

基于块 A*正交匹配追踪的多传感器数据联合重构算法

练秋生* 刘芳 陈书贞

(燕山大学信息科学与工程学院 秦皇岛 066004)

摘要: 针对 A*正交匹配追踪(A*OMP)算法计算复杂度高,且不能利用信号的结构稀疏性这一缺陷,本文提出了块 A*OMP 算法并将其用于解决分布式压缩感知中的信号联合重构问题。该算法用原子块取代单个原子作为搜索树中的节点,在计算路径代价时用搜索树中所有路径的最大长度取代信号的稀疏度。然后在块 A*OMP 算法的基础上,选择与残差矩阵投影误差最小的原子块作为新的节点,得到了一种用于解决 MMV(Multiple Measurement Vector, MMV)问题的块 A*OMP 算法,并利用该算法对相邻区域内的多个传感器所测的温度信号进行了联合重构。实验结果表明,该算法的重构性能优于 MMV 正交匹配追踪(OMPMMV)算法。

关键词: 分布式压缩感知; 联合重构; A*正交匹配追踪算法; 块稀疏

中图分类号: TP393; TP391

文献标识码: A

文章编号: 1009-5896(2013)03-0721-07

DOI: 10.3724/SP.J.1146.2012.01013

A Joint Reconstruction Algorithm for Multiple Sensor Data Based on Block A* Orthogonal Matching Pursuit

Lian Qiu-sheng Liu Fang Chen Shu-zhen

(Institute of Information Science and Technology, Yanshan University, Qinhuangdao 066004, China)

Abstract: Considering the disadvantage of the high complexity and ignoring signal's structural sparsity in A* Orthogonal Matching Pursuit (A*OMP) algorithm, a block A*OMP algorithm is proposed for block-sparse signals, and it is improved to solve the joint reconstruction problem for multiple signals in distributed compressed sensing. In the proposed algorithm, the single atom is replaced by a block that is composed of several atoms, and the sparsity is replaced by the maximum length of all the paths on the search tree when calculating the path cost. Then, on the basis of block A*OMP algorithm, a block A*OMP algorithm for Multiple Measurement Vector (MMV) problem is presented by projecting all blocks onto the residual matrix and selecting the block with the smallest projection error as a new node. With this algorithm, the temperature signals which are measured by sensors in the adjacent region are jointly reconstructed perfectly. Experiments show that the reconstruction performance of this algorithm outperform Orthogonal Matching Pursuit for MMV (OMPMMV) algorithm.

Key words: Distributed Compressed Sensing (DCS); Joint reconstruction; A*Orthogonal Matching Pursuit (A*OMP); Block sparsity

1 引言

无线传感器网络(Wireless Sensor Networks, WSN)通过布置在环境中的多个传感器节点能够灵活方便地监测环境的变化,但它却受能量、存储空间和计算处理能力的限制。如何有效地利用 WSN 节点感知数据的时间相关性和空间相关性来减少 WSN 中的能量消耗,节省存储空间,降低计算的复杂度就显得尤为重要。

压缩感知(Compressed Sensing, CS)理论是近年来信息处理领域发展迅速的一个理论分支^[1],它的出现为 WSN 的研究提供了一个新思路。该理论表明,如果信号是可压缩的或在某个变换域是稀疏的,那么就可以用一个与变换基不相干的观测矩阵将变换所得的高维信号投影到一个低维空间上,然后通过求解一个稀疏优化问题来实现信号的精确重构。在 WSN 中利用 CS 技术能有效地减少数据采集和传输过程中的能量消耗,降低传输带宽^[2]。

传统的获取传感器信号的方法是先均匀采样,然后再经过压缩和编码后发送到汇聚节点(Fusion Center, FC),采样得到的大部分数据在压缩时都会被丢弃,造成了采样资源的严重浪费。而 CS 则通

2012-08-06收到, 2012-10-22改回

国家自然科学基金(61071200, 60772079)和河北省自然科学基金(F2010001294)资助课题

*通信作者: 练秋生 lianqs@ysu.edu.cn

过随机采样直接获得少量的采样值,大大提高了效率,节约了资源。CS研究单个信号的重构问题,只利用了信号的时间相关性或空间相关性,未能充分利用信号的相关性。而分布式压缩感知(Distributed Compressive Sensing, DCS)^[3]研究多个信号的联合重构问题,同时利用了信号的时间相关性和空间相关性,进一步减少了准确重构所需的采样点数。在一定的小范围内具有时间和空间相关性的多个传感器所测的温度、湿度等信号在用稀疏基(如小波)表示时,其显著系数出现的位置大致相同,因此小范围内多个传感器信号的联合重构问题可以看作一个MMV (Multiple Measurement Vector)问题。此外,显著系数是成块出现的,即可将信号看作是块稀疏的。

MMV问题是压缩感知理论研究中的一个重要的领域^[4,5],MMV问题解决了由共享支撑集的多个稀疏信号所组成的信号集的联合重构问题。现有解决MMV问题的许多方法,如可压缩的多信号分类算法(Compressive Multiple Signal Classification, C-MUSIC)^[6]、向前压缩子空间拟合算法(forward Compressive Subspace Fitting, forward CSF)、向后压缩子空间拟合算法(backward Compressive Subspace Fitting, backward CSF)^[7]都需要预先确定信号的稀疏度,而对于大多数信号而言,稀疏度却很难预先确定。因此研究未知稀疏度的重构算法就显得尤为重要。

本文首先简要地介绍了CS理论及MMV问题。为了利用信号的块稀疏特性,本文在A*OMP算法^[8,9]的基础上提出了块A*正交匹配追踪(Block A* Orthogonal Matching Pursuit, BA*OMP)算法。该算法在路径扩张时用原子块取代单个原子作为新的节点,大大降低了算法的复杂度,在计算路径代价时用搜索树中所有路径的最大长度取代信号的稀疏度。为了将BA*OMP算法用于解决MMV问题,本文将与残差矩阵投影误差最小的原子块作为新的节点并用矩阵范数代替向量范数,得到了一种针对MMV问题的无需预先确定信号稀疏度的块A*正交匹配追踪算法(Block A* Orthogonal Matching Pursuit for MMV, BA*OMPMMV)算法。在实验部分,分别利用合成信号和Intel Berkeley Research lab^[10]的WSN所测的温度信号对BA*OMPMMV算法进行了验证。结果表明该算法的重构性能优于OMPMMV算法^[11]的重构性能。

2 CS理论及MMV问题

2.1 CS理论

CS理论解决了如何利用少量的观测值从一个

欠定的系统中高效地恢复稀疏信号的问题。CS模型中并不是直接测量信号 x 本身,而是将信号 x 投影到观测矩阵 Φ 上得到观测向量 y :

$$y = \Phi x \quad (1)$$

式中 $x \in R^N$, $y \in R^M$, $\Phi \in R^{M \times N}$, $M < N$ 。而信号 x 在稀疏基 $\Psi = [\psi_1, \psi_2, \dots, \psi_N]$ 上是 k 稀疏的,即

$$x = \Psi s \quad (2)$$

式中 $s \in R^N$, $\Psi \in R^{N \times N}$, $\|s\|_0 \leq k$, $k < N$ 。将式(2)代入式(1)得

$$y = \Phi x = \Phi \Psi s = A s \quad (3)$$

式中 $A = \Phi \Psi \in R^{M \times N}$ 称为全息基。由于 $M < N$,式(3)是一个非适定问题,一般情况下有无穷多种解。但由于信号是 k 稀疏的,因此在一定条件下当观测值的维数大于信号的稀疏度时就可以通过非线性的方法重构出信号。信号的重构问题可表示为 l_0 范数最小优化问题:

$$\hat{s} = \arg \min \|s\|_0, \quad \text{s.t. } y = \Phi x = A s \quad (4)$$

当考虑噪声时,CS模型可表示为

$$\hat{s} = \arg \min \|s\|_0, \quad \text{s.t. } y = A s + z \quad (5)$$

式中 $z \in R^M$ 表示观测噪声。

2.2 块稀疏信号

块稀疏信号^[12],即非零系数成簇出现的稀疏信号。为了定义块稀疏,可将 s 看作块的级联,即

$$s = \left[\underbrace{s_1 \cdots s_d}_{s^T[1]} \underbrace{s_{d+1} \cdots s_{2d}}_{s^T[2]} \cdots \underbrace{s_{N-d+1} \cdots s_N}_{s^T[H]} \right]^T \quad (6)$$

式中 d 表示块的长度, $s[h]$ 代表第 h 块, $N = Hd$ 。假设 $M = Wd$ (W 为整数),则类似式(6)可将 A 看作大小为 $M \times d$ 的列块 $A[h]$ 的级联,即

$$A = \left[\underbrace{a_1 \cdots a_d}_{A[1]} \underbrace{a_{d+1} \cdots a_{2d}}_{A[2]} \cdots \underbrace{a_{N-d+1} \cdots a_N}_{A[H]} \right] \quad (7)$$

如果信号 x 是块 k 稀疏的,则 $s[h]$ ($h = 1, \dots, H$)的 l_2 最多有 k 个是非零的。当 $d = 1$ 时,块稀疏性退化为式(2)所示的普通意义上的稀疏性。

2.3 MMV问题

MMV问题解决了拥有相同稀疏支撑集的多个信号的联合重构问题。标准形式的MMV定义^[6]为:

假设给定多观测向量 $Y = [y_1, \dots, y_n] \in R^{M \times n}$,共享稀疏支撑集的信号集 $X = [x_1, \dots, x_n] \in R^{N \times n}$ 和观测矩阵 $A \in R^{M \times N}$ 。正整数 M, N, n ($M < N$),分别表示每个观测向量的维数,每个信号的维数,信号的个数。 x_i 和 x^i 分别表示 x 的第 i 列和第 i 行。那么标准形式的MMV问题可以表示为下面的最优化问题:

$$\text{minimize } \|X\|_0, \quad \text{s.t. } Y = AX \quad (8)$$

式中 $\|\mathbf{X}\|_0 = |\text{supp}\mathbf{X}| = k$, $|\text{supp}\mathbf{X}| = \{1 \leq i \leq N : \mathbf{x}^i \neq 0\}$, 观测矩阵 \mathbf{Y} 是满秩的, 即: $\text{rank}(\mathbf{Y}) = n \leq \|\mathbf{X}\|_0$ 。由于每个 MMV 问题都能通过奇异值分解和降维操作转换成标准形式的 MMV 问题^[6], 所以假定本文中 MMV 问题均为标准形式。

当考虑噪声时, MMV 模型可表示为

$$\text{minimize } \|\mathbf{X}\|_0, \text{ s.t. } \mathbf{Y} = \mathbf{A}\mathbf{X} + \mathbf{Z} \quad (9)$$

式中 $\mathbf{Z} \in R^{M \times n}$ 表示观测噪声。

3 BA*OMP 算法

3.1 A*OMP 算法

A*OMP 算法在 CS 重构的过程中利用 A*search 搜索稀疏解, 其重构性能优于 BP(Basis Pursuit) 算法^[13], OMP(Orthogonal Matching Pursuit)算法^[14]以及 SP(Subspace Pursuit)算法^[15]。它是一种基于搜索树的半贪婪算法。树的每个节点代表一个原子, 将由原子集合组成的树枝或路径, 作为表示稀疏信号 \mathbf{x} 的候选。A*OMP 算法在初始化时, 选出 I 个与观测向量 \mathbf{y} 最匹配的原子作为 I 条初始路径并分别计算出这 I 条路径的代价, 而将剩下的 $P - I$ 条空路径的代价初始化为 $\|\mathbf{y}\|_2$ 。它们是最坏的路径, 将会首先被移除。在每次迭代过程中, 该算法首先利用代价函数在搜索树的所有路径中选出最好的一条路径。然后选出与最好路径的残差内积最大的 B 个原子并分别用它们来扩张最好的路径, 从而创建 B 条候选路径。如果这些候选路径中最好的路径不在搜索树中, 则用此路径取代已选出的最好路径。如果其它的候选路径不在搜索树中且其代价小于最坏的路径, 则用它们取代搜索树中最坏的路径。当最好的路径达到所需的长度, 即信号的稀疏度 k 时, 算法终止。

对于所有新的路径, A*OMP 算法通过将观测向量正交投影到路径所选的原子集上的方法来计算残差 \mathbf{r} :

$$\mathbf{r} = \mathbf{y} - \sum_{j=1}^l c^j \mathbf{s}^j \quad (10)$$

式中 \mathbf{s}^j 为第 j 次所选的字典原子, c^j 为第 j 次所选字典原子对应的系数。

A*OMP 算法的代价函数依赖于每条路径的残差的 l_2 范数, 即 $\|\mathbf{r}\|_2$ 。为了对不同的路径长度进行补偿, 代价函数同时采用了一个辅助结构。在文献[8]提出的 A*OMP 算法中, 提出了加性、乘性和自适应 3 种不同的模型。其中乘性的代价函数模型为

$$F(\mathbf{S}_i^l) = \alpha^{(k-l)} \|\mathbf{r}_i^l\|_2 \quad (11)$$

式中 l 为第 i 条路径的长度, k 为信号的稀疏度, $0 < \alpha < 1$ 为调整参数。 \mathbf{r}_i^l 表示第 i 条路径在经过 l 次

扩张后的残差。 \mathbf{S}_i^l 表示长度为 l 的路径。

A*OMP 算法在选择新的节点进行路径扩张时采用如下方法:

$$\mathbf{a}_j^l = \arg \max_{\mathbf{a}_j \in \mathbf{A} \setminus \mathbf{T}} \langle \mathbf{r}^{l-1}, \mathbf{a}_j \rangle \quad (12)$$

式中 \mathbf{T} 为已选过的原子, \mathbf{a}_j 为构成观测矩阵 \mathbf{A} 的原子。

3.2 BA*OMP 算法

为了便于描述, 首先定义路径的实际长度如下:

$$L = |J_1 \cup J_2 \cdots \cup J_m| \quad (13)$$

式中 m 表示路径中的节点数, $|\cdot|$ 表示集合的势, $J_i (i = 1, 2, \dots, m)$ 表示由路径中第 i 个原子块中的原子对应于 \mathbf{A} 的列索引所组成的集合。

文献[9]指出, 当采用动态的代价函数时, A*OMP 算法的重构性能优于 BP 算法, OMP 算法以及 SP 算法, 但该算法需事先确定信号的稀疏度, 而且计算复杂度较高。另外 A*OMP 算法无法利用信号的结构稀疏信息, 不适合块稀疏信号重构。针对块稀疏信号, 本文提出了 BA*OMP 算法。该算法事先将所有原子分成重叠的大小相等的原子块, 然后以原子块为单位来选取新的节点进行路径扩张, 每个节点代表一个原子块, 这样就大大降低了计算的复杂度, 提高了运算效率。为了计算不同长度的路径的代价, BA*OMP 算法在每次迭代结束时, 找出所有路径的最大长度 l_{\max} , 若此次迭代的路径最大长度与上次迭代的路径最大长度不同, 则更新长度未达到最大长度的路径的代价, 将路径的长度补偿为 l_{\max} 时所得到的残差作为代价, 否则不更新路径的代价。此外, BA*OMP 算法将截止条件设为 $\|\mathbf{r}\|_2 \leq \varepsilon$ (ε 的大小与噪声有关), 这样就可以完全回避信号的稀疏度。当原子块的大小 $d = 1$, 重叠的原子数 $o = 0$ 时, BA*OMP 算法退化为无需事先确定信号稀疏度的 A*OMP 算法。

BA*OMP 算法在选择新的节点进行路径扩张时, 采用如下方法^[16]:

$$\mathbf{A}[\hat{h}]^l = \arg \max_{\mathbf{A}[h] \in \mathbf{A} \setminus \mathbf{T}} (\|\mathbf{A}^T[\mathbf{h}]\mathbf{r}^{l-1}\|_2) \quad (14)$$

式中 \mathbf{T} 为已选过的原子块, $\mathbf{A}[h]$ 为构成观测矩阵 \mathbf{A} 的原子块。

实验结果表明该算法的性能优于 A*OMP 算法, 而且该算法不需要事先确定信号的稀疏度, 更贴近于实际应用。

3.3 BA*OMPMV 算法

为了将 BA*OMP 算法用来解决 MMV 问题, BA*OMPMV 算法延续了 BA*OMP 算法的路径扩张机制, 但改变新节点的选择方法, 将与残差矩阵投影误差最小的原子块作为新的节点, 并用

Frobenius 范数代替向量范数。该算法的截止条件为最好路径的残差 $\|R\|_F \leq \varepsilon$ 。该算法适用于解决共享支撑集且非零系数成簇出现的多个信号的联合重构问题。

对于所有新的路径，BA*OMPMMV 算法采用 OMPMMV 算法^[1]计算残差的方法来得到残差 R ：

$$R = P_S^\perp Y = (I - SS^\dagger)Y \quad (15)$$

式中 I 表示单位矩阵， S 表示由路径中所有原子块的并集所组成的矩阵， P_S^\perp 表示到 S 的值空间的正交补空间的正交投影矩阵。 $S^\dagger = S^H(SS^H)^{-1}$ 表示 Moore-Penrose 广义逆。

BA*OMPMMV 算法的代价函数也有 3 种不同的模型。其中乘性的代价函数模型为

$$F(S_i^j) = \alpha^{(l_{\max} - 1)} \|R_i^j\|_F \quad (16)$$

式中 l 为第 i 条路径的长度， l_{\max} 为所有路径的最大长度。 $0 < \alpha < 1$ 为调整参数。 R_i^j 表示第 i 条路径在经过 j 次扩张后的残差。 S_i^j 表示由 j 个节点所组成的路径 i 。

BA*OMPMMV 算法在选择新的节点进行路径扩张时，采用如下方法：

$$A[\hat{h}]^j = \arg \max_{A[h] \in A, T} (\|A^T[h]R^{j-1}\|_F) \quad (17)$$

式中 T 为已选过的原子块， $A[h]$ 为构成观测矩阵 A 的原子块。

BA*OMPMMV 算法是在 BA*OMP 算法的基础上扩展而来的。该算法首先选择 I 个原子块作为 I 条初始路径并计算路径代价；然后在每次迭代时选择代价最小的一条路径进行延伸，每次延伸选择 B 个原子块作为 B 个枝节点，每个枝节点与要延伸的路径一起构成一条候选路径，若候选路径满足替代条件则进行路径替代，否则删除候选路径。每次迭代结束时，计算一次最大路径长度 l_{\max} 并用 l_{\max} 更新路径代价；当满足截止条件时算法终止。使用的符号如表 1 所示，该算法的伪代码如表 2 所示。

表 1 BA*OMPMMV 算法中使用的符号

定义	注释
P	可搜索的路径数
I	初始搜索路径数
B	每次迭代所延伸的树枝数
S_i	由第 i 条路径中所有原子块的并集所组成的矩阵
$\text{index}(A[h])$	获取第 h 块中的原子对应于 A 中的列索引
Ω_i	由第 i 条路径所选原子在 A 中的列索引所组成的集合
L_i	第 i 条路径的长度
C_i	选择第 i 条路径的代价
R_i	第 i 条路径上的残差

表 2 BA*OMPMMV 算法的伪代码

```

(1)输入:  $P, I, B$ , 多观测矩阵  $Y = AX + Z$ , 原子块的长度  $d$ ,
      所有的原子块  $\{A[h]\}, h = 1, 2, \dots, H$ , 误差容限  $\varepsilon$ 
(2)初始化:
       $T = \emptyset$  //初始化已选过的原子块
       $\Omega_i = \emptyset, \forall i = 1, 2, \dots, P$  //初始化支撑集
      For  $i = 1 : I$  do //长度为  $d$  的  $I$  条初始路径
           $A[\hat{h}] = \arg \max_{A[h] \in A, T} (\|A^T[h]Y\|_F)$  //选择初始节点
           $T = [T \ A[\hat{h}]], \Omega_i = \Omega_i \cup \text{index}(A[\hat{h}])$ 
           $S_i = A[\hat{h}], R_i = P_{S_i}^\perp Y, C_i = \|R_i\|_F, L_i = 1$ 
          //初始化延伸路径的代价和路径长度
      End for
       $C_i = \|Y\|_F, L_i = 0, \forall i = I + 1, I + 2, \dots, P$ 
      //初始化其它未延伸路径的代价和路径长度
       $BP = 1$  //将路径 1 作为最好的路径
       $l_{\text{old}} = d$  //初始化搜索树中最长路径的长度
(3)循环执行步骤:
      While ( $\|R_{BP}\|_F > \varepsilon$ ) do
           $\hat{p} = BP, T = S_{BP}$  //最好的路径将被它的第 1 个延伸替代
          For  $i = 1 : B$  do //B 个枝节点组成 B 条候选路径
               $A[\hat{h}] = \arg \max_{A[h] \in A, T} (\|A^T[h]R_{BP}\|_F)$  //选择新的节点
               $T = [T \ A[\hat{h}]], \hat{S} = [S_{BP} \ A[\hat{h}]]$  //候选路径
              If ( $C(\hat{S}) < C(S_{\hat{p}}) \ \& \ (\hat{S} \neq S_j), \forall j = 1, 2, \dots, P$ ) then
                  //如果候选路径好于最坏的路径且不在搜索树中
                   $S_{\hat{p}} = \hat{S}, \Omega_{\hat{p}} = \Omega_{BP} \cup \text{index}(A[\hat{h}])$  //路径替代
                   $L_{\hat{p}} = L_{\hat{p}} + d$  //更新路径长度
                   $R_{\hat{p}} = P_{S_{\hat{p}}}^\perp Y$  //更新残差
                   $l_{\hat{p}} = \max(L_{\hat{p}}, l_{\text{old}}), C_{\hat{p}} = F(S_{\hat{p}})$  //计算代价
              End if
          End for
           $\hat{p} = \arg \max_{j \in \{1, 2, \dots, P\}} C_j$ 
          //找出最坏的路径作为下一次延伸的替代路径
      End for
       $l_{\max} = \max_{j \in \{1, 2, \dots, P\}} (L_j)$  //计算最长的路径长度
      If  $l_{\max} \neq l_{\text{old}}$  then //若最长的路径长度改变则更新代价
           $C_j = F(S_j), j = 1, 2, \dots, P, L_j \neq 0 \ \& \ L_j \neq l_{\max}$ 
      End if
       $l_{\text{old}} = l_{\max}$ 
       $BP = \arg \min_{j \in \{1, 2, \dots, P\}} C_j$  //选出最好的路径
      End while
(4)输出:
      重构的信号  $\hat{X}(\Omega_{BP}, :) = S_{BP}^\dagger Y$ 

```

4 实验结果及其分析

为了对比算法的重构性能，本文从归一化均方误差和准确重构率两个角度考虑算法的重构性能：

(1) 归一化均方误差

$$\text{nmse} = E \left(\frac{\|\hat{\mathbf{x}} - \mathbf{x}\|_2^2}{\|\mathbf{x}\|_2^2} \right) \quad (18)$$

$$\text{NMSE} = E \left(\frac{\|\hat{\mathbf{X}} - \mathbf{X}\|_F^2}{\|\mathbf{X}\|_F^2} \right) \quad (19)$$

式中 $\hat{\mathbf{x}}, \hat{\mathbf{X}}$ 表示重构的信号。式(18)针对单个信号的重构问题，式(19)针对多个信号的联合重构问题。

对于合成信号而言，仿真次数为 500 次。

(2) 准确重构率

$$\text{RR} = E \left(\frac{|\hat{Q} \cap Q_{\text{supp}}|}{|Q_{\text{supp}}|} \right) \quad (20)$$

式中 \hat{Q} 表示重构信号的支撑集， Q_{supp} 表示原信号的支撑集。仿真次数为 500 次。

在实验中， \mathbf{A} 的元素符合高斯分布(即 $a_{m,n} \sim N(0,1/N)$)。 \mathbf{A} 的每一行经过了归一化处理。合成信号的支撑随机分布在 5 个集中的位置，每个位置的支撑个数是随机的。本文中的信噪比 SNR 为原信号的能量与观测噪声能量的比值。此外，定义信号的采样率为

$$\gamma = M/N \quad (21)$$

4.1 BA*OMP 算法与 A*OMP 算法的对比

A*OMP 的参数选择 $I = 3, B = 2, P = 200, \alpha = 0.5$ 。BA*OMP 算法的参数选择 $I = 3, B = 2, P = 200, \alpha = 0.9$ ，原子块的大小 $d = 4$ ，重叠的原子数 $o = 2$ 。信号长度 $N = 256$ ，观测向量的维数 $M = 100$ 。信号 \mathbf{x} 的非零值服从[0,1]的均匀分布。

图 1 给出了无噪声的条件下，乘性 BA*OMP 算法与乘性 A*OMP 算法重构性能的对比情况。由图 1 可知，BA*OMP 算法的重构性能优于 A*OMP 算法。与 A*OMP 算法相比，BA*OMP 算法的归一化均方误差低，准确重构率高。

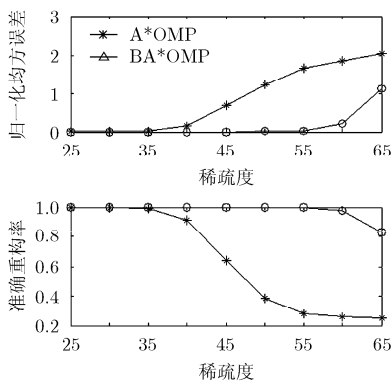


图 1 无噪声时 BA*OMP 算法与 A*OMP 算法的性能对比

4.2 基于合成信号的 BA*OMPMMV 算法与 OMPMMV 算法的对比

BA*OMPMMV 的参数选择 $I = 3, B = 2, P = 200, \alpha = 0.9$ ，原子块的大小 $d = 4$ ，重叠的原子数 $o = 2$ 。仿真实验中采用 $N = 500, n = 10, k = 30$ 。信号 \mathbf{X} 的非零行的每个元素服从[0,1]的均匀分布。

图 2 和图 3 分别给出了无噪声和信噪比为 10 dB 时，BA*OMPMMV 算法和 OMPMMV 算法重构性能的对比情况。由图 2 和图 3 可知，BA*OMPMMV 算法的重构性能优于 OMPMMV 算法的重构性能。前者相对于后者的归一化均方误差低，准确重构率高。达到较准确重构所需的采样率 BA*OMPMMV 算法比 OMPMMV 算法低 0.05 左右。

4.3 利用 BA*OMPMMV 算法重构多个传感器所测的温度信号

BA*OMPMMV 算法的参数选择如 4.2 节所示。本文中的实测数据来源于 Intel Berkeley Research lab 所测的温度值。根据各个传感器之间的相关性，选取了分布在相邻空间内的相关性很强的传感器 1、传感器 2、传感器 3、传感器 4、传感器 6、传感器 7、传感器 8、传感器 9、传感器 10 和传感器 11 这 10 个传感器，即 $n = 10$ 。本文将这些传感器在 2004.03.01-2004.03.07 这一周的温度值大约每 3.5 min 采一个点，得到了长度为 2048 的温度信号，即 $N = 2048$ 。本文选用 sym8 小波对温度信号进行稀疏表示。观测矩阵 Φ 的元素符合高斯分布(即 $\phi_{m,n} \sim N(0,1/N)$)。 Φ 的每一行经过了归一化处理。

图 4 给出了随机选取的 2 个传感器在 2004.03.01 这一天内的温度变化情况。从图 4 可以看出，它们所测的温度值几乎同时达到低谷和峰值，具有很强的空间相关性，而且它们的温度值是缓慢变化的，所以同时也具有很强的时间相关性。当利用小波对

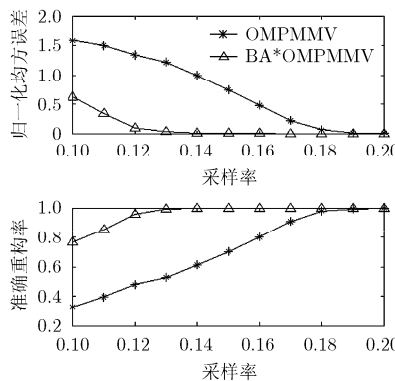


图 2 无噪声时 BA*OMPMMV 算法与 OMPMMV 算法性能对比

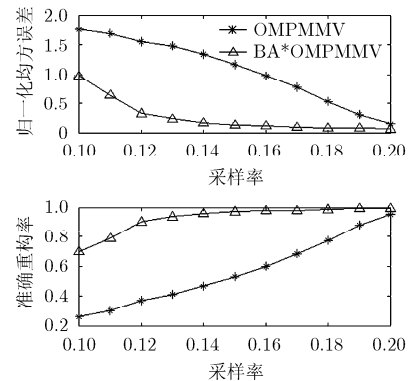


图 3 SNR=10 dB 时 BA*OMPMMV 算法与 OMPMMV 算法性能对比

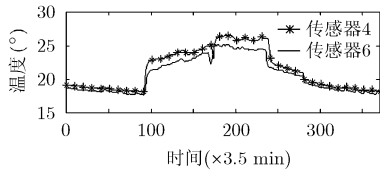


图4 2个传感器的温度变化情况

它们进行稀疏表示时,大部分小波系数的幅值分布在零附近,只有少量的幅值很大而且幅值大的系数呈现聚簇现象,即它们在小波变换下是块稀疏的。此外,幅度大的值所处的位置几乎是相同的,即它们共享支撑集。

图5(a),图5(b)给出了无噪声的情况下,当 γ 为0.12, 0.16时,分别采用BA*OMPMMV算法和OMPMMV算法对信号进行联合重构后传感器1的信号重构情况。

由图5可知,当 $\gamma=0.12$ 时,BA*OMPMMV算法已能较准确地重构信号,而OMPMMV算法却不能重构信号,当 $\gamma=0.16$ 时,前者达到了较好的重构效果,而与其相比后者的重构效果则较差。因此在没有噪声的情况下,BA*OMPMMV算法能够很好地重构原信号,并且其重构性能优于OMPMMV算法。

表3给出了SNR=10 dB时,不同采样率下分别采用BA*OMPMMV算法和OMPMMV算法对信号进行重构的归一化均方误差的对比情况。从表中可以看出,在有噪声的情况下,BA*OMPMMV算法的重构性能优于OMPMMV算法的重构性能。

表3 SNR=10 dB时两种算法的归一化均方误差的对比情况

γ	0.10	0.12	0.14	0.16	0.18	0.20
BA*OMPMMV	0.08	0.05	0.04	0.03	0.03	0.02
OMPMMV	1.84	1.62	1.29	0.66	0.08	0.02

5 结束语

本文将分布在一定范围内的多个传感器所测信号的联合重构问题看作一个MMV问题。针对此问题,本文首先在A*OMP算法的基础上,针对块稀疏信号提出了BA*OMP算法。该算法利用信号的块稀疏结构,在进行路径扩张时用原子块取代单个原子作为新的节点,大大降低算法的复杂度。在计算路径代价时用搜索树中所有路径的最大长度取代信号的稀疏度并将基于稀疏度的截止条件改为基于残差。该算法的重构性能优于A*OMP算法。当原子块的大小为1,重叠的原子数为0时,该算法退化为无需预先确定信号稀疏度的A*OMP算法。为了将BA*OMP算法用于解决MMV问题,本文以与残差矩阵的投影误差最小为原则来选择新的节点,得到了一种用于解决共享支撑集的多个信号的联合重构问题的BA*OMPMMV算法,该算法适用于联合重构具有块稀疏特性的多个信号,而且其性能优于OMPMMV算法。在实验部分,将BA*OMPMMV算法用于重构多传感器所测的温度信号,达到了较好的重构效果。

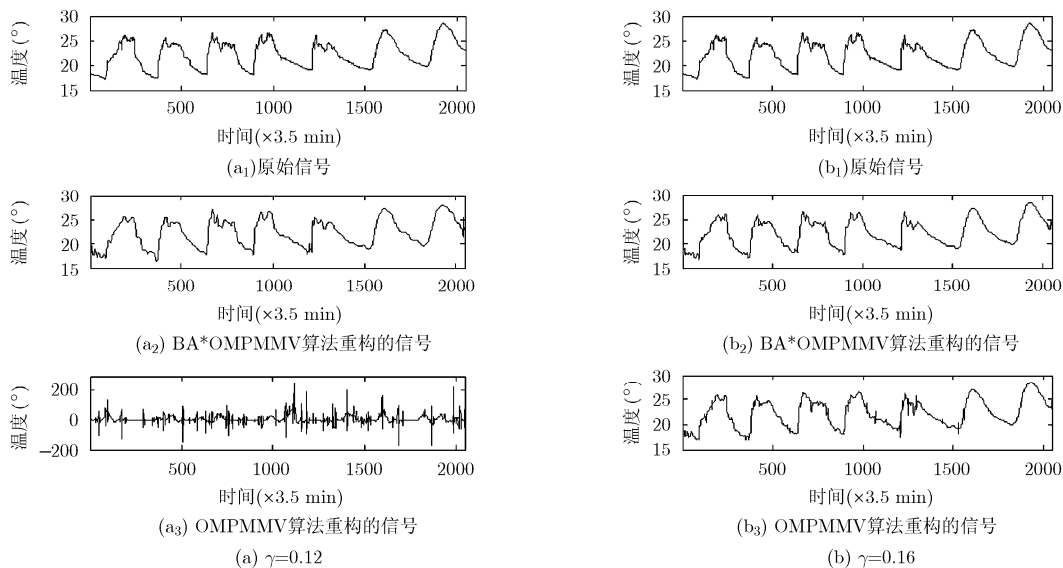


图5 两种算法的信号重构对比

参考文献

- [1] Donoho D L. Compressed sensing[J]. *IEEE Transactions on Information Theory*, 2006, 52(4): 1289-1306.
- [2] 唐亮,周正,石磊,等. 基于能量均衡的无线传感器网络压缩感知算法[J]. *电子与信息学报*, 2011, 33(8): 1919-1923.

- Tang Liang, Zhou Zheng, Shi Lei, *et al.* Energy balance based WSN compressive sensing algorithm[J]. *Journal of Electronics & Information Technology*, 2011, 33(8): 1919-1923.
- [3] Baron D, Duarte M F, Sarvotham S, *et al.* An information-theoretic approach to distributed compressed sensing[C]. Proceedings of the 43d Allerton Conference on Communication, Control, and Computing, Monticello, IL, 2005: 814-825.
- [4] Chen J and Huo X. Theoretical results on sparse representations of multiple measurement vectors[J]. *IEEE Transactions on Signal Processing*, 2006, 54(12): 4634-4643.
- [5] Davies M E and Eldar Y C. Rank awareness in joint sparse recovery[J]. *IEEE Transactions on Information Theory*, 2012, 58(2): 1135-1146.
- [6] Kim J M, Lee O K, and Ye J C. Compressive MUSIC: revisiting the link between compressive sensing and array signal processing[J]. *IEEE Transactions on Information Theory*, 2012, 58(1): 278-301.
- [7] Kim J M, Lee O K, and Ye J C. Noise robust joint sparse recovery using compressive subspace fitting[OL]. <http://arxiv.org/abs/1112.3446>, 2012.
- [8] Karahanoglu N B and Erdogan H. A* orthogonal matching pursuit: best-first search for compressed sensing signal recovery[J]. *Digital Signal Processing*, 2012, 22(4): 555-568.
- [9] Karahanoglu N B and Edogan H. Compressed sensing signal recovery via A*orthogonal matching pursuit[C]. 2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), Prague, 2011: 3732-3735.
- [10] Intel berkeley lab wsn. <http://db.csail.mit.edu/labdata/> labdata.html, 2004.
- [11] Ding J, Chen L, and Gu Y. Performance of orthogonal matching pursuit for multiple measurement vectors[OL]. <http://arxiv.org/abs/1109.6390>, 2011.
- [12] Baraniuk R G, Cevher V, Duarte M F, *et al.* Model based compressive sensing[J]. *IEEE Transactions on Information Theory*, 2010, 56(4): 1982-2001.
- [13] Chen S S, Donoho D L, and Saunders M A. Atomic decomposition by basis pursuit[J]. *SIAM Journal on Scientific Computing*, 1998, 20(1): 33-61.
- [14] Tropp J and Gilbert A C. Signal recovery from random measurements via orthogonal matching pursuit[J]. *IEEE Transactions on Information Theory*, 2007, 53(12): 4655-4666.
- [15] Dai W and Milenkovic O. Subspace pursuit for compressive sensing signal reconstruction[J]. *IEEE Transactions on Information Theory*, 2009, 55(5): 2230-2249.
- [16] Eldar Y C, Kuppinger P, and Bolcskei H. Compressed sensing of block-sparse signals: uncertainty relations and efficient recovery[J]. *IEEE Transactions on Signal Processing*, 2010, 58(6): 3042-3054.
- 练秋生：男，1969年生，教授，博士生导师，研究方向为压缩感知、稀疏表示。
- 刘芳：女，1986年生，硕士生，研究方向为分布式压缩感知。
- 陈书贞：女，1968年生，副教授，研究方向为压缩感知、图像处理。