

HA-DMDB 基于数据模型趋势分析的 Cache 一致性管理策略

石胜飞 李建中 王朝坤 李金宝

(哈尔滨工业大学计算机科学与工程系 哈尔滨 150001)

(Shengfei @hit.edu.cn)

摘要 随着嵌入式微处理器、存储设备、无线通信技术的迅速发展,出现了具有无线通信能力的小体积的无线传感器(wireless sensor). 将大量这样的传感器通过无线通信方式组成传感器网络,可以进行大范围的分布式系统的应用. 提出了一种传感器网络的综合应用系统体系结构和模型,并提出了基于数据模型趋势分析的 Cache 一致性管理策略.

关键词 传感器网络;分布式移动数据库;Cache 一致性管理策略

中图法分类号 TP311.13

Cache Coherence Management Based on Data Trend Analysis in HA-DMDB System

SHI Sheng-Fei, LI Jian-Zhong, WANG Chao-Kun, and Li Jin-Bao

(Department of Computer Science and Engineering, Harbin Institute of Technology, Harbin 150001)

Abstract With the rapid development of embedded microprocessors, memory and wireless communication, the sensor with wireless communication ability has appeared. Network of these sensors will help develop very large scale distributed sensing application. Proposed in this paper is a method to manage cache coherence based on data trend analysis in HA-DMDB system.

Key words sensor net; distributed mobile database; cache coherence management

1 引言

随着嵌入式微处理器、存储设备、无线通信技术的迅速发展,出现了具有无线通信能力的小体积的无线传感器(wireless sensor). 将大量这样的传感器通过无线通信方式组成传感器网络,可以进行大范围的分布式监测. 处于传感器网络内部的传感器结点通过互相合作完成特定的任务. 这种技术可以广泛应用于许多领域,例如森林火灾监测系统、战场环境信息监测等.

由于传感器的计算、存储、通信能力和电源的限制,使得传感器可工作的时间有限,并且不能存储大量的数据也不能进行复杂的运算. 在这样的系统当中,如何有效管理数据、传输数据以及对数据进行有

效地查询是这个领域内主要研究的问题. 目前已经提出了许多有关的算法和系统模型.

文献[1~2]介绍了传感器的物理特性以及提出了专用于传感器结点的嵌入式操作系统. 文献[3~4]提出了在 MAC 层上面减少通信开销的算法,主要是如何减少通信对能量的消耗以及多个结点同时访问网络的时候的公平调度算法.

数据包的路由是这个领域一个很关键的问题. 文献[5~8]提出了一些主要的路由算法.

在传感器网络所处的一个广泛的地理范围内部进行数据的传播也是这个领域所关心的问题. 文献[7,9]提出了 directed diffusion 的数据传播算法. 它主要的特点是在传感器网络内部进行了数据的集成,是一种以数据为中心的存储策略.

但是目前的文章主要集中于传感器网络内部的

收稿日期:2003-06-12

基金项目:国家“八六三”高技术研究发展计划基金项目(2001AA415410);哈尔滨市科技基金项目(2002AA1CG059-10)

各种算法,对于这些数据的访问方式没有更多的讨论.在很多环境下,例如在环境监测系统中,会在很大的范围内布置大量的传感器,分别监测不同的环境参数.整个监测的区域往往被划分为多个不同的地区,由多辆巡逻车在不同的地点收集数据.同时,又有许多配备有个人移动计算设备的人员在区域内进行巡逻.他们向属于自己所在区域的巡逻车发出查询请求,获得本地区或者其他地区的监测数据.在这种应用环境当中,Sensor Net 产生的数据往往是数据流形式的.而另一方面,由于巡逻车和个人的移动性,很难处理全部的数据流.另外,Sensor 结点有限的电源能量又不允许连续工作和对外通信.因此,如何有效地在这样的系统体系结构中进行数据管理是这篇文章重点讨论的问题,我们称这样的系统为 HA-DMDB 系统(hybrid ad-hoc network based distributed mobile database).

我们提出了基于数据变化趋势模型进行数据 Cache 管理的方法.系统根据系统各种参数的历史数据不断地调整各种参数的趋势函数,同时建立多级的 Cache 体系,从而大大减少系统的通信负载,提高了数据的可获得性,延长了 Sensor Net 的生存时间.在 Cache 管理方面已经有过很多的研究成果^[10~12],这些方法都是对于查询结果的缓存,只是组织的形式有所不同.我们提出的 Cache 模型是基于数据变化趋势的,不仅包含了先前的查询结果,而且新的查询也可以通过缓存的趋势函数而获得满足,而且使用趋势函数作为缓存的组织形式,大大减少了所需的存储空间.这篇文章重点讨论了这种 Cache 模型的一致性维护算法.

本文的后续部分结构如下:第 2 节介绍了系统模型以及定义,第 3 节描述了缓存一致性管理的策略,最后给出了结论和将来的工作.

2 HA-DMDB 系统体系结构以及模型定义

HA-DMDB(见图 1)从逻辑上可以看成是一个多层次的分布式移动数据库.整个体系结构的底层是 Sensor Net.每个 Sensor 结点具有有限的计算、存储以及无线通信能力.Sensor 的功能是监测特定的目标或者环境.在实际应用中往往使用一组具有相同功能的 Sensor 同时对某一个目标进行监测,同时,在一个区域内部又分布着多组具有不同功能的 Sensor.从系统的角度来看,Sensor Net 是一个只读的分布式数据库.数据库可以看成关系的模型或者是对象关系模型.在 Sensor Net 上面是基于 ad-hoc 无线通信网络的 Mobile Server(MS).MS 和传统意义上的数据库服务器相比,不同之处在于它的移动性和网络环境是无线网络.在 HA-DMDB 系统当中,MS 处于某些移动设备上面,比如汽车、飞机等等. MS 的计算和存储能力跟传统意义上的服务器相同或相近.在 Sensor Net 所覆盖的广泛区域内,分布着许多 MS,每个 MS 在自己所处的区域内具有数据访问点(access point)或者查询代理(query proxy)的作用.但是,不同于目前已有系统的 Query Proxy,MS 的移动性使得它不固定属于某一个区域,而且它要完成更复杂的功能. HA-DMDB 系统的最终使用者是 Mobile Unit(MU),通常是配带小型移动计算设备的用户,例如使用 PDA、可穿戴计算

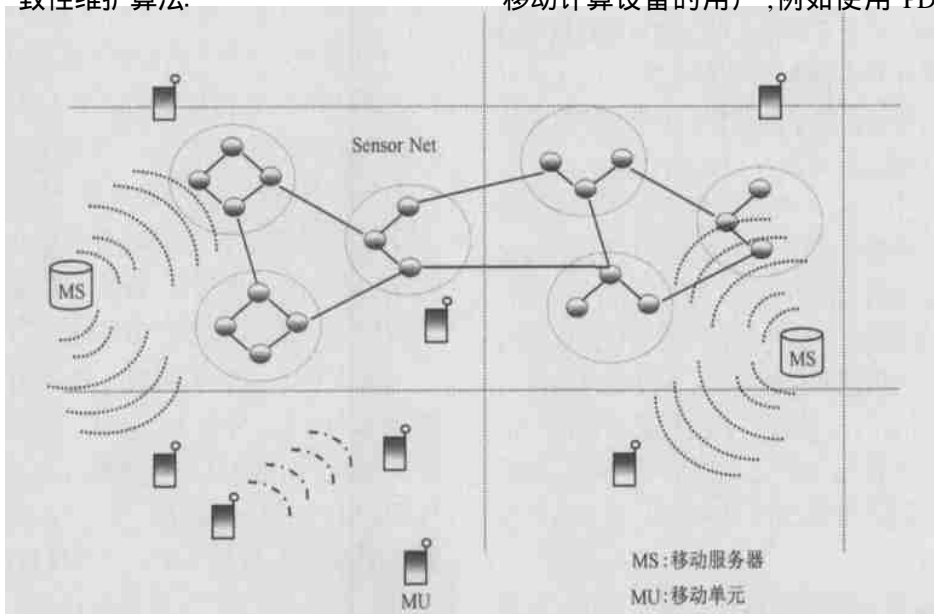


图 1 HA-DMDB 系统体系结构

机、掌上电脑等设备的用户. 与 MS 相比, MU 的计算和存储能力较弱, 能源也受限制, 通信带宽小. 根据以上的描述, 可以将 HA-DMDB 系统的结构形式化地描述如下:

定义 1. HA-DMDB 体系模型是一个四元组 (WS, MS, MU, HAd) , 其中: WS (wireless sensor network) 是无线传感器网络, MS (mobile server) 移动服务器, MU (mobile unit) 移动单元, HAd (hybrid wireless ad-hoc network infrastructure) 混和型的无线 ad-hoc 通信网络. 因为 HA-DMDB 系统当中存在着许多不同的无线通信协议.

定义 2. HA-DMDB 数据对象模型是一个五元组: $(D_{\text{sensor}}, D_{\text{static}}, D_{\text{time}}, D_{\text{model}}, D_{\text{cache}})$, 其中:

$$(1) D_{\text{sensor}} = (S, F, L, M)$$

D_{sensor} 是系统当中的动态数据集合, 是全部 S_{sensor} 集合 S 所对应的数据集合以及在上面进行的各种操作. 根据每个 S_{sensor} 所具备的功能, D_{sensor} 又可以分为多个不相交的数据集合, 分别表示不同功能的 S_{sensor} 所得到的数据. F 是所有 S_{sensor} 功能的集合. D_{sensor} 按照功能进行的划分定义为 $F_{\text{div}}(D_{\text{sensor}}) = \{D(S_i) \mid \forall_i S_i, F(S_i) = f_i, f_i \in F\}$, 我们把每一个划分记为 $F(D)_i$. 另一方面, 每一组数据集合又分布在不同的地点. F 是地理区域划分的集合. 类似地, 我们定义将 $F(D)_i$ 按照地点进行划分: $F_{\text{div}}(F(D)_i) = \{D(S_i) \mid \forall_i S_i, S_i \in F(D)_i, L(S_i) = l_i, l_i \in L\}$, 我们把每一个这样的划分称为具有功能 f_i 的 Cluster. 在每个 Cluster 内部, 多个 S_{sensor} 互相协作完成特定的任务. 每个 Cluster 根据内部每个 S_{sensor} 的计算能力、能源情况、通信能力进行任务的分配, 同时, 还要根据预先设定的局部数据模型决定数据的处理方式, 包括将数据发送出去、丢弃以及如何进行局部数据模型的调整. 这些策略定义为集合 M_{method} . $M_{\text{method}} = \{Role, M_{\text{local}}, Action\}$. 其中, $Role$ 定义了一个 Cluster 内部的角色分配. M_{local} 定义了局部的数据模型, 这个模型跟具体应用相关, 它将 S_{sensor} 产生的数据进行融合. $M_{\text{local}} = (f, \dots)$, 其中, f 是一个函数, 它根据当前得到的数据以及历史数据形成一个数据变化的趋势, 并以此对数据的发展进行一个估计, \dots 是新产生的数据和使用 f 估算值之间的误差值, \dots 是事先指定的一个阈值, 如果误差小于 \dots , 则不产生数据的更新消息, 用来减轻系统的负载.

(2) D_{static} 是一些静态描述信息的集合, 包括 S_{sensor} 的分布情况、性能指标、路由策略等等, 还包

括空间的划分、MU 以及 MS 的描述性信息.

(3) D_{time} 时间参数在 HA-DMDB 系统中具有重要的作用, S_{sensor} Net 产生的数据可以看成是时间序列. 随着时间的变化, 整个数据模型各个参数的发展趋势会发生变化. 各种数据处理的方法都和时间参数密不可分. D_{time} 还包括全局时钟的同步策略.

(4) D_{model} 是 HA-DMDB 系统内部存在于不同层次上面的数据融合模型.

$$D_{\text{model}} = (P, F, H, \dots).$$

P 是模型中所有参数的集合, 参数包括各种传感器产生的不同类型的数据.

F 是模型的数据融合函数, 它综合各种参数的值给出系统局部或者全局的数据模型的发展趋势.

H 表示模型的层次. 根据模型产生的位置, 可以分为 4 个层次: S_{sensor} Net 层、MS 层、MU 的 Cache 层、全局层. S_{sensor} Net 层是每一个 Cluster 产生的局部模型, MS 层是每一个 Mobile Server 在一定时间内对于一个或若干个区域内所形成的数据模型, MU 的 Cache 层是每一个 MU 根据以往查询结果所得到的与位置相关的区域的局部模型, 全局层是通过综合全部 MS 层的数据模型得到的整个系统模型的发展趋势.

是根据当前构造的数据模型所估算的各种参数值和实际测量值之间的误差集合.

是系统规定的各种参数误差允许范围的阈值的集合.

(5) D_{cache} 是位于系统不同层次上面的数据缓存集合.

$$D_{\text{cache}} = (P_{\text{para}}, S_Q, T, F_t, H, B_{\text{uffer}}).$$

P_{para} 是查询中涉及到的系统参数, 即查询 S_Q 在 D_{sensor} 的 S_{sensor} 属性集合的某个子集上面的投影. T 是一个时间段, 表示 Cache 数据项的建立时间和有效持续时间的估计值, 这个估计值是从某个层次上的 D_{model} 根据历史数据趋势的变化得到的经验值. F_t 是一个函数集合, 每个函数是对应于集合 P_{para} 中某个参数发展趋势的预测函数.

H 表示 D_{cache} 所处的层次. 根据 D_{cache} 所处的位置, 可以将 D_{cache} 分为 3 个层次: S_{sensor} 结点层. 每个 S_{sensor} 结点除了产生数据以外, 还负责路由来自邻居结点的查询和数据. 每个 S_{sensor} 结点保存少量的查询结果, 这种 Cache 具有很短的有效期. MS 层. 每个 MS 将曾经回答的 Query 进行缓存, 并按照 Query 的语义信息进行索引. 当 S_{sensor} 数据更

新后,如果影响到了某些缓存中的查询结果,则 MS 会根据参数 T, F_t , 决定是否发出缓存失效报告(IR). MU 层. Mobile Unit 具有的“断接性”、“移动性”以及有限的通信能力这些特点,使得 MU 不能频繁地通过 MS 获得数据,因此,Cache 对于 MU 的数据管理具有重要的作用.

MU 将以前发出的查询获得的结果保存在本地的 Cache 当中,当发出一个新的查询时,首先根据缓存中的各种参数决定当前的缓存项是否有效,如果查询的结果包含在缓存当中,则直接从本地获得结果. 否则,MU 向 MS 发出查询,获得新的结果.

定义 3. HA-DMDB 系统中的趋势查询. 除了目前已有的对于 Sensor Net 数据的查询形式以外^[13], HA-DMDB 还定义了基于数据变化趋势的查询结构:

```
SELECT ( attributes | Trend-Func( attributes ) )
FROM ( Dsensor )
WHERE ( predicate, )
DURATION ( time interval )
```

Trend-Func 是不同的趋势函数,例如曲线函数、线段拟合函数、Hash 函数等等,用来描述属性的变化趋势. 另外,在 WHERE 子句中增加描述误差范围的谓词,用来限制根据趋势计算得到的数据的可接受的误差范围.

3 HA-DMDB 基于数据模型趋势分析的 Cache 一致性管理策略

3.1 Sensor Net 上面的 Cache 管理策略

Sensor Net 中,根据系统采用的路由策略的不同,每个结点使用的收发查询或者数据的方法也不同. 但是,无论采取何种方法,为了提高查询的效率,每个 Sensor 结点都保存少量的缓存信息,包括以前查询请求获得的结果,以及曾经使用的路由信息等. 如果新到的一个查询和缓存当中的信息匹配,则可以直接返回结果. 但是以往的缓存策略都是保存一些确切的数据值,由于 Sensor 结点的存储容量非常有限,不可能保存太多的信息,因此这些缓存策略所起的作用很有限. 在 HA-DMDB 系统中,采用了与以往方法不同的缓存模型,它缓存的不再是具体的数值,而是某个数据源在一个时间段的变化趋势以及这种变化趋势的有效期的估计值. 当一个查询到达的时候,根据查询的数据源、时间区间和允许的误差范围判断本结点上面的缓存项是否满足查询. 每个 Sensor 结点具有两种不同的角色,一种

是作为数据源,产生自身监测到的数据. 另一种是作为路由过程中的中间结点. 下面描述了 Sensor 结点上面的数据管理策略.

Cache 管理策略 1.

WHILE(工作状态)

```
{
  IF (产生新的数据 Ditem) / *本地产生的数据 */
  {
    根据 F, Ditem 计算 ;
    IF ( > )
    {
      修改 F, 将 F, 以及新的有效期估计值 T 沿着先前建立的路径发送出去;
    }
  }
  IF (收到新的更新消息) / *收到需要转发的更新消息 */
  {
    IF (本地 Cache 中存在对应的数据项)
    {
      更新相应的 F 和有效期;
    }
    转发此消息;
  }
  IF (收到新的查询 Q)
  {
    IF ( Q 中的 attributes 包含在 Cache 中 )
      (此结点是 attributes 数据源)
    {
      IF ( F 对应的 值小于 Q 中的 )
      {
        根据 Cache 中的 F 计算 attributes 值;
        返回 F, ;
      }
      ELSE
      {
        转发查询或者本地获得新的数据并返回;
      }
    }
  }
}
```

3.2 MS 的 Cache 管理策略

由于 MS 具有 Sensor Net 数据访问代理和数据库服务器的作用,同时它的客户是经常移动和断接的 MU,因此 MS 的 Cache 管理策略主要考虑以下几方面:

(1) Cache 的一致性管理,通知 MU 本地 Cache 失效;

(2) 查询的处理.

考虑 MU 的移动性,一般 Cache 一致性策略采用广播失效报告的方法 (IR). 同时, MU 经常断接的特点使得设计 IR 广播的时候,通常广播一个时间窗口 (w) 长度的数据变化情况^[12]. 但是,目前的 IR 方法中很少考虑到数据的变化趋势. HA-DMDB 系统提出了基于数据模型变化趋势的一致性管理策略. 在这种策略中, IR 中包含的是时间窗口 w 内,若干个数据集变化趋势函数的变化信息.

Cache 管理策略 2.

IF(收到来自 Sensor 的更新消息 Mup)

```
{
    IF(本地 Cache 包含此数据项)
    {
        根据  $Mup$  包含的  $F$ , 以及新的有效期估计值  $T$  修改本地 Cache;
        IF (此数据项访问频率大于预先规定的阈值) && ( $T > IR$  广播周期)
            将变化记录放到 IR 队列当中;
    }
    ELSE
    {
        IF(Cache 缓冲区满)
            删除过期的和访问频率最低的数据项;
        将新的数据项加入本地 Cache;
        初始化此数据项的访问频率;
    }
}
IF(收到新的查询  $Q$ )
{
    IF ( $Q$  的  $D_{sensor} \cdot Attributes$  在本地 Cache 中) && ( $< Q.$ ) && ( $T$  没过期)
        返回 ( $F, T$ ); /* 返回新的趋势函数和有效期的估计值 */
    ELSE
        转发子查询到 Sensor Net 中;
}
```

IF(到达新的广播周期)

```
{
    根据路由表得到查询每个数据项的源结点;
    将 IR 队列按照数据项对应的源结点传播出去;
}
```

3.3 MU 上面的 Cache 维护、查询策略

MU 将每次查询的结果保存在本地 Cache 当中,如果新的查询能够在允许的精度范围内从本地获得,那么将节省大量的网络通信开销和能源消耗. 在 HA-DMDB 系统中, MU 的 Cache 当中保存的是每个数据源趋势变化函数 F 、有效期 T 、精度. 影响 MU 的 Cache 一致性管理策略的主要因素是 MU 的经常断接性和移动性. 当 MU 关机或者移动到 MS 不能通信的地方,它将不能收到 Cache 的更新信息. 因此, MU 要充分利用 Cache 的信息,构造各种信息源的趋势模型,尽可能满足在断接情况下的查询请求. 当在线的时候, MU 要收听来自 MS 的 IR 通知,更新 Cache 的内容.

Cache 管理策略 3.

IF(收到 IR)

```
{
    IF(IR 影响本地 Cache)
    {
        根据 IR 更新 Cache 相应数据项的各种参数;
        从查询队列取出未处理的查询;
    }
    ELSE
        根据路由信息转发此 IR;
}
IF (产生新的查询  $Q$ ) && ( $Q$  的  $D_{sensor} \cdot Attributes$  在本地 Cache 中)
{
    根据本地 Cache 的  $F$  计算查询的结果;
    IF ( $< Q.$ ) && ( $T$  没过期)
        从本地得到查询结果;
    ELSE IF(在线)
        向 MS 发出查询请求;
    ELSE
    {
        将  $Q$  放入查询队列;
        进入工作状态;
        向邻居结点请求 IR 队列的内容;
```

```

    }
}
IF(收到邻居结点 IR 查询请求)
{
    IF (本结点收到过相应时间窗口的 IR 广播)
        从本地 Cache 当中构造相应的 IR 内容返回;
    ELSE
        将此请求转发;
}

```

4 结论及将来工作

本文提出了 HA-DMDB 系统的体系结构以及数据模型,并提出了基于此模型的 Cache 一致性管理策略。这些策略充分考虑了以 Sensor Net 为基础的分布式移动数据库的特点,提高了数据的传输和访问效率。

今后将进一步加强对于 HA-DMDB 系统的查询以及优化策略的研究。

参 考 文 献

- 1 J Hill, R Szwedczyk, *et al.* System architecture directions for networked sensors. In: Proc of the Int'l Conf on Architectural Support for Programming Languages and Operating Systems. Cambridge, MA: ACM Press, 2000
- 2 G Pottie, W Kaiser. Wireless sensor networks. Communications of the ACM, 2000, 43(5): 43~50
- 3 A Woo, D Culler. A transmission control scheme for media access in sensor networks. In: Proc of the 7th Annual ACM/ IEEE Int'l Conf on Mobile Computing and Networking. Rome, Italy: ACM Press, 2001
- 4 W Ye, J Heidemann, D Estrin. An energy efficient MAC protocol for wireless sensor networks. In: Proc of IEEE Infocom. NY, USA: IEEE Press, 2002
- 5 B Karp, H Kung. Greedy perimeter stateless routing. In: Proc of Mobicom. Boston, MA: ACM Press, 2000
- 6 Anindo Banerjee. A taxonomy of dispersity routing schemes for fault tolerant real-time channels. In: Proc of ECMAST. Louvain-la-Neuve, Belgium: Springer, 1996. 129~148
- 7 Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin.

Directed diffusion: A scalable and robust communication paradigm for sensor networks. In: Proc of Mobicom. Boston, MA: ACM Press, 2000

- 8 K Ishida, Y Kakuda, T Kikuno. A routing protocol for finding two node-disjoint paths in computer networks. In: Int'l Conf on Network Protocols. Tokyo, Japan: IEEE Press, 1992. 340~347
- 9 John Heidemann, Fabio Silva, *et al.* Building efficient wireless sensor networks with low-level naming. In: Proc of the Symp on Operating Systems Principles. Alberta, Canada: ACM Press, 2001
- 10 Qun Ren, Margaret H Dunham. Using semantic caching to manage location dependent data in mobile computing. In: Proc of Mobicom. Boston, MA: ACM Press, 2000
- 11 S Dar, *et al.* Semantic data caching and replacement. In: Proc of VLDB. Bombay, India: Morgan Kaufmann, 1996
- 12 Guohong Cao. A scalable low-latency cache invalidation strategy for mobile environments. In: Proc of Mobicom. Boston, MA: ACM Press, 2000
- 13 Yong Yao, Johannes Gehrke. Query processing for sensor networks. In: Proc of the 2003 CIDR Conf. Asilomar, CA: ACM Press, 2003



石胜飞 男,1972年生,博士研究生,主要研究方向为分布式移动数据库技术、无线传感器网络。



李建中 男,1950年生,教授,博士生导师,主要研究方向为数据库、Web技术、数据仓库、移动数据库等。



王朝坤 男,1976年生,博士研究生,主要研究方向为音乐数据库、Peer-to-Peer技术等。



李金宝 男,1969年生,副教授,主要研究方向为分布式计算、无线传感器网络等。