SINBAD Consortium 2010

Compressed sensing using Kronecker products

Rayan Saab

SLIM
University of British Columbia

Compressed sensing (quick review)

- Compressed sensing: signal acquisition paradigm that allows for reconstructing N-dimensional signals from n measurements, where $n \ll N$.
- Suppose $x \in \mathbb{R}^N$ is (approximately) k-sparse and let A be an $n \times N$ (Gaussian) random matrix with $n \gtrsim k \log(N/k)$.
- Now suppose we collect the measurements $b = Ax + e$ (here e is noise with $\|e\|_2 \leq \epsilon$) and we wish to recover x from b.
- Solve:

$$x^* = \arg\min_v \|v\|_1 \text{ subject to } \|b - Av\|_2 \leq \epsilon$$

- Then $\|x - x^*\|_2 \lesssim \epsilon + \|x - x_k\|_1/\sqrt{k}$.
- Useful tool in proving this is the restricted isometry propery (RIP): A matrix A is said to satisfy the RIP with constants δ_k if for all k-sparse vectors x, we have

$$(1 - \delta_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_k)\|x\|_2^2.$$

Compressed sensing (quick review)

- Compressed sensing: signal acquisition paradigm that allows for reconstructing N-dimensional signals from n measurements, where $n \ll N$.
- Suppose $x \in \mathbb{R}^N$ is (approximately) k-sparse and let A be an $n \times N$ (Gaussian) random matrix with $n \gtrsim k \log{(N/k)}$.
- Now suppose we collect the measurements $b = Ax + e$ (here e is noise with $\|e\|_2 \leq \epsilon$) and we wish to recover x from b.
- Solve:

$$x^* = \arg\min_v \|v\|_1 \text{ subject to } \|b - Av\|_2 \leq \epsilon$$

- Then $\|x - x^*\|_2 \lesssim \epsilon + \|x - x_k\|_1/\sqrt{k}$.
- Useful tool in proving this is the restricted isometry propery (RIP): A matrix A is said to satisfy the RIP with constants δ_k if for all k-sparse vectors x, we have

$$(1 - \delta_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_k)\|x\|_2^2.$$

Compressed sensing (quick review)

- Compressed sensing: signal acquisition paradigm that allows for reconstructing N-dimensional signals from n measurements, where $n \ll N$.
- Suppose $x \in \mathbb{R}^N$ is (approximately) k-sparse and let A be an $n \times N$ (Gaussian) random matrix with $n \gtrsim k \log(N/k)$.
- Now suppose we collect the measurements $b = Ax + e$ (here e is noise with $\|e\|_2 \leq \epsilon$) and we wish to recover x from b.
- Solve:
$$x^* = \arg\min_v \|v\|_1 \text{ subject to } \|b - Av\|_2 \leq \epsilon$$
- Then $\|x - x^*\|_2 \lesssim \epsilon + \|x - x_k\|_1/\sqrt{k}$.
- Useful tool in proving this is the restricted isometry propery (RIP): A matrix A is said to satisfy the RIP with constants δ_k if for all k-sparse vectors x, we have
$$(1 - \delta_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_k)\|x\|_2^2.$$

Compressed sensing (quick review)

- Compressed sensing: signal acquisition paradigm that allows for reconstructing N-dimensional signals from n measurements, where $n \ll N$.
- Suppose $x \in \mathbb{R}^N$ is (approximately) k-sparse and let A be an $n \times N$ (Gaussian) random matrix with $n \gtrsim k \log(N/k)$.
- Now suppose we collect the measurements $b = Ax + e$ (here e is noise with $\|e\|_2 \leq \epsilon$) and we wish to recover x from b.
- Solve:

$$x^* = \arg\min_v \|v\|_1 \text{ subject to } \|b - Av\|_2 \leq \epsilon$$

- Then $\|x - x^*\|_2 \lesssim \epsilon + \|x - x_k\|_1/\sqrt{k}$.
- Useful tool in proving this is the restricted isometry propery (RIP): A matrix A is said to satisfy the RIP with constants δ_k if for all k-sparse vectors x, we have

$$(1 - \delta_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_k)\|x\|_2^2.$$

Compressed sensing (quick review)

- Compressed sensing: signal acquisition paradigm that allows for reconstructing N-dimensional signals from n measurements, where $n \ll N$.
- Suppose $x \in \mathbb{R}^N$ is (approximately) k-sparse and let A be an $n \times N$ (Gaussian) random matrix with $n \gtrsim k \log(N/k)$.
- Now suppose we collect the measurements $b = Ax + e$ (here e is noise with $\|e\|_2 \leq \epsilon$) and we wish to recover x from b.
- Solve:

$$x^* = \arg\min_v \|v\|_1 \text{ subject to } \|b - Av\|_2 \leq \epsilon$$

- Then $\|x - x^*\|_2 \lesssim \epsilon + \|x - x_k\|_1/\sqrt{k}$.
- Useful tool in proving this is the restricted isometry propery (RIP): A matrix A is said to satisfy the RIP with constants δ_k if for all k-sparse vectors x, we have

$$(1 - \delta_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_k)\|x\|_2^2.$$

Compressed sensing (quick review)

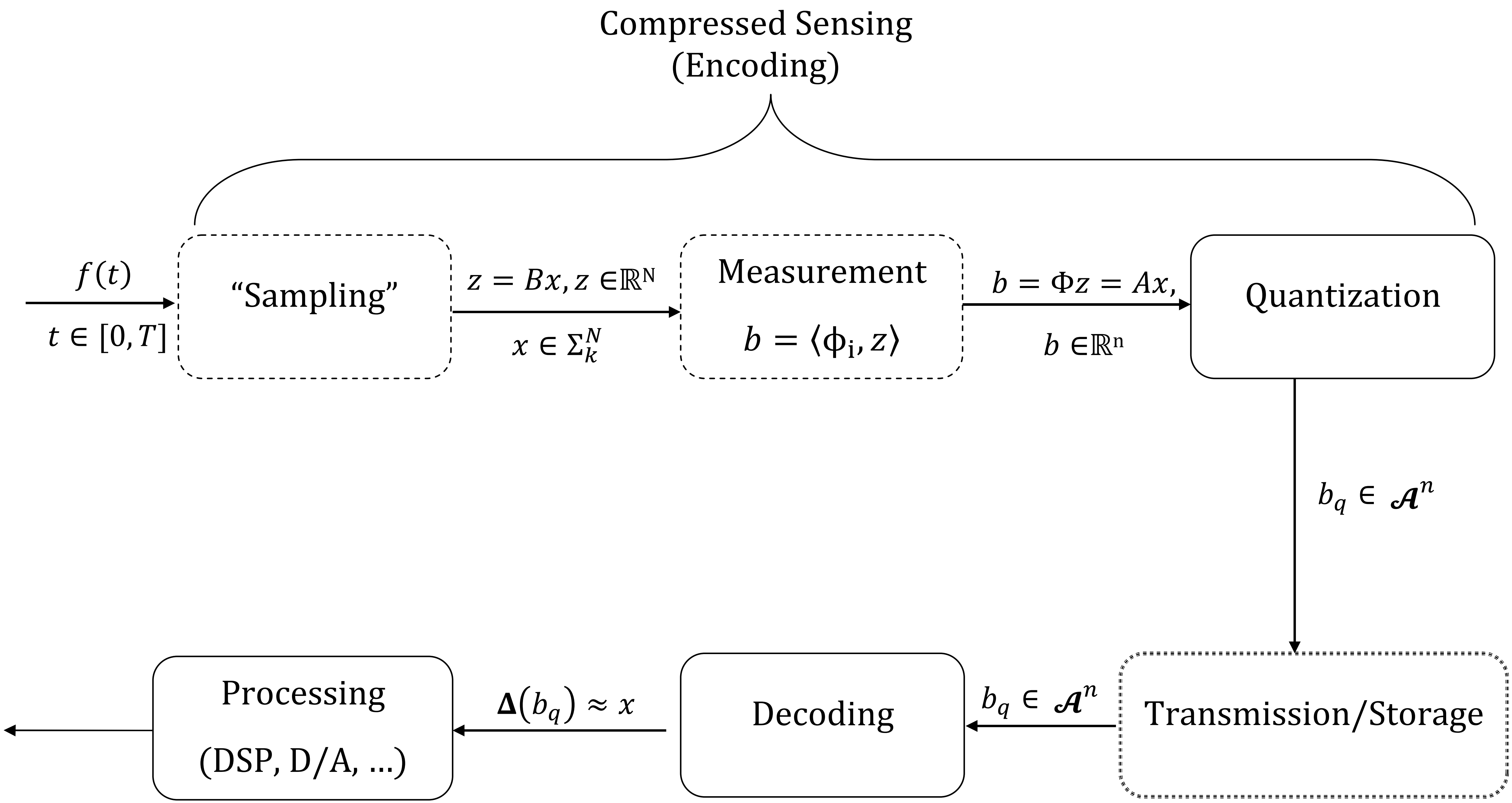
- Compressed sensing: signal acquisition paradigm that allows for reconstructing N-dimensional signals from n measurements, where $n \ll N$.
- Suppose $x \in \mathbb{R}^N$ is (approximately) k-sparse and let A be an $n \times N$ (Gaussian) random matrix with $n \gtrsim k \log{(N/k)}$.
- Now suppose we collect the measurements $b = Ax + e$ (here e is noise with $\|e\|_2 \leq \epsilon$) and we wish to recover x from b.
- Solve:

$$x^* = \arg\min_v \|v\|_1 \text{ subject to } \|b - Av\|_2 \leq \epsilon$$

- Then $\|x - x^*\|_2 \lesssim \epsilon + \|x - x_k\|_1/\sqrt{k}$.
- Useful tool in proving this is the restricted isometry propery (RIP): A matrix A is said to satisfy the RIP with constants δ_k if for all k-sparse vectors x, we have

$$(1 - \delta_k)\|x\|_2^2 \leq \|Ax\|_2^2 \leq (1 + \delta_k)\|x\|_2^2.$$

Compressed sensing (block diagram)



Compressed sensing in "more" dimensions

- Compressed sensing is concerned with recovering *vectors* in, say $\mathbb{R}^N$.
- What if we want to recover objects, say, in $\mathbb{R}^N \times \mathbb{R}^N$? More generally, in d-dimensions (e.g., the Green's function)?
- No problem if we we can treat "object" as vector in $\mathbb{R}^{N^2}$. Example: Single pixel camera of Rice - the dimensions are inherently similar.
- On the other hand, if we were to take a video, "the time axis" cannot be compressively sensed in the same way that pixels are!
- In (some) seismic applications, the object of interest is the Green's function.
- We have source "dimensions", receiver "dimensions", and time. The dimensions are inherently different!

Compressed sensing in “more” dimensions

- Compressed sensing is concerned with recovering *vectors* in, say $\mathbb{R}^N$.
- What if we want to recover objects, say, in $\mathbb{R}^N \times \mathbb{R}^N$? More generally, in d-dimensions (e.g., the Green’s function)?
- No problem if we we can treat “object” as vector in $\mathbb{R}^{N^2}$. Example: Single pixel camera of Rice - the dimensions are inherently similar.
- On the other hand, if we were to take a video, ”the time axis” cannot be compressively sensed in the same way that pixels are!
- In (some) seismic applications, the object of interest is the Green’s function.
- We have source ”dimensions”, receiver ”dimensions”, and time. The dimensions are inherently different!

Compressed sensing in "more" dimensions

- Compressed sensing is concerned with recovering *vectors* in, say $\mathbb{R}^N$.
- What if we want to recover objects, say, in $\mathbb{R}^N \times \mathbb{R}^N$? More generally, in d-dimensions (e.g., the Green's function)?
- No problem if we we can treat "object" as vector in $\mathbb{R}^{N^2}$. Example: Single pixel camera of Rice - the dimensions are inherently similar.
- On the other hand, if we were to take a video, "the time axis" cannot be compressively sensed in the same way that pixels are!
- In (some) seismic applications, the object of interest is the Green's function.
- We have source "dimensions", receiver "dimensions", and time. The dimensions are inherently different!

Compressed sensing in “more” dimensions

- Compressed sensing is concerned with recovering *vectors* in, say $\mathbb{R}^N$.
- What if we want to recover objects, say, in $\mathbb{R}^N \times \mathbb{R}^N$? More generally, in d-dimensions (e.g., the Green’s function)?
- No problem if we we can treat “object” as vector in $\mathbb{R}^{N^2}$. Example: Single pixel camera of Rice - the dimensions are inherently similar.
- On the other hand, if we were to take a video, ”the time axis” cannot be compressively sensed in the same way that pixels are!
- In (some) seismic applications, the object of interest is the Green’s function.
- We have source ”dimensions”, receiver ”dimensions”, and time. The dimensions are inherently different!

Compressed sensing in “more” dimensions

- Compressed sensing is concerned with recovering *vectors* in, say $\mathbb{R}^N$.
- What if we want to recover objects, say, in $\mathbb{R}^N \times \mathbb{R}^N$? More generally, in d-dimensions (e.g., the Green’s function)?
- No problem if we we can treat “object” as vector in $\mathbb{R}^{N^2}$. Example: Single pixel camera of Rice - the dimensions are inherently similar.
- On the other hand, if we were to take a video, ”the time axis” cannot be compressively sensed in the same way that pixels are!
- In (some) seismic applications, the object of interest is the Green’s function.
- We have source ”dimensions”, receiver ”dimensions”, and time. The dimensions are inherently different!

Compressed sensing in seismic acquisition

- We have "source dimensions", "receiver dimensions", and time. The dimensions are inherently different!
- Experiments that we are allowed to conduct consist of:
  1. firing (known combinations of) sources, collecting the response at each receiver over a period of time.
  2. repeating the above for several source configurations.
- Possible objective (?) reduce cost (=time, money, storage space, computational complexity, cost of randomness) without compromising accuracy (too much).

Compressed sensing in seismic acquisition

- We have "source dimensions", "receiver dimensions", and time. The dimensions are inherently different!
- Experiments that we are allowed to conduct consist of:
  1. firing (known combinations of) sources, collecting the response at each receiver over a period of time.
  2. repeating the above for several source configurations.
- Possible objective (?) reduce cost (=time, money, storage space, computational complexity, cost of randomness) without compromising accuracy (too much).

Compressed sensing in seismic acquisition

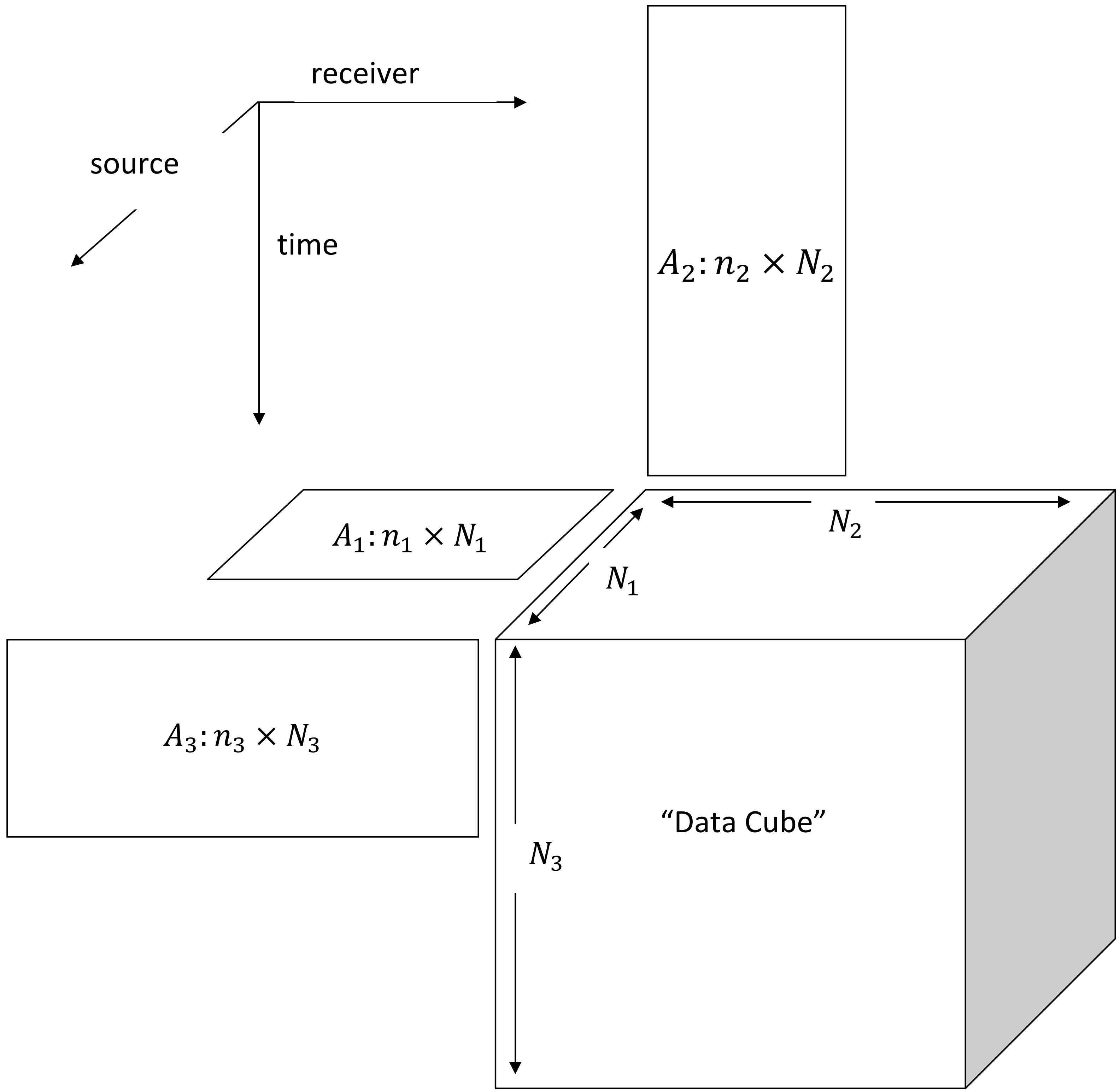
- We have "source dimensions", "receiver dimensions", and time. The dimensions are inherently different!
- Experiments that we are allowed to conduct consist of:
  1. firing (known combinations of) sources, collecting the response at each receiver over a period of time.
  2. repeating the above for several source configurations.
- Possible objective (?) reduce cost (=time, money, storage space, computational complexity, cost of randomness) without compromising accuracy (too much).

Compressed sensing in seismic acquisition

- So how can we reduce cost:
  1. option 1: Fire a fewer total number of sources, e.g.,
     - jitter sampling of a fine grid of sources
     - firing combinations of sources simultaneously $\rightarrow$ repeat a few times.
  2. option 2: Combining receiver recordings and storing/transmitting the combinations, with fewer combinations than receivers (save battery life, computation time).
  3. option 3: Do both of the above - possibly treat the time axis similarly.

Compressed sensing in seismic acquisition

Below, "data cube" corresponds to seismic line



Mathematical formulation

- In two dimensions: collect the measurements B of X:

$$B = A_1 X A_2^*.$$

- Define: $M \otimes A = \begin{bmatrix} M(1,1)A & M(1,2)A & \dots & M(1,N_2)A \\ M(2,1)A & M(2,2)A & \dots & M(2,N_2)A \\ \vdots & \vdots & \ddots & \vdots \\ M(n_2,1)A & M(n_2,2)A & \dots & M(n_2,N_2)A \end{bmatrix}$

- Vectorize:

$$B(:) = (A_2 \otimes A_1)X(:).$$

- In d-dimensions :

$$B(:) = \underbrace{(A_d \otimes A_{d-1} \otimes \dots \otimes A_1)}_{A} X(:).$$

Mathematical formulation

- In two dimensions: collect the measurements B of X:

$$B = A_1 X A_2^*.$$

- Define: $M \otimes A = \begin{bmatrix} M(1,1)A & M(1,2)A & \dots & M(1,N_2)A \\ M(2,1)A & M(2,2)A & \dots & M(2,N_2)A \\ \vdots & \vdots & \ddots & \vdots \\ M(n_2,1)A & M(n_2,2)A & \dots & M(n_2,N_2)A \end{bmatrix}$

- Vectorize:

$$B(:) = (A_2 \otimes A_1) X(:).$$

- In d-dimensions :

$$B(:) = \underbrace{(A_d \otimes A_{d-1} \otimes \dots \otimes A_1)}_{A} X(:).$$

Mathematical formulation

- In two dimensions: collect the measurements B of X:

$$B = A_1 X A_2^*.$$

- Define: $M \otimes A = \begin{bmatrix} M(1,1)A & M(1,2)A & \dots & M(1,N_2)A \\ M(2,1)A & M(2,2)A & \dots & M(2,N_2)A \\ \vdots & \vdots & \ddots & \vdots \\ M(n_2,1)A & M(n_2,2)A & \dots & M(n_2,N_2)A \end{bmatrix}$
- Vectorize:

$$B(:) = (A_2 \otimes A_1)X(:).$$

- In d-dimensions :

$$B(:) = \underbrace{(A_d \otimes A_{d-1} \otimes \dots \otimes A_1)}_{A} X(:).$$

Mathematical formulation

- In two dimensions: collect the measurements B of X:

$$B = A_1 X A_2^*.$$

- Define: $M \otimes A = \begin{bmatrix} M(1,1)A & M(1,2)A & \dots & M(1,N_2)A \\ M(2,1)A & M(2,2)A & \dots & M(2,N_2)A \\ \vdots & \vdots & \ddots & \vdots \\ M(n_2,1)A & M(n_2,2)A & \dots & M(n_2,N_2)A \end{bmatrix}$

- Vectorize:

$$B(:) = (A_2 \otimes A_1) X(:).$$

- In d-dimensions :

$$B(:) = \underbrace{(A_d \otimes A_{d-1} \otimes \ldots \otimes A_1)}_{A} X(:).$$

Back to compressed sensing

- When A is $n \times N$, random, we can (with high probability) recover all k-sparse vectors with $k \lesssim n/\log N$.
- How? ℓ_1-minimization!
- What about when A is a Kronecker product of several A_i's?
- No straightforward answer! In general, the restricted isometry constant of A is larger than the worst RIP constant of the A_i's.
- From now on let us assume that all A_i have the same dimensions and RIP constants.
- Bad news: This can at best guarantee the recovery of k-sparse vectors by ℓ_1 minimization
(terrible, because we have N^d entries in our "object")
- Ideally, we would like to recover k^d-sparse vectors.
- Cannot do that (place all k^d non-zeros on one dimension.)
- What about trying to recover a $k \times k \times ... \times k$ "hypercube" (or dimension-wise permutations of it)?
- Can easily do it!! Solve d ℓ_1-minimization problems sequentially!!

Back to compressed sensing

- When A is $n \times N$, random, we can (with high probability) recover all k-sparse vectors with $k \lesssim n/\log N$.
- How? ℓ_1-minimization!
- What about when A is a Kronecker product of several A_i's?
- No straightforward answer! In general, the restricted isometry constant of A is larger than the worst RIP constant of the A_i's.
- From now on let us assume that all A_i have the same dimensions and RIP constants.
- Bad news: This can at best guarantee the recovery of k-sparse vectors by ℓ_1 minimization
(terrible, because we have N^d entries in our "object")
- Ideally, we would like to recover k^d-sparse vectors.
- Cannot do that (place all k^d non-zeros on one dimension.)
- What about trying to recover a $k \times k \times ... \times k$ "hypercube" (or dimension-wise permutations of it)?
- Can easily do it!! Solve d ℓ_1-minimization problems sequentially!!

Back to compressed sensing

- When A is $n \times N$, random, we can (with high probability) recover all k-sparse vectors with $k \lesssim n/\log N$.
- How? ℓ_1-minimization!
- What about when A is a Kronecker product of several A_i's?
- No straightforward answer! In general, the restricted isometry constant of A is larger than the worst RIP constant of the A_i's.
- From now on let us assume that all A_i have the same dimensions and RIP constants.
- Bad news: This can at best guarantee the recovery of k-sparse vectors by ℓ_1 minimization
(terrible, because we have N^d entries in our "object")
- Ideally, we would like to recover k^d-sparse vectors.
- Cannot do that (place all k^d non-zeros on one dimension.)
- What about trying to recover a $k \times k \times ... \times k$ "hypercube" (or dimension-wise permutations of it)?
- Can easily do it!! Solve d ℓ_1-minimization problems sequentially!!

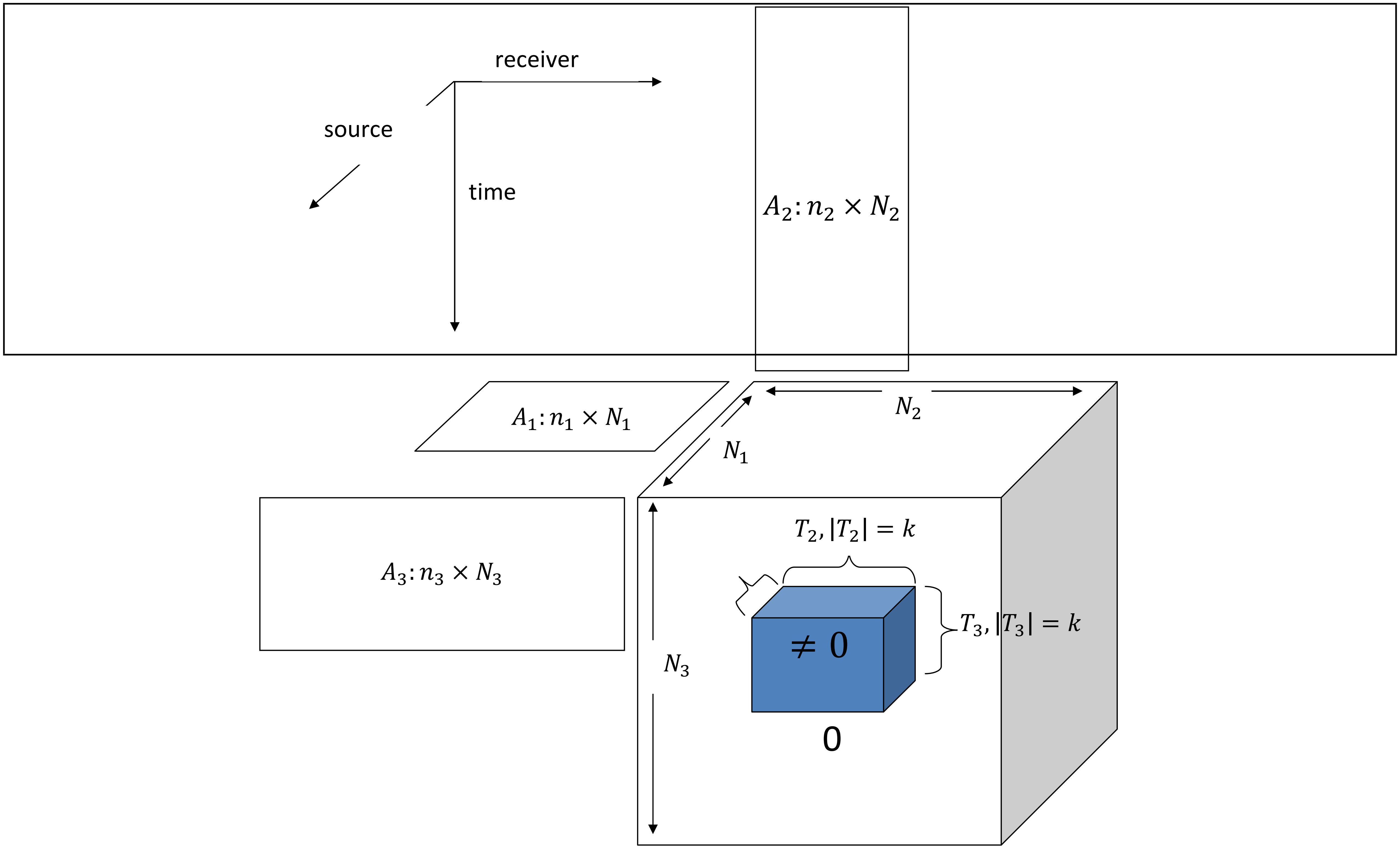
Back to compressed sensing

- When A is $n \times N$, random, we can (with high probability) recover all k-sparse vectors with $k \lesssim n/\log N$.
- How? ℓ_1-minimization!
- What about when A is a Kronecker product of several A_i's?
- No straightforward answer! In general, the restricted isometry constant of A is larger than the worst RIP constant of the A_i's.
- From now on let us assume that all A_i have the same dimensions and RIP constants.
- Bad news: This can at best guarantee the recovery of k-sparse vectors by ℓ_1 minimization
  (terrible, because we have N^d entries in our "object")
- Ideally, we would like to recover k^d-sparse vectors.
- Cannot do that (place all k^d non-zeros on one dimension.)
- What about trying to recover a $k \times k \times ... \times k$ "hypercube" (or dimension-wise permutations of it)?
- Can easily do it!! Solve d ℓ_1-minimization problems sequentially!!

Back to compressed sensing

- When A is $n \times N$, random, we can (with high probability) recover all k-sparse vectors with $k \lesssim n/\log N$.
- How? ℓ_1-minimization!
- What about when A is a Kronecker product of several A_i's?
- No straightforward answer! In general, the restricted isometry constant of A is larger than the worst RIP constant of the A_i's.
- From now on let us assume that all A_i have the same dimensions and RIP constants.
- Bad news: This can at best guarantee the recovery of k-sparse vectors by ℓ_1 minimization
  (terrible, because we have N^d entries in our "object")
- Ideally, we would like to recover k^d-sparse vectors.
- Cannot do that (place all k^d non-zeros on one dimension.)
- What about trying to recover a $k \times k \times ... \times k$ "hypercube" (or dimension-wise permutations of it)?
- Can easily do it!! Solve d ℓ_1-minimization problems sequentially!!

Compressed sensing in seismic acquisition



Kronecker compressed sensing

- Thus, there is an algorithm (involving many ℓ_1-minimization problems) that recovers the hypercube of side k, and all its permutations.
- Above algorithm works with high probability (on the draw of the matrices) for all such structured sparse supports .
- Not a practical algorithm...
- However, this suggests that we can hope to recover structured sparse supports of size k^d, eventhough some sets of size k^d cannot be recovered!!

Kronecker compressed sensing

- Thus, there is an algorithm (involving many ℓ_1-minimization problems) that recovers the hypercube of side k, and all its permutations.
- Above algorithm works with high probability (on the draw of the matrices) for all such structured sparse supports .
- Not a practical algorithm...
- However, this suggests that we can hope to recover structured sparse supports of size k^d, eventhough some sets of size k^d cannot be recovered!!

Kronecker compressed sensing

- Thus, there is an algorithm (involving many ℓ_1-minimization problems) that recovers the hypercube of side k, and all its permutations.
- Above algorithm works with high probability (on the draw of the matrices) for all such structured sparse supports .
- Not a practical algorithm...
- However, this suggests that we can hope to recover structured sparse supports of size k^d, eventhough some sets of size k^d cannot be recovered!!

Kronecker compressed sensing

- Thus, there is an algorithm (involving many ℓ_1-minimization problems) that recovers the hypercube of side k, and all its permutations.
- Above algorithm works with high probability (on the draw of the matrices) for all such structured sparse supports .
- Not a practical algorithm...
- However, this suggests that we can hope to recover structured sparse supports of size k^d, eventhough some sets of size k^d cannot be recovered!!

Kronecker compressed sensing

Proposition

Let A_i, $i \in \{1, ..., d\}$ be $n \times N$ Gaussian random matrices "with RIP constant δ_k". Suppose that X is supported on $\otimes_{i=1}^d T_i$ and denote by $x := X(:)$, then the following holds

$$(1-\delta_k)^d \|x\|_2^2 \le \|(\otimes_{i=1}^d A_i)x\|_2^2 \le (1+\delta_k)^d \|x\|_2^2.$$

If $\delta_k \le 0.25/d$, then $(1-1/3)\|x\|_2^2 \le \|(\otimes_{i=1}^d A_i)x\|_2^2 \le (1+1/3)\|x\|_2^2$.

Lemma

Let A_i, $i \in \{1, ..., d\}$ be $n \times N$ Gaussian random matrices with RIP constant $\delta_{k+1} \lesssim (k/n)^{1/2-\alpha} \le \frac{1}{4d}$ and let $A = \otimes_{i=1}^d A_i$. Denote by $\Omega = \otimes_{i=1}^d T_i$ then the following holds

$$\|A_\Omega^\dagger a_\ell\|_2 \lesssim (k/n)^{1/2-\alpha}.$$

Kronecker compressed sensing

Theorem

Let A and Ω be as above, and let $s = (s_j)_{j \in \Omega}$ be a (Rademacher/Bernoulli) random sign sequence. With high probability every vector x supported on Ω with sign pattern s is the unique solution to the optimization problem

$$\min \|v\|_1 \textit{ subject to } Av = Ax$$

.

- This shows that we can use ℓ_1 minimization on the whole system to recover structured sparse sets with sparsity k^d.
- Ongoing research: extend this to more general support set models ← experimental results indicate that this is true for more general support sets.
- Ongoing research: prove robustness to model-mismatch (compressible signals) and stability to noise.

Kronecker compressed sensing

Theorem

Let A and Ω be as above, and let $s = (s_j)_{j\in\Omega}$ be a (Rademacher/Bernoulli) random sign sequence. With high probability every vector x supported on Ω with sign pattern s is the unique solution to the optimization problem

$$\min \|v\|_1 \text{ subject to } Av = Ax$$

.

- This shows that we can use ℓ_1 minimization on the whole system to recover structured sparse sets with sparsity k^d.
- Ongoing research: extend this to more general support set models $\longleftarrow$ experimental results indicate that this is true for more general support sets.
- Ongoing research: prove robustness to model-mismatch (compressible signals) and stability to noise.

Kronecker compressed sensing

Theorem

Let A and Ω be as above, and let $s = (s_j)_{j \in \Omega}$ be a (Rademacher/Bernoulli) random sign sequence. With high probability every vector x supported on Ω with sign pattern s is the unique solution to the optimization problem

$$\min \|v\|_1 \text{ subject to } Av = Ax$$

.

- This shows that we can use ℓ_1 minimization on the whole system to recover structured sparse sets with sparsity k^d.
- Ongoing research: extend this to more general support set models $\longleftarrow$ experimental results indicate that this is true for more general support sets.
- Ongoing research: prove robustness to model-mismatch (compressible signals) and stability to noise.

Kronecker compressed sensing

Theorem

Let A and Ω be as above, and let $s = (s_j)_{j \in \Omega}$ be a (Rademacher/Bernoulli) random sign sequence. With high probability every vector x supported on Ω with sign pattern s is the unique solution to the optimization problem

$$\min \|v\|_1 \text{ subject to } Av = Ax$$

.

- This shows that we can use ℓ_1 minimization on the whole system to recover structured sparse sets with sparsity k^d.
- Ongoing research: extend this to more general support set models $\leftarrow$ experimental results indicate that this is true for more general support sets.
- Ongoing research: prove robustness to model-mismatch (compressible signals) and stability to noise.

Numerical experiments: synthetic examples

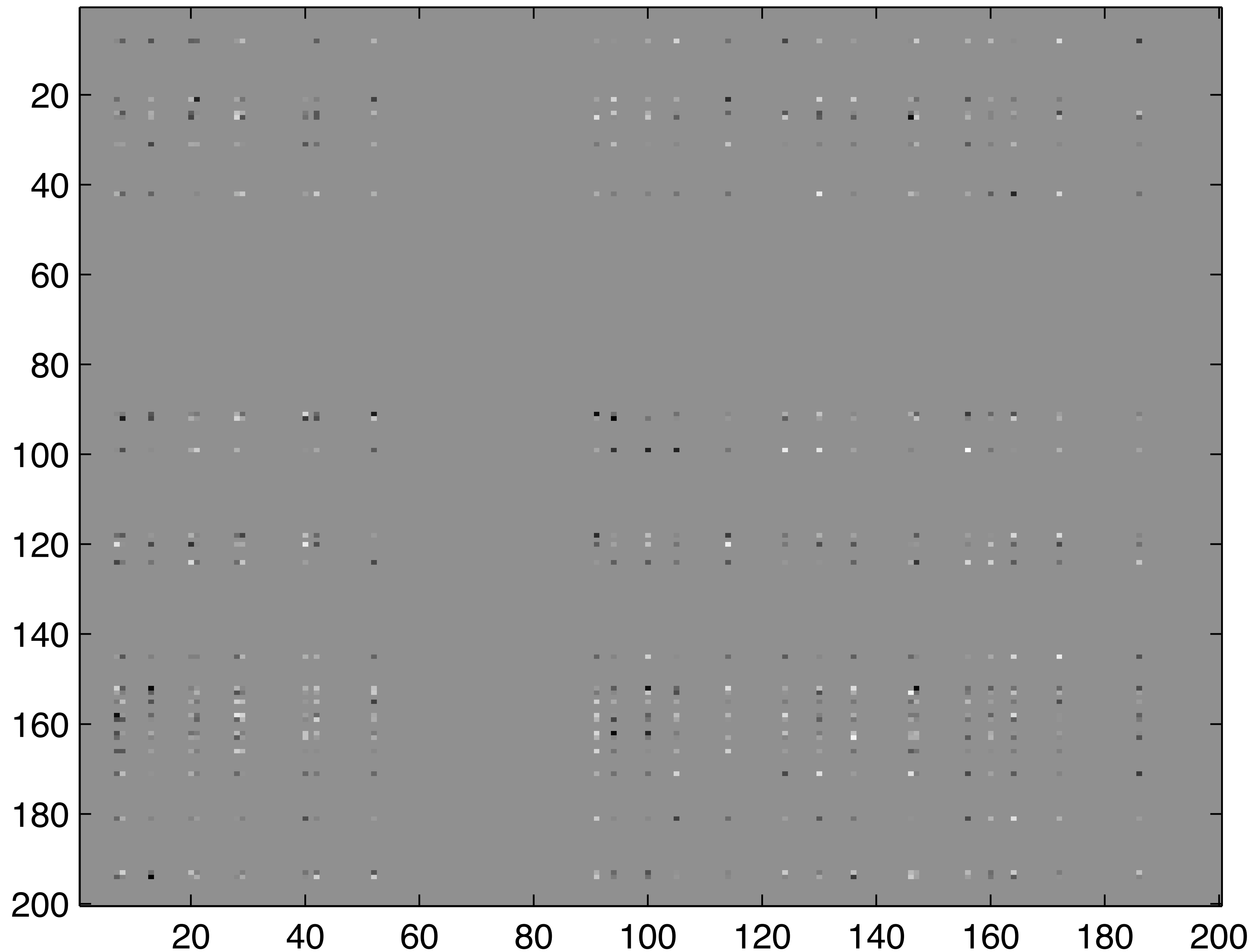


Figure: Original

Numerical experiments: synthetic examples

- 200×200 signal X
- signal is $k \times k$ sparse, $k = 25$, i.e., the supported is "structured".
- A_1, A_2 are Gaussian random 100×200 matrices
- $\implies$ reduction in data size by a factor of 4!!
- sparsity basis used: identity
- Using ℓ_1-minimization, reconstruction is exact!!

Numerical experiments: synthetic examples

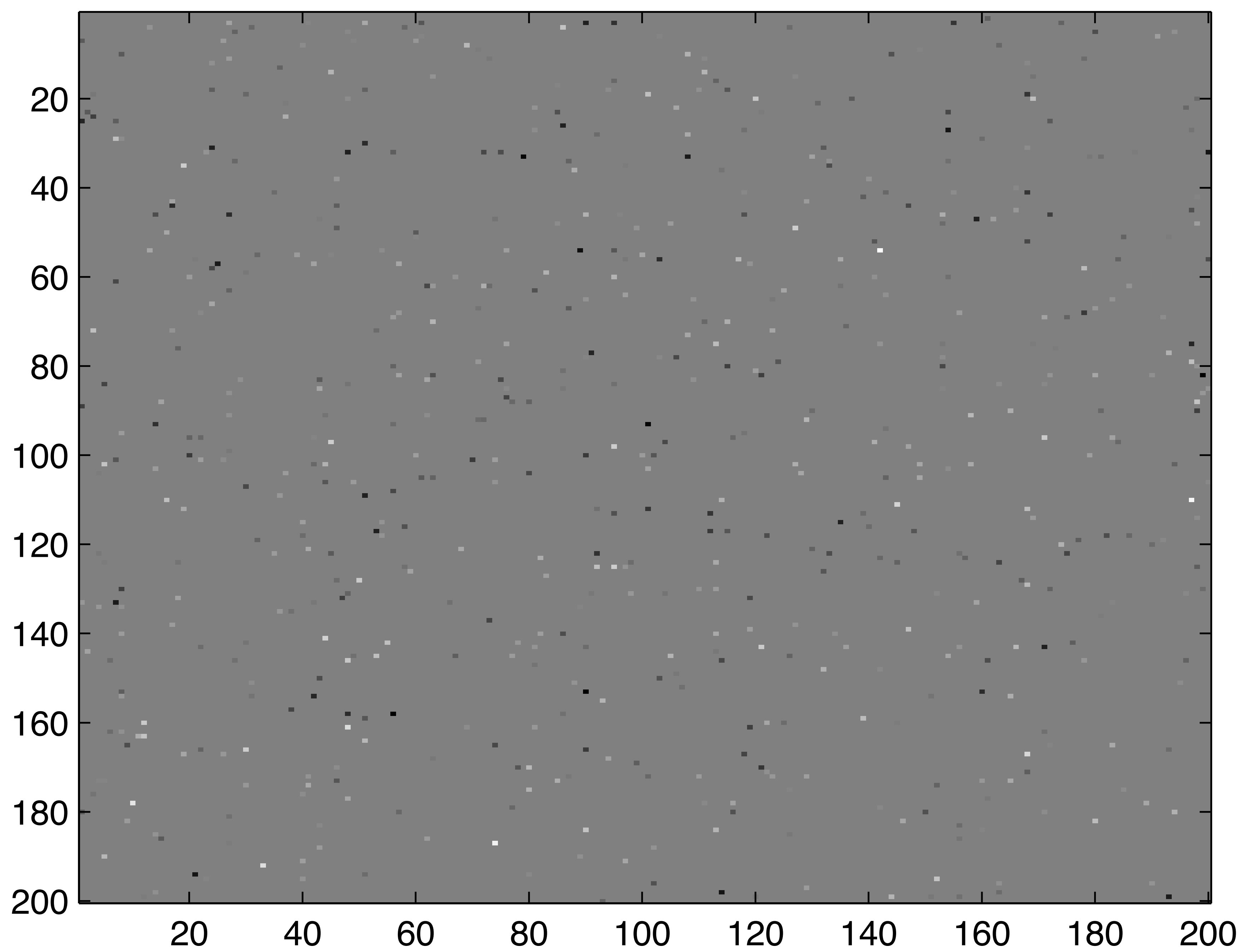


Figure: Original

Numerical experiments: synthetic examples

- 200×200 signal X
- signal is k^2 sparse, $k = 25$, i.e., the support is not "structured"
- A_1, A_2 are Gaussian random 100×200 matrices
- $\Longrightarrow$ reduction in data size by a factor of 4!!
- sparsity basis used: identity
- Using ℓ_1-minimization, reconstruction is exact!!
- It appears that adversarial sets are rare. Proof is an open problem, work in progress.

Numerical experiments: synthetic examples

- 200×200 signal X
- signal is k^2 sparse, $k = 25$, i.e., the support is not “structured”
- A_1, A_2 are Gaussian random 100×200 matrices
- $\Longrightarrow$ reduction in data size by a factor of 4!!
- sparsity basis used: identity
- Using ℓ_1-minimization, reconstruction is exact!!
- It appears that adversarial sets are rare. Proof is an open problem, work in progress.

Numerical experiments: seismic example

- $128 \times 128 \times 512$ data cube (long axis is time).
- A_1, A_2 are Gaussian random 64×128 matrices
- A_3 is the identity.
- $\implies$ reduction in data size by a factor of 4!!
- sparsity basis used: 2D-Curvelets $\otimes$ wavelets.
- Using ℓ_1-minimization, reconstruction SNR ~ 10.2 dB

Common-offset

y-axis: time in increments of 4ms
x-axis: source number

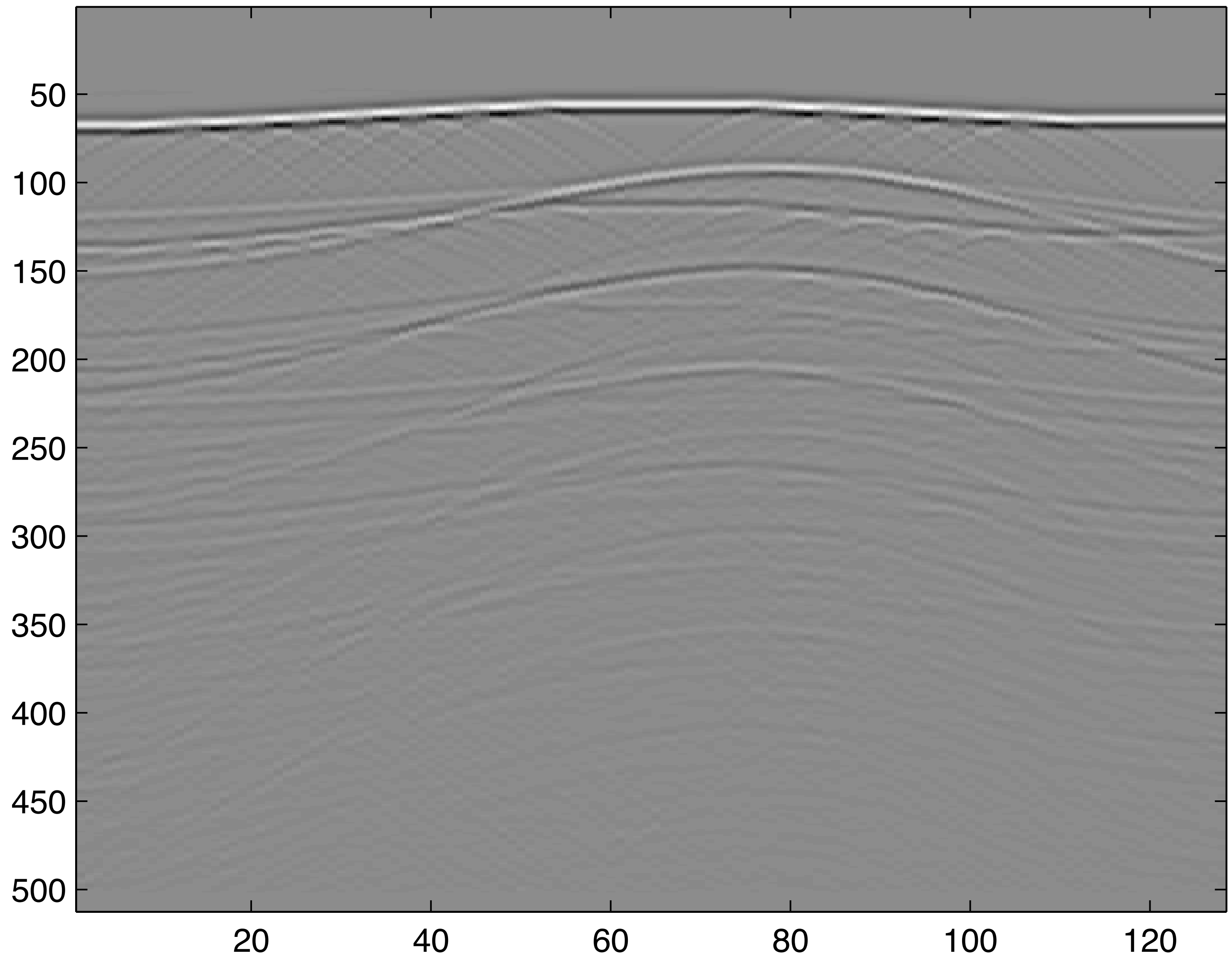


Figure: Original

Common-offset

y-axis: time in increments of 4ms
x-axis: source number

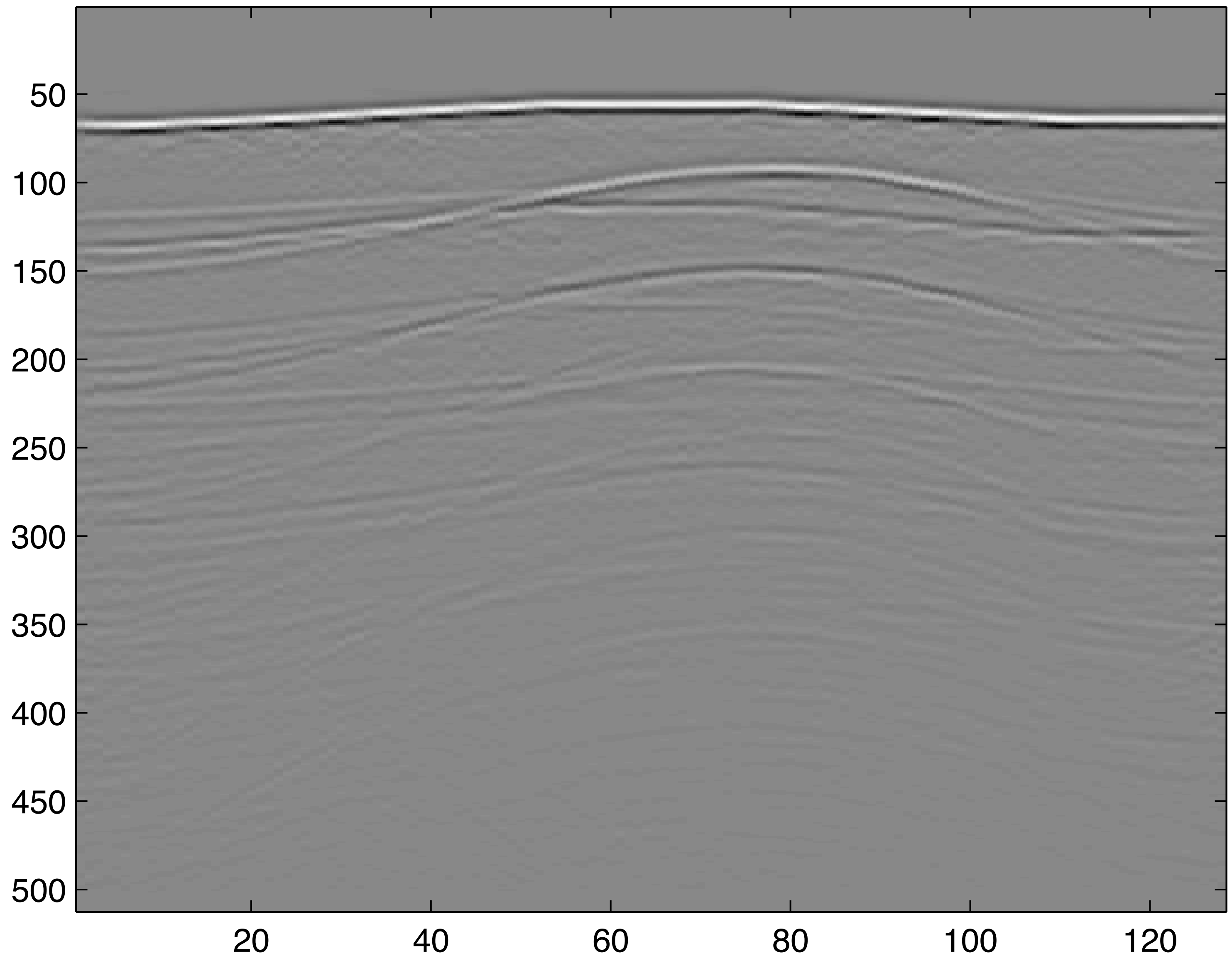


Figure: Reconstructed

Common-offset

y-axis: time in increments of 4ms
x-axis: source number

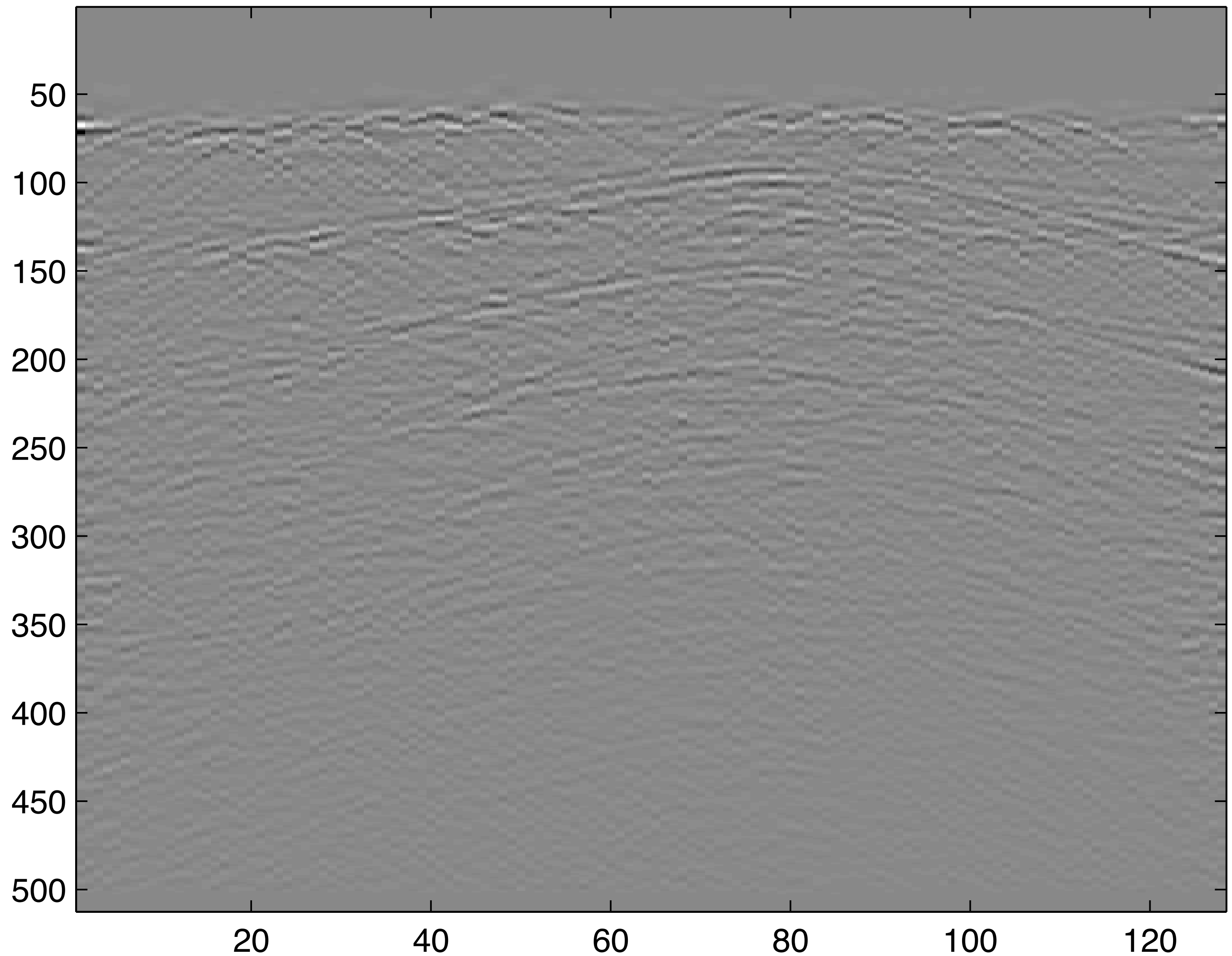


Figure: Difference

Shot record

y-axis: time in increments of 4ms
x-axis: source number

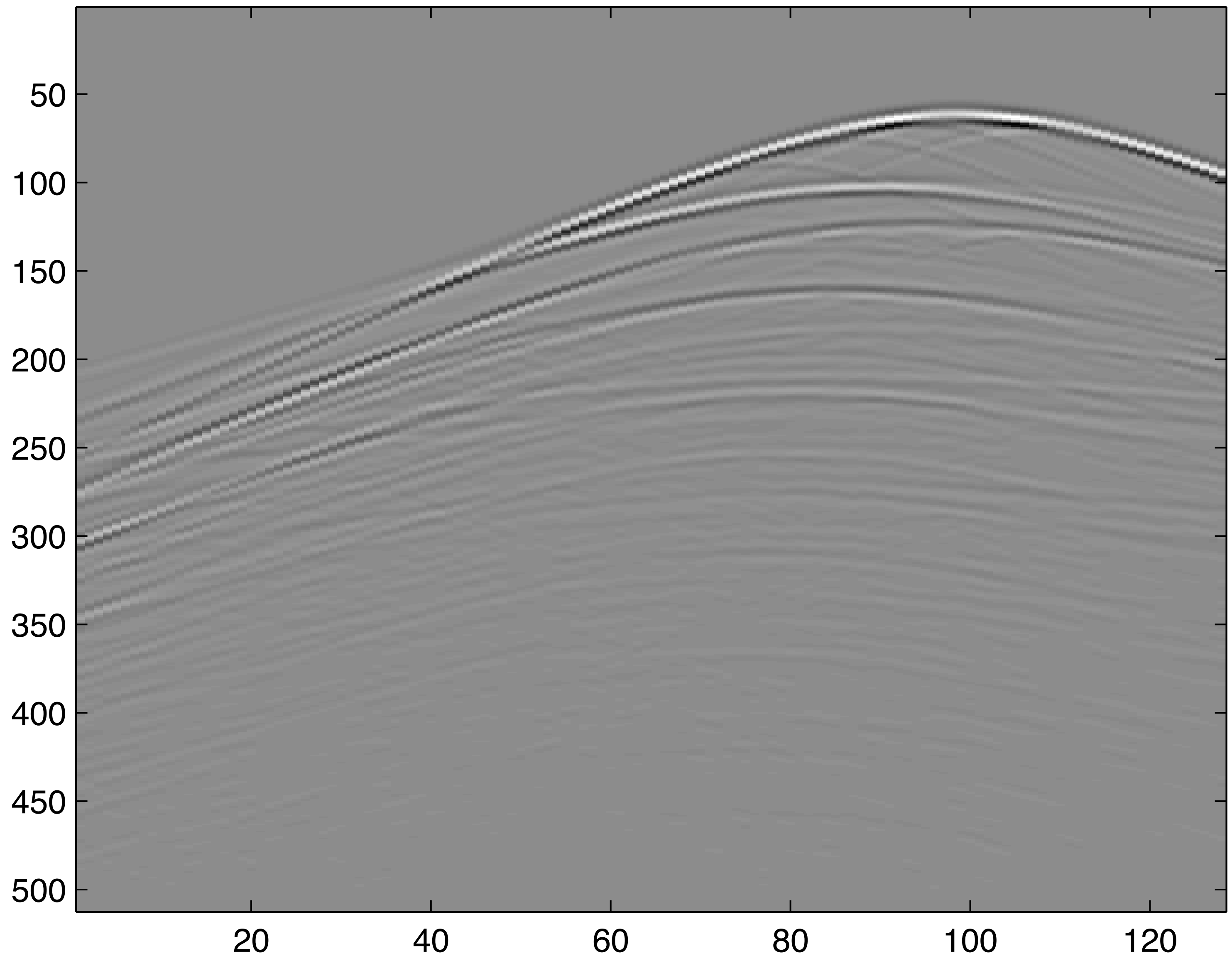


Figure: Original

Shot record

y-axis: time in increments of 4ms
x-axis: source number

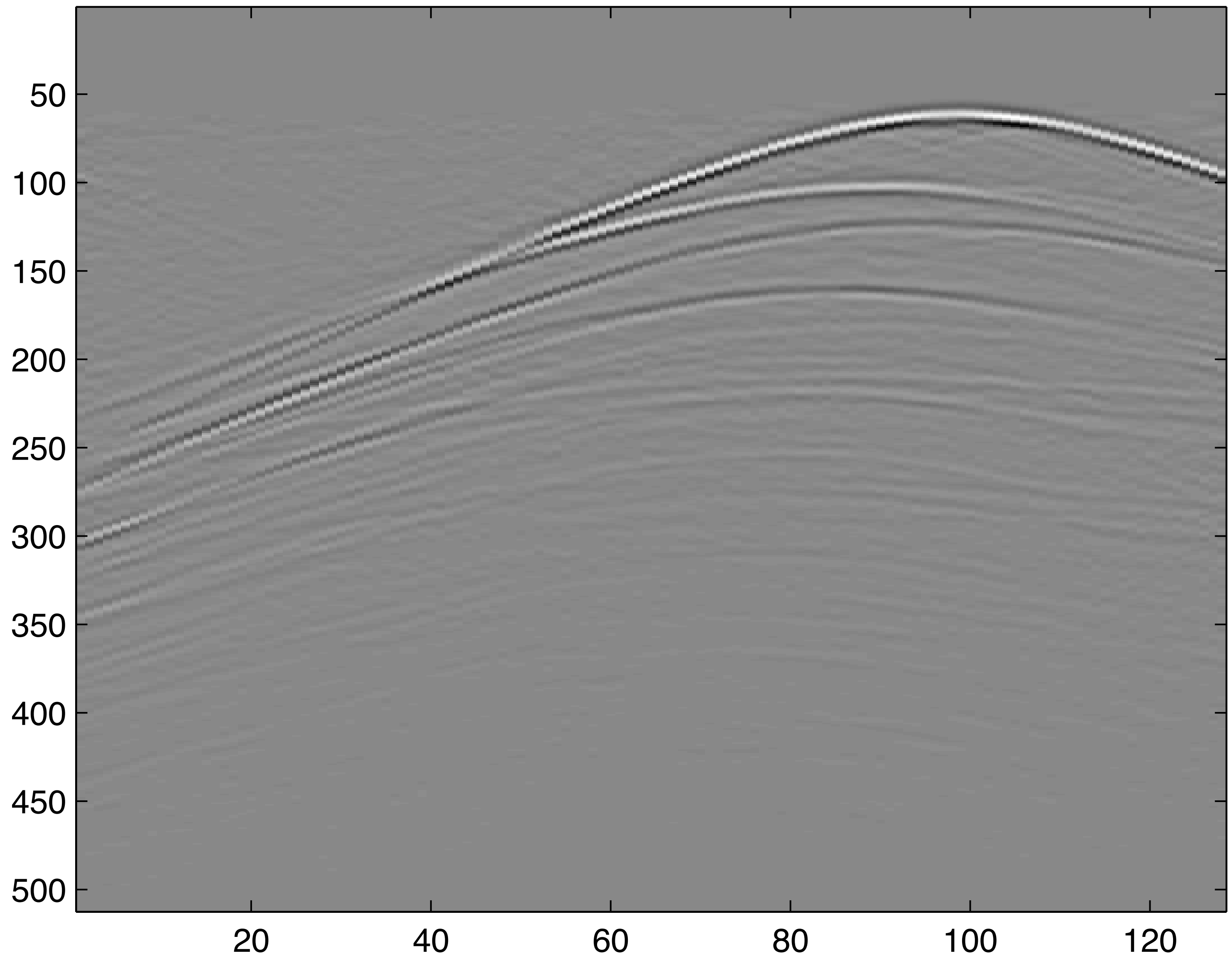


Figure: Reconstructed

Shot record

y-axis: time in increments of 4ms
x-axis: source number

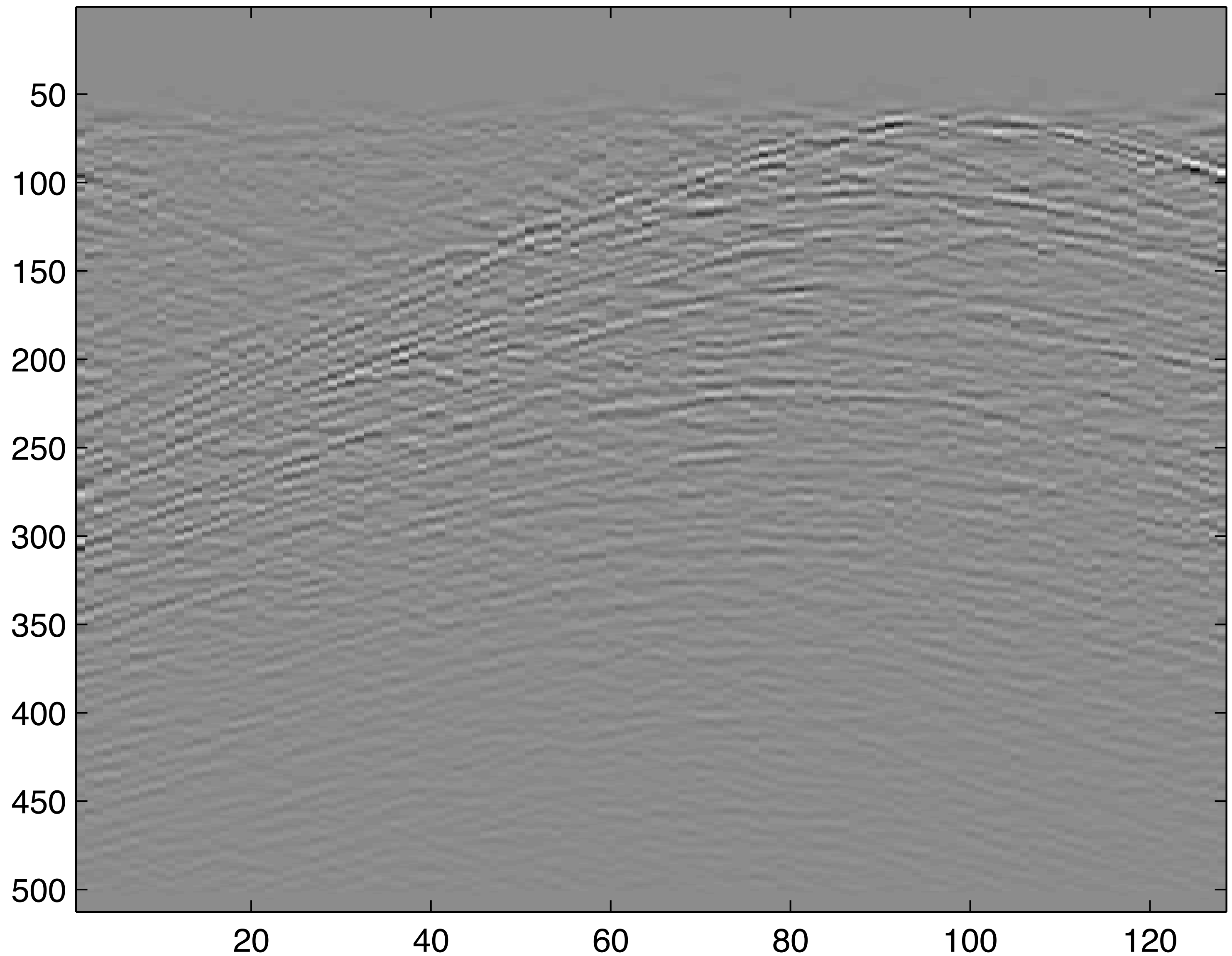


Figure: Difference

time slice, $t = 80 * 4ms$

y-axis: source number
x-axis: receiver number

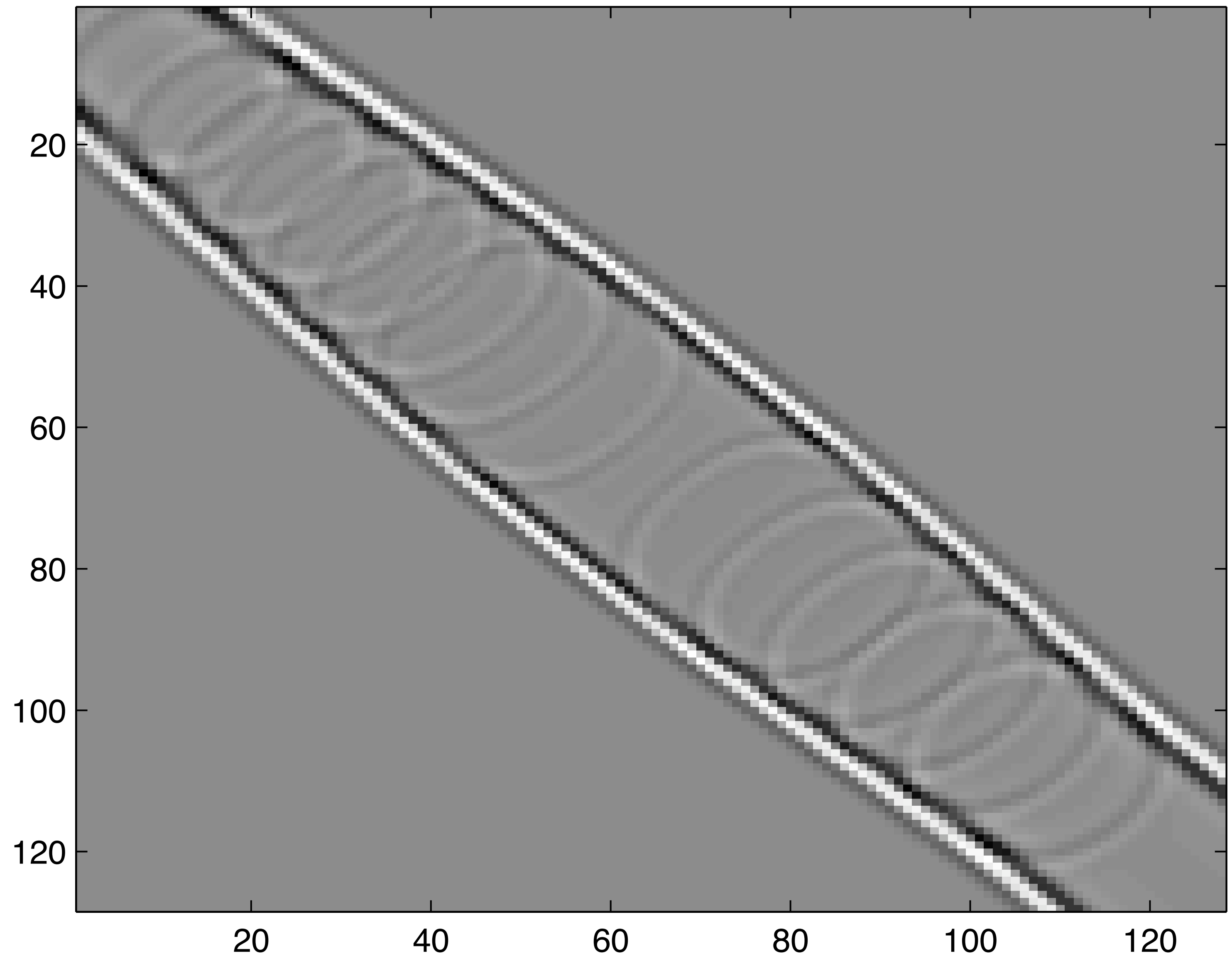


Figure: Original

time slice, $t = 80 * 4ms$

y-axis: source number
x-axis: receiver number

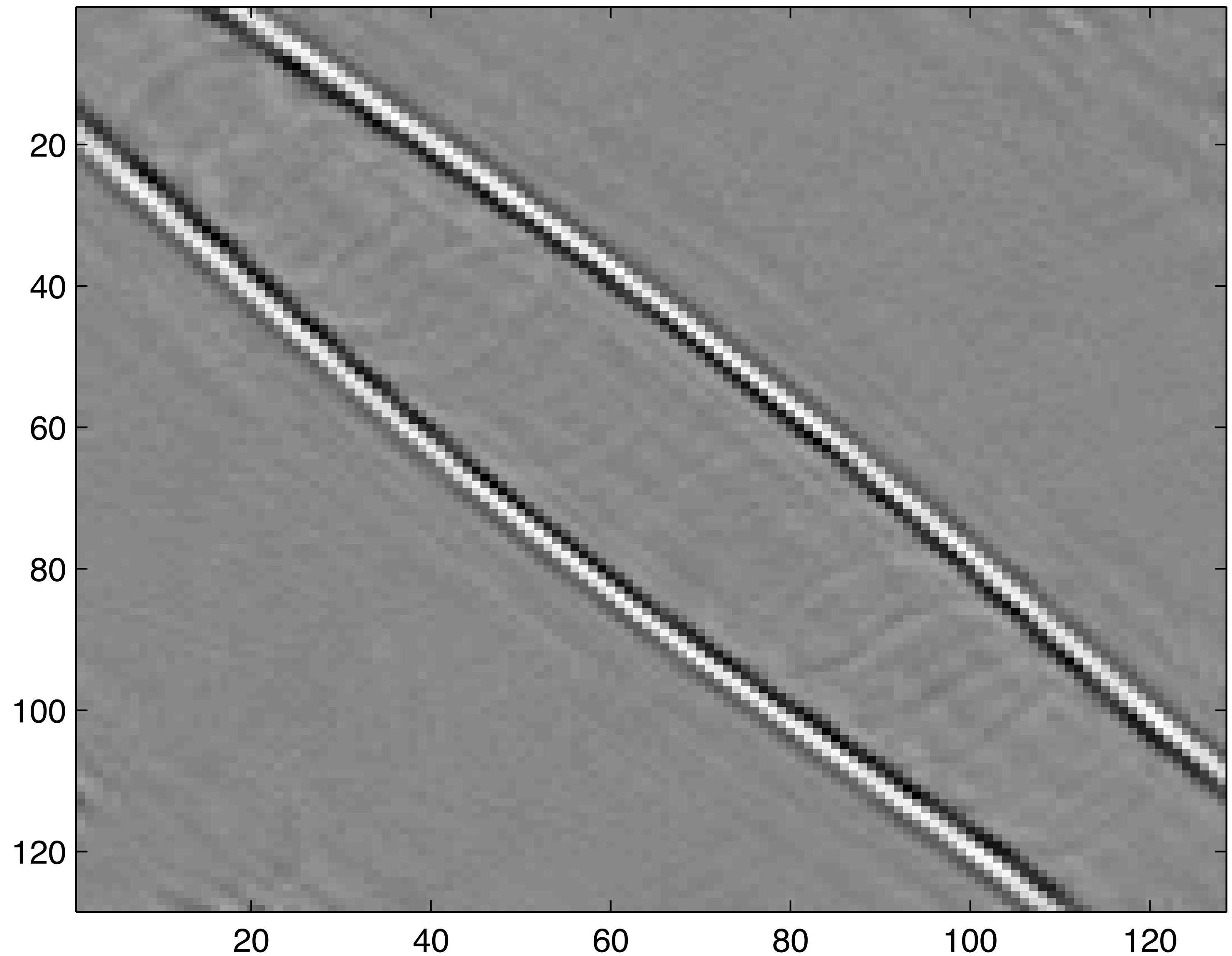


Figure: Reconstructed

time slice, $t = 80 * 4ms$

y-axis: source number
x-axis: receiver number

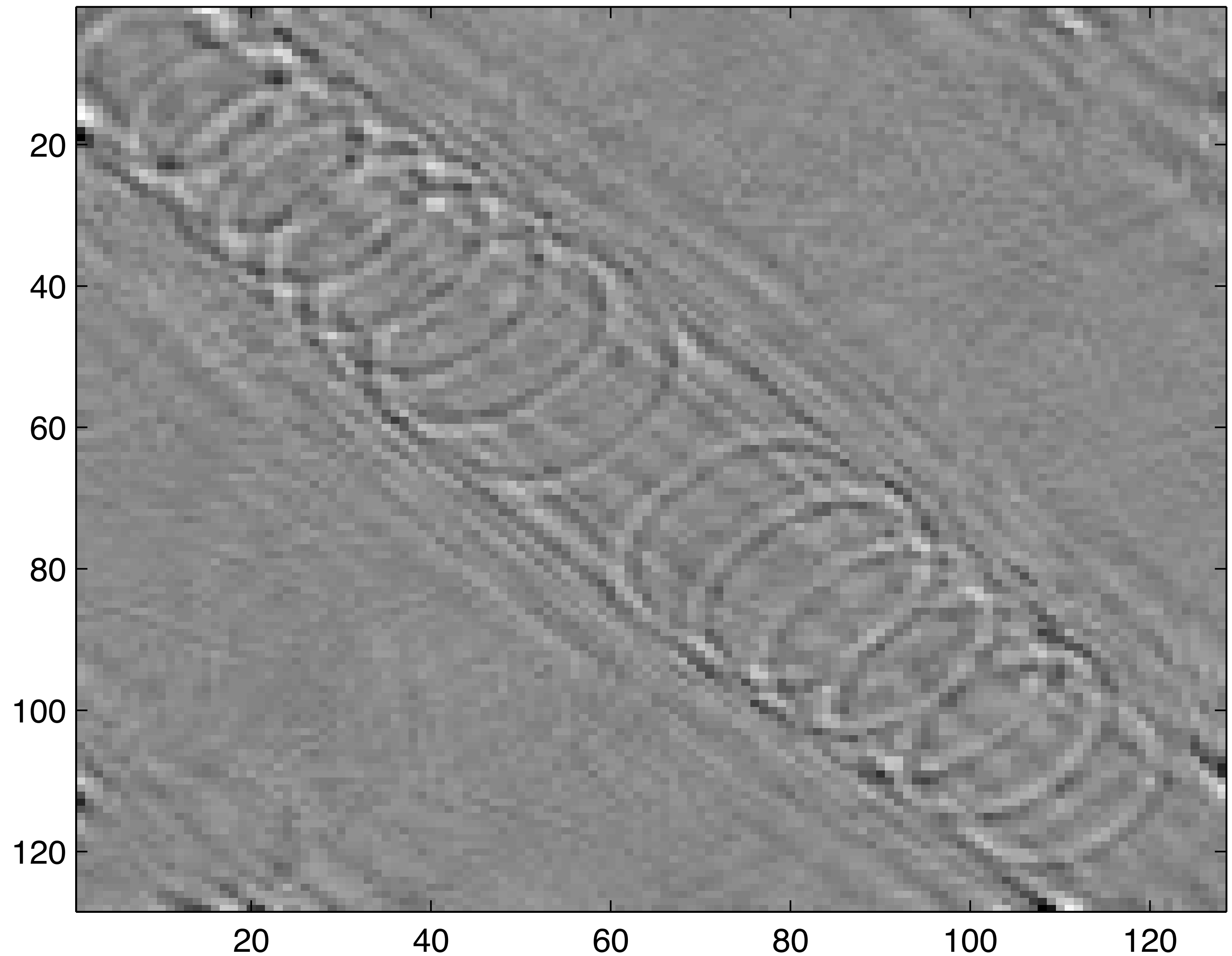


Figure: Difference

Thank you!

NSERC DNOISE II CRD