ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.21, No.6, June 2010, pp.1364−1377
doi: 10.3724/SP.J.1001.2010.03518

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

传感器网络中误差有界的小波数据压缩算法*

张建明[1,2], 林亚平[1,3+], 周四望[3], 欧阳竞成[1]

[1](湖南大学 计算机与通信学院,湖南 长沙 410082)

[2](湖南城市学院 计算机科学系,湖南 益阳 413000)

[3](湖南大学 软件学院,湖南 长沙 410082)

Haar Wavelet Data Compression Algorithm with Error Bound for Wireless Sensor Networks

ZHANG Jian-Ming[1,2], LIN Ya-Ping[1,3+], ZHOU Si-Wang[3], OUYANG Jing-Cheng[1]

[1](College of Computer and Communication, Hu'nan University, Changsha 410082, China)

[2](Department of Computer Science, Hu'nan City University, Yiyang 413000, China)

[3](College of Software, Hu'nan University, Changsha 410082, China)

+ Corresponding author: E-mail: yplin@hnu.cn

Zhang JM, Lin YP, Zhou SW, Ouyang JC. Haar wavelet data compression algorithm with error bound for wireless sensor networks. *Journal of Software*, 2010,21(6):1364−1377. http://www.jos.org.cn/1000-9825/3518.htm

Abstract: Wireless sensor networks usually have limited energy and transmission capacity, and they can't match the transmission of a great deal of data. So, it is necessary to approximate or aggregate raw data sampled by sensors in networks. By designing an error tree and solving the regression equations set, this paper proposes a data compression scheme with infinite norm error bound for wireless sensor networks. The algorithms in the scheme can simultaneously explore the temporal and multiple-streams correlations among the sensory data. The temporal correlation in one stream is captured by the 1D Haar wavelet transform. For multivariate monitoring sensor networks, some streams from one sensor are selected as the bases according to the correlation coefficient matrix, and the other streams from the same sensor node can be expressed with one of these bases using linear regression. Theoretically and experimentally, it is concluded that the proposed algorithms can effectively exploit the temporal and multiple-streams correlations on the same sensor node and achieve significant data reduction.

Key words: wireless sensor network; infinite norm error bound; wavelet compression; regression

摘 要: 无线传感器网络通常能量、带宽有限,难以适应大量数据传输的需求,需要对原始采样数据进行网内近似或聚合.通过设计误差树和解回归方程组,提出了一种无穷范数误差有界的数据压缩方案.该方法可以同时探索传感器数据中的时间相关和多属性间相关.通过一维 Haar 小波变换来消除单个数据流中的时间相关.若单个传感器节点可以采集多种物理量,即产生多个数据流,则根据相关系数矩阵选择其中的若干个数据流作为基信号,其他数据流借助一个基用线性回归参数来表示.实验结果表明,该算法能够有效地利用传感数据中存在的时间相关和多属性间相

* Supported by the National Natural Science Foundation of China under Grant Nos.60973031, 60973127 (国家自然科学基金); the Scientific Research Fund of Hu'nan Provincial Construction Department of China under Grant No.200609 (湖南省建设厅科技计划)

Received 2008-08-17; Accepted 2008-10-27

关,显著减少了冗余数据.

关键词: 传感器网络;无穷范数误差限;小波压缩;回归
中图法分类号: TP393 文献标识码: A

无线传感器网络(WSN)能够协作地实时监测、感知、采集网络分布区域内的各种环境或监测对象的数据,并对这些数据进行处理,获得详尽而准确的信息,传送到需要这些信息的用户[1].无线传感器网络是一种全新的信息感知、获取和处理技术.如果说 Internet 实现了数字世界的联网,那么传感器网络的出现则实现了数字世界与模拟世界的联网,标志了普适计算时代的到来.

WSN 具有电源能量有限、通信能力有限、节点计算能力有限、传感器节点数量大且分布范围广、网络动态性强、感知数据流巨大、以数据为中心等特点.数量众多的传感器节点采集数据后,直接将原始数据传输到基站是不可行的,一是带宽不够,二是能量将很快耗尽.有研究[2]指出,数据通信的耗能远高于数据计算的耗能,传送 1 位数据的耗能是执行 1 次加法运算的 480 倍,数据传输消耗了总能量的 70%.如何有效地减少网络内部的数据量,从而延长网络生命周期并减少数据的传输延迟,是研究人员面临的一个重要课题.

1 相关工作

在有网内处理的情况下,传感器节点采集的数据与要传输的数据是可以不同的.节点通过数据感知部件采集原始数据,经过数据处理部件处理后,再进行数据传输,由此减少待传输的数据量.大致有两类公认的办法[3]:数据汇聚(data aggregation)和数据近似(data approximation).

聚集函数(aggregate function)包括如 COUNT,SUM,AVG 等.使用聚集函数可节省能量,但丢失了数据中大量的原始结构,只提供粗糙的统计量,掩饰了令人感兴趣的局部变化.为获得数据不同粒度的表示,可尝试采用数据挖掘的方法.美国工业与应用数学学会(SIAM)在 SDM 2005 大会第一次组织了 WSN 数据挖掘的 Workshop;而这在传感器网络的研究人员中被称为基于模型的数据汇聚(model-based data aggregation).

数据近似可视为基于模型的数据获取,通过对 WSN 采集到的感知数据进行分布式建模,只要传输模型参数,极大地减少数据传输量,从而节省网络能量,延长网络生命.根据采用模型的不同,现有方法主要包括 4 类:基于概率模型[4,5];基于时间序列分析模型[6,7];基于数据挖掘模型[8];基于数据压缩模型[9−12].

DIMENSIONS[9]是一种层次结构系统,先在底层的各个传感器节点对监测到的数据进行小波压缩,然后基于 WavRoute 路由协议,由中间层的汇聚节点收集底层节点传来的数据,并在汇聚节点进行进一步的小波压缩后传送到上一层节点.DIMENSIONS 在底层节点挖掘数据的时间相关性,在汇聚节点挖掘底层各节点间数据的空间相关性,但在底层节点与中间层的汇聚节点间存在冗余数据的传输.RACE 算法[10]针对单个传感器节点产生的数据,给出了一种比特率(bit rate)自适应的 Haar 小波压缩算法,通过阈值来选择重要小波系数,从而保持 CBR 或 LBR.该算法在单个节点内运行,通过消除时间相关性减少冗余数据的传输,但没有考虑邻近节点间数据的空间相关性及属性间的相关性.Ciancio 等人[11]研究小波的分布式压缩算法,这些算法通过在邻近的节点间交换信息,在数据传送到汇聚节点前分布式挖掘网络中数据的空间相关性,极大地减少了冗余数据的传输.虽然分布式压缩算法有效地减少了网络中冗余数据的传输,然而节点间需要交换信息,由此产生的能量消耗、网络延时等代价尚需要在理论上进行进一步的定性分析.我们针对任意支撑长度的小波函数给出了一种基于环模型的分布式时-空小波数据压缩算法[12],该算法可以同时消除传感器网络中数据的时间和空间相关性.Garofalakis[13]通过引入概率小波大纲第一次提出了数据重构误差有界的小波压缩技术.

目前,多数分布式压缩算法[9,11,12]都是建立在节点之间具有空间相关性的假设之上,实现代价较大.实际上,人们通常希望用最少的节点监测最大的范围,即用最少的投入来获得最大的监测效果,通常使节点的放置尽量彼此独立[14].因此,当节点监测值之间不存在空间相关性或空间相关性不稳定时,设计在各节点上独立运行的算法是更好的选择.本文不考虑数据的空间相关性,仅考虑数据的时间相关和多属性间的相关.

根据不同应用的数据收集模式,WSN 可以分成两类:(1) 持续传输,节点连续周期性地将数据逐跳转发到基

站;(2) 事件触发传输,只有当感兴趣的事件发生了,节点才生成报告传给基站.本文研究持续周期性传输数据时如何尽量减少数据量.假设每个节点采集一种或多种物理量,我们的算法安置于单个的传感器节点.利用时间和属性间的相关性,本文提出了基于小波的误差有界的单属性、多属性数据压缩算法,并采用 C 语言对提出的压缩算法进行实现.模拟实验结果表明,该算法能够减少传输的数据量,比分布式算法具有更好的实时性.

2 预备知识

2.1 一维Haar小波分解算法

设数据向量 s=[2,6,5,11],进行 1D Haar 小波分解如下:逐对计算相邻数据的平均值,得到数据个数为原向量一半的低分辨率的新向量[4,8],称为近似分量.在平均化过程中丢失了一些信息.为了能够重构原来的向量,计算原数据对的差再除以 2,得到[−2,−3],称为细节分量.至此,原向量通过一级分解为[4,8,−2,−3].对近似分量重复以上过程,直至新近似分量只含有 1 个数据.最终,原向量通过二级分解后为[6,−2,−2,−3].

当近似分量含有 2^j 个数据时,定义其分辨率为 2^j,分辨率级(level of resolution)为 j.记 H_jf 为信号 $f(f\in L^2(\Box))$ 在分辨率 2^j 下的逼近,则 H_jf 可以进一步分解为 f 在分辨率 2^{j-1} 下的逼近 $H_{j-1}f$(通过低通滤波器得到)及位于分辨率 2^j 与 2^{j-1} 之间的细节 $D_{j-1}f$(通过高通滤波器得到)之和.其分解过程如图 1 所示,其中 $k\le j$.

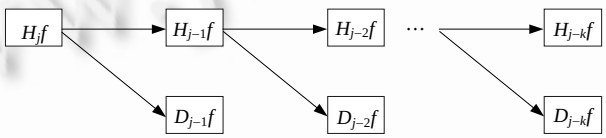

Fig.1 The k-level decomposition of the Mallat's algorithm

图 1 Mallat 算法的 k 级分解

不妨设原始数据向量包含 N 次采样且 $N=2^n$,若原始数据的个数不足 2^n,可以通过平行延拓的方式加以补充.根据能耗模型计算压缩开销和传送开销,可以确定最佳分解级数.考虑到我们的算法是本地执行,没有节点间的协作消息开销,直接根据本地缓存的数据个数来确定分解级数为最大值 $n=\log_2 N$.Micaz 等节点的嵌入式程序采用 nesC 语言开发,为便于实现,本文所有算法用类 C 语言描述并对时间和空间复杂度作了优化.本节各算法中,原始采样数据、小波系数、重构数据都存放在 $s[0..N-1]$中,而 $t[0..N-1]$为辅助空间.

算法 **1**. 1D Haar 分解.

输入:采样数据.

输出:小波系数.

```
1:  for (k=N; k>=2; k=k/2) {        //进行最大分解级数次分解
2:      average_pos=0;              //前半部分放近似分量
3:      detail_pos=k/2;             //后半部分放细节分量
4:      for (i=0; i<k; i=i+2) {
5:          t[average_pos++]=(s[i]+s[i+1])/2;
6:          t[detail_pos++]=(s[i]–s[i+1])/2;   }
7:      for (i=0; i<k; i++) s[i]=t[i];
8:  }   //end of for k
```

显然,算法 1 共要计算 $\sum_{i=1}^{\log_2 N}\left(\frac{N}{2^{i-1}}\right)=2(N-1)$次,时间复杂度为 $O(N)$,空间复杂度为 $O(N)$.

2.2 一维Haar小波重构算法

将图 1 所有箭头改为逆方向即为重构流程,得到算法 2,其时间复杂度为 $O(N)$,空间复杂度为 $O(N)$.

算法 **2**. 1D Haar 重构.

输入:小波系数.

输出:重构数据.

```
1:   for (k=2; k<=N; k=k*2) {        //此 for 语句循环 n 次,n 为重构级数
2:       for (i=0; i<k/2; i++) {     //利用 k/2 个近似分量和 k/2 个细节分量重构出 k 个数据
3:           t[2*i]=s[i]+s[k/2+i];
4:           t[2*i+1]=s[i]–s[k/2+i];   }
5:       for (i=0; i<k; i++) s[i]=t[i];
6:   }
```

传感器节点采集的很多监测属性如温度、湿度、光照、振动等在连续时间内的变化较小,多数邻近的数据相同或相近,用小波分解这样的感知数据时,绝大部分能量集中在低频系数上,高频部分的大量系数为 0 或近似为 0.即使感知对象异常变化,感知数据存在波动异常,小波变换的多级分解特性可以缓解异常波动对整体数据的影响,保证了部分细节分量值仍然近似为 0.压缩(丢弃)值为 0 的细节分量,不会影响数据的重构;压缩值不为 0 的细节分量,会对数据的精度产生影响.压缩越多的细节分量,数据的压缩率越高,但由此带来的数据误差也越大.我们要在保证误差有界的前提下,最大程度地压缩细节分量.

2.3 误差度量

假设采集的 N 个数据为 $s[0],\ldots,s[N-1]$,重构出的近似数据为 $\tilde{s}[0],\ldots,\tilde{s}[N-1]$.

定义 **1**(绝对误差). $e_i^{abs}=|s[i]-\tilde{s}[i]|$.

定义 **2**(相对误差). $e_i^{rel}=|s[i]-\tilde{s}[i]|/s[i]$.

定义 **3**(采样数据的规范化). $s[i]$的规范化为 $norm(s[i])=(s[i]-s_{\min})/(s_{\max}-s_{\min})$,$s_{\max},s_{\min}$ 分别为一段时间内采样数据的最大值、最小值.显然,$norm(s[i])$的值在[0,1]之间.当处理多种物理量的数据时,采样数据规范化可以防止幅度小的属性被幅度大的属性淹没.

定义 **4**(规范化误差). $e_i^{norm}=|norm(s[i])-norm(\tilde{s}[i])|$.易知 $e_i^{abs}=(s_{\max}-s_{\min})e_i^{norm}$.

定义 **5**(**2**-范数平均误差). $\|e\|_2=\sqrt{\frac{1}{N}\sum_{0\le i<N}e_i^2}$.2-范数平均误差体现了两个向量间的整体误差限.

定义 **6**(∞-范数平均误差). $\|e\|_\infty=\max_{0\le i<N}|e_i|$.∞-范数平均误差保证了每个重构数据与对应采样的误差限.

3 单属性的误差有界小波数据压缩算法

传感器网络中的分布式小波压缩算法由多个节点协同完成,小波变换的计算量分散在各个节点,小波系数也分散在各个节点,每个节点的计算量都较小.在分布式算法中,二级小波变换的性能略优于一级小波变换,因为虽然进行第 2 级小波变换会增加额外的耗能与延时,却取得了更好的去相关性[12].但分布式算法不能总是靠提高小波分解级数来提高性能,那样会不断增加通信开销和延时.我们的算法在各传感器节点上独立运行,节点间没有协作,因此可以小波分解到最大级,充分消除数据的时间相关.单属性的误差有界小波压缩算法(SWCEB)分为 4 步:小波分解、系数选择、量化、熵编码,在系数选择时保证了误差限.

3.1 小波系数选择

N 个采样数据通过 n 级小波分解后得到 N 个小波系数,其中第 1 个数据为近似分量,其余 $N-1$ 个数据为细节分量.待编码小波系数由小波系数选择算法确定.在满足误差限的前提下,从 N 个系数里面挑选出最少数量的 m 个系数.针对不同的误差度量,有不同的系数选择算法.

3.1.1 针对$\|e\|_2$的小波系数规范化

要求重构误差$\|e\|_2\le\varepsilon$.阈值化是选择高频系数的一种方法.通过对系数加阈值,节点仅传输特定的高频小波系数.不同小波系数在重构过程中所起的作用不一样,需要先规范化小波系数.近似分量对每个数据的重构都有

影响,低分辨率级的系数比高分辨率级的系数影响的重构数据个数更多,应该具有更高的权.对 Haar 小波而言,设 l 代表分辨率级,对所有 $0\le l\le n-1$,将当前 l 级的细节分量 $s[2^l],\ldots,s[2^{l+1}-1]$都除以 $\sqrt{2^l}$.取规范化后小波系数绝对值最大的 m 个系数即可,已经证明[13]这种系数选择方法对 Haar 小波最小化$\|e\|_2$而言是最优的.为选择最少的系数个数 m,对 m 依次从 1~N 进行穷举:将 $N-m$ 个绝对值较小的小波系数置 0,调用算法 2,直到$\|e\|_2\le\varepsilon$为止.

3.1.2 针对$\|e\|_\infty$的误差树

这里研究为保证重构误差$\|e\|_\infty\le\varepsilon$的小波系数选择算法.为了便于分析重构过程中的误差,我们用误差树(error tree)[10,13]来表示分解和重构过程.如图 2 所示,每个内部节点对应一个小波系数,每个叶节点对应一个原始采样数据.误差树自底向上构造,逐级计算小波系数.

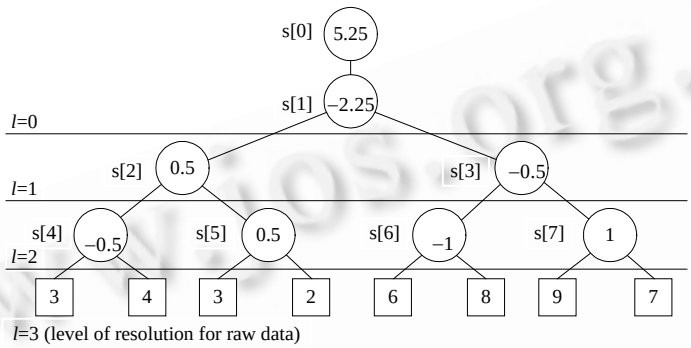

Fig.2 Error tree for our example one-dimensional data vector (N=8)

图 2 数据向量(N=8)的误差树

设 $s[0..N-1]$保存了小波分解后的系数,$\bar{s}[0...N-1]$为重构出的数据.用 *leftleaves*(t)表示以节点 $s[t]$为根的子误差树中 $s[t]$的左孩子的所有叶子节点,*rightleaves*(t)表示以节点 $s[t]$为根的子误差树中 $s[t]$的右孩子的所有叶子节点,*path*(t)表示误差树中从 $s[t]$到根 $s[0]$的路径上 $s[t]$的所有祖先节点.

性质 **1**(节点个数). 误差树有 N 个叶节点和 N 个非叶节点.

性质 **2**(不需**零化**的节点). 根据小波系数的值、误差限ε来判定哪些节点可以被置为 0(称为零化),从而不需要传输:① 若要求$\|e\|_\infty\le\varepsilon$,所有绝对值大于阈值的小波系数不能被零化;② 值等于 0 的小波系数不用考虑零化;③ 近似分量 $s[0]$参与每个叶节点的重构,不能被零化,对分析误差也没有影响.因此,系数零化过程中可以不考虑这 3 种节点.下面的 *candlist* 和 *nodelist* 中都剔除了这 3 种节点.

性质 **3**(重构公式). 重构计算方法为 $\bar{s}[i]=\sum_{s_j\in path(i)}\delta_{i,j}s[j]$,其中 $\delta_{i,j}=\begin{cases}+1, & i\in leftleaves(j)\ \text{or}\ j=0\\ -1, & i\in rightleaves(j)\end{cases}$.

为了根据$\|e\|_\infty\le\varepsilon$来确定要置为 0 的小波系数,定义 3 个辅助数据结构.

```
int flag[N];            //初值为 1;若打算将小波系数 s[i]近似为 0,则置 flag[i]=0
struct {
    int pos[N];         //可能被零化的小波系数(称为候选系数)在 s[0..N-1]中的下标
    int length;         //本列表的长度,显然不超过 N
} candlist;             //候选系数列表
typedef struct {
    float coeff[n+1];   //每个叶节点数据重构时,涉及到的小波系数的值(含参与重构运算的符号)
    int pos[n+1];       //coeff 中对应的小波系数在 s[0..N-1]中的下标
    int length;         //本列表的长度,显然不超过 n+1
} NodeList;             //每个叶节点数据重构时,影响其重构误差的节点列表
NodeList nodelist[N];   //误差树的 N 个叶节点需要 N 个节点列表来表示
```

算法 **3**. 构建候选系数列表.

输入:小波系数 *s*[0..*N*–1],误差限 *eps*.

输出:*candlist*.

```
1:  candlist.length=0;
2:  for (i=1; i<N; i++) if (s[i]!=0 && fabs(s[i])<=eps) candlist.pos[candlist.length++]=i;        //参考性质 2
```

3: 数组 *candlist.pos* 按非递减排序:第 *i* 个元素 *candlist.pos*[*i*]以 *fabs*(*s*[*candlist.pos*[*i*]])为排序关键字.

为了节省内存,没有按常规的方法存储误差树.误差树的内部节点用 *s*[0..*N*–1]表示,其中,*s*[1..*N*–1]是一棵高度为 *n* 的满二叉树,*s*[0]作为根是 *s*[1]的父节点;误差树的叶节点代表重构出的采样数据,每个叶节点需要一个 *NodeList* 类型的节点列表来存储参与该叶节点重构的所有小波系数.算法 4 时间复杂度为 $O(N\log_2 N)$.

算法 **4**. 构建每个叶节点的重构相关节点列表.

输入:*s*[0..*N*–1],*flag*,误差限 *eps*.

输出:*nodelist*,*flag*.

```
1:  for (i=N/2; i<N; i++){   //分辨率级最高的小波系数为 s[N/2..N–1],每个对应 2 个叶节点
2:      k=2*i–N;             //此轮循环创建可重构出第 k 个、第 k+1 个叶节点的 nodelist[k],nodelist[k+1]
3:      if (s[i]==0) flag[i]=0;        //参考性质 2
        else if (fabs(s[i])<=eps) {
          nodelist[k].coeff[nodelist[k].length]=s[i];          //第 k 个叶节点是第 i 个小波系数的左孩子
          nodelist[k].pos[nodelist[k].length++]=i;
          nodelist[k+1].coeff[nodelist[k+1].length]=–s[i];     //第 k+1 个叶节点是第 i 个小波系数的右孩子
          nodelist[k+1].pos[nodelist[k+1].length++]=i;  }
4:      son=i;
5:      j=i/2;                         //沿误差树向上找到当前小波系数节点 i 的父节点 j
6:      while (j>0){
7:        if (s[j]==0) flag[j]=0;
          else if (fabs(s[j])<=eps) {
            if (2*j==son){             //son 是 j 的左孩子
              nodelist[k].coeff[nodelist[k].length]=s[j];   //第 k 个、第 k+1 个叶节点是第 j 个系数的左孩子
              nodelist[k].pos[nodelist[k].length++]=j;
              nodelist[k+1].coeff[nodelist[k+1].length]=s[j];
              nodelist[k+1].pos[nodelist[k+1].length++]=j;     }
            else                       //son 是 j 的右孩子
              同上,将小波系数的下标为 j、值为–s[j]分别保存到 nodelist[k],nodelist[k+1]各域中;   }
8:        son=j;
9:        j=j/2;                       //参考性质 3,依次处理 path(i)的节点 j
10:     }                              //end of while j
11: }                                  //end of for i
```

基于上述数据结构,逐个分析候选列表里的系数:计算与某候选系数相关的所有叶节点的重构值,若所有叶节点重构值都符合精度要求,则该候选系数可零化,相应地,将 *flag* 标志数组中对应元素置 0.算法 5 的时间复杂度为 $O(N^2\log_2 N)$.

算法 **5**. 确定可以零化的小波系数.

输入:*candlist*,*nodelist*,*flag*,误差限 *eps*.

输出:*flag*.

```
1:  for (i=0; i<candlist.length; i++) {          //依次考察每个候选小波系数 i
2:      small=1;                                //先设可以零化
3:      for (j=0; j<N; j++) {                   //依次考察当前候选系数 i 对每个叶节点 j 重构的误差
4:          error=0;                            //叶节点 j 的累计重构误差
5:          for (k=0; k<nodelist[j].length; k++)  //已零化的系数给 j 带来的误差
6:              if (flag[nodelist[j].pos[k]]==0) error+=nodelist[j].coeff[k];
7:          for (k=0; k<nodelist[j].length; k++)  //若零化 i 给 j 带来的误差
8:              if (candlist.pos[i]==nodelist[j].pos[k]) error+=nodelist[j].coeff[k];
9:          if (fabs(error)>eps) small=0;       //叶节点 j 在误差限外,i 不能零化
10:     }   //end of for j
11:     if (small) flag[candlist.pos[i]]=0;     //所有叶节点都在误差限内,则零化 i 节点
12: }   //end of for i
```

这样,我们只要传输 $flag[i]=1$ 的 $s[i],0\le i\le N-1$.注意,接收方重构数据时也需要 $flag$ 数组.为减少传输的数据量,$flag$ 可以用一个二进制位串 $bitflag$ 来代替,一个 $flag[i]$用一个二进制位表示.

3.2 小波系数的量化

常用的量化方法有矢量量化和标量量化两种.标量量化因为算法简单,更加适合无线传感器网络.标量量化是将小波系数映射到一个整数区域,每个小波系数对应此区域中的一个元素.对一组小波系数,设最大的系数值为 $s_{\max}$,最小的系数值为 $s_{\min}$,采用 m 位均匀量化,则量化步长 $step=(s_{\max}-s_{\min})/2^m$.量化位 m 越大,步长越短,量化精度越高.然而,此时需要更多位表示小波系数,使得数据的压缩比降低.

3.3 量化系数的熵编码

编码方法直接影响到数据的压缩效率.一般来说,不同的编码算法适合具有不同统计特性的数据集.对不同统计特性的数据集采用相同的编码算法,会产生不同的压缩效果.游程编码通过形成串的字符、串的长度及串的位置来进行编码,实现简单,其压缩率取决于数据流中的重复字符出现的次数和平均游程长度.

3.4 本算法的性质及应用

传感器网络一般采用多跳通信,形成一棵以 Sink 为根的汇聚树.Sink 节点附近、事件处理热点区域、数据传输链路重载区域都容易出现能量空洞(energy hole),使网络的生命周期过早结束.能量空洞避免机制研究已引起广泛关注.本节提出的 SWCEB 算法在多跳链路上执行多次压缩后,总误差限等于各次压缩误差限之和.利用该性质,可根据节点及其后继节点数据量、带宽和剩余能量,自适应地提高误差限进一步进行压缩,使离 Sink 近的节点具有较低的数据率,避免能量空洞.

性质 **4**(误差限的可累加性). 在传输过程的不同节点上执行多次 SWCEB 压缩算法,保持累加的误差限.

证明:设 $Sensor_1$ 采集的原始数据为向量 S_0,小波分解后得到误差树 T_0,在误差限为 ε_1 进行小波系数零化,得到误差树 T_1,重构可得到近似序列 S_1.根据算法要求有 $\|S_1-S_0\|_\infty\le\varepsilon_1$.

$Sensor_1$ 将数据 T_1 和一个 $bitflag$ 传输给下一跳.在多跳传输过程中,某节点 $Sensor_i$ 发现自己或其后续节点的能量偏低(可根据接收到的数据 T_1 和 $bitflag$ 重构得到序列 S_1),对接收的数据 T_1 再次启动压缩,在误差限为 ε_2 进行小波系数零化,得到误差树 T_2,重构可得到序列 S_2.根据算法要求有 $\|S_2-S_1\|_\infty\le\varepsilon_2$.

$Sensor_i$ 将数据 T_2 和一个新的 $bitflag$ 传输给下一跳.利用范数的三角不等式 $\|X+Y\|\le\|X\|+\|Y\|$,此时误差限为 $\|S_2-S_0\|_\infty=\|(S_2-S_1)+(S_1-S_0)\|_\infty\le\|S_2-S_1\|_\infty+\|S_1-S_0\|_\infty\le\varepsilon_1+\varepsilon_2$. □

4 多属性的误差有界小波数据压缩算法

设每个节点在 N 个时刻可采集 M 种属性数据.传感器节点第 i 个属性 $Attr_i$ 监测到的数据是一个长度为 N 的时间序列 $s_i=(s_{i,0},s_{i,1},\ldots,s_{i,N-1})$,其中 $s_{i,j}$ 表示第 i 个属性在第 j 时刻采集的数据.这样,该传感器上的原始数据被

抽象为一个矩阵 S^0,

$$S^0=\begin{bmatrix}s_0\\ s_1\\ \vdots\\ s_{M-1}\end{bmatrix}=\begin{bmatrix}s_{0,0} & s_{0,1} & \cdots & s_{0,N-1}\\ s_{1,0} & s_{1,1} & \cdots & s_{1,N-1}\\ \vdots & \vdots & \ddots & \vdots\\ s_{M-1,0} & s_{M-1,1} & \cdots & s_{M-1,N-1}\end{bmatrix}\square[t_0,t_1,...,t_{N-1}].$$

4.1 方案概述

方案 1:把每个时间序列 s_i 包含的数据按定义 3 进行规范化,把多个序列连接成一个序列(s_0,s_1,…,s_{N-1})(或按时间把数据归并成一个序列 $(t_0^T,t_1^T,...,t_{M-1}^T)$),将规范化误差作为误差限,使用第 3 节的 SWCEB 压缩算法即可.

方案 2:用二维 Haar 小波变换.传感器节点载有的传感模块个数一般不多,M 远小于 N,而二维小波变换做完全分解要求矩阵最好为方阵且大小为 2 的幂次方.因此,二维小波变换适用于传感模块尽可能多,或者考虑传感器节点间的空间相关性时,即尽量增大 M 的值.多维小波的标准分解方法,对于二维信号就是先对每一行进行求均值和差值的过程,即对每一行进行一维 Haar 小波变换;然后使用同样的方法对每一列进行求均值和差值.Chakrabarti[15]提出了非标准分解,交替地在各维上逐级进行求均值和差值.

方案 3:选出基信号,回归表示其他信号.不同属性的感知数据以及同一属性不同时段的数据具有相关性.Antonios[3]指出,以股票市场的工业和保险指数分别作为 X 轴和 Y 轴坐标作散点图,直观地看,近似一条直线.每个时间序列本身不是线性的,他提出的 SBR 算法将根据所有序列数据分布特征挑出的基信号为自变量,使用回归模型分段近似其他序列.原始的多个时间序列可以表示为基信号加上一些回归参数.当传感器测量的属性间相关性较大时,此方案比较好.但 SBR 没有考虑误差限问题,只是在满足数据压缩的条件下最大程度地压缩数据,可能导致两个问题:误差还没有满足要求算法也终止;误差已经满足要求却还在不断压缩.

4.2 基于回归的多属性误差有界小波数据压缩算法

基于回归的多属性的误差有界小波数据压缩算法(MWCEB)分为 3 大步:挑选基信号;用 SWCEB 算法处理基;其他信号用回归参数表示.把某属性的 N 次采样看成是一个离散型的随机变量的 N 次实验,属性 $\boldsymbol{X}$ 的 N 次实验的测量值记为(x_0,x_1,…,x_{N-1}).各属性间的关系用相关系数来度量.

定义 **7**(方差). 离散型随机变量 $\boldsymbol{X}$ 的数学期望为 $E(X)=\sum_i p_i x_i=\frac{1}{N}\sum_{i=0}^{N-1}x_i$,方差为 $D(\boldsymbol{X})=E((\boldsymbol{X}-E(\boldsymbol{X}))^2)$.

定义 **8**(协方差). 随机变量 $\boldsymbol{X}$ 和 $\boldsymbol{Y}$ 的协方差 $Cov(\boldsymbol{X},\boldsymbol{Y})=E((\boldsymbol{X}-E(\boldsymbol{X})(\boldsymbol{Y}-E(\boldsymbol{Y})))$.协方差用来衡量两个样本之间的相关性有多少,也就是一个样本的值偏离程度会对另外一个样本的值偏离产生多大的影响.

定义 **9**(相关系数矩阵). 样本的相关系数 $r_{xy}=\frac{Cov(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}}=\frac{\sum_{i=0}^{N-1}[(x_i-E(X))(y_i-E(Y))]}{\sqrt{\sum_{i=0}^{N-1}(x_i-E(X))^2}\sqrt{\sum_{i=0}^{N-1}(y_i-E(Y))^2}}$.

如 $\boldsymbol{X}$,$\boldsymbol{Y}$ 呈正(负)相关,则 r 为正(负)值;r=1(–1)时为完全正(负)相关,所有散点都在回归直线上.散点分布在回归直线上下越离散,r 的绝对值越小,相关越不密切.若每个节点可采集 M 种属性,则用相关系数矩阵 R 表征属性间的关系,矩阵中第 i 行 j 列个元素代表第 i 个属性 $Attr_i$ 与第 j 个属性 $Attr_j$ 的相关系数.

定义 **10**($\boldsymbol{X_j}$在基集合上的最佳相关). 设有 M 个时间序列 X_0,X_1,…,X_{M-1},相互之间可能存在相关,已将它们划归到基集合 *BaseSet* 和候选集合 *CandSet*,在集合中用序号代表序列.定义候选集合里元素 X_j 在基集合上的最佳相关为 $bestfit_j=\max(|r_{ij}|),i\in BaseSet$.候选集合中的元素将用基集合中与自己相关系数绝对值最大的元素回归近似,如果误差太大则自己变成基.显然,基集合里元素的最佳相关为 1.为实现后面的算法,定义辅助数组 *bestfit* 表示各个序列在已选出的基集合上的最佳相关.

```
struct {
    double r;        //bestfit[i].r 表示 X_i 与已选出基集合的元素之间具有的相关系数绝对值的最大值
    int pos;         //bestfit[i].pos 表示 X_i 将用 X_pos 作基进行回归,X_i 通过基集合中的 X_pos 达到最佳相关
} bestfit[M];
```

```
struct {
    int rowno[M];     //候选属性的序号.初始时保存了所有时间序列的序号
    int len;          //初始时长度为 M
} cand;               //候选集合
double rowsum[M]={0};     //最初表示相关系数矩阵的行元素和,后来表示期望收益数组 income
```

定义 **11**(增加 $\boldsymbol{X_j}$ 为基的期望收益). 设有 M 个时间序列已划分为基集合和候选集合.此时若添加 X_j 到基集合,定义期望收益 $income_j=\sum_k(|r_{jk}|-bestfit_k),k\in\{|r_{jk}|>bestfit_k\}$.最初,候选集合包含 M 个序列,基集合为空.将相关系数矩阵元素的绝对值按行求和,将和最大对应的时间序列加入基集合作为第 1 个基.继续寻找其他基,每次选择候选集合中期望收益最大的 X_j 加入基集合;同时修改 *bestfit* 为最新值,这样,*bestfit*[*i*].*pos* 始终指明候选集合里的 X_i 将用 X_{pos} 作基进行回归.

算法 **6**. 基信号挑选算法.

输入:M 种属性在 N 个时刻的采集数据 S^0,收益界 *eps*.

输出:*bestfit*.

```
1: 使用 S^0,根据定义 9 计算相关系数矩阵 r[0..M−1][0..M−1];
2: 计算 rowsum[0..M−1]各元素为相关系数矩阵 r 每行各元素的绝对值和;
3: 初始化候选集合 cand,其中数组 cand.rowno 保存了所有时间序列的序号;
4: 数组 cand.rowno 按非递减排序:第 i 个元素 cand.rowno[i]以 rowsum[cand.rowno[i]]为排序关键字;
5: temp=cand.rowno[−−cand.len];     //挑选 rowsum 中值最大的为第 1 个基 temp
6: for (i=0; i<M; i++) {            //初始化最佳相关数组
     bestfit[i].r=fabs(r[temp][i]);
     bestfit[i].pos=temp; }
7: do {                             //算法主体:根据期望收益挑选基
     for (i=0; i<M; i++) rowsum[i]=0;
     for (i=0; i<cand.len;i++)      //挑选出了新基,每个候选基计算更新自己的期望收益
       for (j=0; j<cand.len; j++)   //期望收益只与候选基之间的相关系数有关
         if (fabs(r[cand.rowno[i]][cand.rowno[j]])>bestfit[cand.rowno[j]].r)
           rowsum[cand.rowno[i]]+=fabs(r[cand.rowno[i]][cand.rowno[j]])−bestfit[cand.rowno[j]].r;
     double rtemp=0;                //用 temp 表示期望收益最大的属性的序号
     for (i=0; i<M ;i++) if (rowsum[i]>rtemp) {rtemp=rowsum[i];temp=i; }  //挑期望收益最大的作新基
     if (rowsum[temp]>=eps){        //期望收益大于收益界,选择 X_temp 作基
       对于每个候选基,如果最佳相关发生变化,修改 bestfit;
       将 X_temp 从候选集合 cand 删除,同时候选集合元素减少 1 个; }
   } while ((cand.len>0) && (rowsum[temp]>=eps));
```

一般地,若 $M<N$,则算法 6 的时间复杂度为 $O(M^2N)$.逐个处理算法 6 的输出 *bestfit*[*i*],若 *bestfit*[*i*].*pos*==*i*,则表示 X_i 是基,使用本文第 3 节的一维压缩算法即可;否则,X_i 用 $a\times X_{pos}+b$ 近似,计算出 a,b 再传输即可.

回归过程:首先对基的采样数据进行小波变换,然后在误差限内对变换后的小波系数进行零化,直接用重构出的数据作为回归的基信号 X,这就避免了基的重构误差传播给其他候选属性.候选属性 $\tilde{Y}=aX+b$,求 a,b 使 $\|Y-\tilde{Y}\|_2$ 最小.最小化 $Q=\sum_i(y_i-ax_i-b)^2$,则 $\begin{cases}\frac{\partial Q}{\partial a}=0\\ \frac{\partial Q}{\partial b}=0\end{cases}$,有 $\begin{cases}\sum_i(y_i-ax_i-b)(-x_i)=0\\ \sum_i(y_i-ax_i-b)(-1)=0\end{cases}$,解这个二元一次方程组求 a,b.这样,候选属性用若干对回归参数表示即可.

性质 5(误差有界). MWCEB 算法通过调整收益界和每次处理的数据个数,可以确保$||e||_\infty$满足误差限要求.

证明:基信号挑选算法将不同的属性按相关性分到某个基代表的组中,组中的其他信号用回归参数表示.收益界的选取决定了基信号的个数,收益界越小,需要的基越多,误差越小.通过减少收益界,使相关性较小的组分裂成多个组内更相关的组.极端情况,每个属性一组,自己作基直接传输给接收方,可以确保$||e||_\infty$满足误差限要求,但这退化成用 SWCEB 算法处理的单属性情况.另外,当回归重构误差较大时,减少每次处理的数据个数,即将基信号分段作为多次回归的基,用更多的线段去近似,理论上也可以减少误差,但这会使压缩效果变差. □

5 实 验

数据集 *A* 由 Catterall 等人提供(http://www.comp.lancs.ac.uk/~catterae/alife2002/),包含 5 个 Smart-It 无线传感器节点同时采集的 1 690 次数据.Smart-It 节点可以采集声强、温度、光强等 6 种属性.数据集 *B* 由 Samuel Madden 等人提供(http://db.csail.mit.edu/labdata/labdata.html),包含 54 个 Mica2Dot 节点同时采集的 230 多万次数据.每个 Mica2Dot 节点采集 4 种属性数据:温度、湿度、光强和电压,我们分别记为 0#,1#,2#,3#属性.根据定义 9 计算相关系数矩阵,数据集 *A* 的属性间数据相关性很小,数据集 *B* 的属性间相关性大.

Micaz 节点存储的采样数据、小波系数、回归参数、段的均值等均需要 2 个字节进行存储,而对于序号、计数器等不大的整数可用 1 个字节存储.采用 VC++ 6.0 在 PC 机上实现算法.没有进行量化和熵编码.数据压缩性能用空间节省率(space savings)度量,定义为压缩后减少的数据量除以原始数据大小.

5.1 单属性或属性间相关性小时

文献[16]通过构造数据流的分段常量近似提出了两种朴素的在线压缩算法,消除了单属性数据中的时间相关并保证误差有界.我们选择其中性能更好的 PMC-MEAN 算法与本文第 3 节提出的 SWCEB 算法进行比较.PMC-MEAN 算法用段中所有数据的均值及此段的结束时间来表示一个段,整个数据流由若干个段组成.为节省空间,段的结束时间用 1 个字节表示,若段的结束时间用 2 个字节表示,则 PMC-MEAN 性能会比下面的结果更差.此外,SWCEB 要用一个等于原始数据长的二进制位串 *bitflag* 来说明零化系数的位置.

使用数据集 *A* 第 1 个传感器开始的 1 024 次采样.我们研究节点缓冲区分别积累 16,32,64,…,1 024 次采样再压缩时算法的性能:

(1) 选择光强作为待传输的数据.光强波动大,数据范围为[1,171],均值为 67.485 4,均方差为 74.796 9.实验结果如图 3、图 4 所示;

(2) 选择温度为待传输数据.温度取值仅为{22,23},均值为 22.1025,均方差为 0.303 4.温度波动很小,精度不高.实验结果表明,PMC-MEAN 性能更优;

(3) 选择声强为待传输数据.声强波动大且频繁,数据范围为[1,104],均值为 9.030 3,均方差为 17.399 8.实验结果如图 5、图 6 所示.

实验结果表明:

① 为获得较高空间节省率,批处理压缩算法应以节点缓冲区大小为上限尽可能多积累一次处理的数据;

② SWCEB 一次处理数据的最佳数目与数据集本身的分布、要求的$||e||_\infty$有关.当要求的$||e||_\infty$变小或原序列数据波动变大时,相关数据范围缩小,一次处理的数据量要减少;但是数据太少会降低消除时间相关的效果;

③ 当要求$||e||_\infty$较小、原序列数据波动大时,SWCEB 可以比较精确地逼近原序列;而 PMC-MEAN 效果较差,甚至出现待传输数据反而变多的情况;

④ 当要求$||e||_\infty$较大、原序列数据波动小时,PMC-MEAN 作为一种粗略估计算法开销更小.

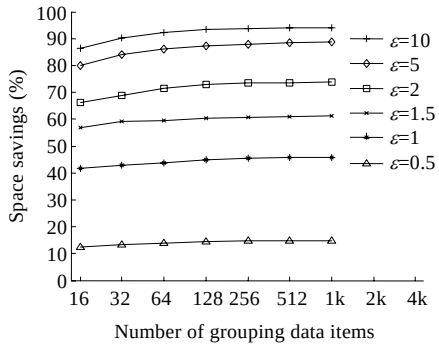

Fig.3 Compressing light sensor data using PMC-MEAN
图 3 用 PMC-MEAN 算法压缩光强数据

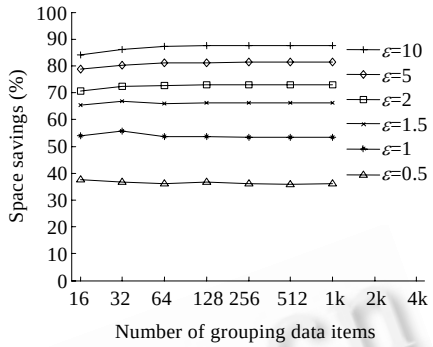

Fig.4 Compressing light sensor data using SWCEB
图 4 用 SWCEB 算法压缩光强数据

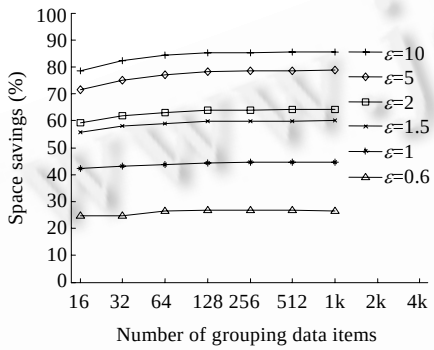

Fig.5 Compressing sound sensor data using PMC-MEAN
图 5 用 PMC-MEAN 算法压缩声强数据

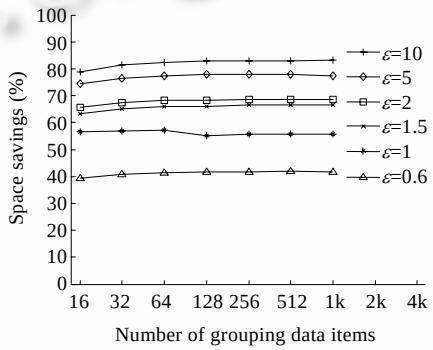

Fig.6 Compressing sound sensor data using SWCEB
图 6 用 SWCEB 算法压缩声强数据

5.2 属性间相关性大时

从数据集 *B* 取出第 1 个传感器的全部采样,按日期时间排序,选出最开始的 1 024 次采样作为实验数据.如果每次处理数据的个数小于等于 2,则回归不能带来任何压缩;另外,为了便于用小波压缩挑选出的基,数据长度最好为 2 的幂次方,所以每次处理数据的个数最少为 4.

采用 MWCEB 算法,实验过程是:① 为提高整体节省率,传感器节点根据全体 1 024 次采样的整体相关性来挑选基,并分组;② 每组的基信号分别执行 SWCEB 算法,进行小波分解、零化后传输.接着在本地重构出基信号用于回归,这样,基的重构误差就不会影响回归误差,并且重构出的基信号比原来更规整,更适合于作为基;③ 非基属性使用自己所在组的基执行线性回归算法,计算出回归参数后传输.非基属性的原始数据不用再传输.另外,由于各种属性数据的分布不同,采用规范化误差作为误差限.

如果同一组中的非基属性用基信号回归重构时不能满足误差限,则将信号等分成段再用基信号中同时间的段进行回归.研究在不同收益界、误差限下,回归基信号长度为 4,8,16,…,1 024 时的空间节省率.见表 1,利用算法 6 得到:方式 1 收益界太大,误差太大;方式 4 收益界太小,退化为单独传输各属性,没有利用属性间相关.只考虑方式 2、方式 3.

按方式 2 实验:1#属性(湿度)是基信号,执行 SWCEB 算法.图 7 表明,不同规范化误差下,空间节省率与每次处理的数据长度的关系,因此在压缩各组的基时,选择每次处理长度为 1 024.实验发现,压缩基时使用的规范化误差越小,后面回归时效果越好,因此选择ε=0.01.在满足误差限的前提下,使用尽可能长的数据段可获取更高的空间节省率.图 8 是规范化误差为 0.145 47 时温度的重构数据与原始采样的对比,按 512 个元素一段进行回归,

只需传输 2×2 个回归参数.图 9 为规范化误差为 0.280 07 时电压的重构数据与原始采样的对比,1 024 个元素一段,只需传输 2 个回归参数.2#属性(光强)是单独的基信号,执行 SWCEB 算法即可.按方式 2 压缩时,各属性的空间节省率、规范化误差见表 2.

按方式 3 实验:0#属性(温度)是基信号,执行 SWCEB 算法.图 10 为规范化误差为 0.202 39 时电压的重构数据与原始采样的对比,按 1 024 个元素一组进行回归.1#(湿度),2#(光强)作为基信号需要单独传输(见表 3).

Table 1 Relation between benefit bound and attributes grouping

表 1 收益界与分组的关系

Scheme	Benefit bound	*bestfit*[*i*].*pos*, *i*=0,1,2,3
1	≥0.158	1 1 1 1
2	[0.085,0.157]	1 1 2 1
3	[0.030,0.084]	0 1 2 0
4	[0,0.029]	0 1 2 3

Table 2 Summary of the 2nd schme

表 2 方式 2 小结

Attribute	Space savings (%)	Normalized error
0# (temperature)	99.609 4	0.145 47
1# (humidity)	88.476 6	0.010 00
2# (light)	85.937 5	0.050 00
3# (voltage)	99.804 6	0.280 07

Table 3 Summary of the 3rd schme

表 3 方式 3 小结

Attribute	Space savings (%)	Normalized error
0# (temperature)	91.113 3	0.020 00
1# (humidity)	88.476 6	0.010 00
2# (light)	85.937 5	0.050 00
3# (voltage)	99.804 6	0.202 39

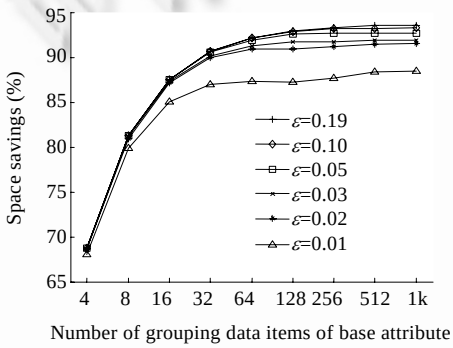

Fig.7 Compressing base data (1# attribute) using SWCEB

图 7 用 SWCEB 算法压缩基(1#属性)

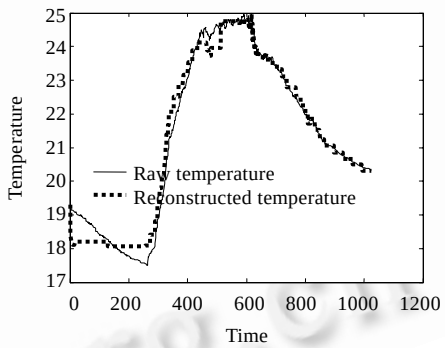

Fig.8 Reconstruction by regression (0# based on 1#)

图 8 回归重构(0#属性以 1#属性为基)

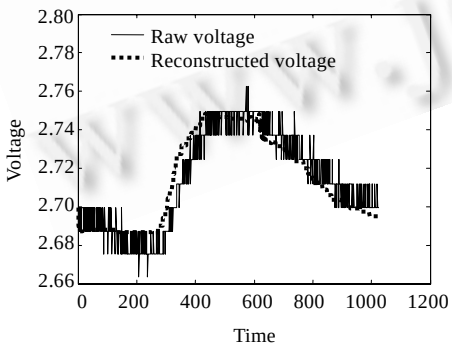

Fig.9 Reconstruction by regression (3# based on 1#)

图 9 回归重构(3#属性以 1#属性为基)

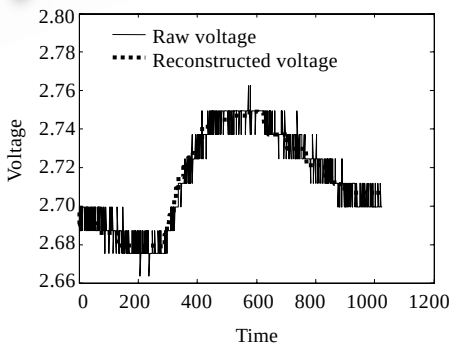

Fig.10 Reconstruction by regression (3# based on 0#)

图 10 回归重构(3#属性以 0#属性为基)

实验结果表明:

① MWCEB 算法的压缩性能优良.实验中,通过选择合适参数,基属性的空间节省率可以达到 85%以上,非

基属性的空间节省率达 99.8046%.

② 非基属性的规范化误差较大,但 RACE[10],PMC-MEAN[16]等算法即使规范化误差比本文算法大也达不到这么高的空间节省率.RACE 在规范化误差为 0.191 时空间节省率仅为 87.5%.

③ 将基信号等分分段作多次回归,类似于分段线性近似,理论上可以减少误差.但我们发现,实验中回归数据段长至少为 512.原因是当回归区间较短时,求回归参数的线性方程组为病态方程组,方程组系数的微小误差导致根(回归参数)较大变化,虽然可以用迭代改善法等计算方法求解,考虑到传感器节点资源有限就不作特别处理.这样当基分段太多时,任何一段计算回归参数出了问题,很容易造成该段回归结果超过规范化误差.

④ MWCEB 适合于属性间相关性较大、非基属性空间节省率非常高但精度要求不太高的场合.进一步地,可以根据非基属性允许的误差限,动态确定每次参与回归计算的采样数据个数,即进行自适应分段回归.

6 结 论

针对传感器节点的数据采样之间不存在空间相关性或空间相关性不稳定,本文设计了在各节点上独立运行的误差有界的数据压缩算法.当单属性或属性间相关性小时,采用第 3 节提出的一维 Haar 小波压缩算法 SWCEB;当属性间相关性大时,采用第 4 节提出的基于线性回归和 Haar 小波的压缩算法 MWCEB.在 WSN 基于模型的数据获取中,如何构造误差小并能随时间动态演化的模型值得深入探讨.如何同时考虑数据的空间相关,特别是不规则空间分布时的相关,突破现有采用 Voronoi 多边形的方法,也是我们下一步研究的重点.

References:

[1] Li JZ, Gao H. Survey on sensor network research. Journal of Computer Research and Development, 2008,45(1):1−15 (in Chinese with English abstract).

[2] Kimura N, Latifi S. A survey on data compression in wireless sensor networks. In: Proc. of the Int'l Conf. on Information Technology: Coding and Computing, Vol.2. Piscataway: IEEE Press, 2005. 8−13.

[3] Deligiannakis A, Kotidis Y, Roussopoulos N. Dissemination of compressed historical information in sensor networks. The VLDB Journal, 2007,16(4):439−461. [doi: 10.1007/s00778-005-0173-5]

[4] Chu D, Deshpande A, Hellerstein JM, Hong W. Approximate data collection in sensor networks using probabilistic models. In: Proc. of the 22nd Int'l Conf. on Data Engineering. Piscataway: IEEE Press, 2006. 48−59.

[5] Kanagal B, Deshpande A. Online filtering, smoothing and probabilistic modeling of streaming data. In: Proc. of the IEEE 24th Int'l Conf. on Data Engineering. Piscataway: IEEE Press, 2008. 1160−1169.

[6] Najafi H, Lahouti F, Shiva M. AR modeling for temporal extension of correlated sensor network data. In: Proc. of the Int'l Conf. on Software in Telecommunications and Computer Networks. Piscataway: IEEE Press, 2006. 117−120.

[7] Tulone D, Madden S. PAQ: Time series forecasting for approximate query answering in sensor networks. In: Römer K, Karl H, Mattern F, eds. Proc. of the Wireless Sensor Networks (EWSN 2006). LNCS 3868, Berlin: Springer-Verlag, 2006. 21−37.

[8] Borgne YL, Bontempi G. Unsupervised and supervised compression with principal component analysis in wireless sensor networks. In: Ganguly AR, *et al*., eds. Proc. of the Knowledge Discovery from Sensor Data (Sensor-KDD 2007). Boca Raton: CRC Press, 2008.

[9] Ganesan D, Estrin D, Heidemann J. DIMENSIONS: Why do we need a new data handling architecture for sensor networks? SIGCOMM Computer Communication Review, 2003,33(1):143−148. [doi: 10.1145/774763.774786]

[10] Chen HM, Li J, Mohapatra P. RACE: Time series compression with rate adaptive and error bound for sensor networks. In: Proc. of the 2004 IEEE Int'l Conf. on Mobile Ad-Hoc and Sensor Systems. Piscataway: IEEE Press, 2004. 124−133.

[11] Ciancio A, Pattem S, Ortega A, Krishnamachari B. Energy-Efficient data representation and routing for wireless sensor networks based on a distributed wavelet compression algorithm. In: Proc. of the IPSN. New York: ACM Press, 2006. 309−316.

[12] Zhou SW, Lin YP, Zhang JM, Ouyang JC, Lu XG. A wavelet data compression algorithm using ring topology for wireless sensor networks. Journal of Software, 2007,18(3):669−680 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/18/669.htm [doi: 10.1360/jos180669]

[13] Garofalakis M, Gibbons PB. Wavelet synopses with error guarantees. In: Proc. of the ACM SIGMOD. New York: ACM Press, 2002. 476−487.

[14] Pan LQ, Li JZ, Luo JZ. An approximate query processing algorithm with confidence based on model fitting in sensor networks. Journal of Computer Research and Development, 2008,45(1):73−82 (in Chinese with English abstract).

[15] Chakrabarti K, Garofalakis M, Rastogi R, Shim K. Approximate query processing using wavelets. The VLDB Journal, 2001,10(3): 199−223.

[16] Lazaridis I, Mehrotra S. Capturing sensor-generated time series with quality guarantees. In: Proc. of the 19th Int'l Conf. on Data Engineering. Piscataway: IEEE Press, 2003. 429−440.

附中文参考文献:

[1] 李建中,高宏.无线传感器网络的研究进展.计算机研究与发展,2008,45(1):1−15.

[12] 周四望,林亚平,张建明,欧阳竞成,卢新国.传感器网络中基于环模型的小波数据压缩算法.软件学报,2007,18(3):669−680. http://www.jos.org.cn/1000-9825/18/669.htm [doi: 10.1360/jos180669]

[14] 潘立强,李建中,骆吉洲.无线传感器网络中基于模型拟合的可信近似查询处理算法.计算机研究与发展,2008,45(1):73−82.

张建明(1976－),男,湖南益阳人,博士生,副教授,CCF 学生会员,主要研究领域为传感器网络中的数据管理,数据挖掘.

周四望(1971－),男,博士,副教授,主要研究领域为传感器网络中的信号处理,小波分析.

林亚平(1955－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为计算机网络,机器学习.

欧阳竞成(1967－),男,博士生,主要研究领域为 P2P 中的信息检索与安全.