

7-2-2020

Algorithms for Join Queries in Multiple Wireless Sensor Networks

Shangfeng Mo

1. Hunan University of Science and Technology, Xiangtan 411201, China;;

Yongjian Fan

2. Hebei University of Engineering, Handan 056038, China;;

Xiaoliang Wang

1. Hunan University of Science and Technology, Xiangtan 411201, China;;

Qingchun Yu

1. Hunan University of Science and Technology, Xiangtan 411201, China;;

See next page for additional authors

Follow this and additional works at: <https://dc-china-simulation.researchcommons.org/journal>



Part of the Artificial Intelligence and Robotics Commons, Computer Engineering Commons, Numerical Analysis and Scientific Computing Commons, Operations Research, Systems Engineering and Industrial Engineering Commons, and the Systems Science Commons

This Paper is brought to you for free and open access by Journal of System Simulation. It has been accepted for inclusion in Journal of System Simulation by an authorized editor of Journal of System Simulation.

Algorithms for Join Queries in Multiple Wireless Sensor Networks

Abstract

Abstract: The perception layer of Internet of Things (IoTs) is mainly composed of Multiple Wireless Sensor Networks (MWSNs). Data query processing is a very important research area in MWSNs. Join queries can monitor similar network environments in different positions. *In general join algorithm, each ordinary node sends its join attribute value to the database server. The database server finds out the result set which meets the join conditions. In distributed join algorithm, the ordinary nodes predict their filters of next round and the database server obtains the global, common filters. The nodes which meet the join conditions send their data to the database server and the database server obtains the final result set of matching nodes.* Experiments on real-world data set show that the algorithms outperform the typical algorithm.

Keywords

internet of things, multiple wireless sensor networks, join queries, filters

Authors

Shangfeng Mo, Yongjian Fan, Xiaoliang Wang, Qingchun Yu, and Yinglong Li

Recommended Citation

Mo Shangfeng, Fan Yongjian, Wang Xiaoliang, Yu Qingchun, Li Yinglong. Algorithms for Join Queries in Multiple Wireless Sensor Networks[J]. Journal of System Simulation, 2016, 28(4): 833-841.

多无线传感器网络中连接查询处理

莫尚丰¹, 范永健², 王晓亮¹, 余庆春¹, 李英龙³

(1.湖南科技大学, 湘潭 411201; 2.河北工程大学, 邯郸 056038; 3.浙江工业大学, 杭州 310014)

摘要: 物联网(IoTs: Internet of Things)感知层通常由多个无线传感器网络(MWSNs: Multiple Wireless Sensor Networks)组成。数据查询处理是多无线传感器网络中很重要的一个研究领域。连接查询能在不同的位置监视相似的网络环境。通用连接算法中, 各节点上报连接属性值给数据库服务器, 由数据库服务器找出符合连接条件的匹配节点结果集。分布式连接算法中, 普通节点预测自己的过滤器区间, 数据库服务器获得全局过滤器区间, 符合条件的普通节点上报信息给数据库服务器, 数据库服务器获得最终的匹配节点结果集。在真实数据集上验证了各算法, 并体现出了通用连接算法和分布式连接算法的优越性。

关键词: 物联网; 多无线传感器网络; 连接查询; 过滤器

中图分类号: TP311.13

文献标识码: A

文章编号: 1004-731X (2016) 04-0833-10

Algorithms for Join Queries in Multiple Wireless Sensor Networks

Mo Shangfeng¹, Fan Yongjian², Wang Xiaoliang¹, Yu Qingchun¹, Li Yinglong³

(1. Hunan University of Science and Technology, Xiangtan 411201, China;

2. Hebei University of Engineering, Handan 056038, China; 3. Zhejiang University of Technology, Hangzhou 310014, China)

Abstract: The perception layer of Internet of Things (IoTs) is mainly composed of Multiple Wireless Sensor Networks (MWSNs). Data query processing is a very important research area in MWSNs. Join queries can monitor similar network environments in different positions. *In general join algorithm, each ordinary node sends its join attribute value to the database server. The database server finds out the result set which meets the join conditions. In distributed join algorithm, the ordinary nodes predict their filters of next round and the database server obtains the global, common filters. The nodes which meet the join conditions send their data to the database server and the database server obtains the final result set of matching nodes.* Experiments on real-world data set show that the algorithms outperform the typical algorithm.

Keywords: internet of things; multiple wireless sensor networks; join queries; filters

引言

物联网广泛应用于安防、电力、交通、医疗等许多领域^[1]。物联网体系结构包括应用层、通信网

络层和感知层。其中, 感知层主要由多个无线传感器网络 (MWSNs: Multiple Wireless Sensor Networks), RFID 网络或其他网络构成。

通信网络层通过 2G, 3G, 4G 等移动通信网络及互联网把感知层原子网络(Atom network, AN)的数据和信息传输给数据库服务器, 由数据库服务器对数据进行处理, 并把处理后的信息发送给应用层不同的用户^[2]。



收稿日期: 2014-07-12 修回日期: 2014-11-17;
基金项目: 国家自然科学基金(61502421), 河北省自然科学基金(F2013402031), 湖南省教育厅资助科研项目(14C0436), 浙江省自然科学基金 (No.LY15F020026);
作者简介: 莫尚丰(1977-), 男, 湖南, 博士, 研究方向为无线传感器网络及物联网复杂查询技术。

<http://www.china-simulation.com>

物联网数据管理可以被看作一种新型分布式数据库系统,用户向物联网发送查询消息,物联网返回监视区域中满足查询要求的信息。数据查询处理是物联网中很重要的一个研究领域。目前,物联网中查询处理的算法还比较少。典型的查询处理算法包括: Top-k 查询、skyline 查询、连接(join)查询等。在这些查询技术中,连接查询能够将不同节点或位置的数据关联起来。典型的连接查询应用是在不同位置监视相似的网络环境。实际应用中,研究

人员希望找到来自不同湿地的感知数据的相似特征,从而分析这些区域环境变量的相似性。

例如:图 1 所示物联网的感知层由 6 个原子无线传感器网络(Atom-WSN)构成,每个原子无线传感器网络包括 1 个基站和若干个普通节点,用户希望连续地从 N 个原子无线传感器网络(图 1 中阴影部分的无线传感器网络 2, 3, 4 和 5)中找到连接属性值彼此近似相等的节点集。

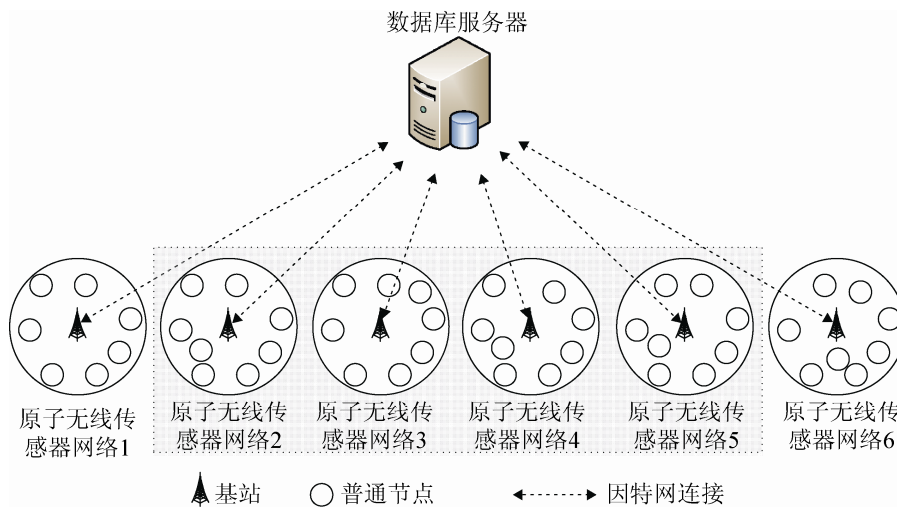


图 1 多无线传感器网络中连接查询处理实例

又如:用户希望连续地从 4 块湿地中找到环境温度(AT : Ambient Temperature) 采样值彼此近似相等(差值 $\tau \leq 1$ °C); 地表温度 (ST : Surface Temperature) 采样值彼此近似相等(差值 $\tau \leq 1$ °C); 相对湿度(RH : Relative Humidity) 采样值彼此近似相等(差值 $\tau \leq 1\%$) 的节点。假设一个采样周期是 10 分钟,该查询可以如下表示:

```
SELECT AN1.*, AN2.*, AN3.*, AN4.*
FROM AN1, AN2, AN3, AN4
WHERE |ANi. AT-ANj. AT| ≤ 1 °C
|ANi. ST-ANj. ST| ≤ 1 °C
|ANi. RH-ANj. RH| ≤ 1% (i≠j, i∈(1, 2, 3, 4),
j∈(1, 2, 3, 4))
SAMPLING PERIOD 10 min
```

这类查询的特点是,假设每个传感器(或 RFID)

节点能感知多种属性数据,用户希望连续地从多个原子网络中找到连接属性值近似相等的节点。我们称这类查询为多无线传感器网络中的多连接查询,它可以如下表示:

```
SELECT AN1.*, ..., ANn.*
FROM AN1, ..., ANn
WHERE |ANi.join-attrs-ANj.join-attrs| ≤ τ (i≠j,
i (1,2, ..., n), j ∈ (1,2, ..., n))
SAMPLING PERIOD p min
```

针对如上所述多连接查询,一个简单方法是在每一轮数据采集过程中,每个节点发送相关属性值给基站(The sink 或 The base station)。基站发送相关的数据给物联网通信网络层的数据库服务器,由通信网络层的数据库服务器找出符合连接条件的匹配节点结果集。这种方法我们称之为集中式算

法。在集中式算法中, 大部分节点都不属于最终多连接结果。而让每个节点发送数据给基站, 无疑会浪费很多能量。

本文关注多无线传感器网络中连接查询处理技术, 并提出了通用(General)连接算法和分布式(Distributed)连接算法。然后, 在真实数据集上验证了这些算法。

1 相关工作

目前, 物联网中查询算法还不多。以下是一些相关的物联网中查询算法。

文献[3]调查研究了物联网的概念、基于 SOA 的体系结构、物联网的应用领域、以及一些开放的研究问题等等。文献[4]分析了物联网中的查询挑战和查询类型, 讨论了物联网中查询处理方法并提出了一些可能的解决方案。在文献[5-6]中, 作者首先分析了物联网数据管理的挑战, 然后提出了基于海云的大规模传感器数据管理系统框架 SeaCloudDM。在这个框架的云数据管理层次, 每个元组有一个指针指向海计算节点(基站), 保证上层节点能访问底层节点。文献[7]中, 作者提出了物联网中的复杂事件处理方法。该方法基于查询计划, 也基于面向服务的体系结构(SOA)。

在无线传感器网络中, 目前已有一些连接查询算法[1,8-15]。REED^[8]和 NEJA^[9]算法处理的是外部连接问题, 让网络中产生的元组和存放在基站的外部表进行连接匹配。PEJA^[10]算法解决了不同区域连接查询问题。PEJA 算法由 3 个阶段组成, 包括: 初始阶段、递增连接处理阶段和清除阶段。TPSJ^[11]算法解决了窗口自连接问题, 目的是探测感兴趣的事件。SENS-Join^[12]处理了通用连接问题。有可能符合匹配条件的节点将发送连接属性值给基站, 基站根据收集到的连接属性值计算连接结果, 属于最终连接结果的节点将被通知发送完整的元组信息给基站。INJECT^[13]算法处理了 2 个区域连接问题。INJECT 提出了 baseJoin, coverJoin, sideJoin 等多种连接实现方案, 并根据不同的应用

场景选择不同的连接实现方案。在文献[1]中, 作者首先提出了 top-k 连接查询的概念。top-k 连接查询根据得分函数计算匹配结果的组合得分, 并报告组合得分结果最大(小)的 k 个匹配节点对。然后提出了基本 top-k 连接算法 BTJQ、集中式 top-k 连接算法 CTJQ、优化的集中式 top-k 连接算法 OCTJQ 和分布式 top-k 连接算法 DTJQ。文献[14]提出了一个基于过滤器的最近多个采样周期内单属性连接查询 SJQP 算法。SJQP 算法在每个节点上安装过滤器, 如果节点的采样值落入了过滤器区间, 其采样值将不会发送给基站。TJA^[15]算法是查找满足用户定义函数的 k 个最高排名结果, 例如: 查找过去 3 个月内 k 个最高平均温度的时间段。

这些查询算法^[8-15]主要应用于单个无线传感器网络中, 多无线传感器网络的体系结构和单个无线传感器网络不同, 这些查询算法不能直接应用于多无线传感器网络中, 但值得借鉴。

2 系统设置

假设多连接查询涉及 N 个原子无线传感器网络, 每个原子无线传感器网络中有 I 个普通传感器节点, 每个传感器节点采集 K 个属性值, 多连接查询连续地执行 R 轮。每个原子无线传感器网络中的普通节点以自组织方式构成网络, 并周期性地采集数据, 每一个采样周期称为一轮。

我们做如下假设: (1) 所有传感器节点是同构的, 都有相同的计算、存储和通信能力。部署后节点静止不动, 同时每个节点有自己唯一的 ID 号。(2) 每个原子无线传感器网络中只有一个基站, 该基站能被重新充电, 因此, 不考虑基站能量消耗问题。(3) 通信链路对称。如果节点 s_{ni} 能和节点 s_{nj} 通信, 反过来, 节点 s_{nj} 也能和节点 s_{ni} 通信。(4) 传感器节点能量有限, 并且不能补充能量。(5) 每个原子无线传感器网络中的普通节点只能和原子无线传感器网络中其他普通节点或基站相互通信; 基站能和通信网络层的数据库服务器通信, 并且不计算他们之间的通信消耗。

3 连接查询处理算法

3.1 定义

定义 1 $sink_n$ 代表第 n 个原子无线传感器网络 Atom-WSN $_n$ 中的基站。 s_{ni} 代表原子无线传感器网络 Atom-WSN $_n$ 中第 i 个普通传感器节点。

定义 2 v_{nikr} 代表原子无线传感器网络 Atom-WSN $_n$ 中, 节点 s_{ni} 采集的第 k 个属性值, 当前轮是第 r 轮。这些属性值可以是环境温度、地表温度、相对湿度等等。所有属性值 v_{nikr} 被封装在元组 t_{nir} 中, 节点 s_{ni} 在需要时发送元组 t_{nir} 给基站 $sink_n$ 。 $t_{nir} = \{v_{nikr} | k=1, 2, \dots, K\}$ 。

定义 3 原子无线传感器网络 Atom-WSN $_n$ 中, 节点 s_{ni} 在第 k 个属性值上设置了一个过滤器 $filter_{nikr}$, 当前轮是第 r 轮。每个过滤器 $filter_{nikr}$ 有上下边界 $[a_{nikr}, b_{nikr}]$, a_{nikr} 和 b_{nikr} 分别表示下边界和上边界。 fil_{nir} 代表原子无线传感器网络 Atom-WSN $_n$ 中, 节点 s_{ni} 在第 r 轮的所有过滤器, $fil_{nir} = \{filter_{nikr} | k=1, 2, \dots, K\}$ 。

定义 4 $FilterDis(filter_{nikr}, filter_{njkr})$ 表示过滤器 $filter_{nikr}$ 和 $filter_{njkr}$ 的距离。如果 $\max(a_{nikr}-b_{njkr}, a_{njkr}-b_{nikr}) > 0$, $FilterDis(filter_{nikr}, filter_{njkr}) = \max(a_{nikr}-b_{njkr}, a_{njkr}-b_{nikr})$ 。否则, $FilterDis(filter_{nikr}, filter_{njkr}) = 0$ 。如果 $FilterDis(filter_{nikr}, filter_{njkr}) = 0$, 表示过滤器 $filter_{nikr}$ 和 $filter_{njkr}$ 相互重叠。

定义 5 基站 $sink_n$ 计算过滤器 $filter_{nikr}$ 和 $filter_{njkr}$ 的距离, 如果 $FilterDis(filter_{nikr}, filter_{njkr}) \leq \tau$, 则基站 $sink_n$ 把 $filter_{nikr}$ 和 $filter_{njkr}$ 归并在一起成为一个新过滤器 $filter_{ngkr}$, $filter_{ngkr} = merged(filter_{nikr}, filter_{njkr})$ 。其中, $a_{ngkr} = \min(a_{nikr}, a_{njkr})$, $b_{ngkr} = \max(b_{nikr}, b_{njkr})$ 。基站 $sink_n$ 获得的归并后的过滤器为 $MergedFilter_{nkr}$ 。 $MergedFilter_{nkr} = \{filter_{ngkr} | g \leq I\}$ 。 $\forall filter_{ngkr} \in MergedFilter_{nkr}$ 和 $filter_{mwkr} \in MergedFilter_{nkr}$, $FilterDis(filter_{ngkr}, filter_{mwkr}) \geq \tau$ 。

定义 6 数据库服务器提取出各基站 $sink_n$ 归并后过滤器 $MergedFilter_{nkr}$ 的公共部分, 称之为: $GlobeFilter_{kr}$ 。 $GlobeFilter_{kr} = \{filter_{gkr} | g \leq I * N\}$ 。 \forall

$filter_{gkr} \in GlobeFilter_{kr}$, $filter_{gkr} \in \{MergedFilter_{nkr} | n=1, 2, \dots, N\}$ 。

定义 7 在第 $r-1$ 轮结束时, 基站 $sink_n$ 和普通节点 s_{ni} 能预测第 k 个属性值在下一轮(第 r 轮)中的传感属性值 pv_{nikr} 。 pt_{nir} 表示预测的元组值, $pt_{nir} = \{pv_{nikr} | k=1, 2, \dots, K\}$ 。

3.2 通用连接算法

算法 1 通用连接算法

连接属性值收集阶段:

- 1: for 每个 Atom-WSN $_n \in$ 查询区域
- 2: for 每个普通节点 s_{ni}
- 3: 发送连接属性值 $v_{nikr-join}$ 给 s_{ni} 的父节点;
- 4: end-for
- 5: end-for
- 6: for 每个基站 $sink_n$
- 7: 转发 Atom-WSN $_n$ 内的连接属性值给数据库服务器;
- 8: end-for
- 9: 数据库服务器找出符合连接条件的匹配节点结果集。

结果集转发阶段:

- 1: 数据库服务器转发匹配节点结果集给各基站 $sink_n$ 。
- 2: 基站 $sink_n$ 提取出属于自己网络中的匹配节点结果集, 并在各自的原子网络 Atom-WSN $_n$ 中广播。

完整元组收集阶段:

- 1: 普通节点 $s_{ni}(s_{ni} \in$ 最终匹配节点结果集)发送完整元组 t_{nir} 给 s_{ni} 的父节点;
- 2: for 每个基站 $sink_n$
- 3: 发送完整元组 t_{nir} 给数据库服务器;
- 4: end-for
- 5: 数据库服务器构建最终的、完整的匹配节点结果集。

通用连接算法如算法 1 所示, 包括 3 个阶段: 连接属性值收集阶段、结果集转发阶段和完整元组收集阶段。

在连接属性值收集阶段, 属于查询区域的节点 s_{ni} 上报连接属性值 $v_{nikr-join}$ 给基站, 基站转发连接属性值给通信网络层数据库服务器, 由数据库服务器找出符合连接条件的匹配节点结果集。

在结果集转发阶段, 通信网络层数据库服务器转发匹配节点结果集给各基站, 各基站提取出属于自己网络中的匹配节点结果集, 并在各自的原子网络中广播。

在完整元组收集阶段, 普通节点 s_{ni} 收到广播

消息后, 比较自己是否属于最终匹配节点结果集, 如果属于最终匹配节点结果集, 则发送完整的元组信息 t_{nir} 给基站, 基站转发完整元组信息 t_{nir} 给通信网络层数据库服务器, 数据库服务器发送最终的、完整的匹配节点结果集给用户。

3.3 分布式连接算法

分布式连接算法流程图如图 2 所示, 一共有 5 个阶段, 分别是: 过滤器构建和传播阶段、正常连接属性值收集阶段、数据库构建探索节点集阶段、探索的连接属性值收集阶段和完整元组收集阶段。详细的算法如算法 2 所示。

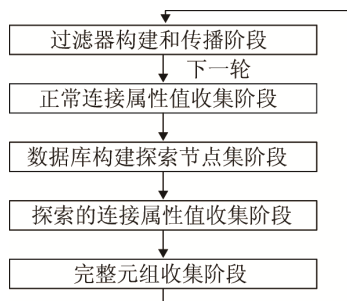


图 2 分布式连接算法流程图

算法 2 分布式连接算法

过滤器构建和传播阶段:

```

1: for 每个基站  $sink_n$ 
2:  $sink_n$  预测下一轮(第  $r$  轮)的预测值  $pv_{nikr}$  并设置过滤器  $filter_{nikr}$ ;
3: for 每个  $filter_{nikr}$  和  $filter_{njkr}$ 
4: if  $FilterDis(filter_{nikr}, filter_{njkr}) \leq \tau$ 
5:    $merged(filter_{nikr}, filter_{njkr})$ ;
6: end-if
7: end-for
8:  $sink_n$  发送归并的过滤器  $MergedFilter_{nikr}$  给数据库服务器;
9: end-for
10: 数据库服务器根据收集的  $MergedFilter_{nikr}$  得出全局的过滤器  $GlobeFilter_{kr}$ 。
11: 数据库服务器转发全局的过滤器  $GlobeFilter_{kr}$  给每个基站  $sink_n$ 。
12: 基站  $sink_n$  广播  $GlobeFilter_{kr}$  给原子网络 Atom-WSN $_n$  中的普通节点。

```

正常连接属性值收集阶段:

```

1: for 每个 Atom-WSN $_n$   $C$  查询区域
2: for 每个普通节点  $s_{ni}$ 

```

```

3:   for  $k=1$  to  $K$ 
4:     if 每个  $v_{nikr-join} \notin filter_{nikr}$  or 每个  $v_{nikr-join} \in GlobeFilter_{kr}$ 
5:       发送连接属性值  $v_{nikr-join}$  给父节点;
6:     end-if
7:   end-for
8: end-for
9: end-for
10: for 每个基站  $sink_n$ 
11: 转发 Atom-WSN $_n$  内的连接属性值  $v_{nikr-join}$  给数据库服务器;
12: end-for
数据库构建探索节点集阶段:
1: for 每个发送了属性值  $v_{nikr}$  给数据库的普通节点  $s_{ni}$ 
2:   for  $k=1$  to  $K$ 
3:     if 一个或多个  $v_{nikr} \notin$  全局过滤器  $GlobeFilter_{kr}$ 
4:       for 每个其他的 Atom-WSN $m$  ( $m \neq n$ )
5:         for  $k=1$  to  $K$ 
6:           if each  $v_{nikr} \in MergedFilter_{mkr}$  of Atom-WSN $m$ 
or  $|v_{nikr} - v_{mjkr}| \leq \tau_k$ 
7:             把值  $v_{nikr}$  和节点  $s_{ni}$  的 ID 加入探索节点集。
8:           end-if
9:         end-for
10:       end-for
11:     end-if
12:   end-for
13: end-for
14: 数据库服务器转发探索节点集给每个基站  $sink_n$ 。
15: 每个基站  $sink_n$  广播属于自己原子网络 Atom-WSN $_n$  范围内的探索节点集给普通节点。
探索的连接属性值收集阶段:
1: for 每个 Atom-WSN $_n$   $C$  查询区域
2:   for 每个普通节点  $s_{ni}$ 
3:     if 连接属性值  $v_{nikr-join}$  和广播的探索节点集的属性值满足连接条件
4:       发送连接属性值  $v_{nikr-join}$  给父节点;
5:     end-if
6:   end-for
7: end-for
8: for 每个基站  $sink_n$ 
9:   转发 Atom-WSN $_n$  内的连接属性值给数据库服务器;
10: end-for
11: 数据库服务器构建匹配节点结果集并转发给各基站  $sink_n$ 。
12: 基站  $sink_n$  提取出属于 Atom-WSN $_n$  中的匹配节点结果集, 并在各自的原子网络 Atom-WSN $_n$  中广播。
完整元组收集阶段:

```

- 1: 普通节点 s_{ni} ($s_{ni} \in$ 最终匹配节点结果集) 发送完整元组 t_{nir} 给 s_{ni} 的父节点;
- 2: for 每个基站 $sink_n$
- 3: 发送完整元组 t_{nir} 给数据库服务器;
- 4: end-for
- 5: 数据库服务器构建最终的、完整的匹配节点结果集。

为了清楚地描述分布式连接算法, 我们首先来看过滤器构建和传播阶段的算法。

当基站 $sink_n$ 收齐各节点上报的连接属性值 $v_{nikr-join}$ 后, 基站 $sink_n$ 和普通节点 s_{ni} 预测 s_{ni} 下一轮(第 r 轮)的预测值 pv_{nikr} 。在分布式连接算法中, 预测值 pv_{nikr} 并不由特定的模型获取。这里只描述我们所使用的模型。在无线传感器网络中, 线性自回归预测模型比移动平均预测模型、指数平滑预测模型和其他时间序列预测模型更有效^[16]。这里给出线性自回归模型的细节。已知时间序列 $\{x_i\}$, 创建一个线性等式 $x_t = a + bx_{t-1} + \varepsilon$ 。基站使用 n (在我们的实验中, $n=6$) 个最近收集的属性值预测第 r 轮预测值 pv_{nikr} 。在文献[17]中有标准的公式估计 a, b 和 ε 。

基站 $sink_n$ 和普通节点 s_{ni} 根据预测值 pv_{nikr} 确定过滤器 $filter_{nikr}$ 。 $filter_{nikr} = [a_{nikr}, b_{nikr}]$, 其中 $a_{nikr} = pv_{nikr} - width_k$, $b_{nikr} = pv_{nikr} + width_k$ 。 ($2 \times width_k$) 表示过滤器的宽度。简单起见, $width_k$ 是第 k 个属性值在原子传感器网络 Atom-WSN n 范围内的平均值, 并且在整个查询期间保持不变。

基站 $sink_n$ 对各节点下一轮的过滤器 $filter_{nikr}$ 进行归并, 若 $FilterDis(filter_{nikr}, filter_{njkr}) \leq \tau$, 则合并这两个过滤器。

如图 3 所示, $FilterDis(filter_{n1kr}, filter_{n2kr}) = 0$, 所以 $filter_{n1kr}$ 和 $filter_{n2kr}$ 被归并为一个新的 $filter$ 。 $FilterDis(filter_{n3kr}, filter_{n4kr}) \leq \tau$, 所以 $filter_{n3kr}$ 和 $filter_{n4kr}$ 被归并为一个新的 $filter$ 。

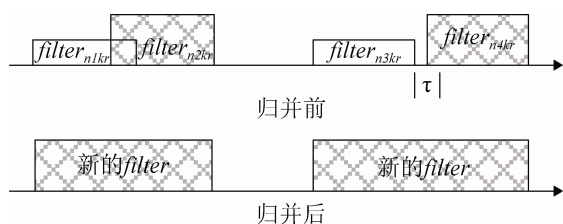


图 3 过滤器合并实例

各基站 $sink_n$ 上报归并后的过滤器 $MergedFilter_{nikr}$ 给通信网络层数据库服务器, 数据库服务器比较各原子网络内的过滤器信息, 得出全局过滤器 $GlobeFilter_{kr}$, 并下发给各基站, 由基站广播给原子网络中的节点。

数据库服务器可以得出全局过滤器 $GlobeFilter_{kr}$, 原子无线传感器网络 1, 2, 3 和 4 自我归并后的过滤器如图 4 所示, 数据库服务器比较各原子网络内的过滤器, 得出公共的全局过滤器。由图 4 可以看出, 全局过滤器 $GlobeFilter_{kr}$ 是每个原子无线传感器网络都具有的、公共的过滤器。

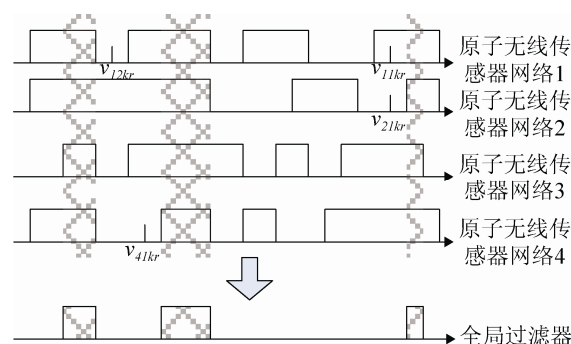


图 4 数据库服务器获得全局过滤器及构建探索节点集实例

下一轮中, 在正常连接属性值收集阶段, 如果普通节点 s_{ni} 的每个连接属性值 $v_{nikr-join}$ 超出了预测的过滤器 $filter_{nikr}$ 区间, 或属于数据库服务器广播的全局过滤器 $GlobeFilter_{kr}$ 区间, 则发送连接属性值 $v_{nikr-join}$ 给基站 $sink_n$, 基站 $sink_n$ 转发给数据库服务器。

数据库服务器收到各基站发送的连接属性值 $v_{nikr-join}$ 后, 进入数据库构建探索节点集阶段。如果节点 s_{ni} 的连接属性值 $v_{nikr-join}$ 不属于全局过滤器 $GlobeFilter_{kr}$ 区间, 且该值 v_{nikr} 在所有其他原子网络 Atom-WSN m ($m \neq n$) 满足连接条件, 即: $v_{nikr} \in$ 原子网络 Atom-WSN m 的归并过滤器 $MergedFilter_{mkr}$ 区间或者和原子网络 Atom-WSN m 的上报值 v_{mjkr} 满足连接条件。则数据库服务器把值 v_{nikr} 和节点 s_{ni} 的 ID 加入探索节点集。数据库服务器转发探索节点集给基站 $sink_n$, 基站 $sink_n$ 广播属于自己原子网络范围内的探索节点集给原子网络 Atom-WSN n 中的节点。

图 4 也显示了数据库服务器构建探索节点集实例。假设连接属性只有一个。如图 4 所示, Atom-WSN1(原子无线传感器网络 1)的上报值 v_{11kr} 和 Atom-WSN2 的上报值 v_{21kr} 满足连接条件, v_{11kr} , v_{21kr} 也在 Atom-WSN3 和 Atom-WSN4 的归并过滤器区间内, 因此 v_{11kr} , v_{21kr} 可能和 Atom-WSN3 和 Atom-WSN4 的未上报值满足连接条件。因此, 需要把值 v_{11kr} , v_{21kr} 以及节点 s_{1l} 和 s_{2l} 的 ID 加入探索节点集。Atom-WSN4 的上报值 v_{41kr} 在 Atom-WSN1、Atom-WSN2 和 Atom-WSN3 的归并过滤器区间内, 因此 v_{41kr} 可能和 Atom-WSN1, Atom-WSN2 和 Atom-WSN3 的未上报值满足连接条件。因此, 需要把值 v_{41kr} 以及节点 s_{4l} 的 ID 加入探索节点集。Atom-WSN1 中的上报值 v_{12kr} 虽然在 Atom-WSN2 的归并过滤器区间内, 但在 Atom-WSN3 和 Atom-WSN4 的上报值或归并过滤器区间内。因此, 不需要将 v_{12kr} 加入探索节点集。

节点 s_{ni} 收到基站 sink_n 广播的探索节点集消息后, 进入了探索的连接属性值收集阶段。如果节点 s_{ni} 的连接属性值 $v_{nikr-join}$ 和广播的探索节点集的属性值满足连接条件, 则上报自己的连接属性值 $v_{nikr-join}$ 给基站 sink_n , 基站 sink_n 再转发给数据库服务器。数据库服务器找出符合连接条件的匹配节点结果集, 并转发匹配节点结果集给各基站, 各基站提取出属于自己网络中的匹配节点结果集, 并在各自的原子网络中广播。

然后进入完整元组收集阶段, 该阶段和算法 1 通用连接算法的完整元组收集方法相同。

最后数据库服务器获得完整的匹配节点结果集后, 进入如前所述的过滤器构建和传播阶段, 当前轮结束。

4 实验

为了分析和验证各算法的性能, 我们在 omnetpp-4.1 平台上实现了本文提出的各个算法。

我们的实验使用了和文献[1, 14]相同的真实数据集 LUCE。该数据集包含各节点的位置信息, 因节点分布不均匀, 破坏了网络连通性, 我们修改了个别节点的位置。在我们的实验中, 如果只有一个连接属性, 则该连接属性是相对湿度; 如果有 3 个连接属性, 则包括环境温度, 地表温度和相对湿度。

在该数据集中, 取 88 个有效节点构成网络, 并从该数据集中为每个节点获取连续 1 000 条记录。我们定义一个采样周期为一轮, 这意味着每个节点采集 1 000 轮数据。消息包大小为 30 byte, 每个属性值占 4 字节, 节点 ID 占 2 字节。

如前所述, 相对于节点计算, 节点之间通信将消耗更多的能量。因此, 我们使用网络通信代价作为衡量网络性能的指标, 该指标包括: 平均发包数和网络生命周期。

平均发包数: 平均每轮每个节点发包数。

网络生命周期: 网络生命周期是指从网络开始运行到有传感器节点失效时的轮数。从网络可用性角度看, 网络生命周期是比平均能量消耗更有用的指标。节点发包数越多, 消耗能量越多, 越容易提前死亡, 网络生命周期越短。所以, 我们以最大发包数衡量网络生命周期, 最大发包数是指网络内所有节点中平均每轮发包数最大的那个节点发包数。

在我们的方案中, 原子无线传感器网络层基于 TAG^[18]路由, 假设通信链路没有错误, MAC 层是理想的。基站位于原子无线传感器网络区域中心。基站通信范围通常远大于普通节点的通信范围, 因此, 假设各基站 sink_n 通信范围能覆盖各原子无线传感器网络 Atom-WSNn 的监测区域。

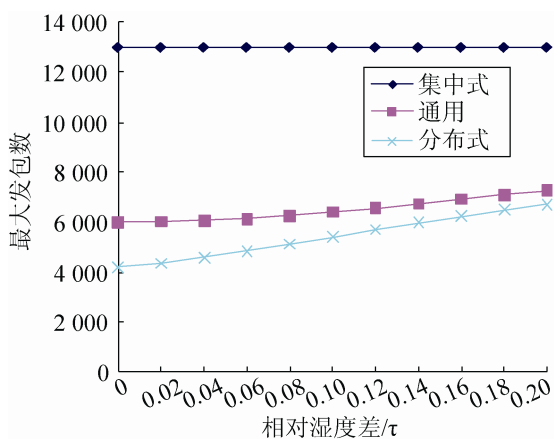
在我们的实验中, 有 4 个原子无线传感器网络参与连接查询; 过滤器宽度 $\text{width}_{AT} = \text{width}_{ST} = \text{width}_T$; 温度差为 $\tau_{AT} = \tau_{ST} = \tau_T$ 。

4.1 单连接属性实验

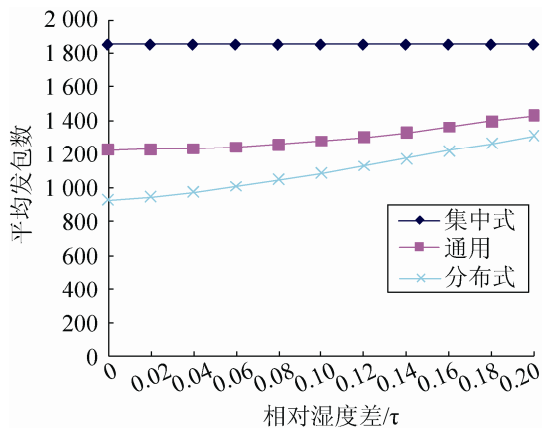
在这个实验场景中, 连接属性数占全部属性数

比例是 33%，即：一条元组包含 3 个属性，其中只有一个连接属性，该连接属性是相对湿度(RH)。

如图 5(a)和(b)所示，随着相对湿度差逐渐增大，通用连接算法和分布式连接算法的最大发包数和平均发包数也逐渐增多。因为，随着相对湿度差逐渐增大，参与连接的元组数也逐渐增多，导致需要发送数据包的节点数增多，所以最大发包数和平均发包数也逐渐增多。在 3 种算法中，分布式连接算法的最大发包数和平均发包数最少，通用连接算法次之，集中式算法最多。



(a)相对湿度差变化对最大发包数的影响



(b) 相对湿度差变化对平均发包数的影响

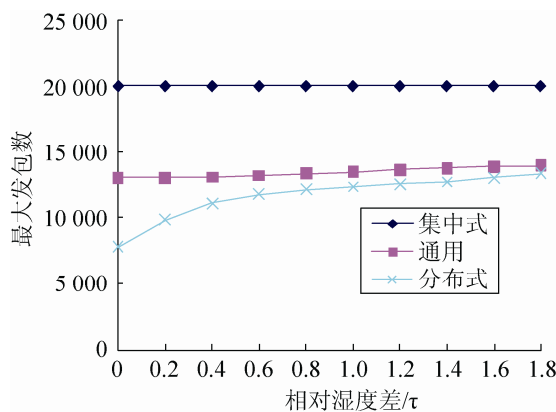
图 5 单连接属性实验中相对湿度差变化对发包数的影响

4.2 多连接属性实验

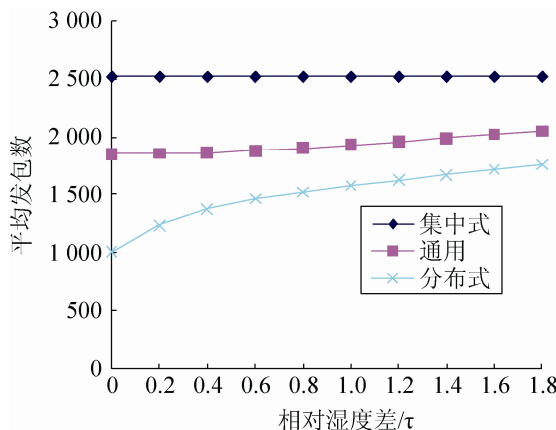
在这个实验场景中，连接属性数占全部属性数比例是 60%，即：一条元组包含 5 个属性，其中有

3 个连接属性，分别环境温度(AT)、地表温度(ST)和相对湿度(RH)。该实验中，默认参数为：过滤器宽度 $width_{AT}=width_{ST}=width_T=0.1, width_{RH}=0.5$ 。温度差为 $\tau_{AT}=\tau_{ST}=\tau_T=1.6$ 。相对湿度差 $\tau_{RH}=1.6$ 。

如图 6(a)和(b)所示，随着相对湿度差逐渐增大，通用连接算法和分布式连接算法的最大发包数和平均发包数也逐渐增多。因为，随着相对湿度差逐渐增大，参与连接的元组数也逐渐增多，导致需要发送数据包的节点数增多，所以最大发包数和平均发包数也逐渐增多。在 3 种算法中，分布式连接算法的最大发包数和平均发包数最少，通用连接算法次之，集中式算法最多。



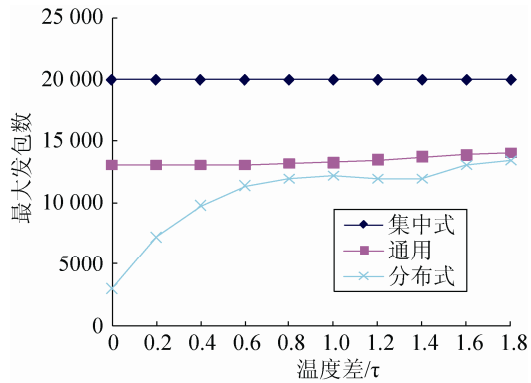
(a) 相对湿度差变化对最大发包数的影响



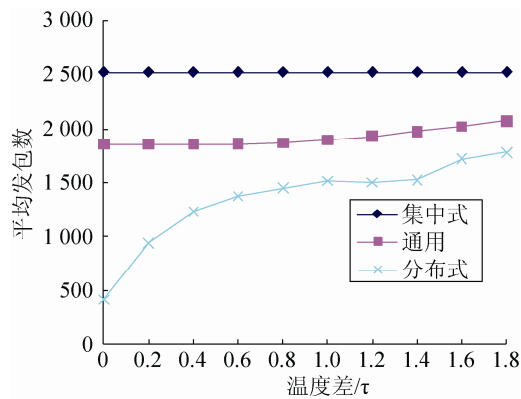
(b) 相对湿度差变化对平均发包数的影响

图 6 多连接属性实验中相对湿度差变化对发包数的影响

如图 7(a)和(b)所示，在 3 种算法中，分布式连接算法的最大发包数和平均发包数最少，通用连接算法次之，集中式算法最多。



(a) 温度差变化对最大发包数的影响



(b) 温度差变化对平均发包数的影响

图 7 多连接属性实验中温度差变化对发包数的影响

5 结论

物联网环境中, 为了从多个原子网络中获得相似的网络环境, 我们提出了通用连接算法和分布式连接算法。这些算法的目标是减少数据包发送量, 减少节点能量消耗, 延长网络生命周期。下一步, 我们计划在物联网环境中实现 Top-k, KNN 等查询, 研究如何减少数据包发送量, 提高查询效率。

参考文献:

- [1] 莫尚丰, 陈丁洁, 陈红, 等. 无线传感器网络中 top-k 连接查询处理 [J]. 计算机学报, 2013, 36(3): 557-570.
- [2] Zhiming Ding, Qi Yang, Hong Wu. Massive Heterogeneous Sensor Data Management in the Internet of Things [C]// 2011 IEEE International Conferences on Internet of Things, and Cyber, Physical and Social Computing. USA: IEEE, 2011.
- [3] Luigi Atzori, Antonio Iera, Giacomo Morabito. The Internet of Things: A survey [J]. Computer Networks (S1389-1286), 2010, 54(15): 2787-2805.

- [4] A K M Azad, Joarder Kamruzzaman, Balasubramaniam Srinivasan, et al. Query Processing over Distributed Heterogeneous Sensor Networks in Future Internet: Scalable Architecture and Challenges [C]// 2010 Second International Conference on Advances in Future Internet. Venice/Mestre, Italy. USA: IEEE Computer Society, 2010: 75-81.
- [5] 丁治明, 高需. 面向物联网海量传感器采样数据管理的数据库集群系统框架 [J]. 计算机学报, 2012, 35(6): 1175-1191.
- [6] Zhiming Ding, Jiajie Xu, Qi Yang. SeaCloudDM: a database cluster framework for managing and querying massive heterogeneous sensor sampling data [J]. Journal of Supercomputing (S0920-8542), 2013, 66(3):1260-1284.
- [7] Fengjuan Wang, Xiaoming Zhang, Yongheng Wang, et al. The research on complex event processing method of Internet of Things [C]// 5th International Conference on Measuring Technology and Mechatronics Automation (ICMTMA), Hong Kong, China. Switzerland: Trans Tech Publications INC, 2013: 1219-1222.
- [8] Daniel J Abadi, Samuel Madden, Wolfgang Lindner. REED: Robust, Efficient Filtering and Event Detection in Sensor Networks [C]// Proceedings of the 31st VLDB Conference, Trondheim, Norway. Germany: Springer Berlin Heidelberg, 2005: 769-780.
- [9] Yongxuan Lai, Hong Chen, Yilong Chen. In-Network Execution of External Join for Sensor Networks [C]// The Ninth International Conference on Web-Age Information Management (WAIM' 08), Zhangjiajie, China. Germany: Springer Berlin Heidelberg, 2008: 78-85.
- [10] Yongxuan Lai, Yilong Chen, Hong Chen. PEJA: Progressive Energy-efficient Join Processing for Sensor Networks [J]. Journal of Computer Science and Technology (S1000-9000), 2008, 23(6): 957-972.
- [11] Xiaoyan Yang, Hock Beng Lim, M Tamer O'zsu, et al. In-Network Execution of Monitoring Queries in Sensor Networks [C]// SIGMOD'07, June 11-14, Beijing, China. USA: Association for Computing Machinery, 2007: 521-532.
- [12] Mirco Stern, Erik Buchmann, Klemens Böhm. Towards Efficient Processing of General-Purpose Joins in Sensor Networks [C]// IEEE International Conference on Data Engineering. USA: IEEE, 2009: 126-137.
- [13] Jun-Ki Min, Heejung Yang, Chin-Wan Chung. Cost based in-network join strategy in tree routing sensor networks [J]. Information Sciences (S0020-0255), 2011, 181(16): 3443-3458.

(下转第 850 页)