

传感器网络中的数据融合^{*}

毕艳忠 孙利民

(中国科学院软件研究所 北京 100080)

摘要 传感器网络通常由大量的能量有限的传感器节点组成,利用无线通信技术从各个节点收集信息,在军事监视、环境监测等诸多方面有广泛的应用前景。如何高效利用传感器节点的能量而获得较长的生存时间是传感器网络需要解决的重要问题。数据融合机制利用节点的运算和存储能力,能够减少网络的数据传输量,是高效利用节点能量的机制之一。本文在简单介绍传感器网络的基础上,说明数据融合的作用和分类,重点分析目前提出的主要数据融合机制。

关键词 传感器网络,数据融合,能量高效

An Overview of Data Aggregation in Sensor Networks

BI Yan-Zhong SUN Li-Min

(Institute of Software, Chinese Academy of Sciences, Beijing 100080)

Abstract Sensor networks, consisting of a large number of sensor nodes with limited battery power and the ability for local computation, gather useful information from the nodes using wireless communication technology. Such systems have proposed for a broad use in many applications including military surveillance and environmental monitoring. It is a critical consideration to collect sensed information in an energy efficient manner for obtaining a long lifetime of the sensor network. As one of the mechanisms that can make use of the energy of the sensor nodes efficiently, data aggregation can reduce the traffic in network by utilizing the abilities of the nodes in local computation and storage. Based on a simple introduction of sensor networks, this paper specifies the functions and the classification of data aggregation, and analyzes the main aggregation schemes proposed in detail.

Keywords Sensor networks, Data aggregation, Energy efficient

1 引言

传感器网络是由部署在监测区域内的大量微型的传感器节点组成,收集并处理各节点采集的信息,并通过无线技术实现节点间的通信。如何有效利用节点的有限能量是传感器网络设计中的重要问题,而减少网络中的数据传输量是降低节点能量消耗的有效手段之一。因此在收集和传输监测信息的过程中,应尽量进行网内处理(in-network processing)。由于每个独立的传感器的监测范围及可靠性是有限的,在放置传感器节点时,必须使传感器节点的监测范围互相交叠,以增强整个网络所能获得的信息的鲁棒性和准确性。这样,传感器节点所采集到的数据就存在一定的冗余性。在各个节点将监测数据通过多跳路由传送给收集数据的 Sink 节点(或称基站)的路径中,需要对数据进行处理以减少冗余信息,这就是数据融合(data aggregation 或 data fusion)。数据融合是网内处理的主要目的之一。

下面首先说明数据融合的作用及分类,然后对目前提出的传感器网络中的数据融合机制进行介绍和分析,最后说明数据融合的成本并总结。

2 数据融合的作用及分类

在传感器网络中,数据融合主要有以下三个主要方面的

作用:

(1) 节省能量 数据融合的直观效果就是减少了数据传输量,从而节省了能量。最理想的融合情形,中间节点可以把 n 个长度相等的输入数据分组合并成 1 个输出分组,其节能效率就是 $(n-1)/n$;最差情况下,融合操作并未减少数据量,但通过减少分组个数,减少了大量协商、争用信道的操作,降低了网络整体的单位传输开销,同样节省了能量。

(2) 获得更准确的信息 传感器的监测误差以及监测环境的噪音或干扰,会导致收集到的数据不够准确。通过对监测同一对象的多个传感器所采集的数据进行综合,能够有效地提高信息的精度和可信度。

(3) 减轻网络拥塞 减少数据量(包括分组大小和分组个数)的另一个直接结果就是减轻网络的传输拥塞,降低了数据传输的延迟,提高了网络收集数据的整体效率。

传感器网络中的数据融合目前主要有两种分类方法。第一种是根据融合前后的数据的信息含量,分为两类^[1]:

(1) 无损融合 所有的细节信息均被保留,此类融合常见做法就是去除信息中的冗余(重复的内容)。根据信息理论,无损融合的信息整体缩减大小受到其熵值的限制;

(2) 有损失融合 通常会省略一些细节信息或降低数据的质量以获得更高等级的能量节省。

数据融合可以在传感器网络协议栈的多层中实现,第二

^{*} 本论文研究得到国家自然科学基金项目(No. 60373049)和国家 863 高科技发展计划项目(No. 2001AA112051)资助。毕艳忠 硕博连读生,研究方向是无线网络。孙利民 博士,研究员,主要研究方向为无线移动网络和宽带接入网。

种分类方法是根据数据融合与各协议层之间的关系,分为三类^[14](如图 1):

(1)依赖于应用的数据融合(Application Dependent Data Aggregation, ADDA);

(2)独立于应用的数据融合(Application Independent Data Aggregation, AIDA);

(3)结合以上两种技术的数据融合。

ADDA 的操作需要跨协议层理解应用层数据的含义,并且可能会导致数据中丢失的信息过多;AIDA 则保持了网络协议层的独立性,处于网络层与数据链路层之间,本身不会导致信息丢失;由于 AIDA 可以和其它协议层内的数据融合技术共存,因此可以综合使用多种机制获得更好的融合效果。

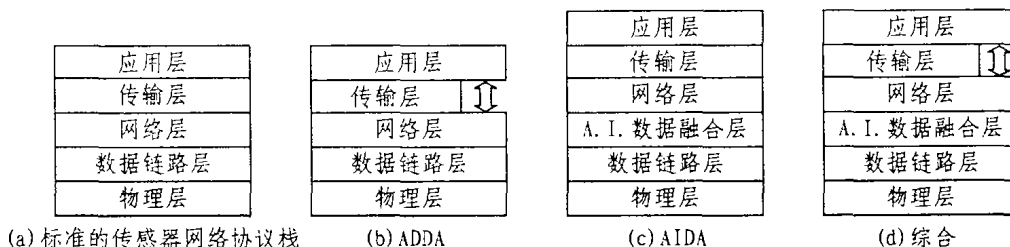


图 1 数据融合根据与传感器网络协议层的关系分类

3 数据融合机制

数据融合通常是应用相关的,影响缩减数据总量的因素包括监测数据的特性与表达形式、网络连接拓扑以及具体应用等。在传感器网络的协议栈中(如图 1),低层协议(包括数据链路层与物理层)是面向传输的,是保障网络中各节点间进行通信的基础,而在上面的三个协议层中,传输层并不是所有的传感器网络都需要。因此研究者主要把目光投向了应用层与网络层:在应用层开发了面向应用的数据融合接口,在网络层开发了与路由相结合的数据融合技术。在现有的协议层之外,又提出了独立于应用的数据融合技术,形成了在网络层与数据链路层之间的数据融合层。

3.1 应用层中的数据融合

传感器网络具有以数据为中心的特点,在应用层需要考虑以下两点:

(1)传感器网络可以实现多任务,因此需要对用户屏蔽底层的工作,并提供方便、灵活的需求提交手段;

(2)通信的代价相对于本地计算的代价要高,因此通过本地计算,可以大幅度减少通信的数据量,有效节省能量。

因此,分布式数据库技术能够用来实现传感器网络的数据汇集。通过让每个节点都理解数据请求,中间节点可以对接收到的数据和自己的数据进行本地运算,并只传送查询或运算结果。

Cougar^[12]和 TAG^[13]就是采用了类 SQL 语言的应用层接口,方便传感器网络的使用者根据应用需要组织收集数据的请求,各节点也通过分布式的数据库查询逐步汇集查询结果。但是在应用层开发的数据融合技术只适用于网络规模较小,且单个节点的计算与存储能力较充足的传感器网络。

3.2 网络层中的数据融合

3.2.1 路由方式 传感器网络中的路由根据是否考虑数据融合可以分为两类^[8]:

(1)以地址为中心的路由(Address-Centric Routing,简称 AC 路由):每个源节点沿着到 Sink 节点的最短路径转发数据,是不考虑数据融合的路由;

(2)数据为中心的路由(Data-Centric Routing,简称 DC 路由):数据在转发的路途中,中间节点根据数据的内容,对来自多个数据源的数据进行融合或合并的操作。

AC 路由与 DC 路由对能量消耗的影响视数据的可融合程度而不同。如果所有数据源的数据之间没有任何冗余信息,

则无法进行数据融合,AC 与 DC 路由在能量消耗方面没有差别。如果所有数据源的数据都相同,信息冗余量最大,则 AC 路由由只需要简单修改就可以达到 DC 路由的节能效果甚至更省能量。例如在重复数据的产生要维持一段时间的情况下,修改 AC 路由使得当 Sink 节点收到第一份数据时,就立即通知其它数据源和正在转发数据的中间节点立即停止发送数据。但是,如果不对 AC 路由进行修改,在这种情况下,两种路由对能量的消耗的差距最大,即 DC 路由节能的优势最明显。如果不同的数据源发出的数据之间都有一定程度的冗余,则 AC 路由无论怎样修改都无法达到 DC 路由的节能效果。

3.2.2 DC 路由中的数据融合

(1)基于查询的路由。定向扩散(Directed Diffusion, DD)^[5]路由中的数据融合包括路径建立阶段的任务(又称为兴趣,interest)融合和数据发送阶段的数据融合,这两种融合都通过缓存机制得以实现。DD 中的兴趣融合得益于它基于属性的命名方式,类型相同、监测区域完全覆盖的兴趣在某些情况下就可以融合成一个兴趣。DD 的数据融合采用的是“抑制副本”的方法,即对转发过的数据进行缓存,发现重复的数据将不予转发。这种方法虽然简单,但与其路由技术相结合,能够有效地减少网络中的数据量。

(2)基于层次的路由。LEACH^[2]与 TEEN^[4]都是基于层次的路由,它们使用分簇的方法使得数据融合的地位突显出来,每个簇首在收到本簇成员的数据后进行融合处理,并将结果发送给 Sink 节点。LEACH 算法仅强调了数据融合,但并未涉及具体的融合方法。根据传感器网络的功能模式,传感器网络可简单地分为两类:传感器节点周期性发送监测信息的主动网络(proactive network)和监测突发事件的反应网络(reactive network)。TEEN 是 LEACH 的一种改进,应用于反应网络。TEEN 与 DD 一样通过缓存机制抑制不需要转发的数据,但它利用阈值的设置使抑制更加灵活,对与前一次采集结果差值较小的数据也进行了抑制。

(3)基于链的路由。PEGASIS^[10]及其高阶算法^[5]是对 LEACH 的融合方式进行了改进。它基于两个假设:一是所有节点距离 Sink 节点都很远;二是每个节点都能将接收到的数据分组与自己的数据融合成一个大小不变的分组。PEGASIS 的做法是在收集数据前,利用贪心算法将网络中的所有节点连接成一条单链(如图 2),然后随机选取一个节点作为首领(图中为节点 3),它向链的两端发出收集数据的请求,数据从单链的两个端点向首领流动。中间节点在传递数据前要执行

融合操作,最终由首领节点将结果数据传送给 Sink 节点。

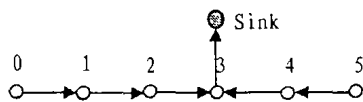


图 2 PEGASIS 的单链结构

PEGASIS 的优点在于单链的结构使得每个节点发送数据的距离几乎都是最短的,而且最终只有一个节点进行远距离的数据传输,因此比 LEACH 算法更节省能量。PEGASIS 的主要缺陷是平均延迟较大且鲁棒性较差。

3.2.3 数据融合树 (aggregation tree) 的构造 传感器网络中, Sink 节点在收集数据时,是通过反向组播树的形式从分散的传感器节点逐步将监测数据汇集起来的。如图 3 所示, A、B、C、D 四个传感器节点监测到突发事件,如果传送数据的路径形成一棵反向组播树(即数据融合树),树上的每个中间节点都对收到的数据进行融合处理,那么数据便得到了即时的且最大程度的融合。

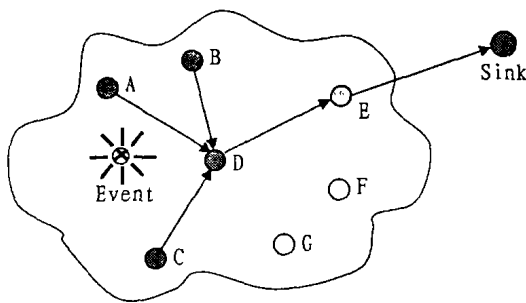


图 3 数据融合树图

对于 DC 路由,文[8]证明了对于一个给定的节点任意放置的传感器网络,对每个数据传输次数都最少的 DC 路由可以转化为最小 Steiner 树,构造这样的路由是一个 NP 难的问题。文[8]涉及到了三种实用的次优方案:

(1)近源汇集(Center at Nearest Source, CNS):距离 Sink 节点最近的源节点充当数据的汇聚节点,所有其它的数据源都将数据发送给这个节点,由这个节点将融合后的数据发送给 Sink 节点;

(2)最短路径树(Shortest Paths Tree, SPT):每个数据源都各自沿着到达 Sink 节点的最短路径传送数据,这些最短路径的交叠形成融合树;

(3)贪心增长树(Greedy Incremental Tree, GIT):此种方案中融合树是逐步建立的,最初只有 Sink 节点与距离它最近的源节点之间的一条最短路径,然后每一步都从剩下的源中选出距离这条最短路径树最近的节点连接到树上,直到所有的源节点都连接到树上。

这三种方案比较适合应用于反应网络,因为这样的网络环境具备源节点数目少、位置相对集中以及数据相似性大的特点,可以在发动远距离(事件源到 Sink)传输前尽早地进行数据融合处理,有效减少数据的传输量。在数据的可融合程度一定的情况下,它们之间的节能效果关系为:GIT>SPT>CNS。

3.3 独立的数据融合协议层

在网络层开发的数据融合技术主要有以下缺陷:首先是破坏了各层协议的完整性,上下层协议不透明;其次,为了跨协议层理解数据,需要对数据进行命名,而命名机制导致来自

同一源节点的不同类型的数据之间不能融合;此外,这些数据融合方法都默认数据融合的程度越高越好,采用了网内处理的手段,这虽然带来了较高的融合程度,但会导致信息丢失过多,且容易引入较大的延迟。

文[14]的作者提出了独立于应用的数据融合(AIDA)机制,其基本思想就是不关心数据的内容,而根据下一跳地址进行多个数据单元的合并,通过减少数据封装头部的开销以及 MAC 层的发送冲突来达到节省能量的效果。提出 AIDA 的目的除了要摒除依赖于应用的融合方案(ADDA)的弊端外,还将增强数据融合对网络负载状况的适应性,即当网络负载较轻时不进行融合或进行低程度的融合,而在网络负载较重,MAC 层发送冲突较严重时进行较高程度的融合。

AIDA 协议层位于网络层和 MAC 层之间,对上下协议层透明,其体系结构如图 4 所示。

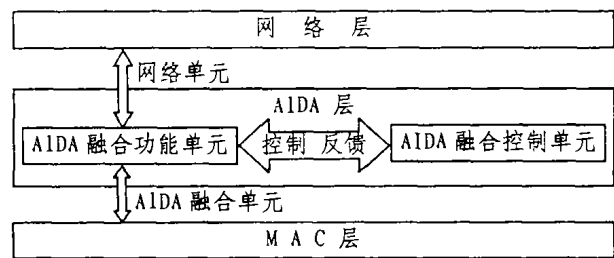


图 4 AIDA 的体系结构

AIDA 工作过程简述如下:

(1)发送方向(从网络层到 MAC 层):AIDA 融合功能单元根据设定的融合度(合并的最大分组数)和数据分组的下一跳地址,将从网络层到来的多个数据分组融合成一个 AIDA 数据单元并递交给 MAC 层进行传输;融合度的确定以及何时调用融合功能则由 AIDA 融合控制单元决定;

(2)接收方向(从 MAC 层到网络层):融合功能单元将 MAC 层递送上来的数据单元拆散为原来的网络层分组传递给网络层;这样做虽然会在一定程度上降低效率,但其目的是为了保证协议层的模块性,并且允许网络层对每个数据分组重新路由。

AIDA 提出的出发点并不是将网络的生存时间最大化,而是要构建一个能够适应网络负载变化,独立于其它协议层的数据融合协议层,能在保证不降低信息的完整性和不降低网络端到端延迟的前提下,以数据融合为手段,减轻 MAC 层拥塞冲突,减少能量的消耗。

结束语 数据融合技术在节省能量的同时,要牺牲其它方面的性能作为代价。首先是延迟的代价。数据传送过程中,寻找易于进行数据融合的路由,执行数据融合函数,有时还要为进行融合而等待其它数据的到来,这三个方面都会增加网络的平均延迟。其次是鲁棒性的代价。传感器网络相对于传统网络有更高的节点失效率以及数据丢失率。数据融合可以大幅度降低数据的冗余性,但相对而言也降低了网络的鲁棒性。

数据融合可以在传感器网络汇集数据的过程中减少数据传输量,从而节省能量。现有的数据融合机制大部分都与网络层的路由机制相结合;同时,在应用层中,可以利用分布式数据库查询技术,对采集到的数据进行逐步筛选;此外,还提出了独立于其它协议层的数据融合协议层,通过减少 MAC 层的发送冲突和头部开销达到节省能量的目的。数据融合技术在节省能量的同时,是以延迟的增加和鲁棒性的降低为代价的。

(下转第 117 页)

编译的,快速实现也比较困难。有些优化算法与其执行平台有关。因而也就不能象传统的静态编译器可以在中间表示级做大量的与目标机无关的优化。鉴于运行时特定化的优缺点,将运行时特定化作为一种优化技术嵌入到传统的编译器中。这样一方面可以充分发挥其优势,同时可以有效地弥补其不足。

(2)执行延迟问题。引入运行时编译开销是运行时特定化和 just-in-time 编译都存在的问题。如何在获得较好的性能的同时引入最少的运行时编译开销是动态编译系统需要解决的一个关键问题。减少执行延迟的有效途径主要有两个方面,一是减少运行时编译的代码,即 Lazy compilation;另一个在多处处理器系统中利用空闲的处理器来执行运行时编译任务,使得程序的运行和编译可以并行执行,即后台编译^[10]。

(3)可重定向问题。开放式平台的显著特点之一是它的平台无关性。动态编译技术的另一个问题是如何解决重定向的问题,真正做到与平台无关,并能根据目标机执行相关的优化。由于动态编译系统运行时编译的特点,因此可以更加充分地利用机器信息进行编译的优化和决策,甚至可以在运行时收集部分机器信息,从而获得更好的可重定向性。但是现有的动态编译系统在这些方面做得都还很少,因此可重定向问题也是动态编译技术研究中的核心问题。

(4)决策模型的自适应问题。由于应用环境千差万别,因此对目标代码的要求也各不相同。而一般的编译优化技术的目标主要在于提高目标代码的执行性能,而对代码占用的存储空间和执行空间以及功耗方面则考虑较少。编译优化是一个多目标优化问题,对一个动态编译优化系统来说,如果能够自适应地实现多目标优化则可以大大提高其自适应性。而现有的系统大多采用由固定优化方法集合组成的双重编译器,不支持不同应用程序特点和不同优化目标的综合考虑,因此,自身的灵活性较差,难以适应目标环境及应用模式的变化。

结束语 动态编译作为一种新兴的编译优化技术,有广阔的发展前景。本文探讨了运行时特定化和 JIT 编译的特点及其代表性系统。动态编译能够扩大优化范围,提高系统性能,但是动态编译优化引入运行时的编译开销,从而可能会产生执行延迟。因此,利用动态编译优化技术,要在动态编译开

销和动态编译代码的质量间做出折衷考虑。

参考文献

- 1 Masuhara H, et al. Dynamic Compilation of a Reflective Language Using Run-Time Specialization. Principles of Software Evolution, 2000. In: Proc. Intl. Symposium on, Nov. 2000. 128~137
- 2 Noel F, et al. Automatic, Template-Based Run-Time Specialization: Implementation and Experimental Study. Computer Languages, 1998. In: Proc. 1998 Intl. Conf. on, May 1998. 132~142
- 3 Auslander J, et al. Fast, Effective Dynamic Compilation. In: Proc. of the ACM SIGPLAN'96 conf. on Programming language design and implementation, May 1996
- 4 Grant B, et al. DyC: An Expressive Annotation-Directed Dynamic Compiler for C. Theoretical Computer Science, 2000, 248(1-2)
- 5 Mock M, et al. Calpa: A Tool for Automating Dynamic Compilation. Microarchitecture, 2000. MICRO-33. In: Proc. 33rd Annual IEEE/ACM Intl. Symposium on, Dec. 2000. 291~302
- 6 Gupta R, Mehofer E, et al. Profile Guided Compiler Optimizations. The Compiler Design Handbook: Optimizations & Machine Code Generation, Auerbach Publications
- 7 Michal C, et al. Practicing JUDO: Java Under Dynamic Optimizations. In: Proc. of the ACM SIGPLAN '00 conf. on Programming language design and implementation, Aug. 2000
- 8 Michael G B, et al. The Jalapeno Dynamic Optimizing Compiler for Java™. ACM. In: Proc. of the ACM 1999 conf. on Java Grande, June 1999
- 9 Matthew A, et al. Adaptive Optimization in the Jalapeno JVM. In: Proc. of the conf. on Object-oriented programming, systems, languages, and applications, Oct. 2000
- 10 Chandra K, et al. Reducing the Overhead of Dynamic Compilation. Software: Practice and Experience, 2000, 31(8): 717~738
- 11 Krantz C. Coupling On-Line and Off-Line Profile Information to Improve Program Performance. Code Generation and Optimization, 2003. CGO 2003. In: Intl. Symposium on, Mar. 2003. 69~78
- 12 Ali-Reza A, et al. Fast, Effective Code Generation in a Just-In-Time Java Compiler. In: Proc. of the ACM SIGPLAN '98 conf. on Programming language design and implementation, May 1998
- 13 Michal C, et al. The Open Runtime Platform: A Flexible High-Performance Managed Runtime Environment. In: Proc. of the 2002 joint ACM-ISCOPE conf. on Java Grande, Nov. 2002
- 14 Matthew A, et al. Online Feedback-Directed Optimization of Java. In: Proc. of the 17th ACM conf. on Object-oriented programming, systems, languages, and applications, Nov. 2002
- 15 Bowen A, et al. The Jalapeno virtual machine. IBM Systems Journal, Java Performance Issue, 2000, 39(1)
- 16 Bruening D, et al. An Infrastructure for Adaptive Dynamic Optimization. Code Generation and Optimization, 2003. CGO 2003. In: Intl. Symposium on, Mar. 2003. 265~275

(上接第 103 页)

参考文献

- 1 Akyildiz I F, Su W, Sankarasubramaniam Y, Cayirci E. A survey on sensor networks. IEEE Communications Magazine, 2002, 40(8): 102~114
- 2 Heinzelman W R, Chandrakasan A, Balakrishnan H. Energy-efficient communication protocol for wireless microsensor networks. In: Proc. of the 33rd Intl. Conf. on System Sciences (HICSS '00), Jan. 2000. 1~10
- 3 Intanagonwiwat C, Govindan R, Estrin D. Directed diffusion: A scalable and robust communication paradigm for sensor networks. In: Proc. of the Sixth Annual ACM/IEEE Intl. Conf. on Mobile Computing and Networking (Mobicom'2000), Boston, Massachusetts, August 2000
- 4 Manjeshwar, Agarwal D P. TEEN: a routing protocol for enhanced efficiency in wireless sensor networks. In: 1st Intl. Workshop on Parallel and Distributed Computing Issues in Wireless Networks and Mobile Computing, April 2001
- 5 Lindsey S, Raghavendra C S, Sivalingam K. Data Gathering in Sensor Networks using the Energy * Delay Metric. In: Proc. of the IPDPS Workshop on Issues in Wireless Networks and Mobile Computing, 2001
- 6 Heidemann J, et al. Building efficient wireless sensor networks

- with low-level naming. In: Proc. of the ACM Symposium on Operating Systems Principles, Banff, Canada, Oct. 2001
- 7 Intanagonwiwat C, et al. Impact of network density on data aggregation in wireless sensor networks. [Technical Report 01-750]. University of Southern California, Nov. 2001
- 8 Krishnamachari B, Estrin D, Wicker S. Modelling data-centric routing in wireless sensor networks. In: Proc. of IEEE Infocom, 2002
- 9 Krishnamachari B, Estrin D, Wicker S. Impact of data aggregation in wireless sensor networks. In: Intl. Workshop on Distributed Event-Based Systems, Vienna, Austria, July 2002
- 10 Lindsey S, Raghavendra C S. PEGASIS: Power Efficient Gathering in Sensor Information Systems. In: Proc. of IEEE Aerospace Conf. 2002
- 11 Kalpakis K, Dasgupta K, Namjoshi P. Maximum Lifetime Data Gathering and Aggregation in Wireless Sensor Networks. In: Proc. of IEEE Networks'02 Conf. 2002
- 12 Yao Y, Gehrke J E. The Cougar Approach to In-Network Query Processing in Sensor Networks. Sigmod Record, 2002, 31(3)
- 13 Madden S R, et al. TAG: a Tiny Aggregation Service for Ad-Hoc Sensor Networks. OSDI, Dec. 2002
- 14 He T, et al. AIDA: Adaptive Application Independent Data Aggregation in Wireless Sensor Networks. ACM Trans. on Embedded Computing System Special issue on Dynamically Adaptable Embedded Systems, 2003