之后,能否可靠地向网关传送信息是评价一个路由协议设计是否完善的重要标志。在笔者介绍的协议中,一个节点到了新的区域之后,只要向自己的一跳邻居发出“节点注册”信息,获得自己的最小跳数值和转发邻居集,就可以进行信息发送,实现非常简单,需要传送的广告信息数量也很少。

5 算法性能评估

假设一个无线传感器网络中有 300 个节点,如果每个节点发出的信息包都沿着最短路径向网关发送,那么一个跳数为 k 的节点向网关发送一个信息包在网络内引起的信息包共为 k 个。由此,可以得到 300 个节点各发一个信息包在网络内所引起的信息包数量为: $\sum_{k=1}^{300} N_k$,其中 N_k 为节点 k 的跳数。

仿真实验也证实也这一点,例如在一次仿真中,所有节点跳数之和为 2903,采用基于最小跳数的路由协议,每个节点向网关发送一个信息包,在网络内所引起的信息包数量也是 2903 个。

而相同的实验条件下,采用基于最小代价的路由协议,网络内信息包数量达到了 6000 余个。图 5 显示了两种算法在不同的节点数量情况下,每个节点各发送一个信息包在网络内所引起的总的信息包数量。从图中可以看出,协议所需要的信息包数量大约是基于代价场信息转发协议的 1/3。通过实验发现随着网络内节点平均连接度的增大,算法的优势越明显。

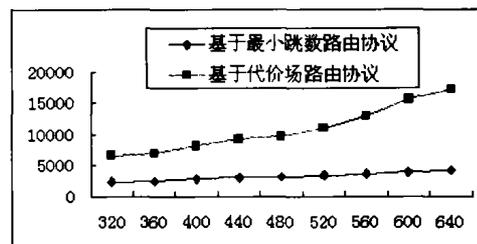


图5 网络内节点数量—网络内总信息量

6 结论

路由协议是无线传感器网络的重要研究课题,它的性能直接影响整个网络的运行效率。该文提出了一种基于最小跳数的信息转发协议,采用这个协议,网络内任何节点向网关发送的信息都沿着最短路径传送。协议的设计和实现都非常简单,节点不需要维持庞大的路由信息表,只需要记录自己转发节点集,就可以完成信息转发,因此可以在大规模网络中应用。基于最小跳数的路由协议只需要作简单的处理,就可以实现移动节点的信息转发。

该文介绍了最小跳数场的建立过程,从理论上分析了每个节点只会收到一次导致自己跳数变小的信息。作者详细分析了

(下转 198 页)

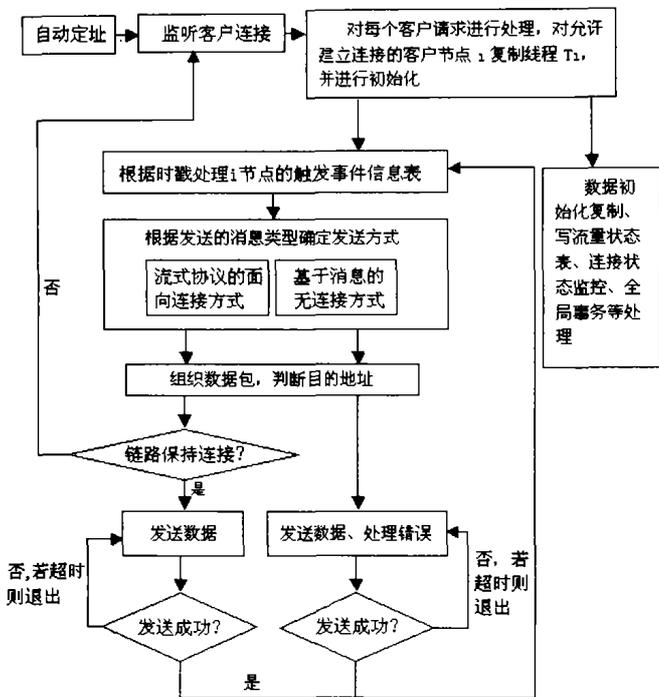


图5 通信服务器端的程序流程

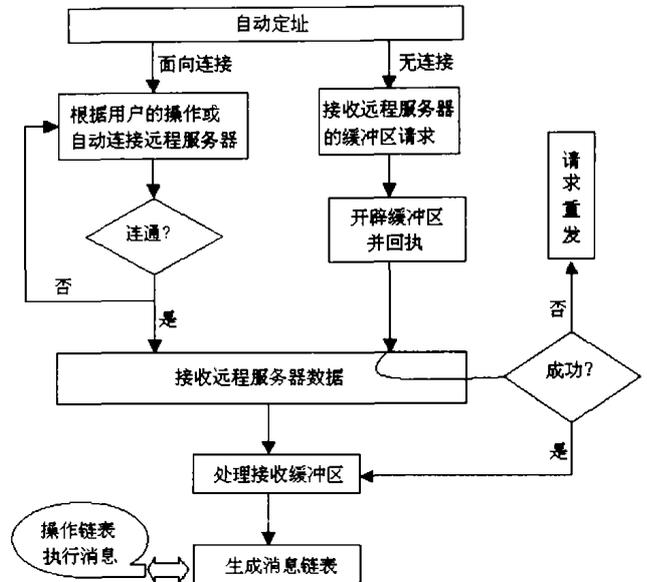


图6 通信客户端的程序流程

该文首次提出了主数据对象和影子数据对象的概念模型, 在异构数据库环境中, 利用主数据对象和影子数据对象的映射关系可以灵活地定义数据的路径空间, 并通过主动数据库中“事件-条件-动作”的数据触发机制、数据缓冲技术以及分布事务处理方法, 较好地解决了多节点群中异构数据库的问题。(收稿日期: 2003年6月)

5 系统的优点

- (1) 将节点群系统或模型的开发与数据的交互完全分开;
- (2) 采用触发器和消息解释机制后, 不但减轻了数据库的负担, 而且实现了异构数据库的数据同步, 目前支持的系统有: ORACLE、INFORMIX、Microsoft SQL Server、Sybase 等大型数据库管理系统;
- (3) 在分布式仿真应用中, 能比较容易地实现模型的控制检验点、暂停、继续、再启动以及事后重演等功能;
- (4) 方便地实现 DR 算法, 减少了网络传输流量;
- (5) 数据传输的实时性、安全性、可靠性得到保证。

6 小结

(上接 167 页)

在跳数场建立过程中, 随机延时对跳数场建立的影响及相应的处理措施。同时, 说明了基于最小跳数的信息转发协议的工作过程, 并分析了如何处理移动节点的信息转发问题。通过仿真实验, 将提出的算法与基于代价场路由协议进行性能比较, 可以看出基于最小跳数的路由协议更为合理有效。

当然, 基于最小跳数的路由协议是基于这样一个事实: 即无线传感器网络内的信息总是从节点向网关发送, 网关除了向整个网络发布广告信息之外, 不会指定某个节点作为目的地址进行信息传送。也就是说, 该协议是一种单向的信息传送协议。(收稿日期: 2003年6月)

参考文献

1. 京京工作室译. Windows 网络编程技术[M]. 机械工业出版社, 2000
2. 汪诗林等译. 数据结构、算法与应用[M]. 机械工业出版社, 2000
3. 徐新华编著. Delphi4 核心技术. 北京希望电脑公司, 1999
4. 卢正鼎等. 多数据库系统中的一致性维护[J]. 计算机研究与发展, 2001; 38
5. 雪飞, 熊平, 刘锦得. 异构数据库互操作性的概念以及面临的问题[J]. 计算机科学, 1994; (6)

参考文献

1. Akylidiz I F, Su W, Sankarasubramanian Y et al. A Survey on Sensor Networks[J]. IEEE Communications Magazine, 2002-08
2. Praveen Rentala, Ravi Musunnuri, Shashidhar Gandham et al. Udit Survey on Sensor Networks. University of Texas at Dallas, Survey Paper Submitted as per the Requirements of Mobile Computing (CS 6302) Course
3. Fan Ye, Alvin Chen, Songwu Lu et al. Scalable solution to minimum cost forwarding in large sensor networks[C]. In: Computer Communications and Networks, 2001, Proceedings Tenth International Conference on 2001: 304
4. Andras Vargar. OMNET++ Version 2.2 User Manual[M]. 2002

(上接 190 页)

参考文献

1. Paul J Perrone et al. J2EE 构建企业系统专家级解决方案[M]. 清华大学出版社
2. 梁允荣. 异构数据库集成环境下数据的直接转换[N]. 计算机世界, 1998; 198 2004.11 计算机工程与应用

(35)

3. 梁鹰等. 异构数据库的数据转换在大型信息系统中的实现[J]. 计算机工程与应用, 2000; 36(9): 103~105
4. David M Geary. Java2 图形设计 卷 II: SWING[M]. 机械工业出版社
5. Bruce Eckel. Java 编程思想[M]. 机械工业出版社
6. Hector Garcia-Molina. 数据库系统实现[M]. 机械工业出版社