

大规模无线传感器网络(ϵ, δ)-近似计数算法

朱敬华^{1,2}, 管学敏^{1,2}

(1. 黑龙江大学 计算机科学技术学院, 黑龙江 哈尔滨 150001;
2. 黑龙江省数据库与并行计算重点实验室, 黑龙江 哈尔滨 150001)

摘 要: 研究了大规模无线传感器网络中的近似计数问题, 提出 2 个基于数字二叉树 (DBT, digital binary tree) 协议的近似计数算法 DBT-ACA 和 DBT-BACA。算法能够以 $O(\log \log n)$ 的时间复杂性返回 (ϵ, δ) -精度保证的近似计数结果。DBT-BACA 采用了二分搜索、逐层转发和延迟响应等技术, 有效地减少了查询时间和数据通信量。理论分析和实验结果表明, 提出的算法在近似结果的精准度、时间效率和能量开销等方面均优于现有的近似计数算法。

关键词: 无线传感器网络; 数据聚集; 近似算法; 数字二叉树

中图分类号: TP393.01

文献标识码: A

文章编号: 1000-436X(2013)06-0174-10

(ϵ, δ)-approximate counting algorithm for large scale wireless sensor networks

ZHU Jing-hua^{1,2}, GUAN Xue-min^{1,2}

(1. Department of Computer Science and Technology, Heilongjiang University, Harbin 150001, China;
2. Key Laboratory of Database and Parallel Computing Heilongjiang Province, Harbin 150001, China)

Abstract: The problem of approximate counting for large scale wireless sensor networks was studied. Two approximate counting algorithms, DBT-ACA and DBT-BACA, based on DBT (digital binary tree) protocol were also proposed. The algorithms presented could attain the counting result in $O(\log \log n)$ time while meeting the (ϵ, δ) accuracy requirement. DBT-BACA exploits binary search, level-by-level forwarding and delay response technique to effectively reduce the query delay and transmission cost. Theoretical analysis and experimental results show that the proposed algorithms outperform existing approaches in terms of estimation accuracy, time efficiency and energy cost.

Key words: wireless sensor networks; data aggregation; approximate algorithms; digital binary tree

1 引言

随着通信技术、嵌入式技术以及传感器技术的飞速发展和日益成熟, 无线传感器网络 (WSN, wireless sensor network) 在国防军事、入侵检测、环境监控、交通管理和珍惜动物保护等领域都有广泛的应用前景^[1]。大规模传感器网络通常部署在广阔的区域, 由一个或多个汇聚节点和成千上万个廉价传感器节点组成。具有感知能力、计算能力和通

信能力的传感器节点周期地采集温度、湿度、光照等环境数据, 以无线传输的方式将数据路由到汇聚节点, 为用户提供查询结果或决策分析数据。

目前, 能量是制约传感器网络服务于实际应用的主要原因。传感器网络的物理层、MAC 层、路由层、应用层都涉及能源高效的问题, 人们在这些方面开展了很多研究, 例如多跳路由、睡眠机制、网内处理等。数据聚集 (如 SUM、COUNT、AVG) 不仅是常见的查询操作, 也是有效的节能手段。基

收稿日期: 2012-07-15; 修回日期: 2012-10-23

基金项目: 国家自然科学基金青年基金资助项目 (61100048); 哈尔滨市科技创新人才专项基金资助项目 (2011RFQXG028); 黑龙江省高校科技创新团队建设计划基金资助项目 (2013TD012)

Foundation Items: The National Science Foundation for Young Scholars of China (61100048); The Foundation of Harbin Technological Innovation (2011RFQXG028); The University Science and Technology Innovation Team Program of Heilongjiang Province (2013TD012)

于生成树的聚集算法^[2]能够以较少的数据传输获得精准的聚集结果, 但是此类算法只适合链路稳定的静态传感器网络。如果由于链路不稳定或节点移动导致网络拓扑随时间动态变化, 则聚集树的维护开销将会相当大。分布式数据聚集算法^[3]虽然不需要维护聚集树, 但为了获得精确的全局聚集结果往往需要大量的数据交换, 并且算法的收敛时间也会延长。

事实上, 传感器网络数据由于测量误差、通信干扰和数据丢失等原因具有内在的不确定性, 即使是最好的算法也不能获得 100% 准确的聚集结果; 另外, 某些决策分析类的应用并不需要精准的聚集结果, 使用近似估计值也可以达到分析目的。因此, 为节省能量, 人们提出各种近似聚集算法^[4-6]。例如, Sketch^[4,5]对数据进行压缩减少数据的通信量, 但仍需要全网数据参与计算; 时空相关近似聚集算法^[7-12]维护数据的预测模型, 动态网络中数据预测模型变化频繁, 维护开销很大; 基于抽样的近似算法^[13-18]根据样本数据估计聚集值, 有效地减少了数据的传输量, 可以满足用户任意精度的需求, 缺点是对于多重集合, 估计值受样本数据中元素重复出现频率的影响, 而且算法的时间复杂性和通信复杂性随样本容量的增加而增加。

计数 (COUNT) 是一种重要的聚集操作, 在移动对象跟踪和珍惜动物保护等应用中, 实时地或周期性地监控指定区域内移动对象的数量, 可以为分析野生动物的聚集规律和栖息习惯等提供依据; 另外, COUNT 对传感器网络的系统功能监控也很重要, 定期统计网络中活节点的数量比例, 如果计数值低于预设阈值则及时补充传感器节点; 而且 COUNT 操作也为其他类型的聚集 (如 AVG) 提供基础。因此, 本文主要研究大规模动态传感器网络中能源高效的近似计数算法, 针对上述已有算法存在的问题, 提出了一种基于数字二叉树 (DBT, digital binary tree) 协议的 (ϵ, δ)-近似计数算法, 该算法能够以 $O(\log \log n)$ 时间复杂度为用户返回满足精度需求的计数结果。为适应网络的大规模性和动态性, 算法中采用了逐层转发和延迟响应的方法, 从而进一步减小了网络的通信能耗。理论分析和实验结果表明, 所提出的算法在近似结果的精准度、时间效率和能量开销等方面均优于现有的近似算法。

本文的主要贡献包括以下 3 点。

1) 提出了 $O(\log \log n)$ 时间复杂度的 (ϵ, δ)-近似计数协议 DBT, 能够高效地解决大规模动态传感

器网络中的近似计数问题。

2) 提出逐层转发和延迟响应方法, 有效地减少查询下发和返回结果时的能量消耗。

3) 对 DBT 协议的性能进行了理论分析, 通过实验验证了所提出算法的节能效率和时间效率。

2 相关工作

基于 Sketch 的近似计数方法^[4,5]借助位图数组存储节点数量, 每个节点使用多个散列函数对数组位进行标记, 如果数组中的标记位数是 b , 则节点总数可以估计为 2^b 。

基于抽样技术的近似计数方法在传统数据库、数据流和 P2P 系统中得到广泛应用。文献[12,13]针对传统数据库提出了基于顺序抽样的集中式近似聚集算法。文献[14]针对数据流提出基于 Bernoulli 抽样的近似聚集算法。上述算法的集中式特征和多次抽样性都不适合大规模分布式传感器网络。文献[16]针对 P2P 系统提出了基于抽样的分布式近似聚集算法, 以随机游走的方式进行抽样从而提高算法的效率。但是 P2P 系统一般没有能量约束, 数据相对稳定, 不需要样本更新和维护, 这与 WSN 有很大的区别, 因此并不适合 WSN。文献[17]针对 WSN 提出基于 Bernoulli 抽样的 (ϵ, δ)-近似聚集算法, 可以满足用户任意的精度需求, 同时提出精度需求变化时聚集结果的维护算法。但是, 基于抽样技术的近似计数结果受样本数据中元素重复出现频率的影响, 而且算法的时间复杂性和通信复杂性随样本容量的增加而增加。CountTorrent^[3]是适合传感器网络拓扑动态变化时的一种分布式数据聚集协议, 通过标号分配和数据合并 2 个阶段分布式的计算聚集结果。主要优点是保证了动态网络中的数据可靠性和聚集结果的精度, 但其缺点是数据通信量大, 并且算法的收敛时间长。

多播网络中的节点计数问题已经得到广泛深入的研究, 基本方法是在多播树中以泛洪方式传播查询请求, 每个节点都直接回应根节点。这种方法引起“反馈爆炸”的现象, 即在根节点附近产生严重的通信瓶颈。文献[18]中解决通信瓶颈的方法是为节点设置反馈概率, 节点总是以一定的反馈概率响应查询。因此可以用反馈节点个数和相应的反馈概率估计网络中的节点总数。通常需要多轮计算, 初始反馈概率设置较低, 随后每轮概率逐渐增加。

本文研究大规模动态 WSN 中的近似计数问题，提出的基于 DBT 协议的近似计数算法能够将估计性能提高到 $O(\log \log n)$ 。

3 问题的定义

本文研究了大规模动态传感器网络中活节点的近似计数问题。网络由一个或多个汇聚节点 (sink) 以及成千上万的传感器节点组成。节点失效、新节点加入和节点的随机移动都可以导致网络拓扑随时间的动态变化。因此要求近似计数算法不仅能够满足用户的精度需求，还应该适应网络的动态性和大规模性。另外，在资源严重受限的传感器网络中，节点的能量十分珍贵，无线传输能耗在各种能耗中占很大的比例，因此要求算法具有较低的通信开销。同时，较短的查询响应时间也是近似算法的一个重要性能指标，算法的时间复杂性不应该随网络规模的增大而快速增长。综上所述，快速、准确和能源高效是本文设计近似计数算法的 3 个主要目标。

近似结果的精准度可以用相对误差上限 ϵ 和失误概率上限 δ 来度量。假设某时刻网络中实际的活跃节点数是 n ，近似算法得到的估计值是 \tilde{n} ，则近似结果的精准度由定义 1 给出。

定义 1 ((ϵ, δ) -近似计数值): 给定任意的 ϵ ($\epsilon \geq 0$) 和 δ ($0 \leq \delta \leq 1$)，若估计值 \tilde{n} 满足 $\Pr\{|\tilde{n} - n| \leq \epsilon n\} \geq 1 - \delta$ ，则称 \tilde{n} 是 n 的 (ϵ, δ) -近似计数值。

例如，传感器网络监控区域内活跃节点个数是 20 000，用户的精度需求是 $\epsilon = 0.05$ 和 $\delta = 0.01$ ，则 (ϵ, δ) -近似算法得到的估计值在区间 [19 000, 21 000] 的期望概率不小于 0.99。传感器网络中活跃节点的近似计数问题可以描述如下。

输入：用户精度需求 ϵ ($\epsilon \geq 0$) 和 δ ($0 \leq \delta \leq 1$)
 输出：满足定义 1 的 (ϵ, δ) -近似计数值

本文算法基于以下假设：1) 允许节点随机移动，通过 GPS 定位获取位置信息，所有节点时钟同步，具有唯一的节点 ID；2) 汇聚节点 Sink 位置固定，网络初始化阶段 Sink 将位置广播给其他节点；3) 节点通信半径为 R ，干扰半径为 $2R$ ；4) 节点采用双工通信方式，可以同时发送数据和接收数据。

4 DBT 协议的数学基础

4.1 DBT 协议的设计

DBT 协议的目的是估计多重集 M 中不同元素

的个数，本质上是一个概率计数过程，其主要思想是：设 $hash(x) \rightarrow [0 \cdots 2^{L-1}]$ 是随机均匀散列函数，将 M 中任意非负整数 x 映射为长度为 L 的二进制数字串。 $bit(x, k)$ 表示 x 的二进制数字串的第 k 位，则 $x = \sum_{k \geq 0} bit(x, k) 2^k$ 。设 $\rho(x)$ 为 x 的二进制串中最低

$$\text{位“1”的位置，则 } \rho(x) = \begin{cases} \min bit(x, k) \neq 0, & x > 0 \\ L, & x = 0 \end{cases}$$

因为散列函数 $hash(x)$ 均匀的，所以二进制串 $0^k 1 \cdots$ 的出现概率为 $1/2^{(k+1)}$ 。利用位图向量 $Bitmap[0 \cdots 2^{L-1}]$ 存储所有元素的二进制串。 $Bitmap[k]$ 为 1，当且仅当多重集 M 中存在散列后形式为 $0^k 1$ 的元素。因此，可以用位图中处于最右侧的“0”的位置近似估计出 M 中元素个数。

在 DBT 协议中， $Bitmap[0 \cdots 2^{L-1}]$ 可以用一棵高度为 L 的二叉树表示。如图 1 所示，DBT 中有一个根节点，多个分支节点和叶节点。叶节点以外的其他节点都有 2 个分支：0-分支和 1-分支。从根节点到叶节点的每条路径对应一个唯一的二进制串，为叶节点的编码。传感器节点具有唯一的出厂 ID，通过随机均匀散列函数将 ID 映射为二进制编码。DBT 中的叶节点代表网络中的传感器节点，叶节点的编码与传感器节点的 ID 对应。

假设网络中有 4 个活跃节点，节点 ID 通过均匀随机散列函数 $hash(x) \rightarrow [0 \cdots 2^{L-1}]$ 映射为长度为 $L=4$ 的二进制串：0000、0101、1001、1110。

如图 1 所示，DBT 中的黑色叶子节点代表 4 个活跃节点，白色叶节点代表非活跃节点。下面给出 DBT 协议中的相关概念。

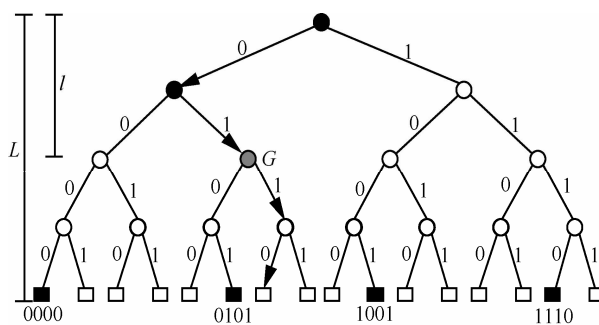


图 1 数字二叉树 DBT

定义 2 节点的层次。令 $level(node_i) = level(node_j) + 1$ ，节点 j 是 i 的父节点，根节点的层次为 0。

定义 3 白子树和白节点。设 v 是 DBT 中的节点，如果以 v 为根的子树， $Sub_T(v)$ 中不含黑色叶

子, 则称 $Sub_T(v)$ 为白子树, 称 v 为白节点。

定义 4 黑子树和黑节点。设 v 是 DBT 中的节点, 如果以 v 为根的子树, $Sub_T(v)$ 中至少含有一个黑色叶子, 则称 $Sub_T(v)$ 为黑子树, 称 v 为黑节点。

为了估计活跃节点个数, 汇聚节点 Sink 产生一个 L bit 的随机二进制串 s , 本例中 $s=0110$, 唯一标识一条从根到叶子的路径。在 DBT 中, 每个节点都有两棵子树, 把随机路径 s 经过的子树标记为 $Sub_T_s^i$, 随机路径 s 未经过的子树记为 $Sub_T_{\sim s}^i$ 。由定义 3 和定义 4 可知, 如果 i 是白节点, 则 $Sub_T_s^i$ 和 $Sub_T_{\sim s}^i$ 必然都是白子树; 否则, $Sub_T_s^i$ 或 $Sub_T_{\sim s}^i$ 必有一个是黑子树。

定义 5 灰节点。设节点 i 是随机路径 s 上的一个黑节点, 若 $Sub_T_s^i$ 是白子树, 即路径 s 上除 i 以外的其他黑节点的 Sub_T_s 都是黑子树, 并且节点 i 是路径 s 上层次最低的黑节点, 则称 i 为灰节点。

如图 1 所示, 节点 G 是灰节点。直观上, DBT 中白色叶节点的比例越大, 则灰节点的层次就越高 (层次 0 为最高层), 因此可以用灰节点的层次 l 估计传感器网络中活跃节点的个数。本文将在 4.2 节通过分析证明 DBT 中灰节点的层次可以作为活跃节点的近似计数结果。为了确定 l , Sink 产生随机路径 s , 本例中 $s=0110$, 随后 Sink 向网络发出随机路径的前缀查询 $0***$, 等待传感器节点的响应。由于本例中 4 个活跃节点的随机二进制编码分别为 0000、0101、1001 和 1110, 所以编码为 0000 和 0101 的传感器节点对 Sink 回应。Sink 接收到节点的回应, 继续发送新的前缀码 $01**$, 编码为 0101 的节点回应。Sink 继续此过程, 直到没有节点回应为止。本例中, Sink 发出 $011*$ 后, 没有节点的前缀码与之匹配, 因此 Sink 未收到任何回应。Sink 由此推断必存在某黑色叶子的前缀码与 $010*$ 匹配, 在 DBT 中找到随机路径 s 中的灰节点 G , G 的层次是 2。本文将在 4.2 节研究如何通过灰节点的层次 l 估计活跃节点的个数以及估计的精度。

实际应用中, L 设置为 32 bit 就可以构建一个大规模 DBT, 容纳数亿个黑色叶子。沿 32 bit 的随机路径进行前缀码查询就能够得到灰节点的层次 l , 进而得到传感器网络活跃节点的计数结果。需要说明的是, DBT 只是用于将传感器节点分组以及对各分组进行计数估计的一种概念数据结构, Sink 不必创建和维护任何 DBT。

4.2 DBT 协议的分析

由 4.1 节可知, DBT 中灰节点的层次 l 对于活跃节点个数的估计有很重要的作用。可以利用 l 估计 DBT 中黑色叶子的个数, 即网络中活跃节点的个数。下面通过理论分析研究如何使用 DBT 协议中灰节点的层次 l 对活跃节点个数进行估计以及如何保证用户的精度需求。

定理 1 灰节点的层次 l 的数学期望是 $E(l) \approx \text{lb}(\varphi n)$, 其中, $\varphi=1.26$ 。灰节点的层次 l 的标准方差是 $\sigma(l) \approx 1.87$ 。

证明 定义随机变量 l 为灰节点 i 在随机路径 s 中的层次。根据 4.1 节灰节点的定义可知 l 的概率为 $\Pr(l) = \Pr\{IsWhite(Sub_T_s^i) \wedge IsBlack(Sub_T_{\sim s}^i)\}$ (1)

设 DBT 中的叶子以 p ($0 \leq p \leq 1$) 概率为白色, 以 $1-p$ 的概率为黑色, 黑色叶子代表网络中的活跃传感器节点。灰节点 i 的两棵子树各包含 2^{L-l-1} 片叶子, 传感器节点的 NID 通过随机均匀散列函数 $hash(x)$ 映射为 DBT 的叶子, 此过程是独立的, 因此有

$$\Pr\{Sub_T_s^i \text{ 是白子树} \} = p^{2^{L-l-1}} \quad (2)$$

$$\Pr\{Sub_T_{\sim s}^i \text{ 是黑子树} \} = 1 - p^{2^{L-l-1}} \quad (3)$$

$$\Pr(l) = p^{2^{L-l-1}} (1 - p^{2^{L-l-1}}) \quad (4)$$

根据期望的线性性, l 的期望为

$$E(l) = \sum_{k=1}^L k \Pr(k) = -Lp^{2^L} + \sum_{k=0}^{L-1} p^{2^k} \quad (5)$$

由于 $p = (1 - 2^{-L})^n \approx e^{-n2^{-L}}$, 故

$$E(l) = -Le^{-n} + \sum_{k=0}^{L-1} e^{-n2^{-k-1}} \approx \sum_{k=0}^{L-1} e^{-n2^{-k-1}} \quad (6)$$

利用 Mellin 变换对式 (6) 化简, 得到近似期望值为

$$E(l) \approx \text{lb}(\varphi n) \quad (7)$$

其中, $\varphi=1.26$ 。

又由标准方差的定义可得

$$\sigma(l) = \sqrt{\text{Var}(l)} = \sqrt{\sum_{k=1}^L [k - E(l)]^2 \Pr(k)} \quad (8)$$

利用 Mellin 变换得到标准差的近似值: $\sigma(l) \approx 1.87$ 。证毕。

由定理 1 可知, 利用 DBT 中灰节点层次 l 的期望, 可以估计网络中活跃节点的个数。虽然 l 的实际值与期望值 $E(l)$ 之间存在偏差, 但根据大数中心定律, 多次实验得到的 l 的均值近似等于 l 的期望。

因此定义随机过程 $\bar{l} = \frac{1}{k} \sum_{i=1}^k l_i$ 为 k 次独立实验的均值。Sink 随机产生估计路径 s , 传感器节点的编码随机分布, 因此获得 l_i 的过程是独立随机过程, 则下面 2 个公式成立。

$$E(\bar{l}) = E\left(\frac{1}{k} \sum_{i=1}^k l_i\right) = \frac{1}{k} E\left(\sum_{i=1}^k l_i\right) = E(l) \quad (9)$$

$$\sigma(\bar{l}) = \sqrt{Var(\bar{l})} = \sqrt{Var\left(\sum_{i=1}^k l_i\right) / k^2} = \sigma(l) / \sqrt{k} \quad (10)$$

根据定理 1 和式 (7) 容易验证定理 2 成立。

定理 2 活跃节点个数的近似值为

$$\tilde{n} = \phi^{-1} \times 2^l = \phi^{-1} \times 2^{\frac{1}{k} \sum_{i=1}^k l_i} \quad (11)$$

由式 (10) 可知, l 的方差随 k 的增大而减小, 因此可以通过多轮执行获得接近实际值的计数结果。下面通过定理 3 的证明分析执行轮数与精度之间的关系。

定理 3 为满足用户 (ϵ, δ) -近似计数查询, 所需要的随机路径前缀查询次数 k 至少为 $\max\{[-c\sigma(l)/\text{lb}(1-\epsilon)]^2, [c\sigma(l)/\text{lb}(1+\epsilon)]^2\}$, 其中, $\epsilon (\epsilon \geq 0)$ 是相对误差上限, $\delta (0 \leq \delta \leq 1)$ 是失误概率上限, c 是常数。

证明 定义随机变量

$$X = (\bar{l} - \mu) / \sigma \quad (12)$$

根据中心极限定理, X 服从均值为 $\mu = E(\bar{l}) = \text{lb}(\phi n)$, 方差为 $\sigma = \sigma(\bar{l}) = \sigma(l) / \sqrt{k}$ 的标准正态分布。因此, 随机变量 X 的概率密度函数为

$$F(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{t^2}{2}} dt \quad (13)$$

给定误差概率 δ , 存在常数 c 使得下式成立。

$$\Pr\{-c \leq X \leq c\} = 1 - \delta = \text{error_}f(c/\sqrt{2}) \quad (14)$$

其中, $\text{error_}f$ 是高斯误差函数。则近似计数的精度需求可以表示为

$$\begin{aligned} \Pr\{|\tilde{n} - n| \leq \epsilon n\} &= \Pr\{(1-\epsilon)n \leq \tilde{n} \leq (1+\epsilon)n\} \\ &= \Pr\{(1-\epsilon)n \leq \phi^{-1} \times 2^{\bar{l}} \leq (1+\epsilon)n\} \\ &= \Pr\{\text{lb}\phi(1-\epsilon)n \leq \bar{l} \leq \text{lb}\phi(1+\epsilon)n\} \end{aligned} \quad (15)$$

由式(12)和式(14)得到 2 个必须满足的条件。

$$(\text{lb}\phi(1-\epsilon)n - \mu) / \sigma \leq -c \quad (16)$$

$$(\text{lb}\phi(1+\epsilon)n - \mu) / \sigma \geq c \quad (17)$$

再由式 (9)、式 (10) 和式 (17) 得到算法执行轮数 k 的下界为

$$k \geq \max\{[-c\sigma(l)/\text{lb}(1-\epsilon)]^2, [c\sigma(l)/\text{lb}(1+\epsilon)]^2\} \quad (18)$$

证毕。

5 基于 DBT 的 (ϵ, δ) -近似计数算法

通过第 4 节的介绍可知, 利用 DBT 结构可以对大规模动态传感器网络中活跃节点个数进行近似估计, 本节首先介绍基于 DBT 协议的基本近似计数算法 DBT_ACA, 包括 Sink 端算法和节点端算法; DBT_ACA 中 Sink 采用广播方式下发查询, 节点通过多跳路由向 Sink 返回响应分组, 因此能耗比较高。5.2 节介绍了一种改进算法 DBT_BACA, 利用二分搜索技术将算法的时间复杂性改进到 $\log\log n$, 并提出逐层转发和延迟响应技术进一步减少网络的能耗。

5.1 DBT_ACA 算法

图 2 描述了 Sink 端的基于 DBT 协议的 (ϵ, δ) -近似计数算法。算法的输入是置信区间 $\epsilon (\epsilon \geq 0)$ 和误差概率上限 $\delta (0 \leq \delta \leq 1)$ 。Sink 首先根据定理 3 的式 (18) 计算出保证用户查询精度需求执行轮数 k (图 2 的 1)), 然后在每轮中执行 3) ~11)。在第 i 轮, Sink 随机生成一条估计路径 s , 按照由低到高的顺序依次设置随机路径 s 的前缀码 4)~6), 然后将 s 和前缀码掩码 mask 广播给网络中的所有节点。如果 Sink 在等待延迟 T 内收到任何节点的响应分组, 则递增前缀码掩码重复上述查询过程; 否则, 存储

输入: 置信度 ϵ , 误差概率 δ 输出: 近似计数值 1) $k \leftarrow \max\{[-c\sigma(l)/\text{lb}(1-\epsilon)]^2, [c\sigma(l)/\text{lb}(1+\epsilon)]^2\}$ 2) $L \leftarrow 32$ 3) for $i \leftarrow 1$ to k do 4) 生成随机路径 s 5) for $j \leftarrow 1$ to L do 6) 设置掩码的高 j bit 7) Broadcast (s, mask) 8) 如果在时间间隔 T 内未得到响应 9) $i \leftarrow L - j + 1$ 10) break 11) end if 12) end for 13) end for 14) 根据式 (11) 计算得到 \tilde{n}

图 2 Sink 端的基于 DBT 协议的 (ϵ, δ) -近似计数算法

此轮灰节点的层次 $l_i=L-j+1$ 。算法循环上述过程 k 次，得到 k 个灰节点的层次值，最后根据定理 2 的式 (11) 返回网络中活跃节点个数的估计值。

DBT_ACA 中，Sink 递增地查询随机路径 s 的前缀码，这相当于在 DBT 中沿随机路径 s 自顶向下地搜索灰节点。本文使用图 3 中的例子分析该搜索过程。给定随机路径 s ，用 $node_i^s$ 表示路径 s 中层次为 i 的节点。图 2 算法从 $node_0^s$ 开始搜索，然后沿着路径 s 进行递增前缀搜索，直到发现灰节点 $node_3^s$ 。在 DBT 中确定灰节点的层次需要 $O(L)$ 次前缀搜索。

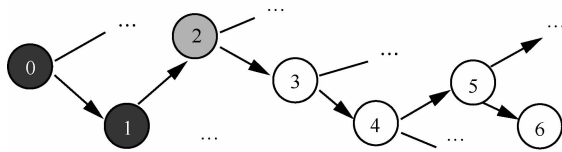


图 3 随机路径中灰节点的位置

图 4 描述了网络中的传感器节点接收到 Sink 的查询请求时的处理过程。节点根据接收到随机种子 $seed$ 和节点的 NID，使用随机散列函数获得一个 L bit 的随机二进制编码（图 4 中的 2）。如果该节点的编码与接收到的查询掩码相匹配，即接收到的掩码是节点编码的前缀码，则该节点立刻向 Sink 节点回应，回应分组通过多跳转发到达 Sink；否则，该节点保持沉默，不作任何回应，等待接收其他查询消息。

```

1) 接收查询消息 ( $s, mask$ )
2)  $code \leftarrow Hash(NID)$  /* 计算随机编码 */
3) while TRUE do
4) if  $code \wedge mask = s \wedge mask$  then
5) 响应 Sink
6) else
7) 不响应 Sink
8) end if
9) end while
    
```

图 4 网络中的传感器节点接收到 Sink 的查询请求时的处理过程

当网络中活跃节点数量规模很大时，DBT 的层次 $L \approx \log n$ ，因此图 2 算法的效率为 $O(\log n)$ 。在大规模传感器网络应用中，Sink 向网络中广播查询需要经过多个节点的转发到达远方节点，同样地，远方节点的查询响应分组也需要多跳路由到达 Sink。因此，图 2 算法和图 4 算法的通信开销和时间开销随着网络规模的增大快速增长。下面本文给出 2 个

改进算法，能够将效率改进到 $O(\log \log n)$ ，进一步地降低了网络的通信开销和时间复杂性。

5.2 DBT_BACA 算法

根据 4.1 节中的定义 3、定义 4 和定义 5，DBT 中的随机路径 s 经过的节点只有 3 种：黑节点、白节点和灰节点。对任意的节点 $node_i^s$ 和 $node_j^s$ ($i < j$)，式(19)是成立的。

$$(Sub_T_s^j \cup Sub_T_{\sim s}^j) \subseteq Sub_T_s^i \quad (19)$$

因此，随机路径 s 上的任意 2 个节点 $node_i^s$ 和 $node_j^s$ ($i < j$) 存在以下 3 种关系。

- 1) 若 $node_i^s$ 是白节点或灰节点，则 $node_j^s$ 是白节点。
- 2) 若 $node_j^s$ 是黑节点或灰节点，则 $node_i^s$ 是黑节点。
- 3) 随机路径 s 中只有一个灰节点。

上述分析表明随机路径 s 中的节点颜色呈现单调性：以灰节点为分界，高层节点全是黑节点，而低层节点全是白节点。如图 3 所示， $node_2$ 是灰节点， $node_{0,1}$ 是黑节点， $node_{3,4,5,6}$ 是白节点。利用随机路径 s 的单调性，采用二分搜索算法可以快速定位灰节点。Sink 生成随机路径之后，对随机路径的编码进行二分法匹配前缀，每次进行 $mid = \lceil (high + low) / 2 \rceil$ bit 前缀码匹配，根据是否接收到节点的响应调整上界 $high$ 和下界 low 。当 $high$ 和 low 重合时，表示遇到灰节点。算法的效率得到改进，只需要 $O(\log L) = O(\log \log n)$ 次前缀匹配就能确定灰节点的层次。

为减少广播查询和查询响应的通信开销，本文提出逐层转发、延迟响应的策略。假设 Sink 位于网络的中心， r 是节点通信半径， R 是传感器网络半径，网络逻辑上被划分成 R/r 个以 Sink 为中心，半径为 r 的同心圆，每个同心圆称为网络的一个层。对于 Sink 发出的每个前缀码，可能会有多个节点作出响应，Sink 接收到其中任何节点的响应都可以作出判断。因此 Sink 采取由内层向外层逐层查询方式：首先向 0 层节点即一跳邻居广播 ($s, mask$)，如果收到任何节点的回应 Reply，则更新前缀掩码 $mask$ 并重新广播新查询；否则，说明一跳邻居中无匹配节点，则等待一跳邻居节点将查询消息转发给两跳邻居，在等待过程中，收到任何节点的 Reply 就立即更新前缀码并广播新查询；如果 Sink 在最长等待延迟后都未收到任何响应，表明此次查询搜索

到了灰节点，从而可以确定此轮的灰节点层次。图 5 具体描述了采用二分搜索和逐层转发技术的近似计数过程。

```

输入：置信度  $\epsilon$ , 误差概率  $\delta$ 
输出：近似计数值  $\tilde{n}$ 
1)  $k \leftarrow \max \{[-c\sigma(l)/\text{lb}(1-\epsilon)]^2, [c\sigma(l)/\text{lb}(1+\epsilon)]^2\}$ 
2)  $L=32$ 
3) for  $i \leftarrow 1$  to  $k$  do
4) 生成随机路径  $s$ 
5)  $low \leftarrow 1, high \leftarrow L$ 
6) while  $low < high$  do
7)  $mid \leftarrow \lceil (low + high) / 2 \rceil$ 
8) 设置掩码的高  $j$  bit
9) One-Hop Broadcast ( $s, mask, hopcount$ )
10) 如果在时间间隔  $T$  内未得到响应
11)  $high \leftarrow mid$ 
12) else
13)  $mid \leftarrow low$ 
14) end if
15) end while
16)  $l_i \leftarrow low$ 
17) end for
18) 根据式 (11) 计算得到  $\tilde{n}$ 
    
```

图 5 采用二分搜索和逐层转发技术的近似计数过程

节点端的算法也做相应改进。在图 6 中，节点接收 Sink 的查询消息，如果随机编码和收到的前缀码匹配，需要响应 Sink，但是可能其他节点也要对 Sink 回应，则匹配查询的节点各自随机等待一段时间，其中，等待时间最短的节点对 Sink 响应，即所谓的延迟响应。其他节点如果在 w 时间后收到 Sink 的新查询，则放弃旧查询，处理新查询；否则，节点将旧查询转发给外层节点，逐层转发查询。

6 实验结果

本节将通过实验考察基于 DBT 的 (ϵ, δ) -近似计数算法的性能。笔者首先考察 DBT 在各种参数设置时的可调节精准度和时间效率；然后和基于 Bernoulli 的概率抽样算法 BSC 以及移动网络中的分布式聚集算法 CountTorrent 进行对比实验，从近似结果的精准度、时间效率和能量开销等方面验证 DBT 算法的性能。

6.1 实验设置和性能指标

笔者利用 NS2 模拟器来模拟规模为 50 000 个节点的传感器网络，节点随机部署在 1 000 m × 1 000 m 的矩形范围内，Sink 位于网络的中心。传感器节点的

通信半径设为 50 m，所有节点都可以通过现有的定位算法或 GPS 获得自身的位置，并且 Sink 的位置对所有节点可知。网络中的每个节点在出厂时都分配了一个唯一的节点 ID。节点采用双工方式通信，发送消息和接收消息可以同时进行。发送和接收 1 bit 信息的能耗分别为 0.014 mJ 和 0.005 7 mJ。由于传感器节点发送 1 bit 信息的能耗可以执行 1 000 条机器指令，因此本文实验对节点执行指令的能耗忽略不计。

```

1) 接收查询消息 ( $s, mask, hopcount$ )
2) if  $n\_hopcount < hopcount$ 
3) 放弃此次查询，接收下一个查询
4) else
5)  $code \leftarrow \text{Hash}(NID)$  /*计算随机编码*/
6) while TRUE do
7) if  $code \wedge mask = s \wedge mask$ 
8) 等待随机时间间隔  $t_{ui}$ 
9) 如果未收到新查询请求
10) 响应 Sink
11) 监听下一个查询
12) else
13) goto 2) /*处理新查询*/
14) else
15) 等待随机时间间隔  $t_{vi}$ 
16) 如果未收到新查询请求
17) Forward ( $s, mask, n\_hopcount$ )
18) else
19) goto 2) /*处理新查询*/
20) end if
21) end if
22) end while
23) end
    
```

图 6 改进的节点端算法

近似结果的精准度是本文算法最重要的性能指标。设 n 表示传感器网络中实际活跃节点的个数， \tilde{n} 表示算法得到的估计值，用 \tilde{n} / n 来度量算法的准确性，比值越接近 1，算法的准确性越好。近似算法的标准方差 $\sigma = \sqrt{E[(\tilde{n} - n)^2]}$ 可以度量算法的估计精度。一般地，标准方差越大，表明算法得到的估计值越分散，标准方差越小，表示算法的估计值越集中于实际值，算法的估计精度也就越高。因此，近似计数算法的目标是以较低的方差获得准确性接近 1 的近似结果。

另一个主要性能指标是算法的时间效率。传感器网络中的近似计数算法要求在满足用户的 (ϵ, δ) 精度需求的同时，能够快速返回查询结果。为了估计活跃节点的数量，DBT 进行了随机

编码的前缀查询，每次执行不同长度的前缀码查询，因此可以用前缀码查询次数度量算法的时间效率。算法的执行时间越短，对大规模网络的可扩展性越强。

最后一个性能指标是网络的能量开销。在资源特别是能量非常有限的传感器网络中，算法的能量开销决定了网络寿命的长短。本节实验主要对比 DBT 算法、BSC 算法以及 CountTorrent 算法的通信能耗。本文所有实验结果都是模拟执行 100 次后得到的平均值。

6.2 DBT 协议的性能

本文首先验证基于 DBT 协议的近似计数算法，通过增加执行的轮数获得任意精度需求的近似结果。第 1 组实验考察算法的准确性，第 2 组实验考察算法的标准方差，第 3 组实验考察算法的归一化标准方差。

如图 7 所示，DBT 通过增加执行的轮数 m 来改进算法的估计精度。当经过 32 轮之后，算法获得的精度接近 1，即估计值近似等于真实值。通过定理 3 的分析可知，算法需要执行的轮数只和用户提出的精度需求有关，因此根据用户的不同精度需求，可以调整算法的执行轮数。从图 7 中可以看到，算法的精度与网络中的节点规模无关，实际上，在未知网络中实际节点数量的前提下，可以利用 DBT 估计大规模网络中的活跃节点个数。

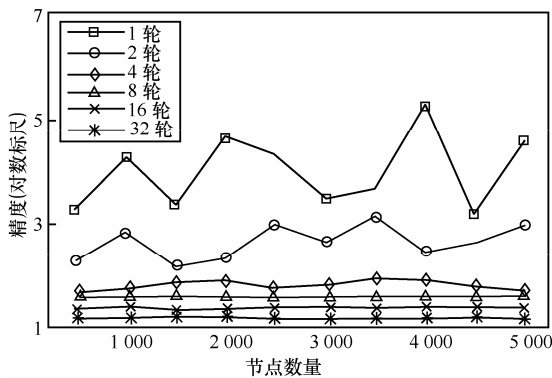


图 7 不同执行轮数下的精度变化

第 2 组实验考察近似结果的标准方差。图 8 和图 9 显示了 DBT 估计的标准方差和归一化的标准方差。执行轮数越多，标准方差越小。图 9 的归一化标准方差进一步验证了此结论，而且标准方差的大小不受节点数量的影响。前 3 组实验说明，DBT 通过调整执行的轮数可以满足用户任意精度的近似计数查询。

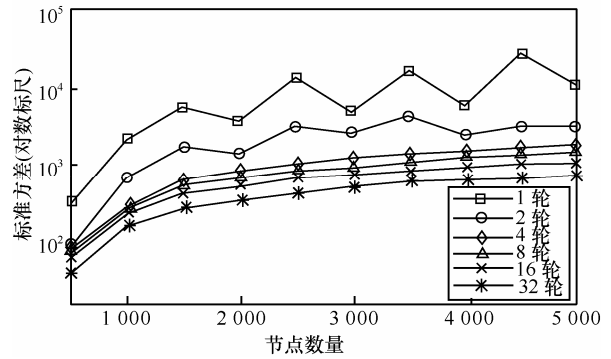


图 8 不同执行轮数下的标准方差变化

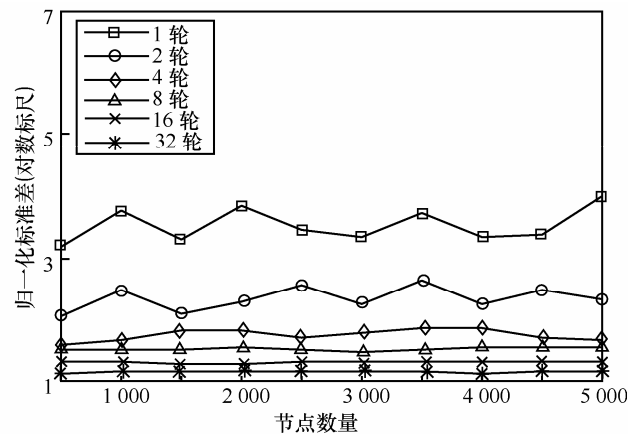


图 9 不同执行轮数下的归一化标准方差的变化

除了近似结果的精度，本文还考察了算法的执行时间效率。在实验中，DBT 的层次设置为 32，由于改进算法采用对分搜索进行前缀码查询，因此在每轮匹配中，Sink 只需要进行 5 次前缀码匹配查询即可。图 10 显示了执行 m 轮查询的总时间。

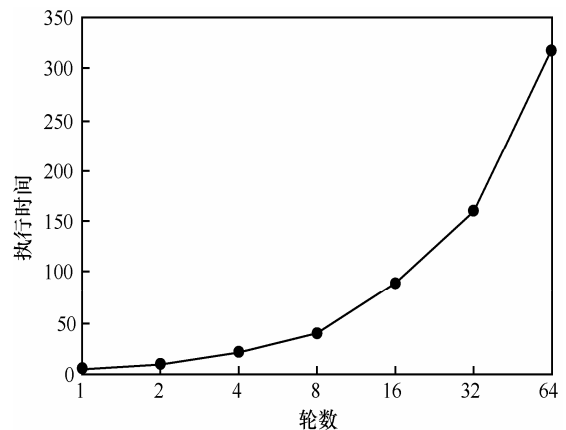


图 10 DBT_BACA 算法的执行时间变化曲线

6.3 性能对比

将 DBT 算法和基于 Bernoulli 的近似聚集算法 BSC 以及移动网络中的分布式聚集算法 CountTorrent

进行对比,考察算法的精度、查询处理时间和能耗,图 11 对比了 3 种算法的精度。可以看出,由于 CountTorrent 是精确算法,总是返回准确的查询结果,因此精度最高。而 BSC 算法和 DBT 算法的精度接近。

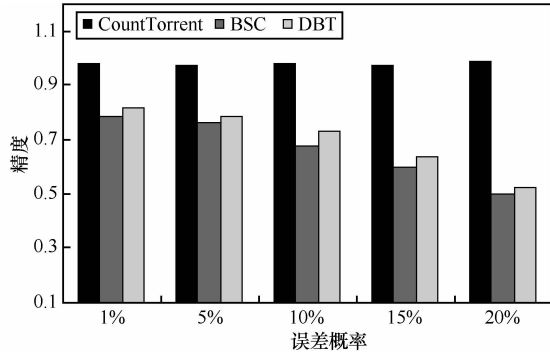


图 11 3 种计数算法的精度对比

图 12 显示了 3 个算法的处理时间。在相同精度需求下, DBT 算法的执行时间更短。这是因为 DBT 采取随机编码前缀匹配查询,而 BSC 基于采样方式进行数据聚集,执行时间和样本数量及网络规模均有关系。CountTorrent 为获得精确结果,分布式地查询处理,局部结果汇集成全局结果的收敛时间最长,处理时间与网络规模及节点的移动都有关系。

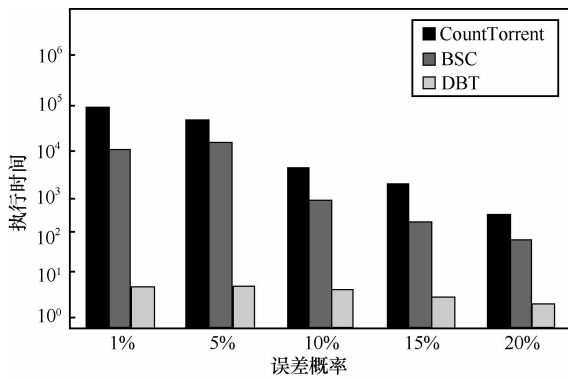


图 12 3 种计数算法的时间性能对比

图 13 显示了 3 个算法的能量消耗情况。DBT 采用随机前缀匹配查询,并不需要收集网络中的样本数据,又采用了逐层转发和延迟响应的技术,进一步减小了传输能耗。CountTorrent 的能耗最大,需要在全网络以类似于洪泛的方式计算聚集值,而且为获得精确结果,全网节点都需要参与计算,节点之间的通信量非常大。DBT 的能耗在三者之间是最小的。

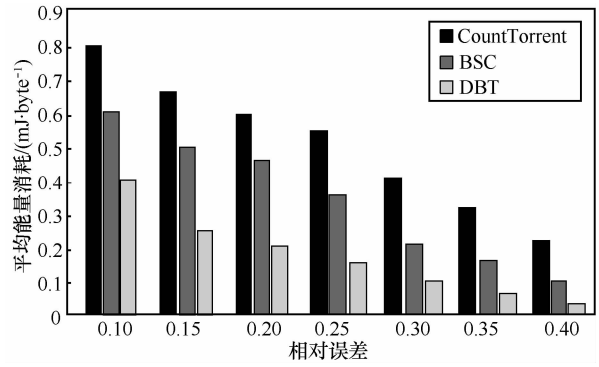


图 13 3 种计数算法的能量消耗对比

7 结束语

本文针对大规模动态传感器网络提出了一种基于 DBT 协议的 (ϵ, δ) -近似计数算法,该算法能够满足用户提出的任意精度需求。改进算法利用二分搜索技术使得时间复杂性降低至 $O(\log \log n)$,提出的逐层转发和延迟响应方法进一步地减小了网络的能耗。理论分析和实验结果表明,本文提出的算法能够有效地解决大规模动态网络中的近似计数问题,和现有的近似聚集算法以及分布式聚集算法相比,在精准度、时间效率和能量开销等方面均具有优越性。接下来将进一步研究大规模动态传感器网络中的其他近似聚集操作。

参考文献:

- [1] XU N. A survey of sensor network applications[J]. IEEE Communications Magazine, 2002, (40):120-135.
- [2] MADDEN S, FRANKLIN M J, HELLERSTEIN J M, et al. TAG: a tiny aggregation service for ad-hoc sensor networks[A]. Proc of the 5th Symp on Operating System Design and Implementation[C]. New York, USA, 2002. 131-146.
- [3] ABHINAV K, VISHAL M, DAN R. Count torrent: ubiquitous access to query aggregates in dynamic and mobile sensor networks[A]. Proc of the 5th International Conference on Embedded Networked Sensor Systems (Sensys 07)[C]. New York:ACM Press, USA, 2007. 137-152.
- [4] CONSIDINE J, LI F, KOLLIOS G, et al. Approximate aggregation techniques for sensor databases[A]. Proc of the 20th Int'l Conf on Data Engineering[C]. Washington: IEEE Computer Society Press, USA, 2004. 449-460.
- [5] NATH S, GIBBONS P B, SESHAN S, et al. Synopsis diffusion for robust aggregation in sensor networks[A]. Proc of the ACM Conf on Embedded Networked Sensor Systems[C]. New York: ACM Press, USA, 2004. 250-262.
- [6] HARTL G, LI B C. Infer: a bayesian inference approach towards energy efficient data collection in dense sensor networks[A]. Proc of the 25th IEEE Int'l Conf on Distributed Computing Systems[C]. Washington:IEEE Computer Society Press, USA, 2005. 371-380.
- [7] DELIGIANNAKIS A, KOTIDIS Y, ROSSOPOULOS N. Processing

- approximate aggregation queries in wireless sensor networks[J]. Information Systems, 2006, 31(8):770-792.
- [8] DELIGIANNAKIS A, KOTIDIS Y, ROUSSOPOULOS Y N. Hierarchical in-network data aggregation with quality guarantees[A]. Proc of the Int'l Conf on Extending Database Technology[C]. Washington: IEEE Computer Society Press, USA, 2004. 658-675.
- [9] CORMODE G, GAROFALAKIS M N, MUTHUKRISHNAN S, *et al.* Holistic aggregates in a networked world: distributed tracking of approximate quantiles[A]. Proc of the 2005 ACM SIGMOD Int'l Conf on Management of Data[C]. New York: ACM Press, USA, 2005. 25-36.
- [10] CHU D, DESHPANDE A, HELLERSTEIN J M, *et al.* Approximate data collection in sensor networks using probabilistic models[A]. Proc of the 22nd Int'l Conf on Data Engineering[C]. Washington: IEEE Computer Society Press, USA, 2006.48-59.
- [11] SILBERSTEIN A, PUGGIONI G, GELFAND A, *et al.* Suppression and failures in sensor networks: a Bayesian approach[A]. Proc of the 33rd Int'l Conf on Very Large Data Base[C]. New York: ACM Press, USA, 2007.842-853.
- [12] HAAS P J, SWAMI A N. Sequential sampling procedures for query size estimation[A]. Proc of the 1992 ACM SIGMOD Int'l Conf on Management of Data[C]. New York: ACM Press, USA, 1992. 341-350.
- [13] LARSON P, LEHNER W, ZHOU J R, *et al.* Cardinality estimation using sample views with quality assurance[A]. Proc of the 2007 ACM SIGMOD Int'l Conf on Management of Data[C]. New York: ACM Press, USA, 2007. 175-186.
- [14] CEMULLA R, LEHNER W, HAAS P J. Maintaining bernoulli sample over evolving multisets[A]. Proc of the 26th ACM SIGMOD-SIGACT-SIGART Symp on Principles of Database Systems[C]. New York: ACM Press, USA, 2007. 93-102.
- [15] BENJAMIN A, GAUTAM D, DIMITRIOS G, *et al.* Approximating aggregation queries in peer-to-peer networks[A]. Proc of the 22nd Int'l Conf on Data Engineering[C]. Washington: IEEE Computer Society Press, USA, 2006. 642-654.
- [16] BAYER K, HAAS P J, REINWALD B. On synopses for distinct-value estimation under multiset operations[A]. Proc of the 2007 ACM SIGMOD Int'l Conf on Management of Data[C]. New York, USA, 2007. 199-210.
- [17] CHENG S Y, LI J Z. (ϵ, δ)-approximate aggregation algorithm in wireless sensor networks[J]. Journal of Software, 2010, 21(8):1936-1953.
- [18] TIMUR F, DON T. Multicast Session Membership Size Estimation[R]. Technical Report, 1998.

作者简介:



朱敬华(1976-),女,黑龙江齐齐哈尔人,博士,黑龙江大学副教授、硕士生导师,主要研究方向为传感器网络、不确定性数据管理。

管学敏(1982-),女,黑龙江牡丹江人,黑龙江大学硕士生,主要研究方向为不确定数据查询处理。