Matrix and Tensor Completion for Large-Scale Seismic Interpolation: A Comparative Study

Okan Akalin, Curt Da Silva, Rajiv Kumar
Ben Recht, Felix Herrmann

WISCONSIN
UNIVERSITY OF WISCONSIN–MADISON

UBC THE UNIVERSITY OF BRITISH COLUMBIA

Quick Summary

- Problem: Large Scale Seismic Data Interpolation
- Approach: Matrix completion and tensor completion on different representations of seismic data
- Contribution: Posing the interpolation problem in the compressed sensing framework that allows scalable algorithms
- Outcome: Large scale interpolation problems can be solved efficiently by simple scalable algorithms

Outline

- Introduction
- Approach
- Experiments
- Conclusion & Future Work

Seismic Data Interpolation Problem

- Data is poorly sampled along a subset of modes
- Different from classical interpolation due to the nature of seismic data
  - Incomplete
  - Large volume
  - High dimensional
- Hence we need a space and time efficient algorithm for feasible analysis

Problem Setting

- 5-D data. Modes are time, source and receiver coordinates.
- Fourier transform is taken in time domain and a certain frequency slice is selected.
  - 4.68Hz downsampled to 68x68x101x101
  - 7.34Hz downsampled to 68x68x101x101
  - 12.3Hz downsampled to 68x68x201x201

Structured Signal Recovery

- Fundamentally different from Shannon-Nyquist based approaches:
  - **Shannon-Nyquist based methods:** periodic sampling with costly sampling rate
  - **Structured recovery methods:** impose stronger structural requirements with milder sampling rates

CS-Based Recovery

- Every successful compressed sensing recovery scheme consists of three main components:
  - **Signal structure - sparsity:** a sparse signal in some dictionary
    - *Seismic images tend to be sparse in curvelet domain*
  - **Structure-destroying sampling operator:** a sampling operator that breaks the assumed structure of the signal
  - **Structure-promoting optimization program:** a formulation which favors the sparsest signal that fits our data

Outline

- Introduction
- Approach
  - Matrix Completion
  - Tensor Completion
  - Low Rank Promoting Organization
  - Structure Promoting Optimization Program
- Experiments
- Conclusion & Future Work

Matrix Completion

- Matrix completion is the problem of filling the missing entries based on the observations
  - Ill posed unless we assume structure: low-rank!
- The CS framework can be easily extended for the matrix completion problem.

Matrix Completion

- The original matrix completion problem

$$\begin{aligned} &\text{minimize} && \text{rank}(\mathbf{X}) \\ &\text{subject to} && X_{ij} = M_{ij} \quad (i,j) \in \Omega \\ & && \mathbf{X} \in \mathbb{R}^{n \times n}, \end{aligned}$$

- Its convex relation that is tractable

$$\begin{aligned} &\text{minimize} && \|\mathbf{X}\|_{\star} \\ &\text{subject to} && X_{ij} = M_{ij} \quad (i,j) \in \Omega. \end{aligned}$$

$$\|X\|_* = \sum_i \sigma_i(X)$$

Matrix Completion

- Three pillars:
  - **Signal Structure - Low Rank:** For a matrix, a direct analogue of sparsity in the signal domain is sparsity in singular value spectrum
  - **Structure-destroying sampling operator:** Random sampling of entries!
  - **Structure-promoting optimization program:**

$$\begin{aligned} &\min \; \|\mathbf{X}\|_* \\ &\text{s.t.} \;\; \|\mathcal{A}(\mathbf{X}) - \mathbf{B}\|_2 \leq \sigma \end{aligned}$$

Matrix Completion

- Three different formulations:
  - Basis Pursuit Denoising (BPDN) formulation

$$\begin{aligned} &\min \ \|\mathbf{X}\|_* \\ &\text{s.t.} \ \ \|\mathcal{A}(\mathbf{X}) - \mathbf{B}\|_2 \leq \sigma \end{aligned}$$

  - Quadratic Programming (QP) Formulation

$$\min_{\mathbf{X}} \frac{1}{2}\|\mathcal{A}(\mathbf{X}) - \mathbf{B}\|_2^2 + \lambda\|\mathbf{X}\|_*$$

  - LASSO Formulation

$$\begin{aligned} &\min_{X} \ \|\mathcal{A}(\mathbf{X}) - \mathbf{B}\|_2^2 \\ &\text{s.t.} \ \|\mathbf{X}\|_* \leq \tau \end{aligned}$$

Matrix Completion

- Basis Pursuit Denoising (BPDN) formulation

$$\begin{aligned} &\min \ \|\mathbf{X}\|_* \\ &\text{s.t.} \ \|\mathcal{A}(\mathbf{X}) - \mathbf{B}\|_2 \leq \sigma \end{aligned}$$

  - The parameter **σ** can be naturally interpreted as the noise level
  - Challenging to solve

Matrix Completion

- (Quadratic Programming) QP Formulation

$$\min_{\mathbf{X}} \frac{1}{2}\|\mathcal{A}(\mathbf{X}) - \mathbf{B}\|_2^2 + \lambda\|\mathbf{X}\|_*$$

  - λ parameter does not have a natural interpretation, hard to pick in the noisy case
  - Time and space efficient scalable algorithms exist to solve the QP formulation

Matrix Completion

- LASSO Formulation

$$\min_{X} \|\mathcal{A}(\mathbf{X}) - \mathbf{B}\|_2^2$$
$$\text{s.t. } \|\mathbf{X}\|_* \leq \tau$$

  - Needs an estimate on the rank of the matrix, much harder than the estimation of the noise level
  - Can be solved efficiently

Outline

- Introduction
- Approach
    - Matrix Completion
    - Tensor Completion
    - Low Rank Promoting Organization
    - Structure Promoting Optimization Program
- Experiments
- Conclusion & Future Work

Tensor Completion

- The low rank idea can also be applied in the tensor completion setting.
- However the problem is much harder:
  - Even tensor rank is very hard to compute (NP-Complete!)
  - Fixed rank tensors do not form a closed set (a lowest rank approximation may not exist at all!)
  - A lack of theoretical framework for the completion problem

Tensor Completion

- An alternative (and popular) method is penalizing the rank of matricizations of the tensor

$$\min_{\mathbf{D}} \sum_{i=1}^{4} \|\mathbf{D}^{(i)}\|_*$$
$$\text{s.t. } \|\mathcal{A}(\mathbf{D}) - b\|_2 \leq \sigma$$

- Since this formulation uses nuclear norm, the problem is tractable
- It is unclear why this combination of nuclear norms is supposed to work

Tensor Completion

- Actually when the sampling operator is Gaussian, the sample complexity can be bounded below by the greatest sample complexity amongst all matricizations
  - It is impossible to outperform the single most successful matricization
- Gaussian sampling does not fit in the tensor completion framework, however empirical evidence suggests a similar argument might still hold.

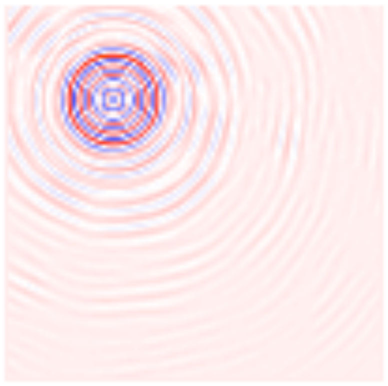
Oymak, S., Jalali, A., Fazel, M., Eldar, Y. C., & Hassibi, B. (2012). Simultaneously structured models with application to sparse and low-rank matrices. arXiv preprint arXiv:1212.3753.

Outline

- Introduction
- Approach
  - Matrix Completion
  - Tensor Completion
  - Low Rank Promoting Organization
  - Structure Promoting Optimization Program
- Experiments
- Conclusion & Future Work

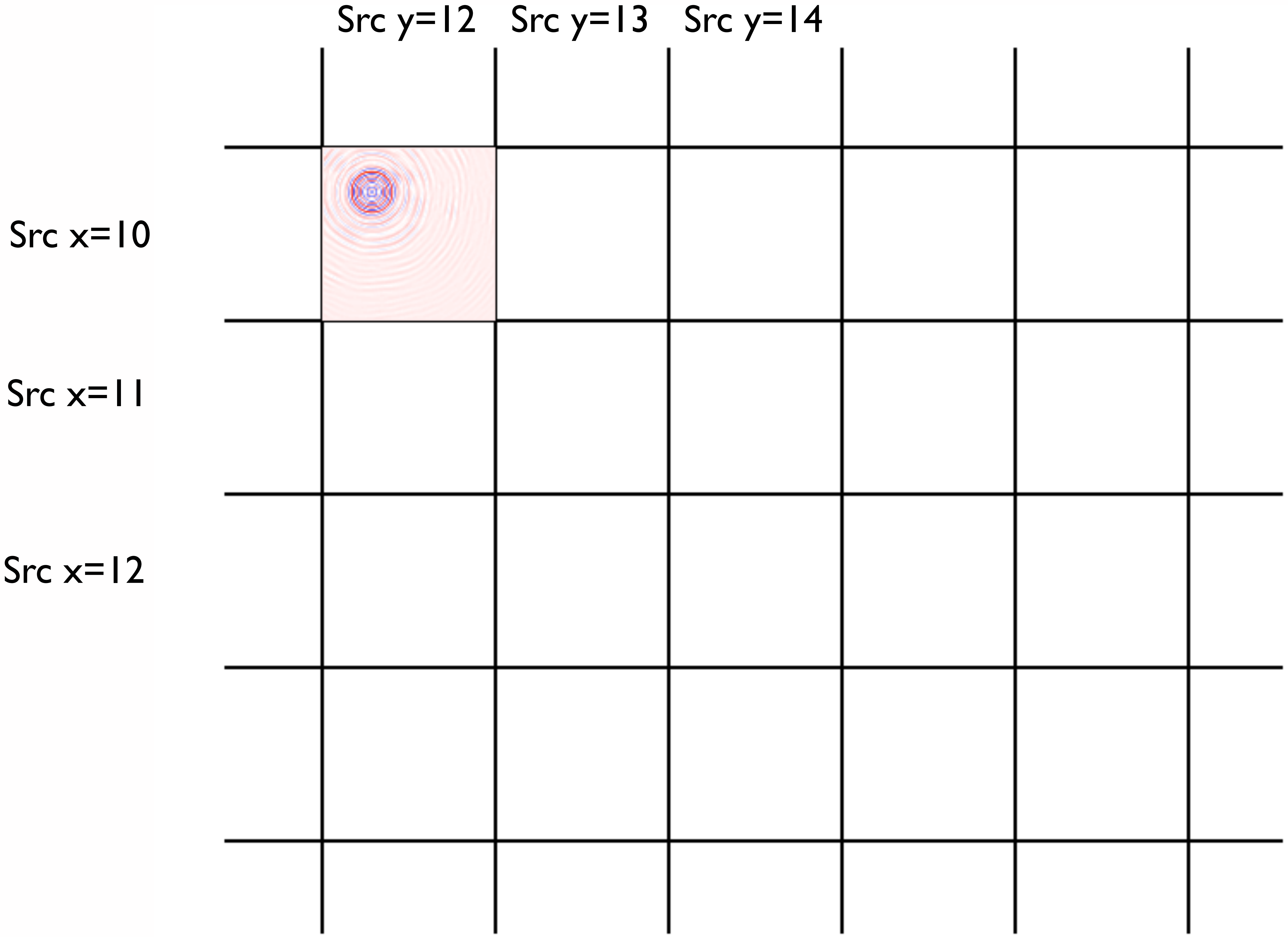
Low Rank Promoting Organization

(src x, src y)=(10,12)

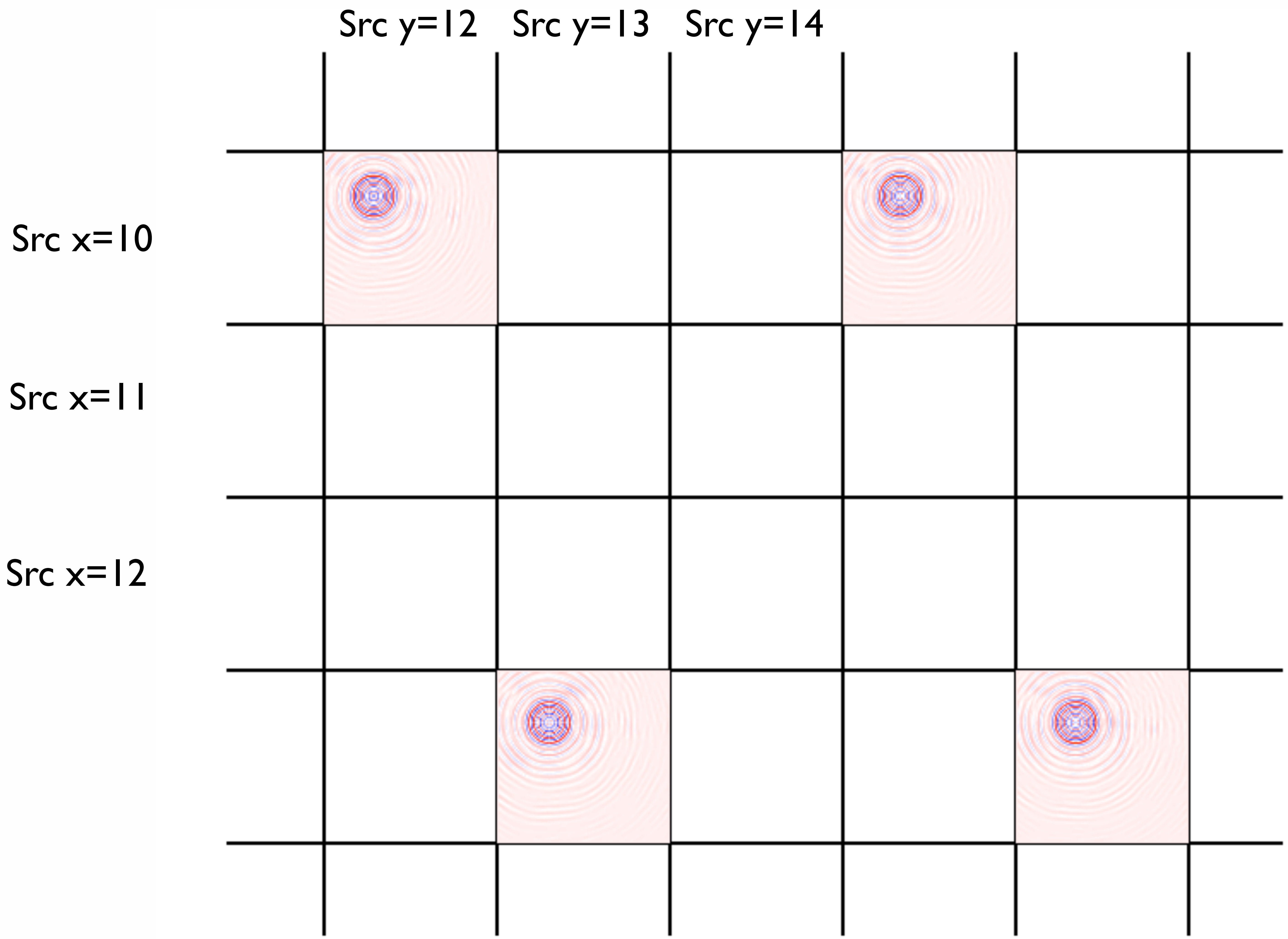


Fixing source coordinates, we obtain a specific shot

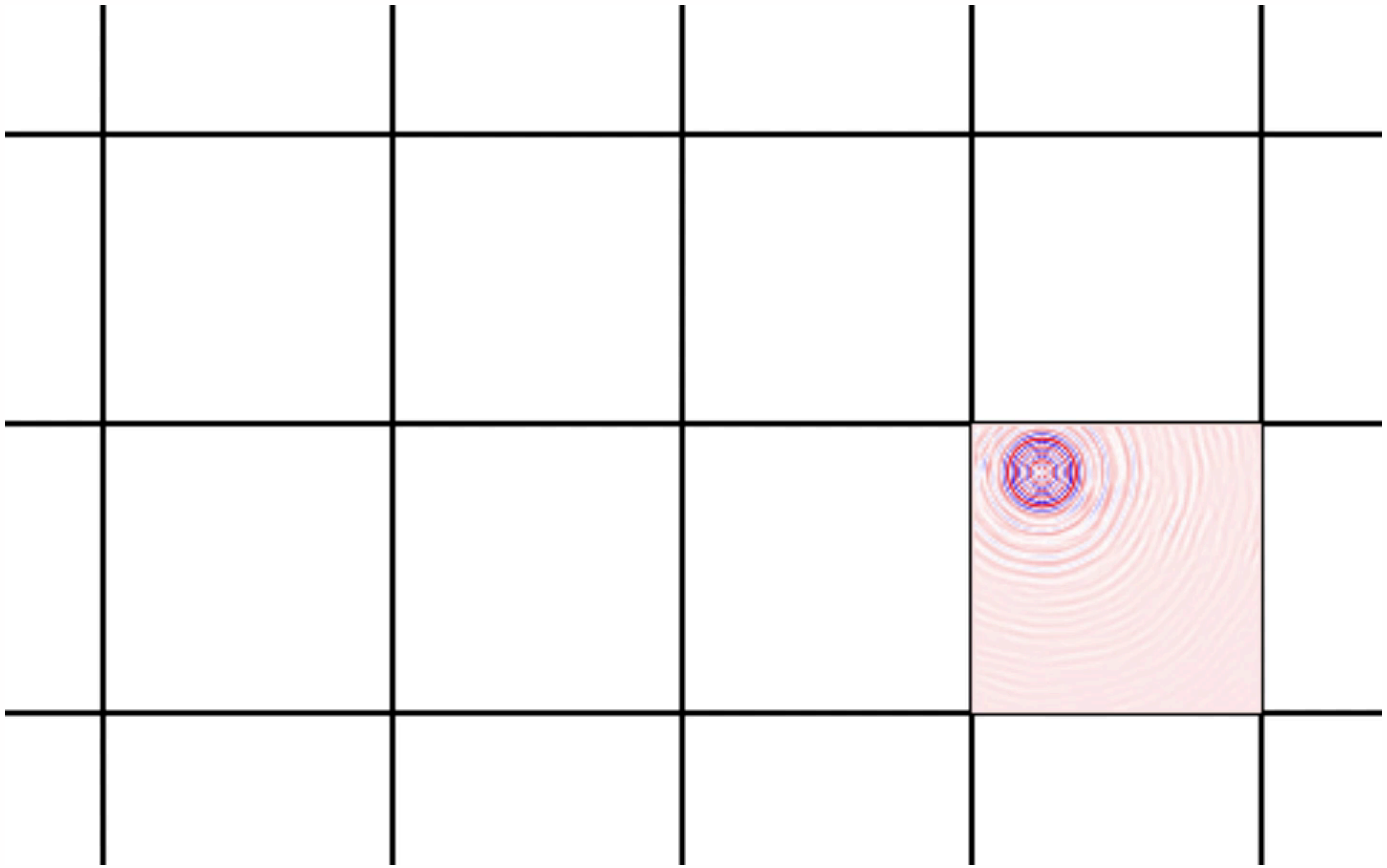
Low Rank Promoting Organization



Low Rank Promoting Organization



Low Rank Promoting Organization



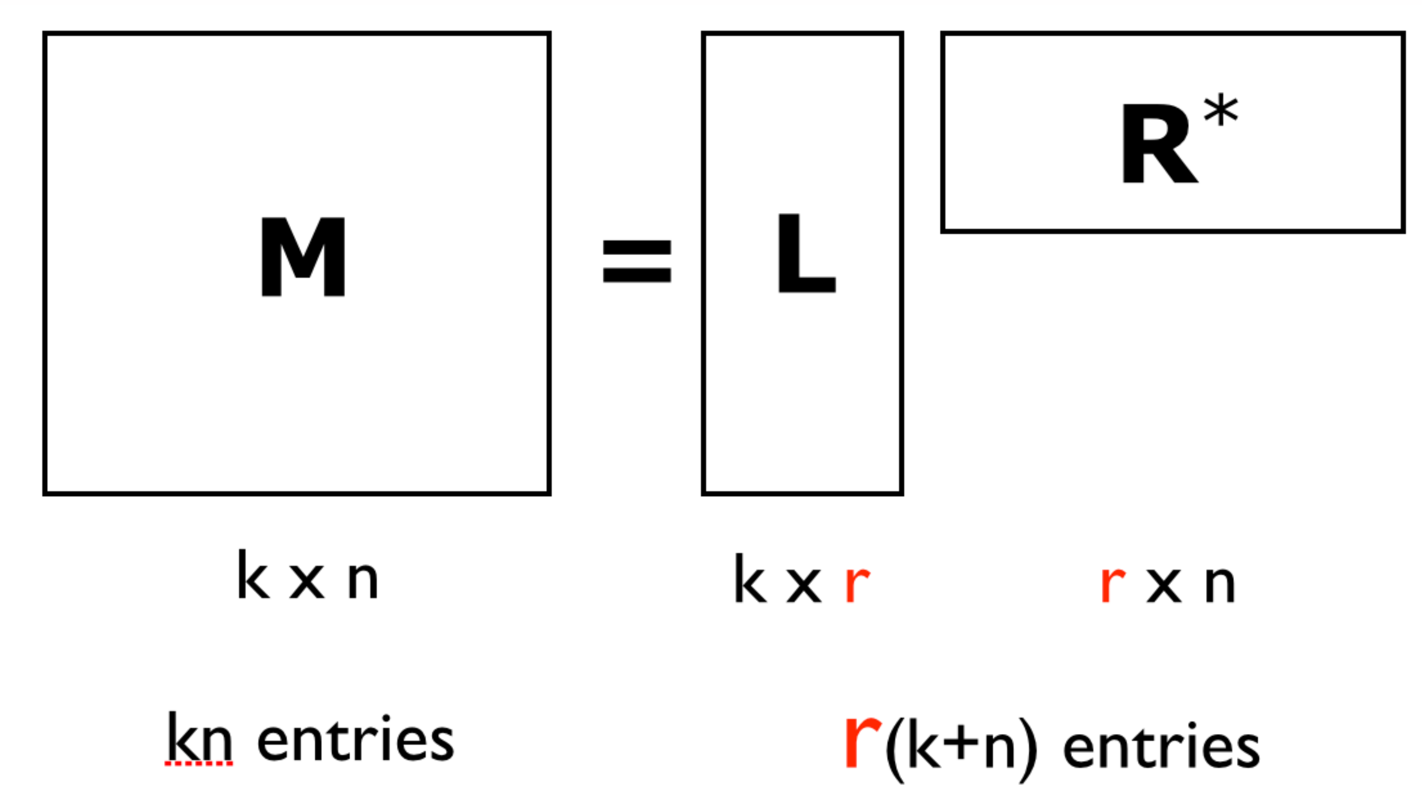
Outline

- Introduction
- Approach
  - Matrix Completion
  - Tensor Completion
  - Low Rank Promoting Organization
  - Structure Promoting Optimization Program
- Experiments
- Conclusion & Future Work

Jellyfish

- Jellyfish solves a relaxation of the QP program which is equivalent to the original program:

$$\text{minimize}_{(\mathbf{L},\mathbf{R})} \sum_{(u,v)\in E} \{(\mathbf{L}_u \mathbf{R}_v^T - M_{uv})^2 + \mu_u \|\mathbf{L}_u\|_F^2 + \mu_v \|\mathbf{R}_v\|_F^2\}$$



Recht, B., & Ré, C. (2011). Parallel stochastic gradient algorithms for large-scale matrix completion. Mathematical Programming Computation, 1-26.

Jellyfish

- This formulation allows application of stochastic gradient descent algorithm which can be highly parallelized by proper sampling of data points
    - Can scale to GB sized matrices on workstations
- Jellyfish explicitly compresses the matrix by factorizing it

SPGL1

- Successively solves LASSO problems, updating **τ** at each iteration:

$$\tau^{k+1} = \tau^k - \frac{v(\tau) - \eta}{v'(\tau)}$$

- where

$$v(\tau) = \min_x \rho(\mathcal{A}(x) - b) \quad \text{s.t.} \;\; \|x\| \leq \tau$$

Friedlander, M., & Van den Berg, E. (2008). SPGL1, a solver for large scale sparse reconstruction. SIAM Journal on Scientific Computing, 31(2), 890-912.
Aravkin, A. Y., Kumar, R., Mansour, H., & Recht, B. (2013). A robust SVD-free approach to matrix completion, with applications to interpolation of large scale data.

Tensor Completion

- For tensor completion, the objective function is:

$$J(\boldsymbol{\mathcal{D}}, \boldsymbol{\mathcal{Y}}_i, \boldsymbol{\mathcal{W}}_i) = \frac{\lambda}{2}\|\mathcal{T}\boldsymbol{\mathcal{D}} - \boldsymbol{d}^{obs}\|_F^2 + \sum_{i=1}^{4}\left(\|\mathbf{Y}_i^{(i)}\|_* - <\boldsymbol{\mathcal{W}}_i, \boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{Y}}_i> + \frac{\beta}{2}\|\boldsymbol{\mathcal{D}} - \boldsymbol{\mathcal{Y}}_i\|_F^2\right)$$

- This is the ADMM reformulation of the original problem.

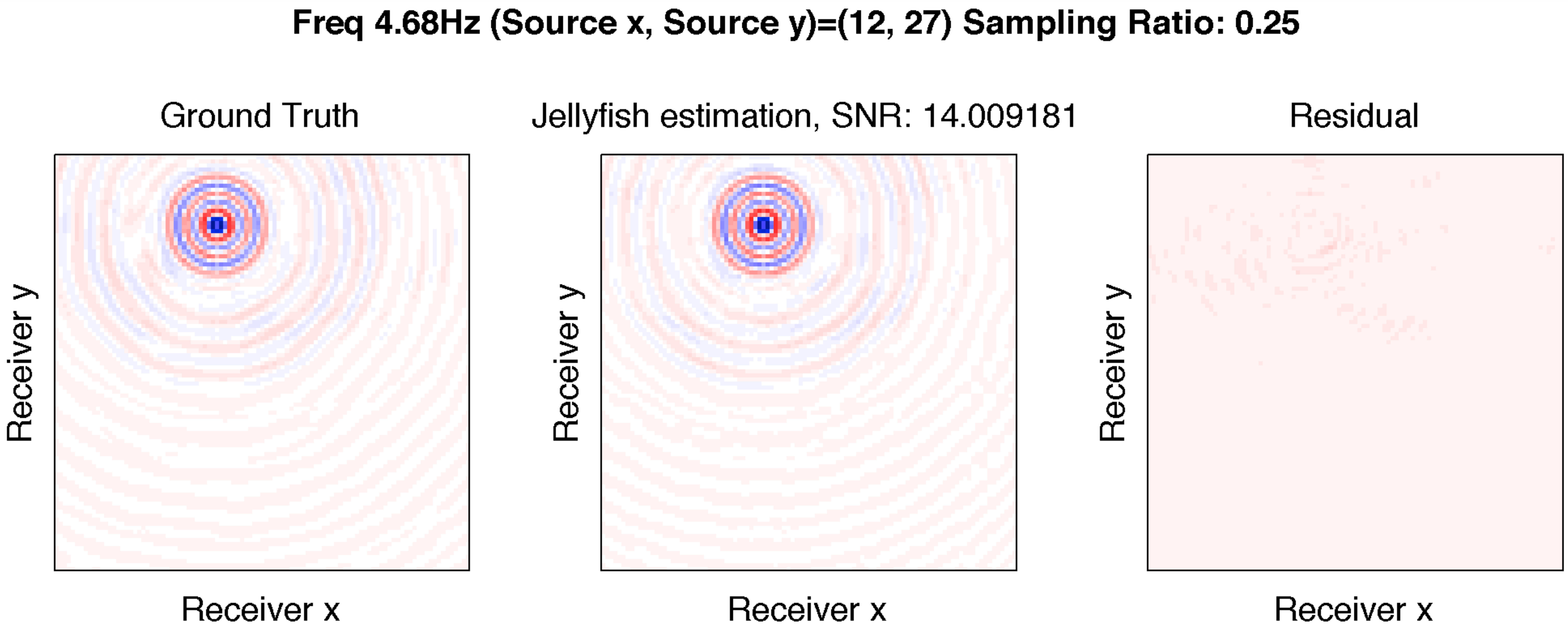
Gandy, S., Recht, B., & Yamada, I. (2011). Tensor completion and low-n-rank tensor recovery via convex optimization. Inverse Problems, 27(2), 025010.

Kreimer, N. , Stanton, A., Sacchi, M. (2013). Tensor completion based on nuclear norm minimization for 5D seismic data reconstruction

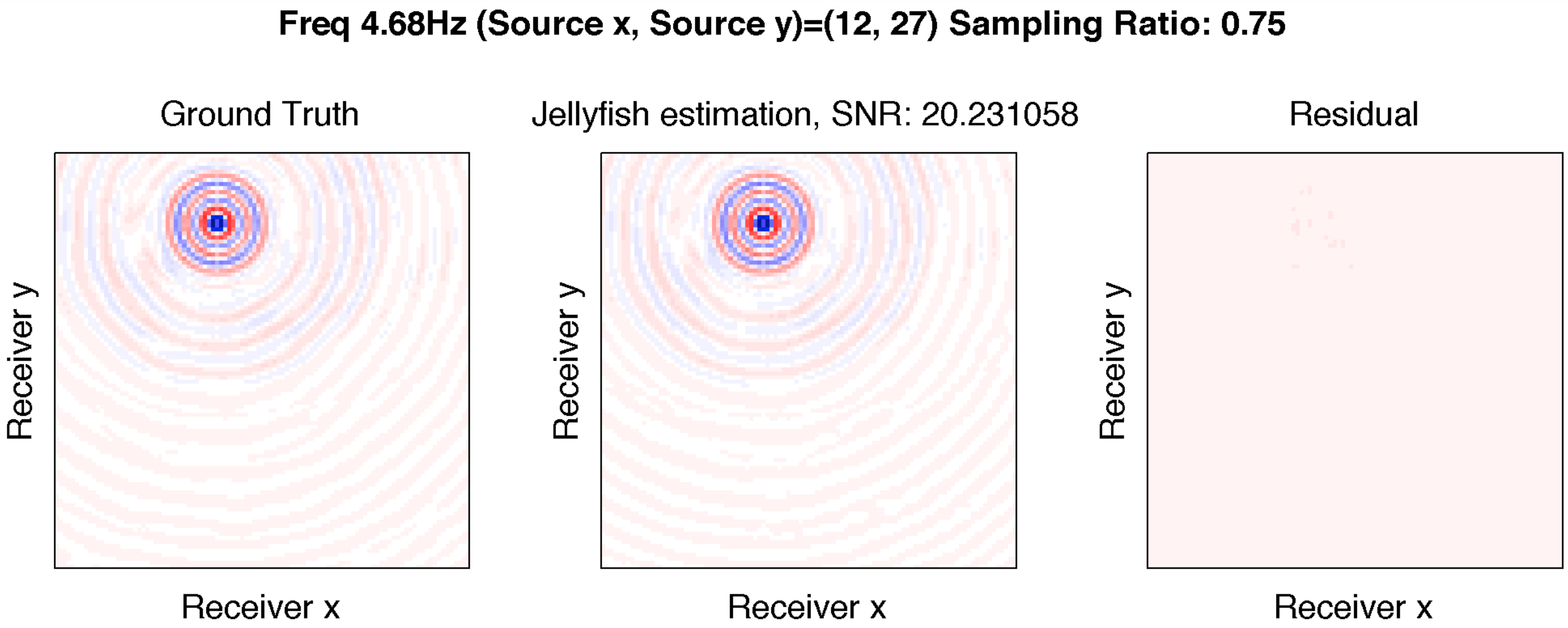
Outline

- Introduction
- Approach
- Experiments
- Conclusion & Future Work

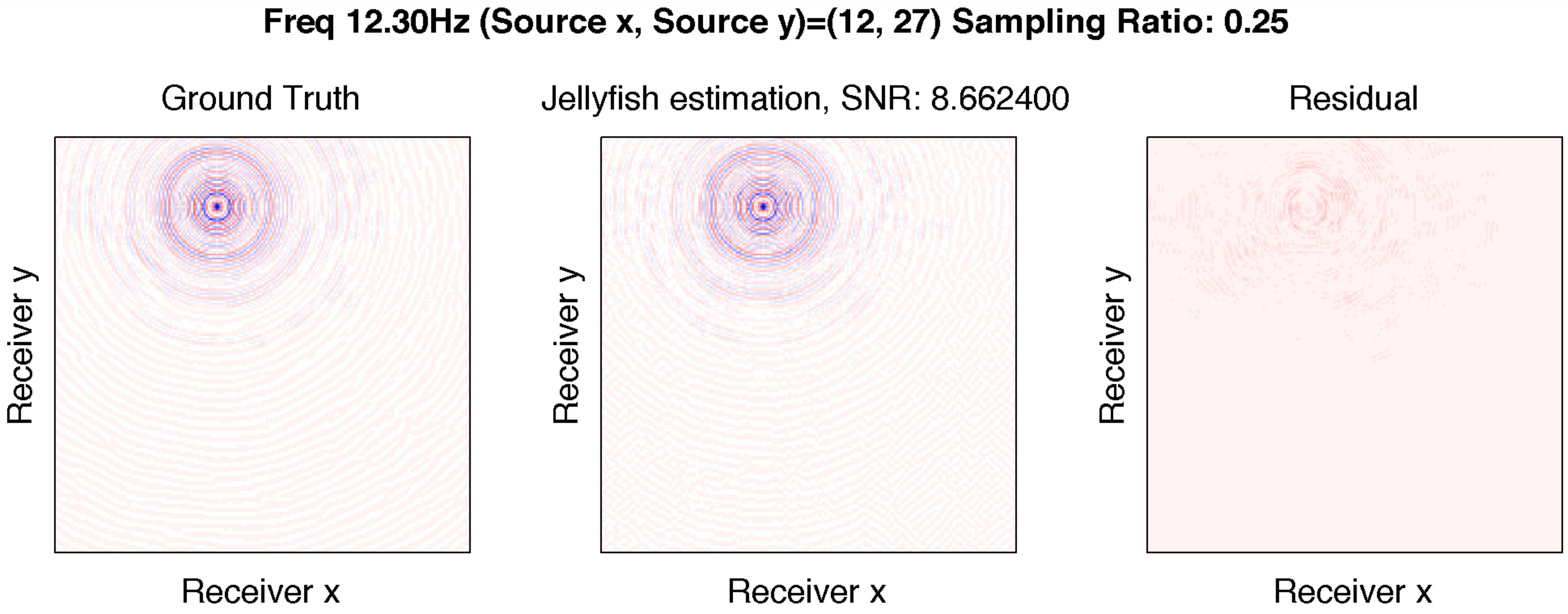
Matrix Completion Results



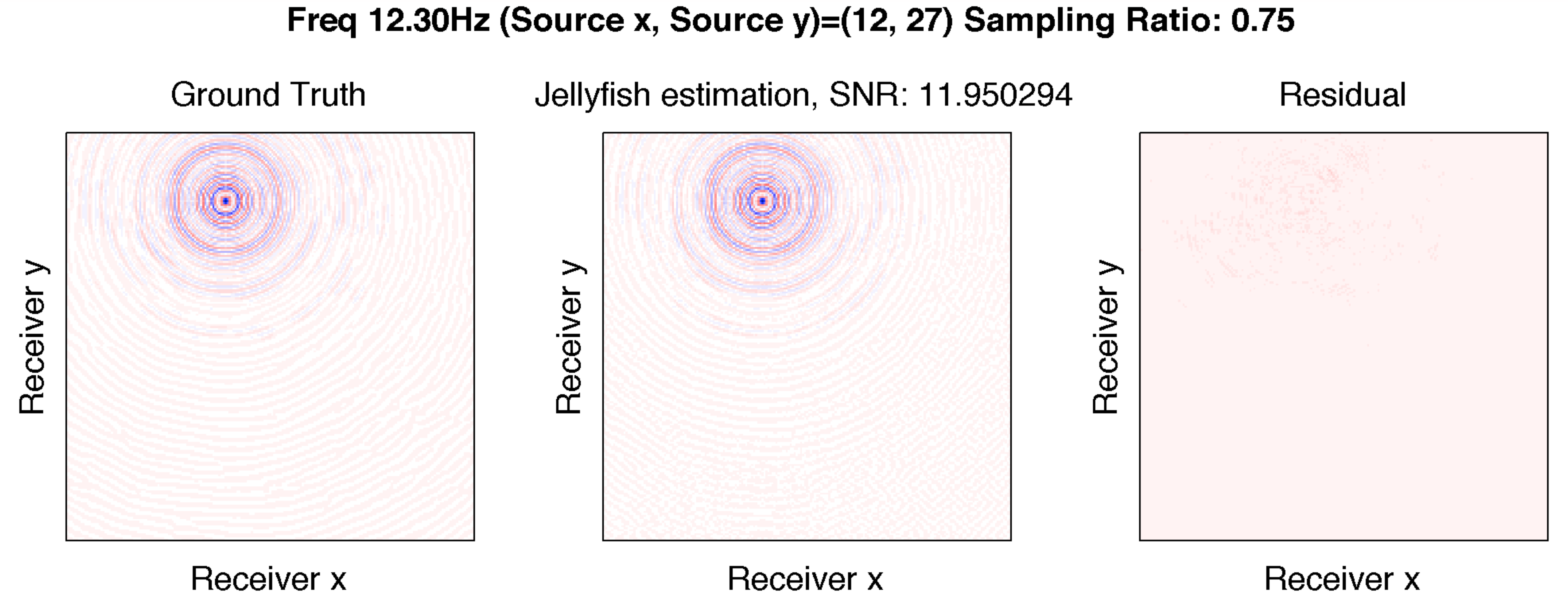
Matrix Completion Results



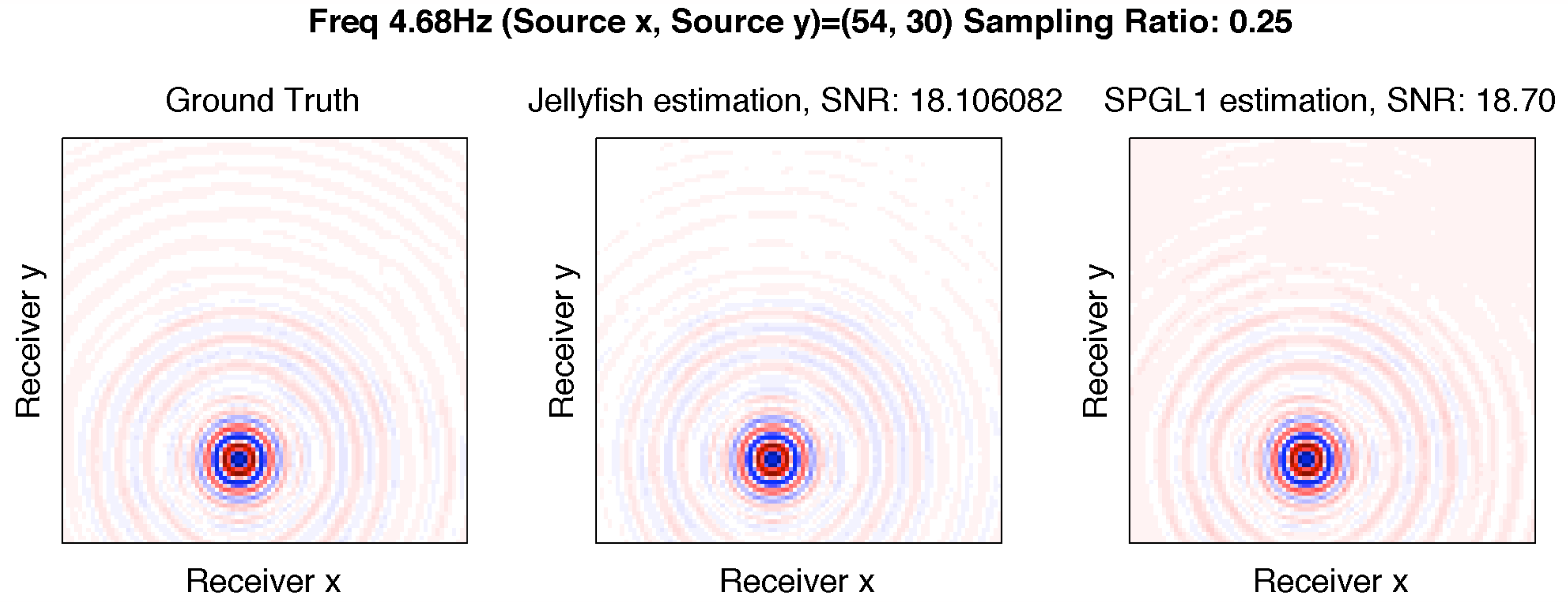
Matrix Completion Results



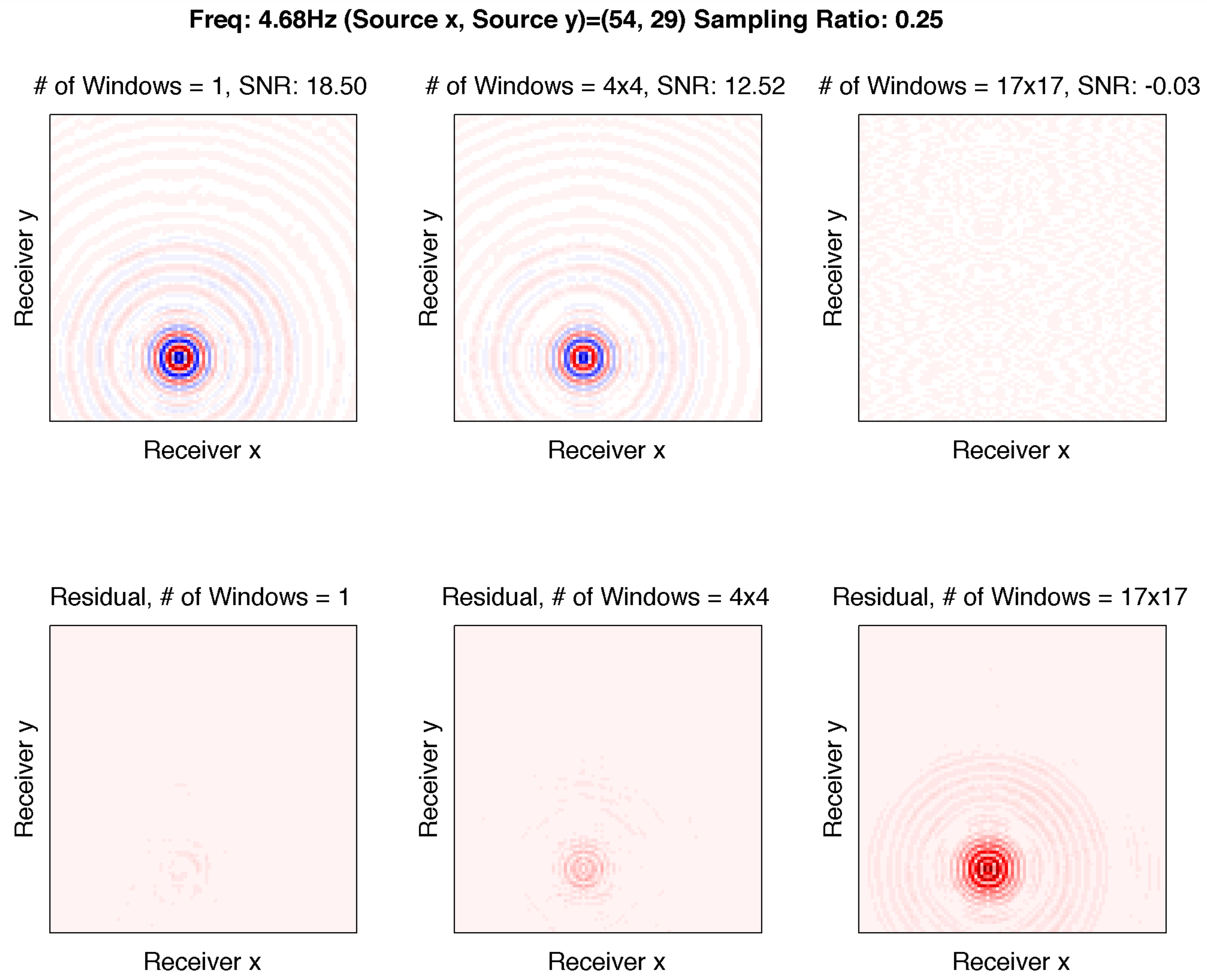
Matrix Completion Results



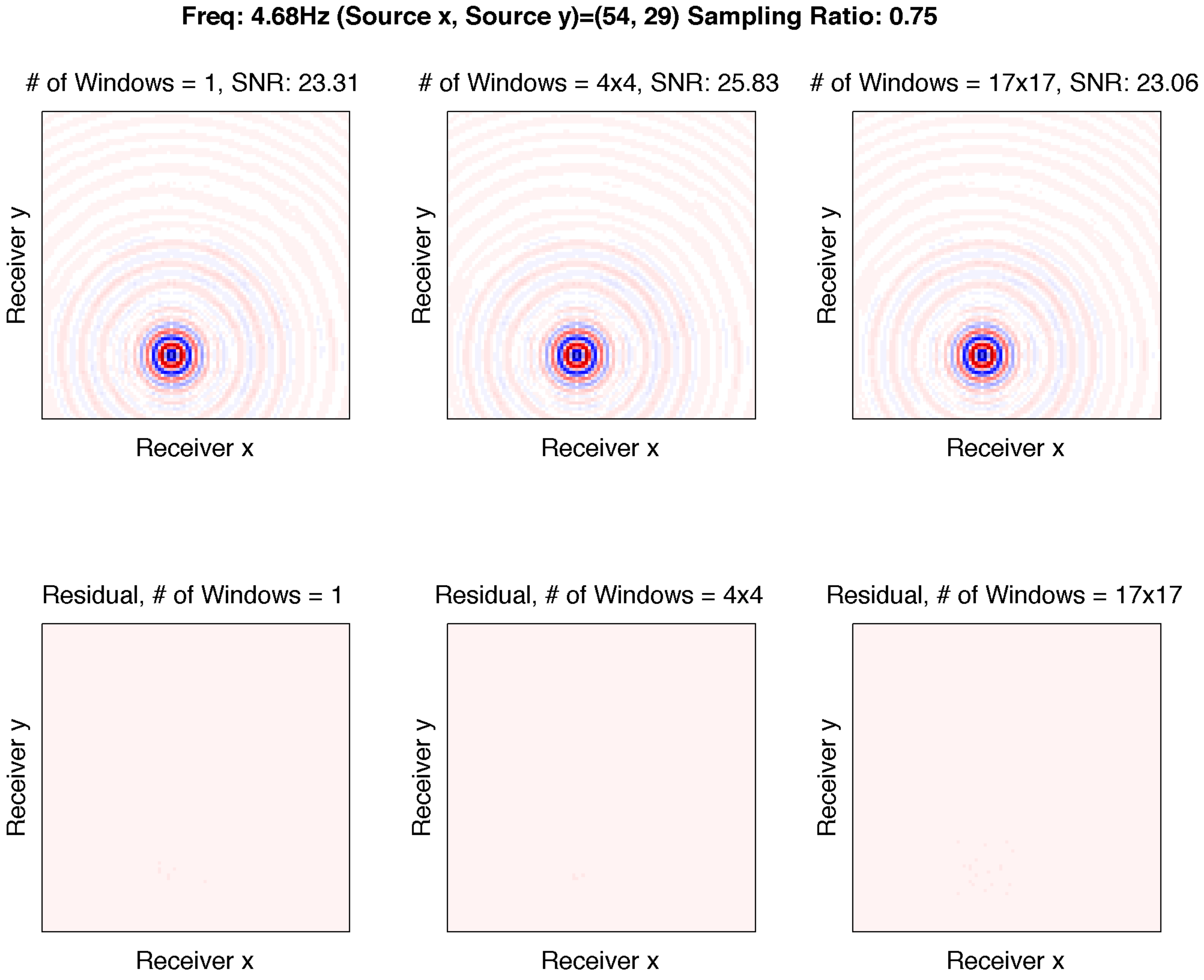
Jellyfish vs. SGPL1



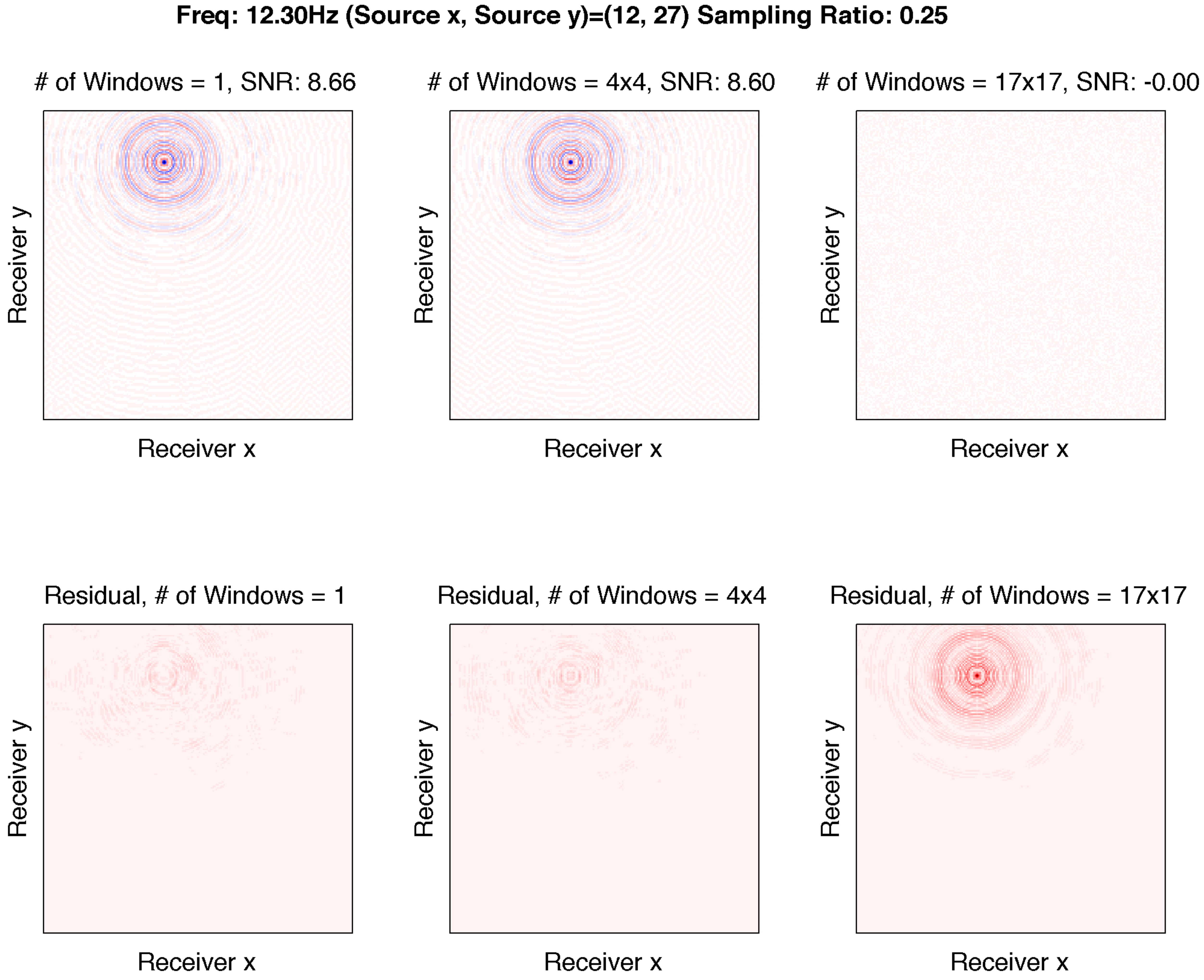
Windowing Results



Windowing Results



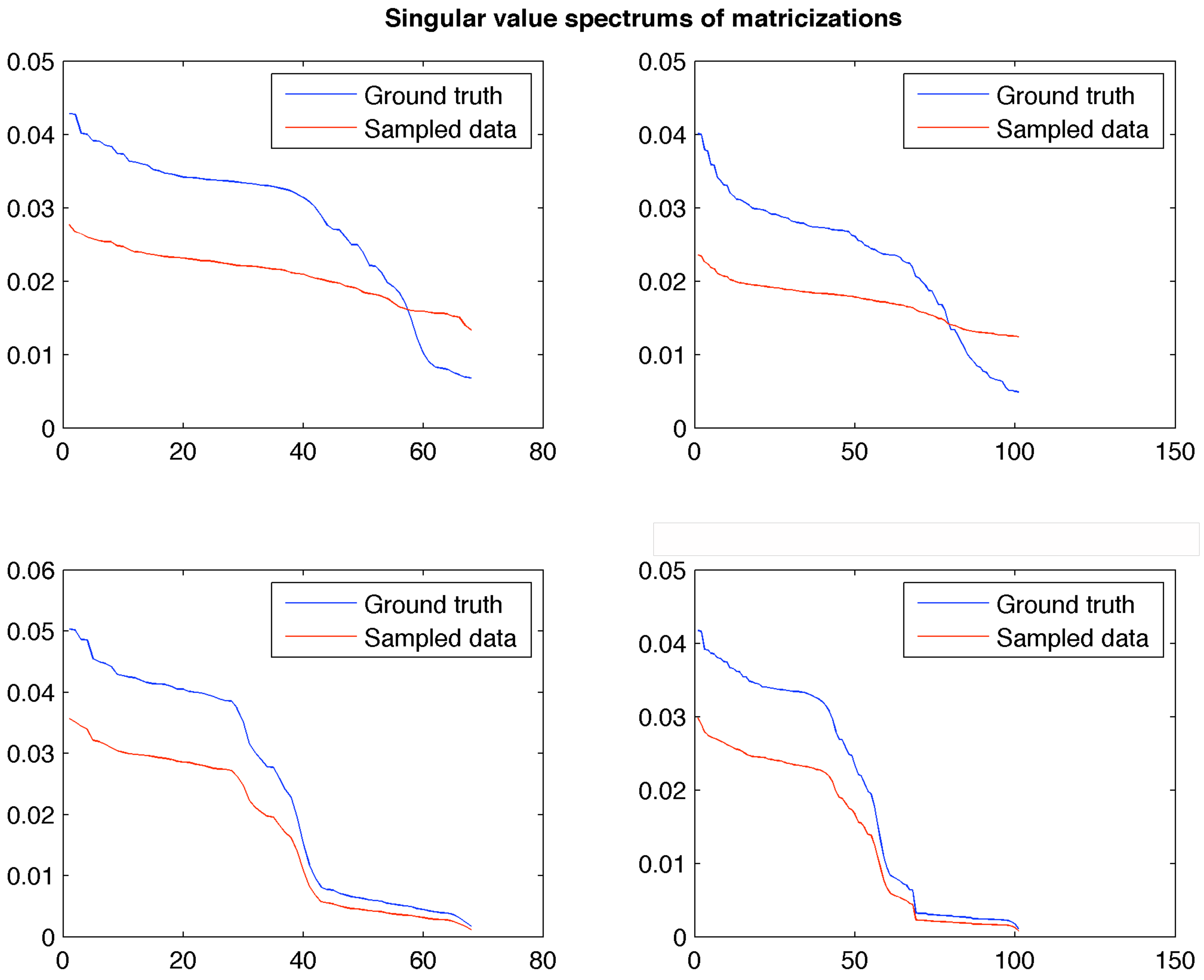
Windowing Results



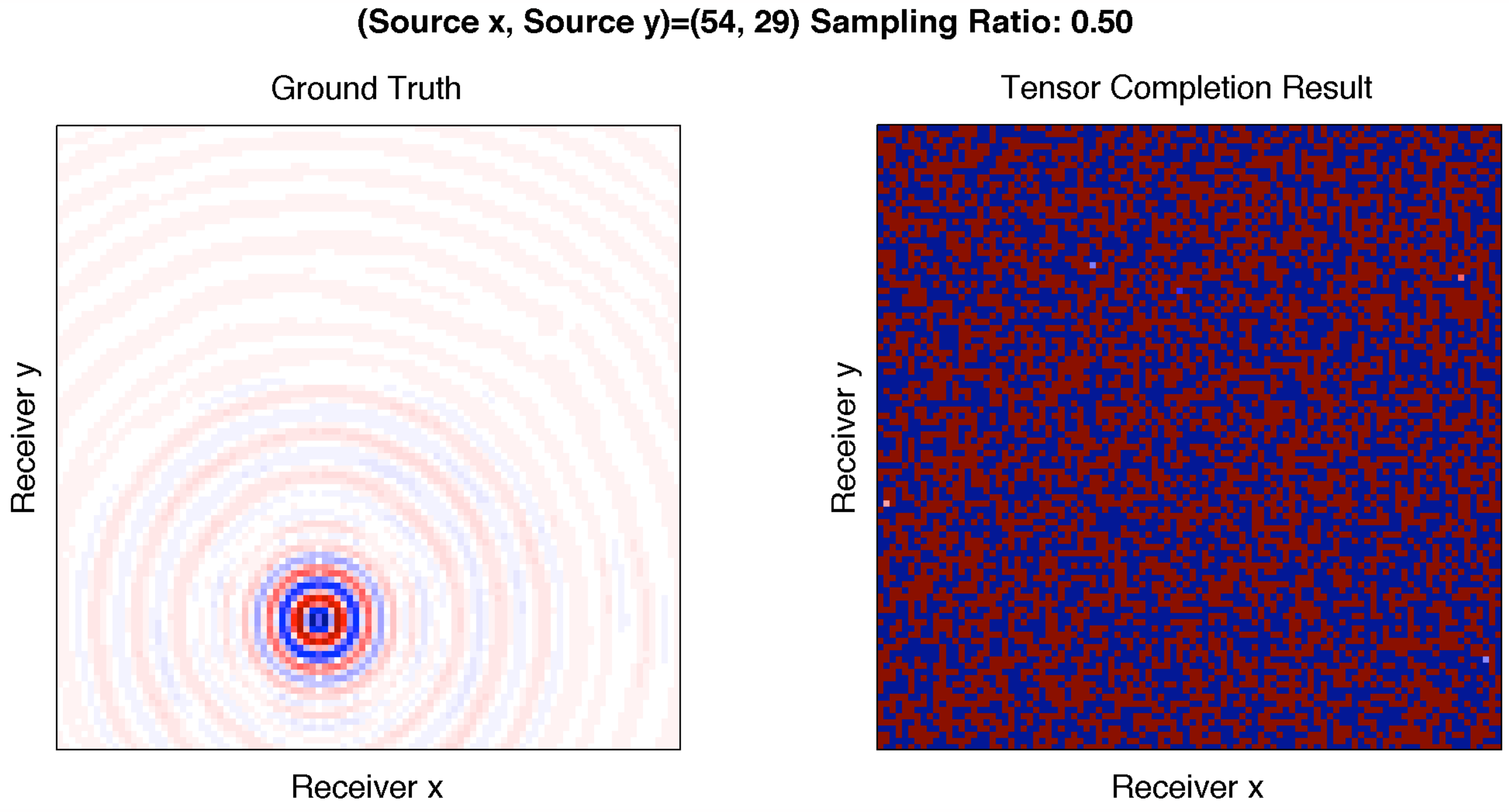
Comparison of Methods

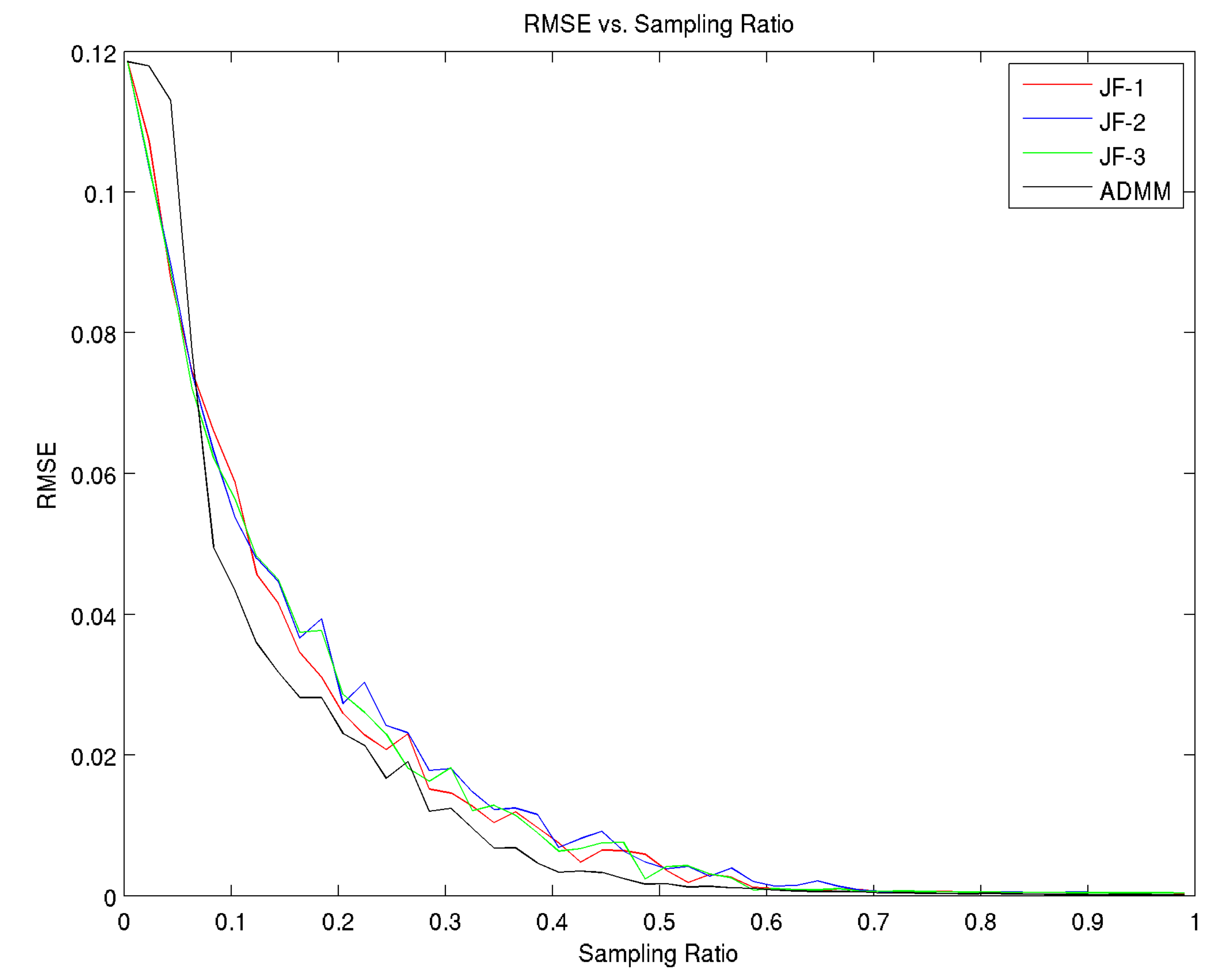
Frequency	% sources	SPGl_1	SPGl_1 time	Jellyfish	Jellyfish time	ADMM	ADMM time
4.68 Hz	25%	15.9	5040	16.34	2160	-80.96	127346
	50%	20.75	5760	19.81	4899	-82.03	132987
	75%	21.47	6840	19.64	7434	-82.25	130309
7.34 Hz	25%	11.2	5040	11.99	3126	-81.12	126984
	50%	15.2	7560	15.05	8767	-82.85	128921
	75%	16.3	8280	15.31	11710	-81.12	133567
12.3 Hz	25%	7.3	19440	9.34	13387	-80.14	174296
	50%	12.6	26280	12.12	42330	-81.19	177145
	75%	14.02	27000	12.90	77670	-81.53	175340

Tensor Completion Results

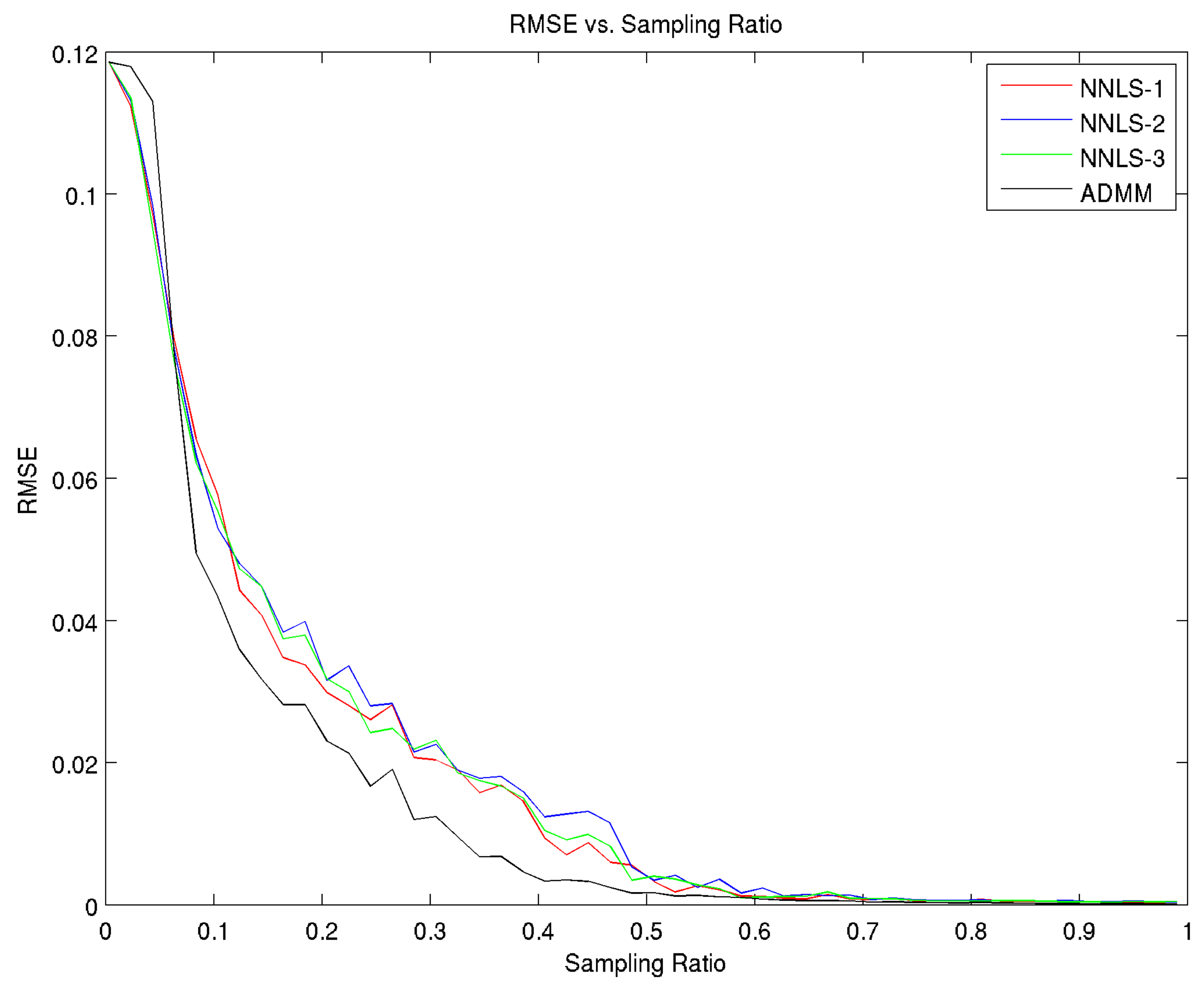


Tensor Completion Results

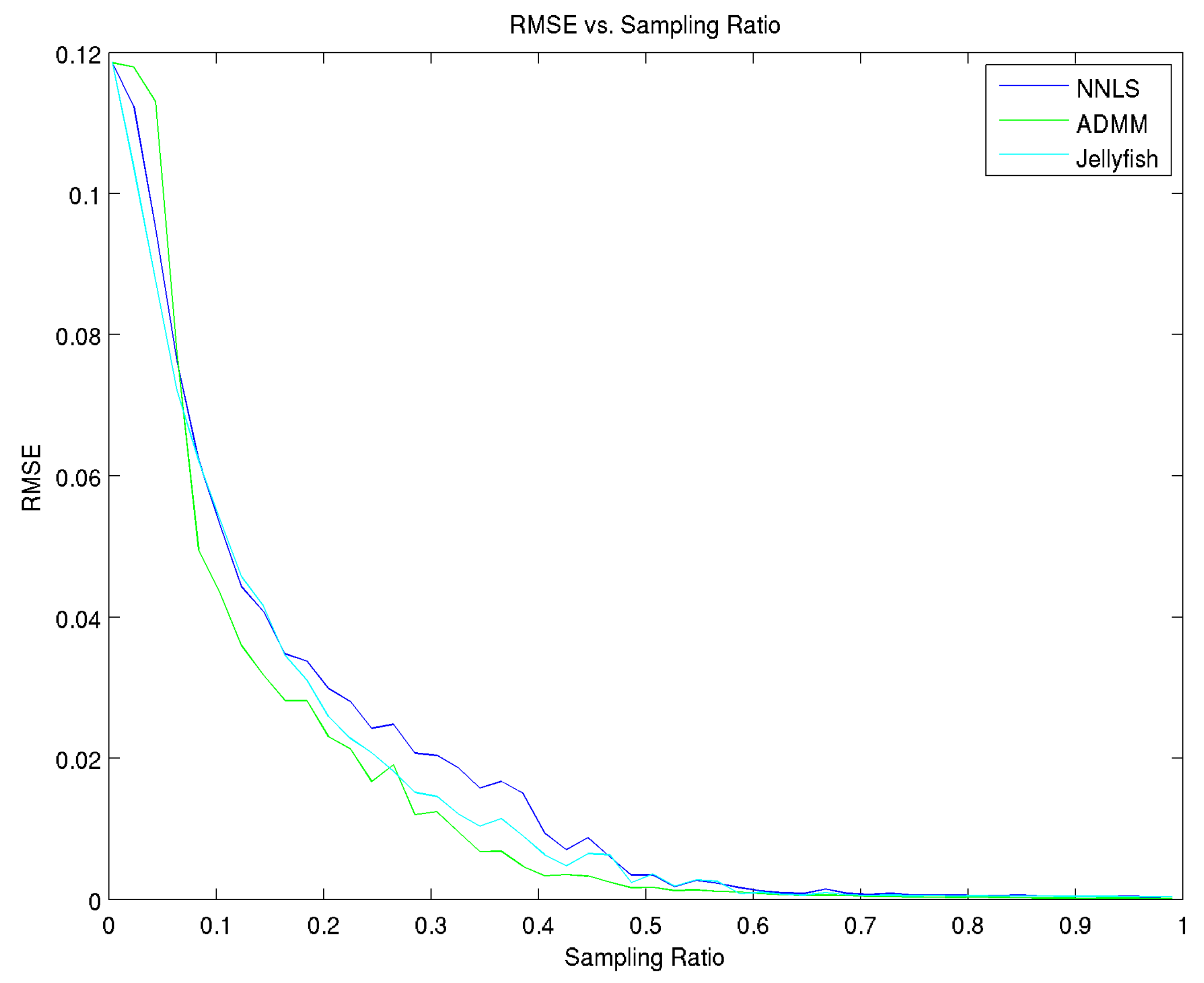




(a) Jellyfish Results

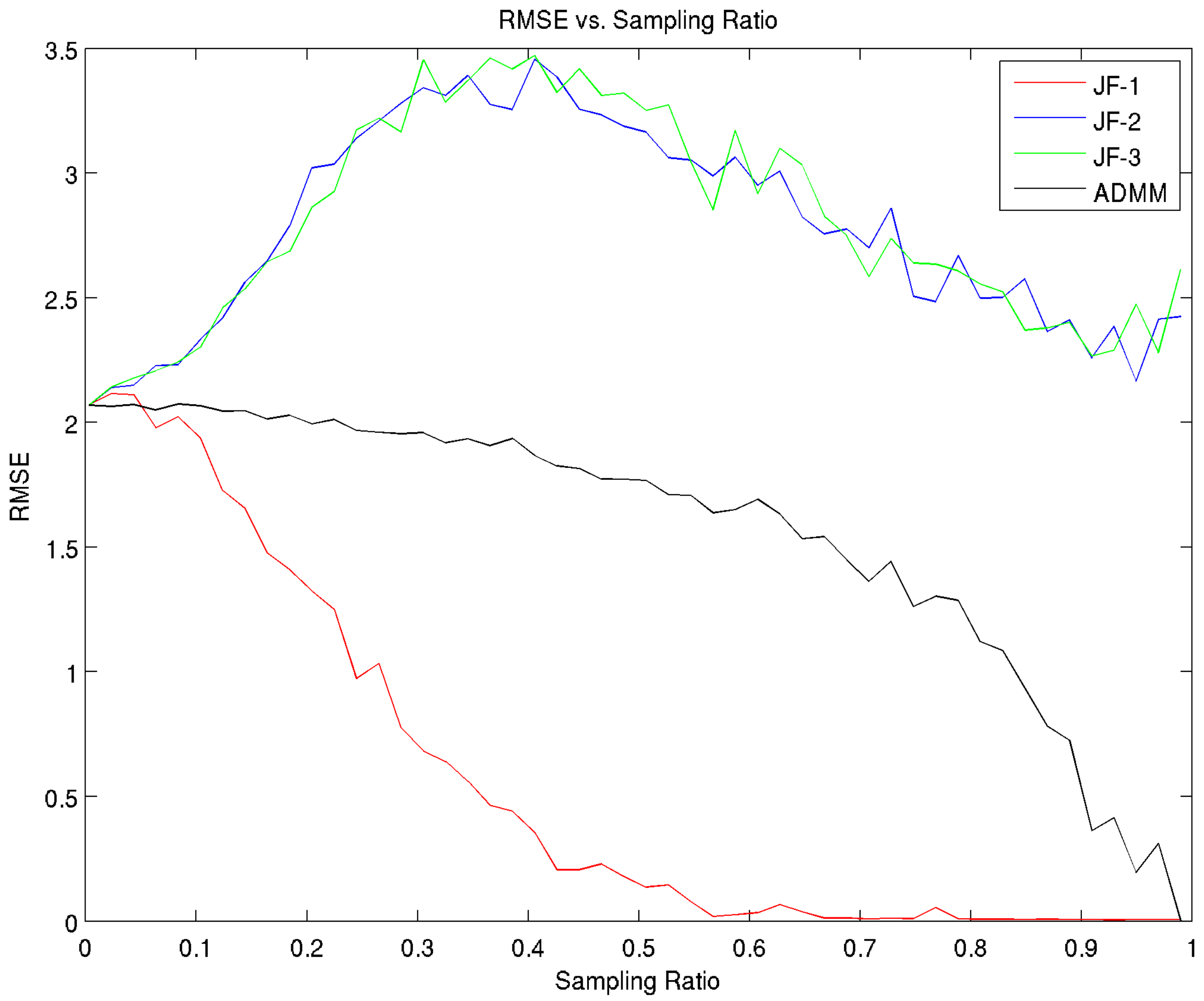


(b) NNLS Results

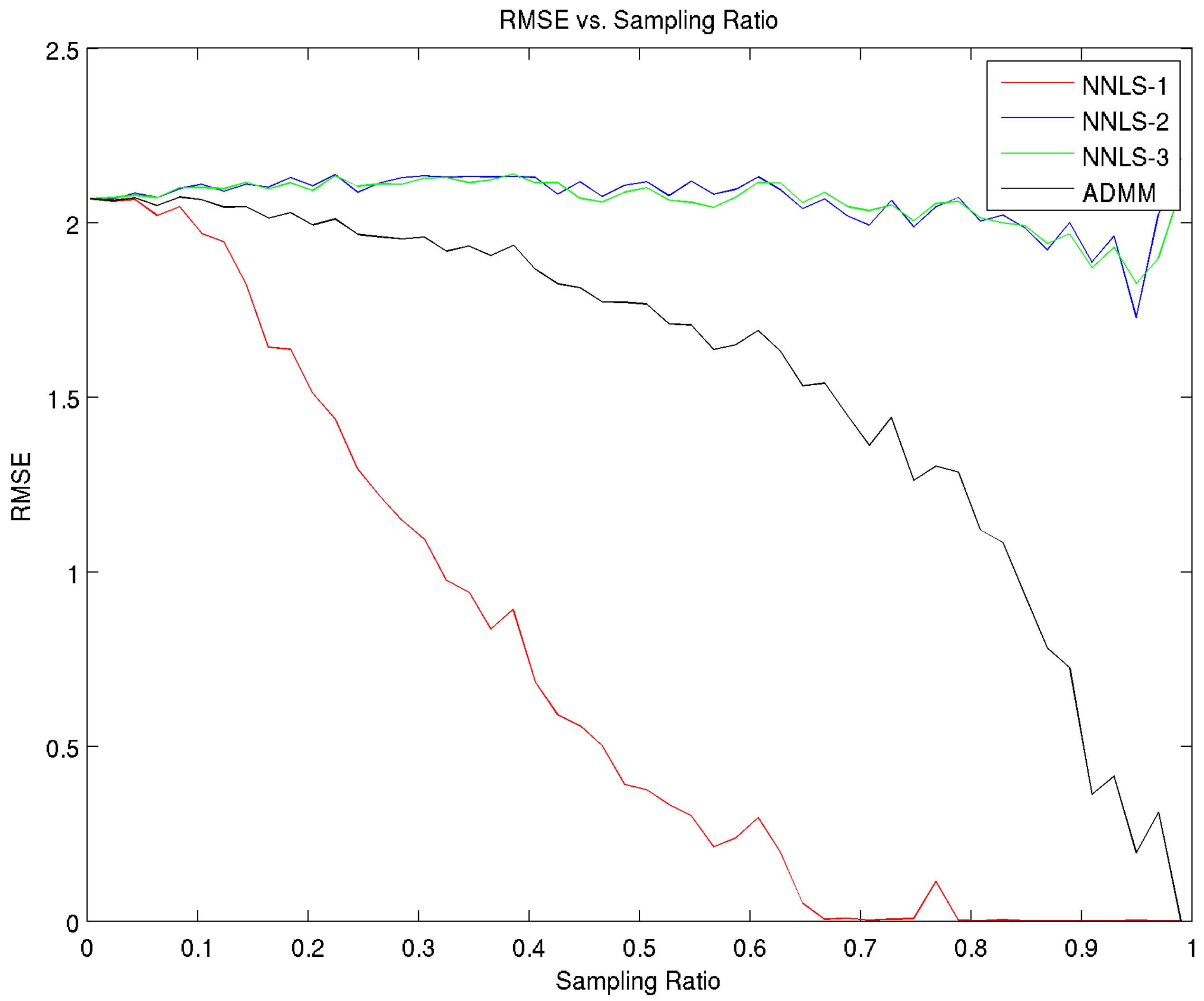


(c) Comparative Results

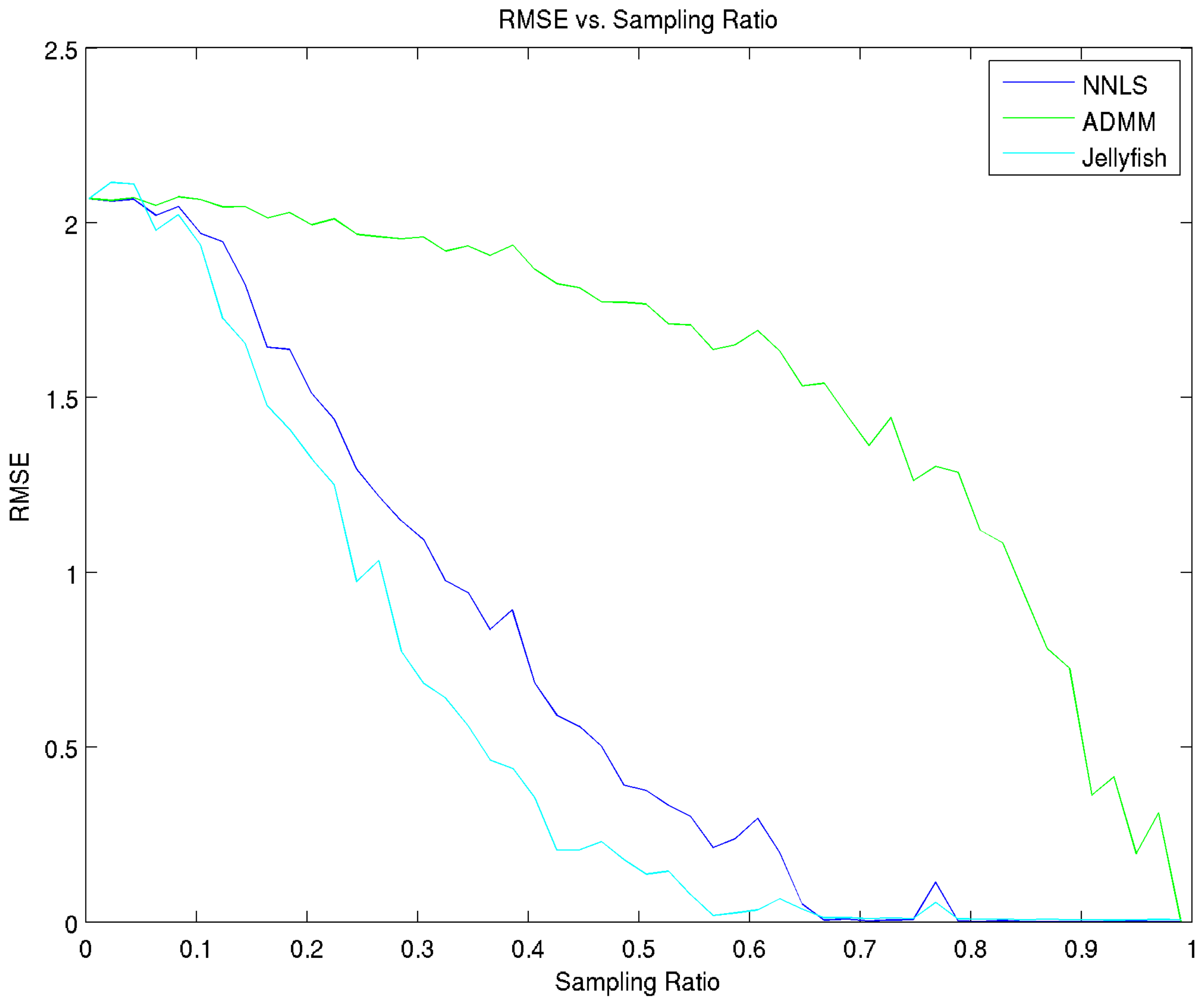
Figure 1: Sample complexity results for a rank 5 tensor of size $20 \times 20 \times 20$.



(a) Jellyfish Results



(b) NNLS Results



(c) Comparative Results

Figure 2: Sample complexity results for tensor of size $20 \times 20 \times 20$ with exactly one low rank unfolding (rank 5).

Recap

- Jellyfish and SPGL1 yield very similar results as expected.
- Matrix completion performs fairly well
- Windowing yields very good results, making it possible to scale to very large data sets
- Tensor completion with nuclear norm averaging does not work
- Parameter selection still takes considerable time, however good parameter combinations work across different problems

Outline

- Introduction
- Approach
- Experiments
- Conclusion & Future Work

Conclusion

- Compressed sensing provides a framework that allows scalable algorithms for the large scale seismic data interpolation problem
- The runtime efficiency can be increased by windowing
- In order to do successful tensor completion, we need a model that agrees with the structure of seismic data
    - Hierarchical Tucker Decomposition yields comparable results

Future Work

- Finding different low rank representations of seismic data
- Sample complexity comparison between tensor completion (Hierarchical Tucker Decomposition) and matrix completion methods
- Using windowing to interpolate high frequencies
- Using windowing for fast parameter selection

Thank you!

- Any questions?