Distributed Data Aggregation for Sparse Recovery in Wireless Sensor Networks

Shuangjiang Li, Hairong Qi

Advanced Imaging and Collaborative Information Processing (AICIP) Lab
University of Tennessee, Knoxville

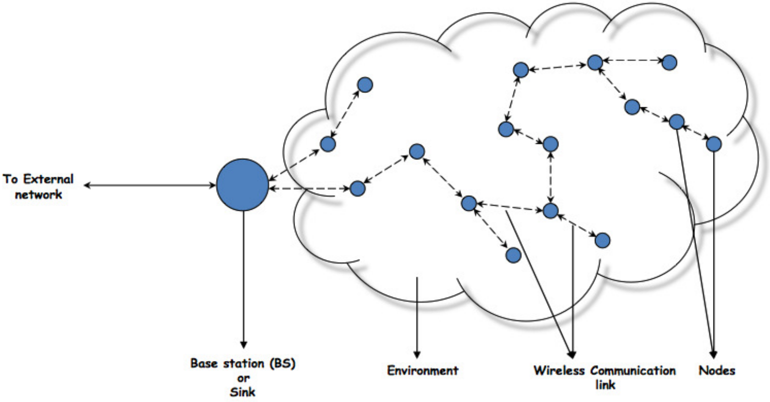
May 21, 2013

The 9th IEEE International Conference on Distributed Computing in Sensor Systems (DCOSS 2013)

Outline

- Background and motivation
- Prior works
- Our approach
- Experiments
- Conclusion

Data aggregation in WSNs



Data aquisiton in WSNs

Traditionally: sample then compress/aggregation

- i.e., average, mean of the data or transform the data in another domain (frequency, wavelet etc)
- to save energy and storage by sampling all the data first and then discarding most of them?

Data aquisiton in WSNs

Traditionally: sample then compress/aggregation

- i.e., average, mean of the data or transform the data in another domain (frequency, wavelet etc)
- to save energy and storage by sampling all the data first and then discarding most of them?
- How to aquire informative data efficiently?

Data aquisiton in WSNs

Compressed Sensing (CS)

- Direclty through $y = \Phi x$, ($y \in \mathbb{R}^m, x \in \mathbb{R}^n$ and $m \ll n$) we can recovery x with no/little informance loss.

Data aquisiton in WSNs

Compressed Sensing (CS)

- Direclty through $y = \Phi x$, ($y \in \mathbb{R}^m, x \in \mathbb{R}^n$ and $m \ll n$) we can recovery x with no/little informance loss.
  - Underdetermined system of linear equations which leads to infinite solutions

Data aquisiton in WSNs

Compressed Sensing (CS)

- Direclty through $y = \Phi x$, ($y \in \mathbb{R}^m, x \in \mathbb{R}^n$ and $m \ll n$) we can recovery x with no/little informance loss.
  - Underdetermined system of linear equations which leads to infinite solutions
  - What if we put conditions on x and Φ??

Data aquisiton in WSNs

Compressed Sensing (CS)

- Direclty through $y = \Phi x$, ($y \in \mathbb{R}^m, x \in \mathbb{R}^n$ and $m \ll n$) we can recovery x with no/little informance loss.
  - Underdetermined system of linear equations which leads to infinite solutions
  - What if we put conditions on x and Φ??
    x is sparse/compressible, Φ satisfies certain property, (RIP, Null space property)

Data aquisiton in WSNs

Compressed Sensing (CS)

- Direclty through $y = \Phi x$, ($y \in \mathbb{R}^m, x \in \mathbb{R}^n$ and $m \ll n$) we can recovery x with no/little informance loss.
  - Underdetermined system of linear equations which leads to infinite solutions
  - What if we put conditions on x and Φ??
    x is sparse/compressible, Φ satisfies certain property, (RIP, Null space property)
    unique solution!!

Data aquisiton in WSNs

Compressed Sensing (CS)

- Direclty through $y = \Phi x$, ($y \in \mathbb{R}^m, x \in \mathbb{R}^n$ and $m \ll n$) we can recovery x with no/little informance loss.
  - Underdetermined system of linear equations which leads to infinite solutions
  - What if we put conditions on x and Φ??
    x is sparse/compressible, Φ satisfies certain property, (RIP, Null space property)
    unique solution!!
- [Pros]: reduces the sample length

Data aquisiton in WSNs

Compressed Sensing (CS)

- Direclty through $y = \Phi x$, ($y \in \mathbb{R}^m, x \in \mathbb{R}^n$ and $m \ll n$) we can recovery x with no/little informance loss.
  - Underdetermined system of linear equations which leads to infinite solutions
  - What if we put conditions on x and Φ??
    x is sparse/compressible, Φ satisfies certain property, (RIP, Null space property)
    unique solution!!
- [Pros]: reduces the sample length
- [Cons]: introduces the dense measurement problem if Φ is dense. (i.e., a linear project would involve all the sensor readings

Data aquisiton in WSNs

Compressed Sensing (CS)

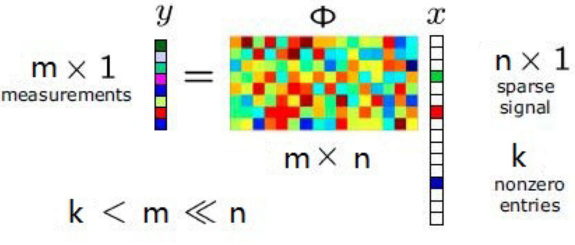


Figure : Compressed sensing [1]

[1] Image courtsey of Professor Richard Baraniuk at Rice University

Prior works

CS based data aggregation and routing in WSNs

Prior works

CS based data aggregation and routing in WSNs

- [Bajwa2007, Luo2009] single-hop comm. to sink, dense CS matrix

Prior works

CS based data aggregation and routing in WSNs

- [Bajwa2007, Luo2009] single-hop comm. to sink, dense CS matrix
- [Quer2009, Wang2011] routing based on measurement matrix, random routing, both require grid networks and not consider fairness of sensor selection

Prior works

CS based data aggregation and routing in WSNs

- [Bajwa2007, Luo2009] single-hop comm. to sink, dense CS matrix
- [Quer2009, Wang2011] routing based on measurement matrix, random routing, both require grid networks and not consider fairness of sensor selection
- [Lee2009] spatially-locailzed sparse projections as measurment matrix, not easy to generate

Prior works

CS based data aggregation and routing in WSNs

- [Bajwa2007, Luo2009] single-hop comm. to sink, dense CS matrix
- [Quer2009, Wang2011] routing based on measurement matrix, random routing, both require grid networks and not consider fairness of sensor selection
- [Lee2009] spatially-localized sparse projections as measurment matrix, not easy to generate

Our approach

- A sparse binary matrix based on unbalanced expander graph for two purposes

Our approach

- A sparse binary matrix based on unbalanced expander graph for two purposes
  - as the CS measurement matrix

Our approach

- A sparse binary matrix based on unbalanced expander graph for two purposes
  - as the CS measurement matrix
    - works as good as ”dense” CS matrices (Random Gaussian, Scrambled Fourier matrices)

Our approach

- A sparse binary matrix based on unbalanced expander graph for two purposes
  - as the CS measurement matrix
    - works as good as ”dense” CS matrices (Random Gaussian, Scrambled Fourier matrices)
    - reduces computational complexity and communication costs

Our approach

- A sparse binary matrix based on unbalanced expander graph for two purposes
  - as the CS measurement matrix
    - works as good as ”dense” CS matrices (Random Gaussian, Scrambled Fourier matrices)
    - reduces computational complexity and communication costs
  - as the sensor selection matrix
    - needs to be uniform selection or be fair

Our approach

- A sparse binary matrix based on unbalanced expander graph for two purposes
  - as the CS measurement matrix
    - works as good as "dense" CS matrices (Random Gaussian, Scrambled Fourier matrices)
    - reduces computational complexity and communication costs
  - as the sensor selection matrix
    - needs to be uniform selection or be fair
- A Distributed Compressive Sparse Sampling (DCSS) algorithm

Our approach

- A sparse binary matrix based on unbalanced expander graph for two purposes
  - as the CS measurement matrix
    - works as good as ”dense” CS matrices (Random Gaussian, Scrambled Fourier matrices)
    - reduces computational complexity and communication costs
  - as the sensor selection matrix
    - needs to be uniform selection or be fair
- A Distributed Compressive Sparse Sampling (DCSS) algorithm
  - randomly choose *m* designated sensors

Our approach

- A sparse binary matrix based on unbalanced expander graph for two purposes
  - as the CS measurement matrix
    - works as good as "dense" CS matrices (Random Gaussian, Scrambled Fourier matrices)
    - reduces computational complexity and communication costs
  - as the sensor selection matrix
    - needs to be uniform selection or be fair
- A Distributed Compressive Sparse Sampling (DCSS) algorithm
  - randomly choose m designated sensors
  - query only a necessary number of measurements (i.e., $\mathcal{O}(k\log(n))$ is enough for guaranteed CS data recovery)

Our approach: measurement matrix design

- Instead of using traditional random CS measurement matrices. We use sparse graph codes (i.e., expander graphs) as CS measurement matrix

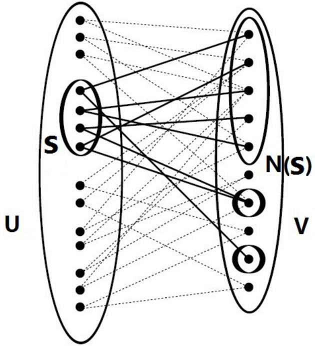
Our approach: measurement matrix design

- Instead of using traditional random CS measurement matrices. We use sparse graph codes (i.e., expander graphs) as CS measurement matrix
- [Berinde2008, Theorem 1,2,3] studied the relationship between the expander graph and CS measurement matrices, which serves as the theoretic foundation of our approach.

Our approach: measurement matrix design

Expander graph

- Let $X \subset U$, $N(X)$ be set of neighbors of X in V
- $G(U, V, E)$ is called (k, ϵ)-expander if
$$\forall X \subset U,\ |X| \leq k \Rightarrow |N(X)| \geq (1-\epsilon)d|X|$$
- Each set of nodes on the left expands to $N(X)$ number of nodes on right



Our approach: measurement matrix design

Sparse binary matrix from expander graph

A sparse binary matrix of m rows and n columns is generated in the following way:

Our approach: measurement matrix design

Sparse binary matrix from expander graph

A sparse binary matrix of m rows and n columns is generated in the following way:

- Step 1: For each column, randomly generate τ integers whose values are between 1 and m and place 1's in those rows indexed by the τ numbers;

Our approach: measurement matrix design

Sparse binary matrix from expander graph

A sparse binary matrix of *m* rows and *n* columns is generated in the following way:

- Step 1: For each column, randomly generate τ integers whose values are between 1 and *m* and place 1's in those rows indexed by the τ numbers;
- Step 2: If the τ numbers in one column are not distinct, repeat Step 1 until they are (this is not really an issue when $\tau \ll m$).

Our approach: measurement matrix design

Sparse binary matrix from expander graph

A sparse binary matrix of *m* rows and *n* columns is generated in the following way:

- Step 1: For each column, randomly generate τ integers whose values are between 1 and *m* and place 1's in those rows indexed by the τ numbers;
- Step 2: If the τ numbers in one column are not distinct, repeat Step 1 until they are (this is not really an issue when $\tau \ll m$).

τ is the degree of the expander graph, $\tau = 8$ in our experiment

Our approach: measurement matrix design

Sparse binary matrix for sensor selection

For each row of the sparse binary matrix $\Phi_i \in \mathbb{R}^n$ $(1 \leq i \leq m)$

1	0	0	0	1	$\cdots$	0	1	0

can be seen as an *n*-dimensional row vector (binary indicator function) for sensor subset selection (i.e., select sensor j to be active for sensing when $\Phi_{ij} = 1$, and inactive otherwise.)

Our approach: measurement matrix design

Fairness of sensor selection

- Each sensor will be selected equally τ times for sensing based on the design of the matrix.

Our approach: measurement matrix design

Fairness of sensor selection

- Each sensor will be selected equally τ times for sensing based on the design of the matrix.
- *m* designated sensors and routing sensors will cause unbalanced energy consumption, however

Our approach: measurement matrix design

Fairness of sensor selection

- Each sensor will be selected equally τ times for sensing based on the design of the matrix.
- *m* designated sensors and routing sensors will cause unbalanced energy consumption, however
  - they are chosen each time when you want to sense the whole environment (i.e., selected very infrequently)

Our approach: measurement matrix design

Fairness of sensor selection

- Each sensor will be selected equally τ times for sensing based on the design of the matrix.
- *m* designated sensors and routing sensors will cause unbalanced energy consumption, however
  - they are chosen each time when you want to sense the whole environment (i.e., selected very infrequently)
  - since $m \ll n$, each time only a random small amount of sensors will be selected.

Our approach: measurement matrix design

Fairness of sensor selection

- Each sensor will be selected equally τ times for sensing based on the design of the matrix.
- m designated sensors and routing sensors will cause unbalanced energy consumption, however
  - they are chosen each time when you want to sense the whole environment (i.e., selected very infrequently)
  - since $m \ll n$, each time only a random small amount of sensors will be selected.

Our approach: measurement matrix design

Fairness of sensor selection

- Each sensor will be selected equally τ times for sensing based on the design of the matrix.
- m designated sensors and routing sensors will cause unbalanced energy consumption, however
  - they are chosen each time when you want to sense the whole environment (i.e., selected very infrequently)
  - since $m \ll n$, each time only a random small amount of sensors will be selected.
- In the long run, the energy consumption can still be balanced

Our approach: DCSS algorithm

Network model

- Consider a wireless network of n sensors with diameter d hops, each measures a real data value x_i ($i = 1, 2, \cdots, n$), which is sparse or compressible under some transformation domains
- $y = \Phi x$ (Φ is sparse binary matrix, i.e., $\phi_{ij} \in \{0, 1\}$)
- Randomly choose m designated sensors from n sensors ($m \ll n$)

Our approach: DCSS algorithm

- Input:
  - $\Phi \in \mathbb{R}^{m \times n}$,
  - sensor neighborhood information,
  - environmental reading vector $x \in \mathbb{R}^n$.

Our approach: DCSS algorithm

- Input:
  - $\Phi \in \mathbb{R}^{m \times n}$,
  - sensor neighborhood information,
  - environmental reading vector $x \in \mathbb{R}^n$.
- Output: recovered vector $x^* \in \mathbb{R}^n$.

Our approach: DCSS algorithm

- Input:
  - $\Phi \in \mathbb{R}^{m \times n}$,
  - sensor neighborhood information,
  - environmental reading vector $x \in \mathbb{R}^n$.
- Output: recovered vector $x^* \in \mathbb{R}^n$.
- Steps:
  - The Fusion Center(FC) generates the sparse binary matrix Φ and broadcasts the matrix and neighborhood information,

Our approach: DCSS algorithm

- Input:
  - $\Phi \in \mathbb{R}^{m \times n}$,
  - sensor neighborhood information,
  - environmental reading vector $x \in \mathbb{R}^n$.
- Output: recovered vector $x^* \in \mathbb{R}^n$.
- Steps:
  - The Fusion Center(FC) generates the sparse binary matrix Φ and broadcasts the matrix and neighborhood information,
  - For each sensor s_j, $(1 \leq j << \leq n)$, sends its reading x_j to the designated sensor D_i through shortest path if $\Phi_{ij} \neq 0$,

Our approach: DCSS algorithm

- Input:
  - $\Phi \in \mathbb{R}^{m \times n}$,
  - sensor neighborhood information,
  - environmental reading vector $x \in \mathbb{R}^n$.
- Output: recovered vector $x^* \in \mathbb{R}^n$.
- Steps:
  - The Fusion Center(FC) generates the sparse binary matrix Φ and broadcasts the matrix and neighborhood information,
  - For each sensor s_j, $(1 \leq j <\leq n)$, sends its reading x_j to the designated sensor D_i through shortest path if $\Phi_{ij} \neq 0$,
  - For the m designated sensors $D_1, \cdots, D_m$, each computes and stores the sum of the reading it receives,

Our approach: DCSS algorithm

- Input:
  - $\Phi \in \mathbb{R}^{m \times n}$,
  - sensor neighborhood information,
  - environmental reading vector $x \in \mathbb{R}^n$.
- Output: recovered vector $x^* \in \mathbb{R}^n$.
- Steps:
  - The Fusion Center(FC) generates the sparse binary matrix Φ and broadcasts the matrix and neighborhood information,
  - For each sensor s_j, $(1 \leq j <\leq n)$, sends its reading x_j to the designated sensor D_i through shortest path if $\Phi_{ij} \neq 0$,
  - For the m designated sensors $D_1, \cdots, D_m$, each computes and stores the sum of the reading it receives,
  - The m designated sensors send their results to the FC and FC performs CS recovery for x^*.

Our approach: DCSS algorithm

$$\begin{pmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1n} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ \phi_{m1} & \phi_{m2} & \cdots & \phi_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \cdots \\ y_m \end{pmatrix}$$

Our approach: DCSS algorithm

$$\begin{pmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1n} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ \phi_{m1} & \phi_{m2} & \cdots & \phi_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \cdots \\ y_m \end{pmatrix}$$

For each sensor:

Our approach: DCSS algorithm

$$\begin{pmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1n} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ \phi_{m1} & \phi_{m2} & \cdots & \phi_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \cdots \\ y_m \end{pmatrix}$$

For each sensor:

- $\phi_{ij} \cdot x_j$ means sensor id s_j sends its readings x_j to the designated sensor D_i, if $\phi_{ij} \neq 0$

Our approach: DCSS algorithm

$$\begin{pmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1n} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ \phi_{m1} & \phi_{m2} & \cdots & \phi_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \cdots \\ y_m \end{pmatrix}$$

For each sensor:

- $\phi_{ij} \cdot x_j$ means sensor id s_j sends its readings x_j to the designated sensor D_i, if $\phi_{ij} \neq 0$

For m designated sensors:

Our approach: DCSS algorithm

$$\begin{pmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1n} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ \phi_{m1} & \phi_{m2} & \cdots & \phi_{mn} \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \cdots \\ x_n \end{pmatrix} = \begin{pmatrix} y_1 \\ y_2 \\ \cdots \\ y_m \end{pmatrix}$$

For each sensor:

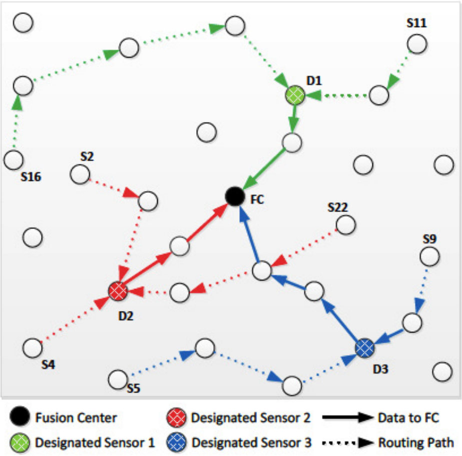
- $\phi_{ij} \cdot x_j$ means sensor id s_j sends its readings x_j to the designated sensor D_i, if $\phi_{ij} \neq 0$

For m designated sensors:

- Designated sensor D_i computes and stores the summation of the sensor reading it receives, for all $1 \leq i \leq m$
- Report $y_1, \cdots, y_m$ as observations $y \in \mathbb{R}^m$

Our approach: DCSS algorithm

An example



Our approach: DCSS algorithm

Communication cost

- Based on avergate bit-hop cost per reading
- Assume ω to be the average row weight of the sparse binary measurement matrix, $\tau\omega$ is the cost of gathering the sensor readings for each projection

Our approach: DCSS algorithm

Communication cost

- Based on avergate bit-hop cost per reading
- Assume ω to be the average row weight of the sparse binary measurement matrix, $\tau\omega$ is the cost of gathering the sensor readings for each projection
- Assume d as the cost to send the projection to FC
- For generation of $\mathcal{O}(k \log(n))$ projection for data recovery, the total communication cost is: $\mathcal{O}(k(\tau\omega + d) \log(n))$

Our approach: DCSS algorithm

Communication cost

TABLE I: Communication cost of different CS algorithms.

Algorithm	Cost
DS	$\mathcal{O}(kn \log n)$
SRP [9]	$\mathcal{O}(kd \log^2 n)$
CDS(RW) [4]	$\mathcal{O}(k(t+d) \log n)$
Sparse Binary	$\mathcal{O}(k(\tau\omega + d) \log n)$

- DS: dense sampling
- SRP: sparse random projection in [Wang2007IPSN]
- CDS (RW): compressive distributed sensing using random walk in [Sartipi2011DCC]

Experiments

Setup

- Data aggregation schemes comparsion
  - Dense sampling matrices (Random Gaussian, Scrambled Fourier measausrement matrices)
  - Sparse random projection matrices in [Wang2007IPSN] with various sparse level s
- Evaluation metrics

$$\varepsilon = \frac{\|\mathbf{x} - \mathbf{x}^*\|_2^2}{\|\mathbf{x}^*\|_2^2}$$

where $\mathbf{x}$ is the value of the original signal, while $\mathbf{x}^*$ is the reconstructed signal.

- Using ℓ_1-magic package for CS recovery

Experiments

Exact sparse signal recovery

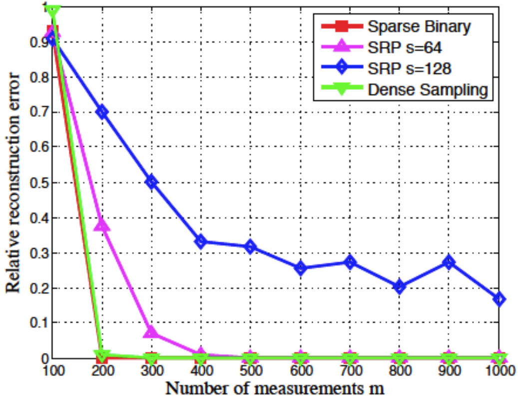


Fig. 5: Recovery result of an $n = 1024$, sparsity $k = 30$ sparse signal $\mathbf{x}$, with an average of 100 experiments using LP recovery method.

 Shuangjiang Li, Hairong Qi, AICIP Lab "Distributed Data Aggregation for Sparse Recovery in Wireless Sensor Networks." DCOSS 2013

Experiments

Noisy sparse signals recovery

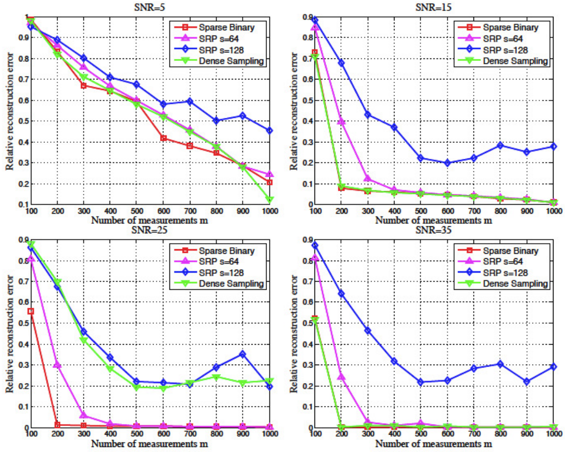


Fig. 6: Noisy recovery result of an $n = 1024$, sparsity $k = 30$ sparse signal $\mathbf{x}$, with different SNRs $(5, 15, 25,$ and $35)$ and an average of 100 experiments using LP recovery method evaluated by different measurements.

Experiments

Compressible signals recovery

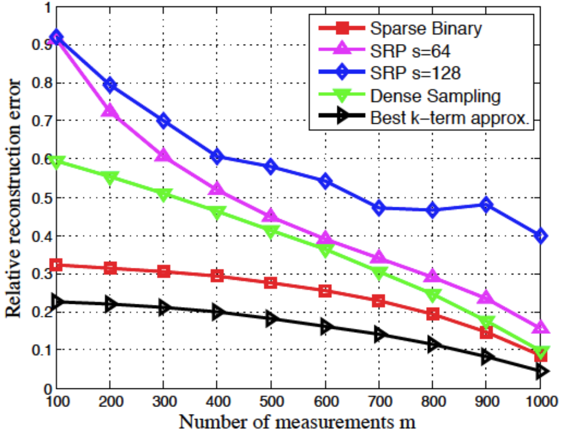


Fig. 7: Recovery result of a sampled compressible signal $x = 4n^{-\frac{7}{10}}$, with an average of 100 experiments using LP recovery method evaluated by different measurements.

Experiments

Real signal: Intel lab data (light intensity at node 19)

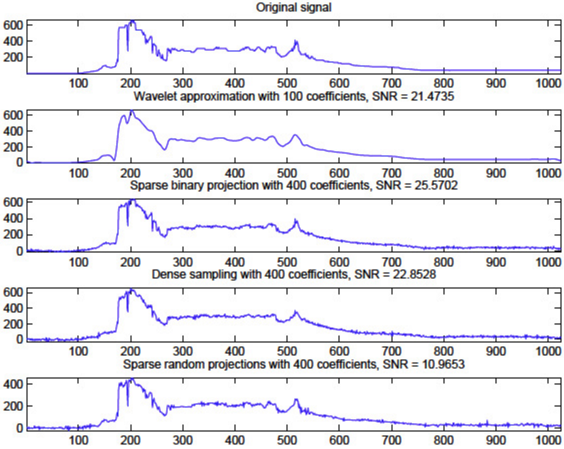


Fig. 8: Recovery result of real Intel lab signal using 100 wavelet coefficients and 400 CS measurements with different measurement matrices.

Experiments

Real signal: Intel lab data (light intensity at node 19)

TABLE II: Recovered SNR of different measurement matrices.

Methods	Wavelet approx.	Sparse binary	Dense sampling	SRP ($s = 64$)
SNR	21.4735	25.5702	22.8528	10.9653

Real signal: intel lab data (light intensity at node 19)

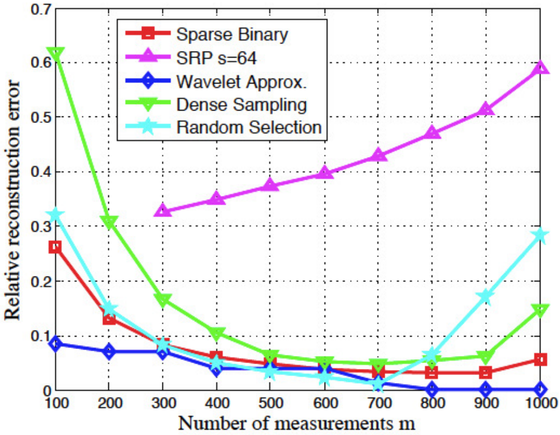


Fig. 9: Recovery result of real Intel lab signal, with an average of 100 experiments using LP recovery method evaluated by different measurements.

Conclusions

- A sparse binary measurement matrix was designed based on expaned graph.
  - Can be used for CS measurment matrix and sensor subset selection.
  - The recovery result is as good as traditional random dense CS measurement matrix and worked the best on compressible data.
  - Resolved the dense measurement problem.

Conclusions

- A sparse binary measurement matrix was designed based on expaned graph.
  - Can be used for CS measurment matrix and sensor subset selection.
  - The recovery result is as good as traditional random dense CS measurement matrix and worked the best on compressible data.
  - Resolved the dense measurement problem.
- A structure free data aggregation algorithm (DCSS) was proposed. Results from both synthetic and real data experiments demonstrated the usefulness of the algorithms.

Key References

- [Berinde2008] R. Berinde, A. Gilbert, P. Indyk, H. Karloff, and M. Strauss, "*Combining geometry and combinatorics: A unified approach to sparse signal recovery*," in Communication, Control, and Computing, 2008 46th Annual Allerton Conference on. IEEE, 2008, pp. 798–805.
- [Wang2007IPSN] W. Wang, M. Garofalakis, and K. Ramchandran, "*Distributed sparse random projections for refinable approximation*," in Proc. Int. Conf. on Info. Processing in Sensor Networks, Cambridge, MA, Apr. 2007.
- [Sartipi2011DCC] M. Sartipi and R. Fletcher, "*Energy-efficient data acquisition in wireless sensor networks using compressed sensing*," in Data Compression Conference (DCC), 2011. IEEE, 2011, pp. 223–232.

Key References

- [Luo2009] C. Luo, F. Wu, J. Sun, and C. Chen,"*Compressive data gathering for large-scale wireless sensor networks*," in Proceedings of the 15th annual international conference on Mobile computing and networking. ACM, 2009, pp. 145-156.
- [Wang2011] X. Wang, Z. Zhao, Y. Xia, and H. Zhang, "*Compressed sensing for efficient random routing in multi-hop wireless sensor networks*," International Journal of Communication Networks and Distributed Systems, vol. 7, no. 3, pp. 275–292, 2011.
- [Lee2009] S. Lee, S. Pattem, M. Sathiamoorthy, B. Krishnamachari, and A. Ortega, "*Spatially-localized compressed sensing and routing in multi-hop sensor networks*," Geosensor Networks, pp. 11–20, 2009.

Key References

- [Quer2009] G. Quer, R. Masiero, D. Munaretto, M. Rossi, J. Widmer, and M. Zorzi, "*On the interplay between routing and signal representation for compressive sensing in wireless sensor networks*," in Information Theory and Applications Workshop, 2009. IEEE, 2009, pp. 206–215.
- [Bajwa2007] W. Bajwa, J. Haupt, A. Sayeed, and R. Nowak, "*Joint source–channel communication for distributed estimation in sensor networks*," Information Theory, IEEE Transactions on, vol. 53, no. 10, pp. 3629–3653, 2007.

Thank you!!