Scaling Network-based Spectrum Analyzer with Constant Communication Cost

Youngjune Gwon H. T. Kung

Presented at the 32nd IEEE International Conference on Computer Communications (INFOCOM'13) in Turin, Italy

April 16, 2013

Introduction

- Dynamic spectrum access (DSA) with cognitive radios
  - Alleviates inefficient spectrum allocation and licensing

- Accurate low-latency spectrum sensing most important to maximize DSA benefits

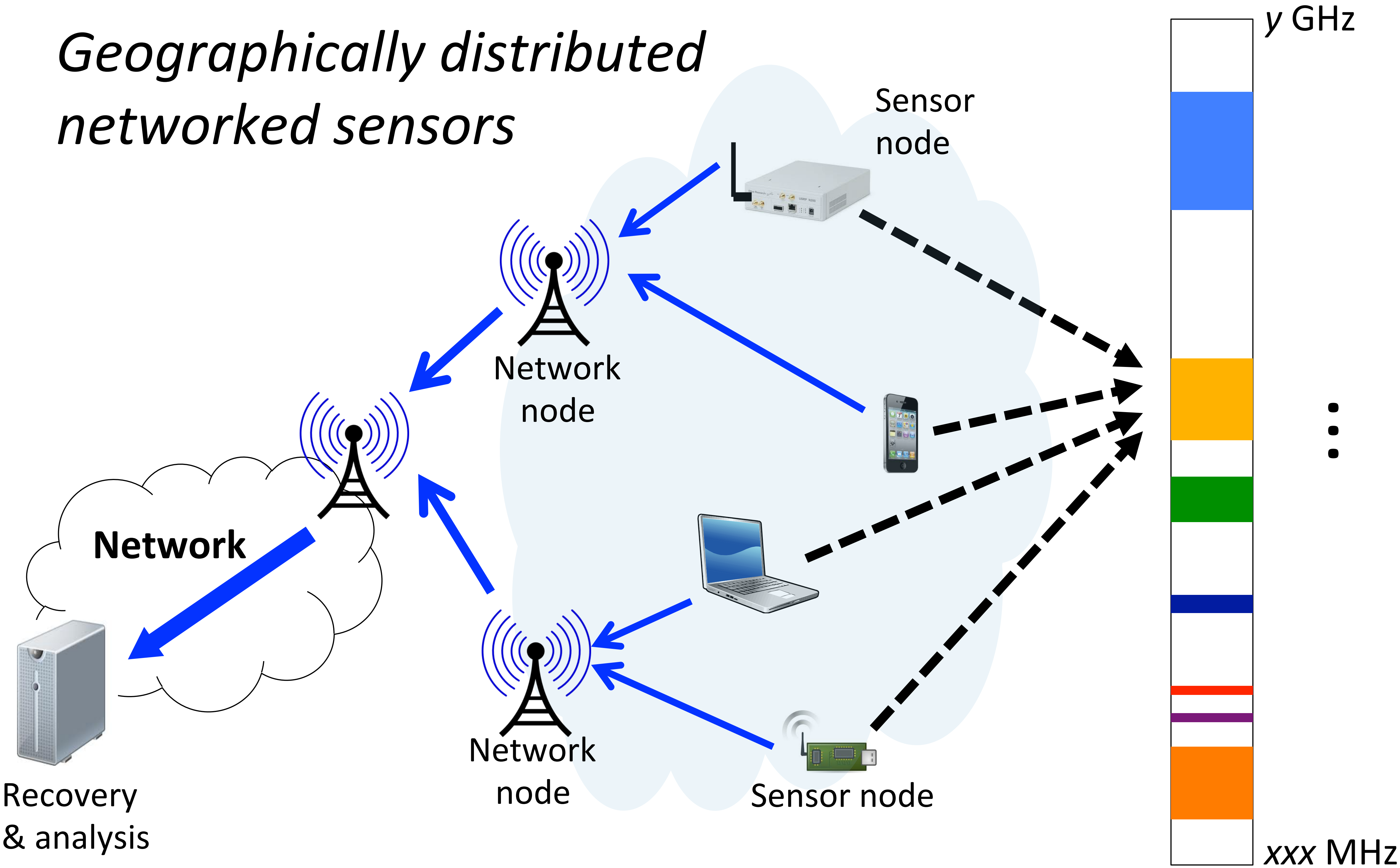
- Conventional spectrum analyzer
  - Can be ideal spectrum sensor
  - Measures amplitude of signals over *time* and converts to power magnitudes across *frequency*
  - FFT at the heart of modern spectrum analyzer equipment
    - Expense of FFT $\Longrightarrow$ true bottleneck is to keep up with Nyquist sampling
      - » *E.g.*, 1-MHz channel: $2\times10^6\times$*sample size* bps $\longrightarrow$ 40 Mbps (if 20-bit sample)

Network-based Spectrum Analysis

- Distributed spectrum sensing
  - Spectrum analyzer & measurers separate entities but networked
- Simple, *in situ* compression of measurement data at acquisition
  - Compressive sensing encode
- In-network processing of data
  - Combine multiple compressed measurements
- Recovery of original data
  - Undo in-network data processing & compressive sensing decode

Case for Distributed Spectrum Sensing (1)

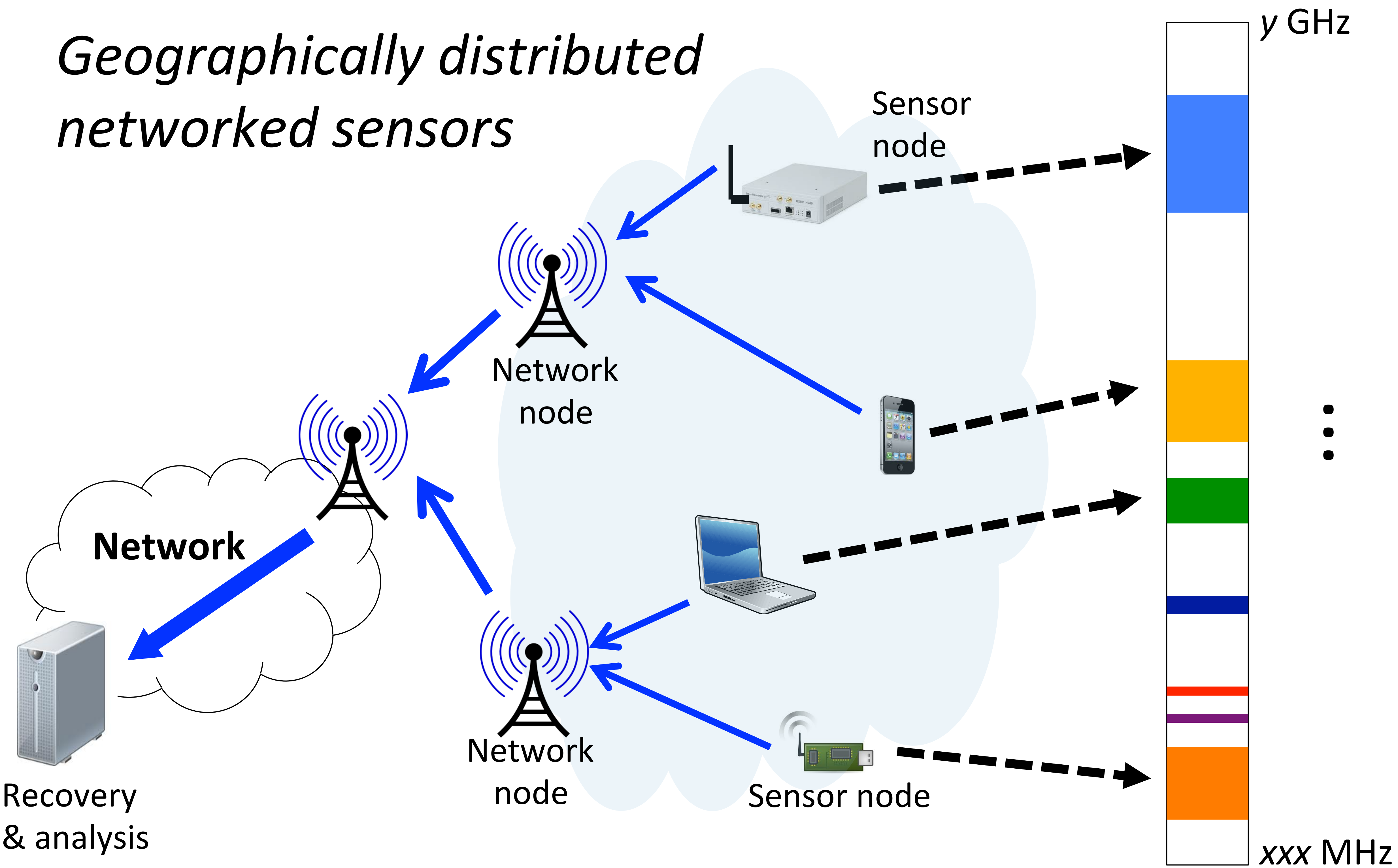
Geographically distributed networked sensors



Improve on spatial diversity $\Longrightarrow$ ***Not our Focus***

Case for Distributed Spectrum Sensing (2)

Geographically distributed networked sensors



Measure disjoint bands and aggregate ⟹ ***Our Focus***

Challenges

- Naïve FFT spectrum analysis bottlenecked by high data rate of Nyquist sampling
  - Use of multiple spectrum sensors each monitoring a sub-band, we have mitigated this problem

- How to minimize network communication cost of sensor measurements propagating network

- Fine-grained spectral analysis of wideband spectrum

Problem Statement

What is the size-reducing operation $\boldsymbol{\theta}$ that makes network-based analyzer feasible?

$$\arg\min_{\theta} \sum_{i=1}^{J} \dim(\mathbf{y}_i = \theta(\mathbf{x}_i)) \quad \text{s.t.} \ \|X(f_k) - \hat{X}(f_k)\|_{\ell_2} \leq \epsilon$$

J: # of partitions in the spectrum

$\mathbf{x}_i$: raw measured data from partition i

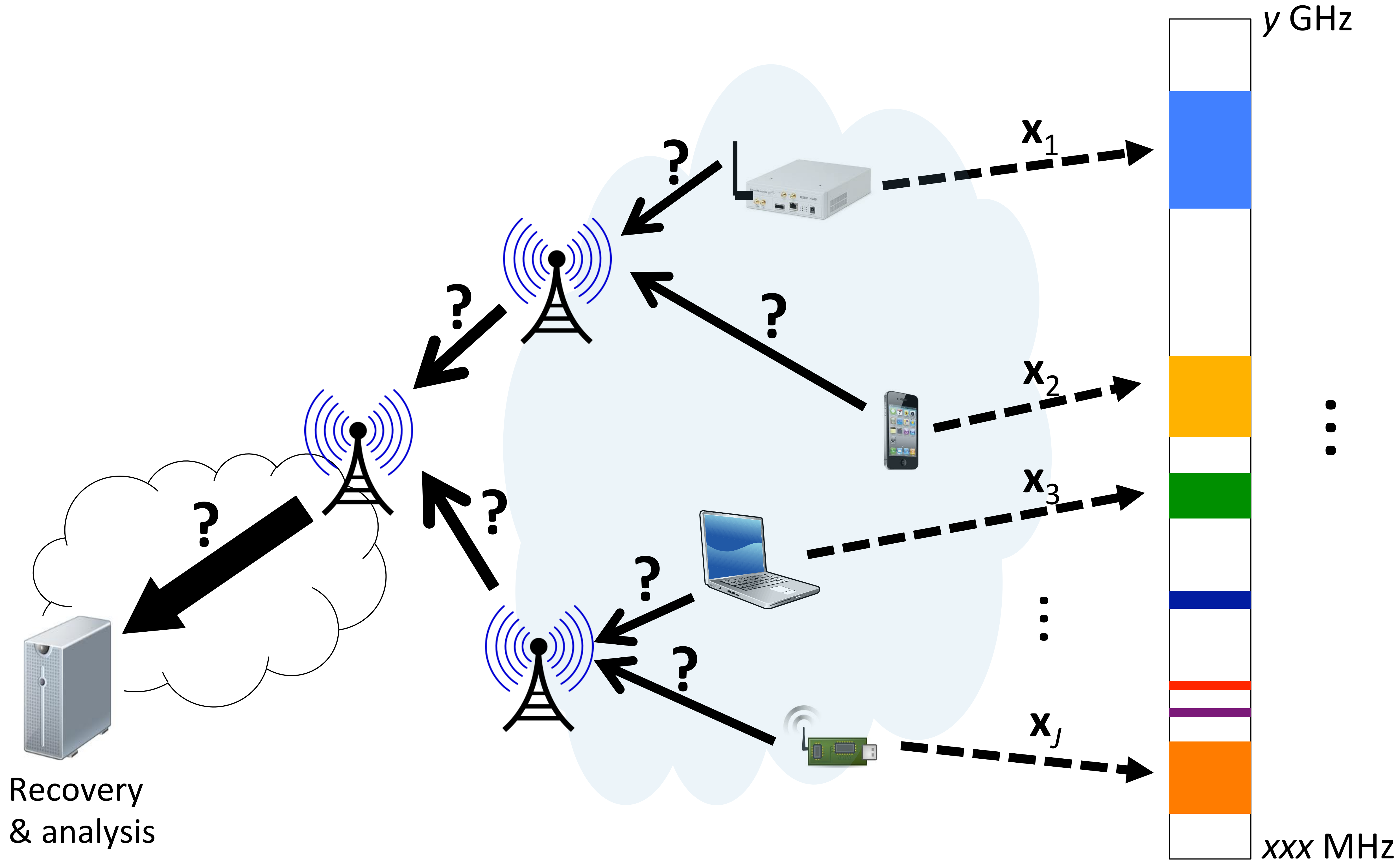
$\mathbf{y}_i$: compressive measurement of $\mathbf{x}_i$

X: frequency response of original $\mathbf{x} = \{\mathbf{x}_i\}_{\forall i}$

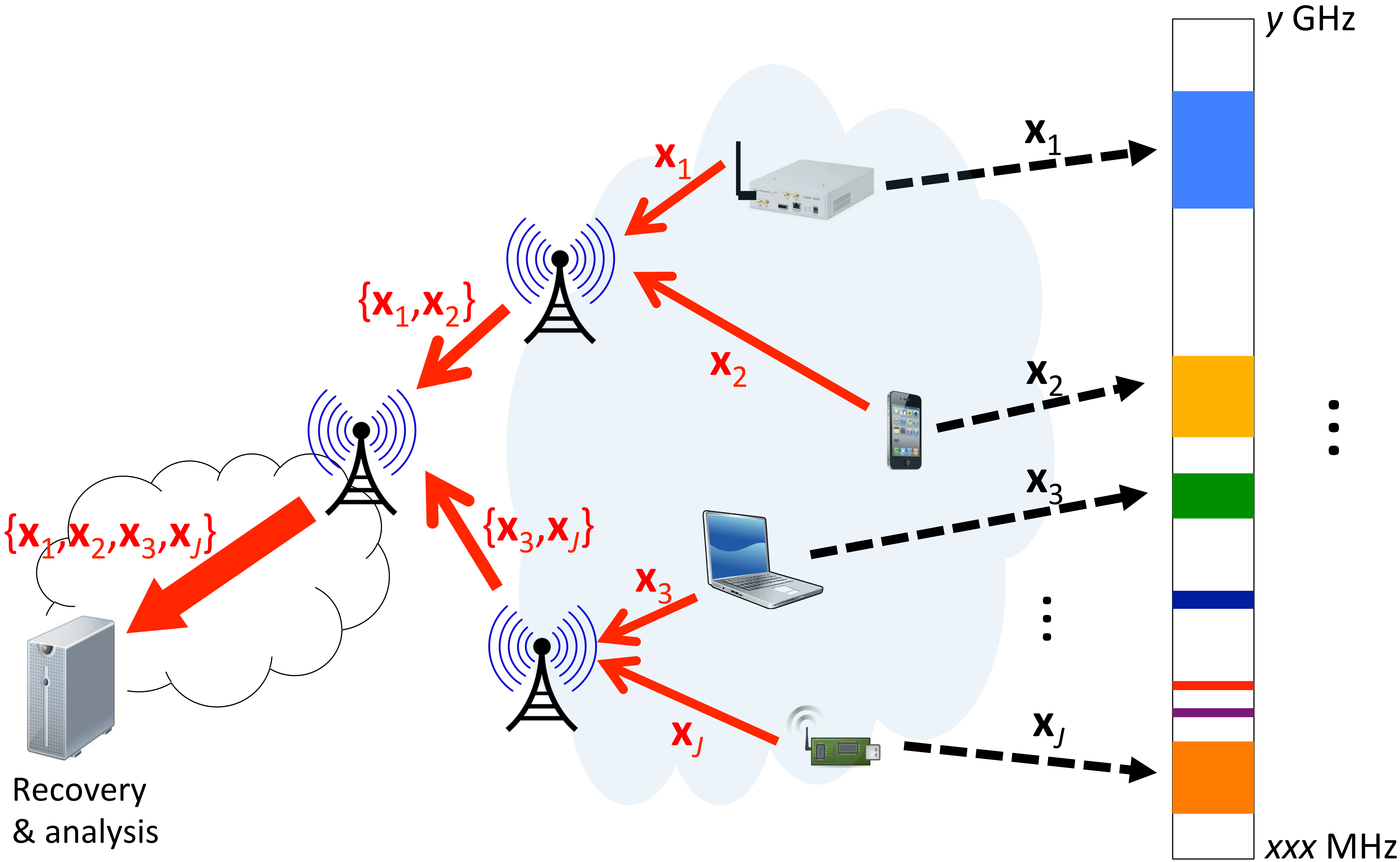
$\hat{X}$: frequency response of restored $\mathbf{x}$

ε: some small error requirement

Problem Illustrated

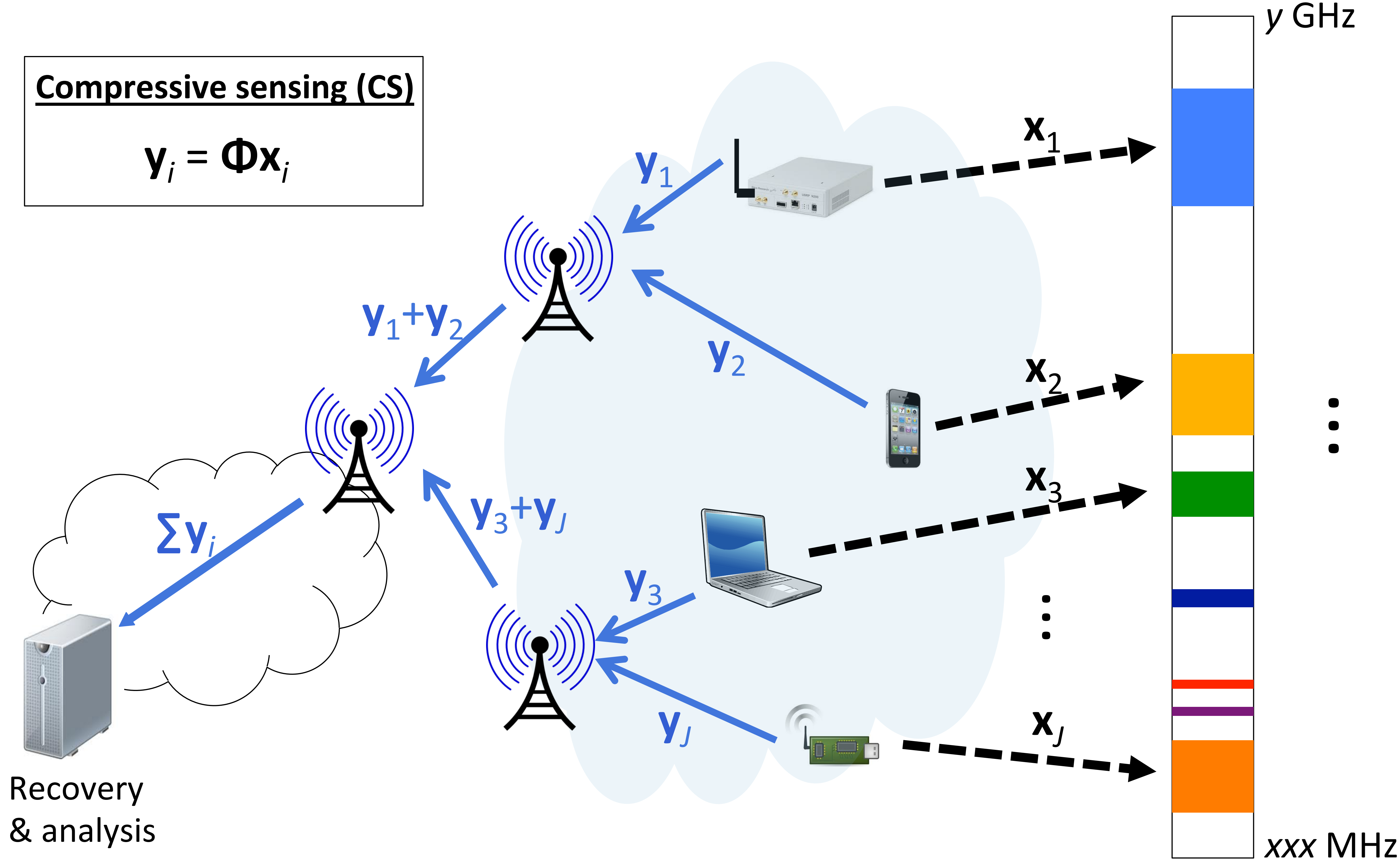


Solution Approach (1)



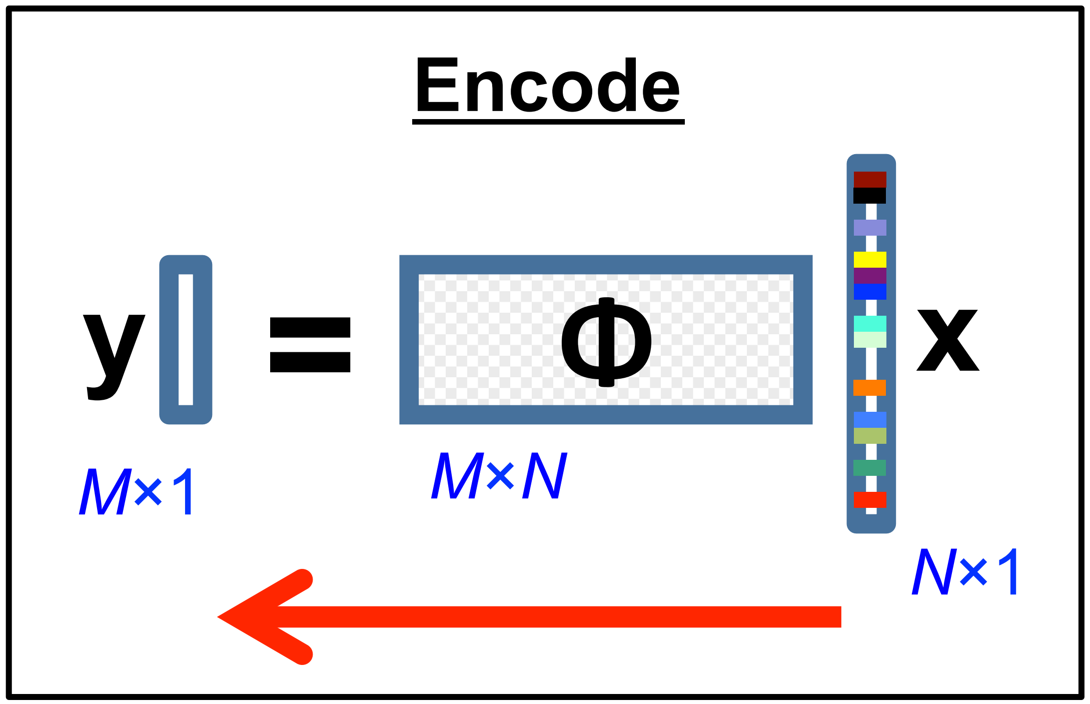
No in-network data reduction

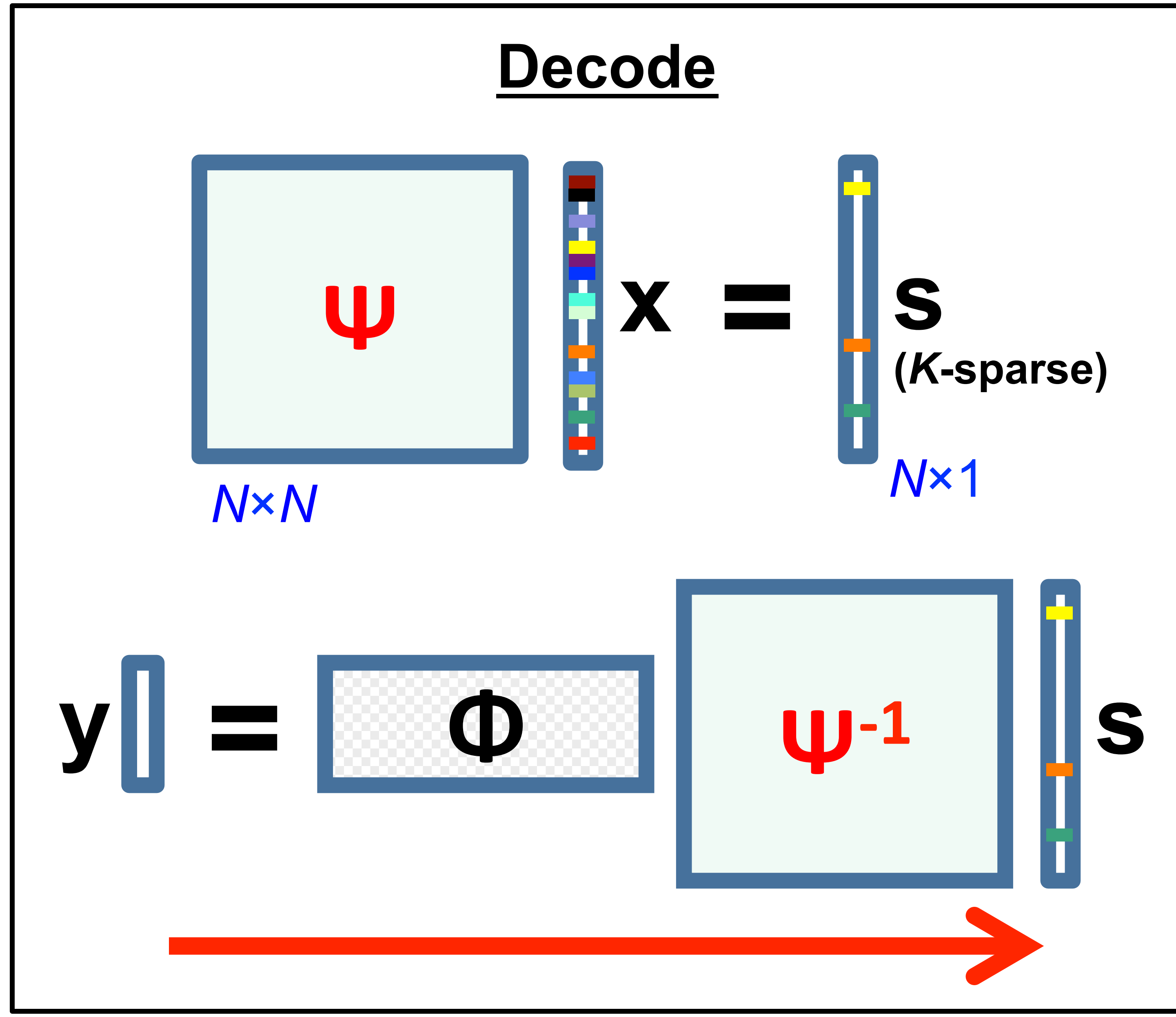
Solution Approach (2)



Compressive sensing (CS) encoding and ***in-network combining of compressed data***

Reminder: Compressive Sensing



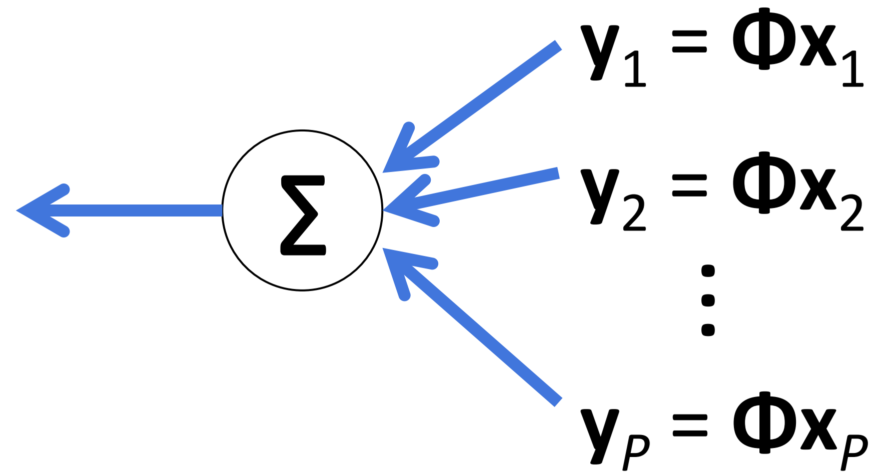


- Encode: $\mathbb{C}^N \longrightarrow \mathbb{C}^M$
  - Simple & data-blind $\Longrightarrow$ N:M compression (M << N) for sparse signal
- Decode
  - Available sparsifying basis (**Ψ**) determines $M \geq c \cdot K \cdot \log(N/K)$
    - Sparsity K revealed by **Ψ**
  - L1-minimization (*e.g.*, linear programming): $\min \|\mathbf{s}\|_1$ s.t. $\mathbf{y} = \mathbf{\Phi}\mathbf{\Psi}^{-1}\mathbf{s}$

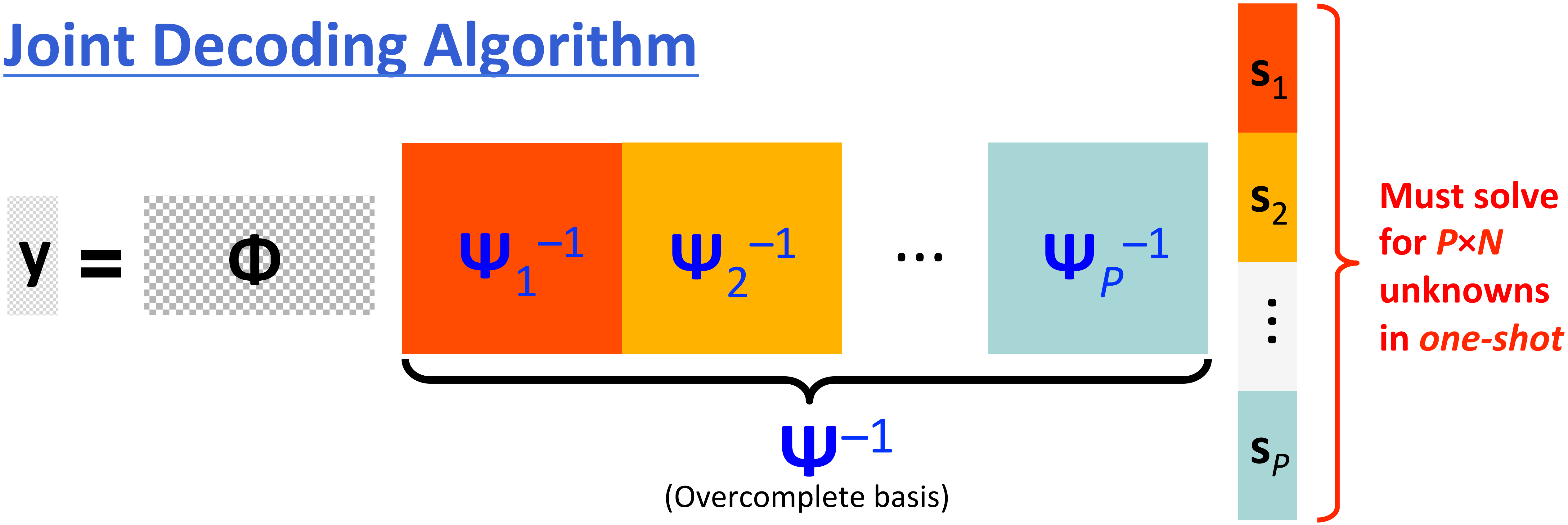
How to Separate Sum of Compressed Measurements?

<u>Generalized *P*-way sum</u>

$$\mathbf{y} = \mathbf{\Phi x}_1 + \mathbf{\Phi x}_2 + \ldots + \mathbf{\Phi x}_P$$



Joint Decoding Algorithm



Compressive sensing decode on $\mathbf{y} = (\mathbf{\Phi}\, \Psi^{-1})\, \mathbf{s}$ to solve for $\mathbf{s}_1, \mathbf{s}_2, \ldots, \mathbf{s}_P$ jointly

Can we do better?

Initial Approximation by Least Squares

$$\boldsymbol{y} = \boldsymbol{\Phi} \, [\tilde{\boldsymbol{Q}}_1 \; \tilde{\boldsymbol{Q}}_2 \; \cdots \; \tilde{\boldsymbol{Q}}_P] \begin{bmatrix} \tilde{\boldsymbol{s}}_1 \\ \tilde{\boldsymbol{s}}_2 \\ \vdots \\ \tilde{\boldsymbol{s}}_P \end{bmatrix}$$

$$\begin{bmatrix} | \; | & 0 & | & 0 & | & 0 \end{bmatrix}$$

Keep only several ***leading*** eigenvectors of $\mathbf{Q}_i$

Equivalent to keeping variables at ***leading*** positions only (others are multiplied by zeros)

- <u>Require</u>
  - $\{\mathbf{Q}_1, \mathbf{Q}_2, \ldots, \mathbf{Q}_P\} \triangleq$ *distinct* sparsifying bases for each channel
    - $\mathbf{Q}$ can be estimated from $\mathbf{R}_x = \mathbb{E}[\mathbf{x}\mathbf{x}^H] = \mathbf{Q}\boldsymbol{\Lambda}\mathbf{Q}^H$
- Leading components have largest eigenvalues
- Remove non-leading components until we have ***overdetermined*** system
  - More equations than unknowns: dim($\mathbf{y}$) > # of unknowns
  - Least squares does this job well

Iterative Refinement by CS Decode

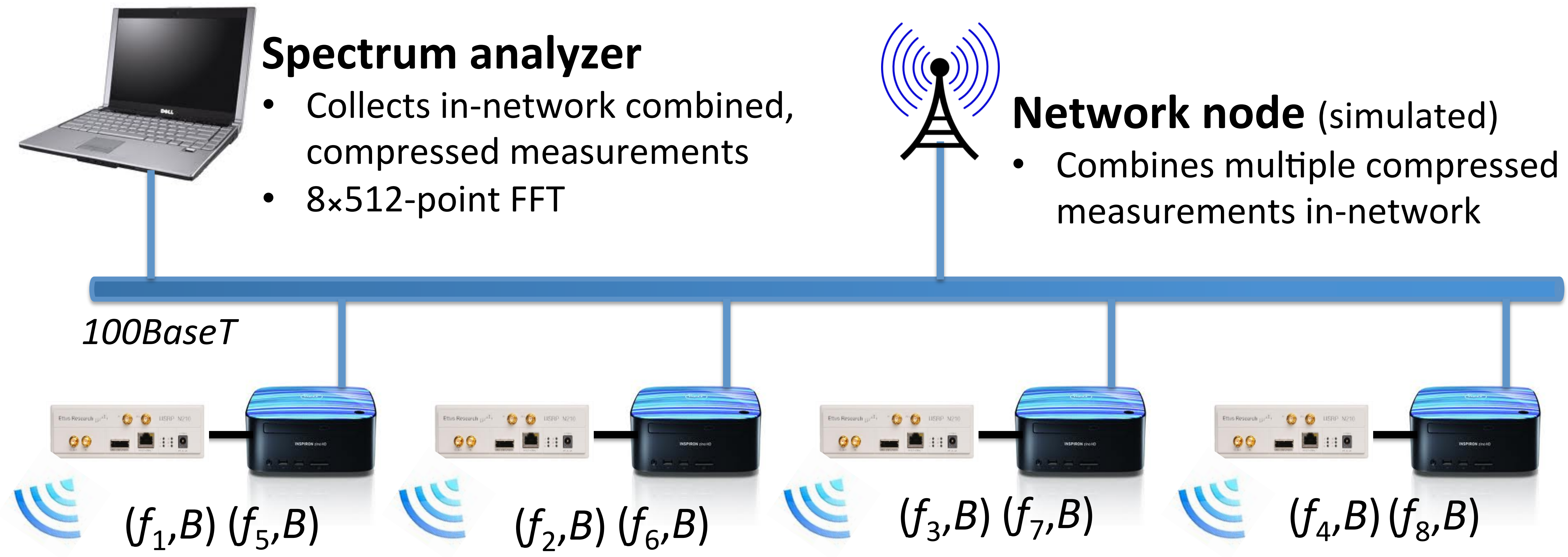
$$\mathbf{y} = \mathbf{\Phi}\left[\mathbf{Q}_1\ \tilde{\mathbf{Q}}_2\ \cdots\ \tilde{\mathbf{Q}}_P\right]\begin{bmatrix} \boldsymbol{s}_1 \\ \hat{\tilde{\boldsymbol{s}}}_2 \\ \vdots \\ \hat{\tilde{\boldsymbol{s}}}_P \end{bmatrix}$$

Relax $\mathbf{s}_1$

1. Back-substitute: $\mathbf{y}' = \mathbf{y} - \mathbf{\Phi}[\tilde{\mathbf{Q}}_2\hat{\tilde{\mathbf{s}}}_2 + \ldots + \tilde{\mathbf{Q}}_P\hat{\tilde{\mathbf{s}}}_P]$ using initial approximates of $\mathbf{s}_2, \ldots, \mathbf{s}_P$
2. Do compressive sensing decode with $\mathbf{y}' = (\mathbf{\Phi}\mathbf{Q}_1)\mathbf{s}_1$ to obtain refined $\mathbf{s}_1$

- Compressive sensing decodes ***underdetermined*** system
  - More unknowns than equations
- Relax $\mathbf{s}_i$'s in *descending* order of their L1-norm
  - Compressive sensing works better on ***largest-first decoding principle***
  - No need to solve for more than *N* unknowns at once
    - $N \triangleq$ length of original, uncompressed measurements ($\mathbf{x}_i$'s) on channel i
- Can be repeated in another stage

Evaluation in Lab Testbed of SW-defined Radios



Four **sensor nodes** (USRP2/USRP-N200) with WBX RF daughterboards

- Measure 8 channels from UHF white space
  f_i = {512.5, 537.5, 562.5, 587.5, 612.5, 637.5, 662.5, 687.5} MHz
- Each channel with B = 25 MHz bandwidth

Some Details

- Sensing & recovery methods
  1. Compressive sensing only (no combining)
  2. *P*-way combined compressed measurements for *P* = 2, 4, 8

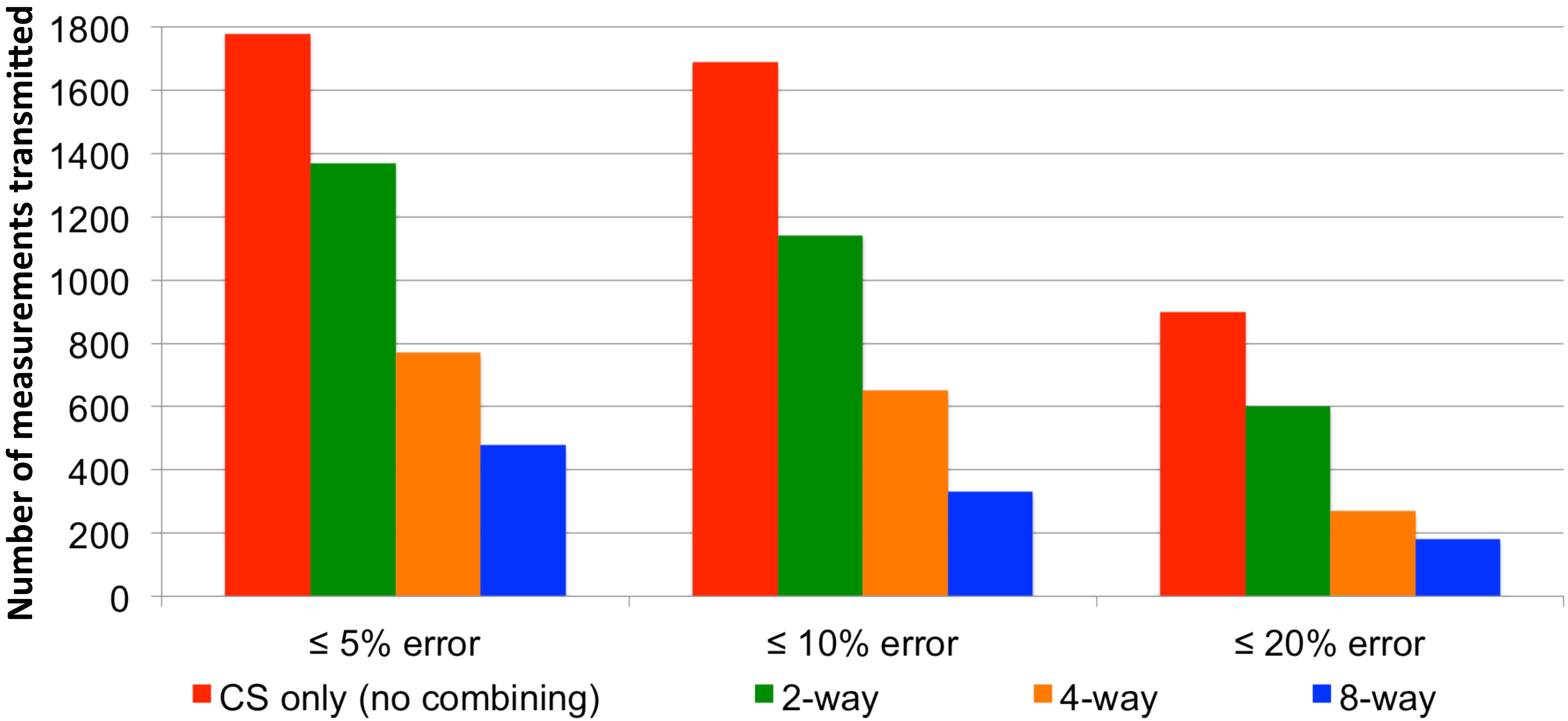
- M = # of compressed measurements (per channel)
  - Varied from 26 (20x compression) to 308 (1.67x)

- Error metric

$$\xi = \frac{1}{L} \sum_{k=1}^{L} \frac{\|X(f_k) - \hat{X}(f_k)\|_{\ell_2}}{\|X(f_k)\|_{\ell_2}}$$

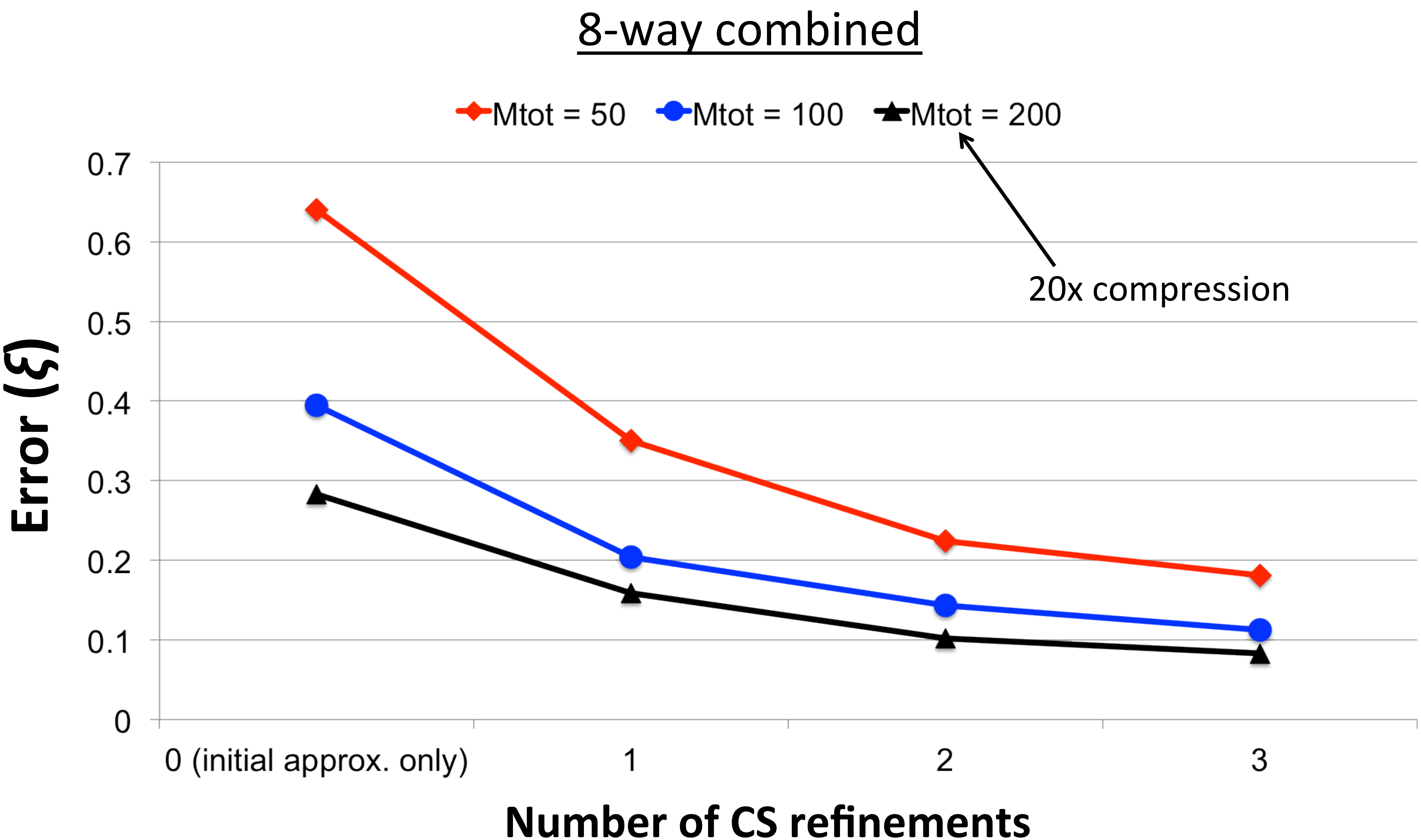
  - Average normalized frequency response error per sample
  - $L = 8 \times 512 = 4096$
  - $f_k \in [500, 700)$ MHz

Error Performance



- Total # of measurements transmitted ∝ communication cost
- *P*-way in-network combining could reduce measurements up to *P*-fold
  - Given error budget, # of measurements can remain constant until some limit
  - This limit depends on ***sparsity*** of channels in spectrum
- Proposed algorithm achieves similar accuracy performance as joint decoding while requiring *P* times less unknowns to solve concurrently

Improvement at Refinement Stages



- Error improvement more significant with smaller M_{tot}
- Small gain on accuracy after 2 stages

Summary

- Network-based spectrum analysis
  - Distributed spectrum sensors employed by distant analyzer operate over network
  - Key is to overcome network communication cost to move spectrum measurements

- Our approach
  - Compressive sensing encoding at sensor nodes
  - Simple in-network summing of multiple compressed measurements to further reduce overhead at network nodes

- New recovery algorithm
  - Least squares on leading principal components to separate individual measurements from the sum
  - Iterative relaxation by compressive sensing decode on each individual data

- **Conclusion: sensors can be added without additional communication cost**
  - Hold true until some limit determined by sparsity
  - Sparsity = true measure for channel information content

Supporting Materials

Remark on Sparsity and Discretization

- Discrete measurements performed by sensors ***preserve*** or ***bring out*** sparsity of original signal in frequency domain or in a custom basis
  - This is fundamental premise of our approach

- Design of better sparsity-inducing discretization schemes is challenging but can hugely enhance our approach

CS Recovery of Complex Signals

- $y = \Phi x = (\Phi\Psi^{-1})x$
  - $x = N\times 1$ *complex*-valued
  - $y = M\times 1$ *complex*-valued
  - $\Phi = M\times N$ *real*-valued
  - $\Psi = N\times N$ *complex*-valued
- $y = \Phi[x_R + j\cdot x_I] = \Phi\Psi^{-1}[X_R + j\cdot X_I] = \Phi[\Psi_R^{-1} + j\cdot\Psi_I^{-1}][X_R + j\cdot X_I]$
  - j = sqrt(-1)
- We want: $y' = AX'$

$$y' = \begin{bmatrix} y_R \\ y_I \end{bmatrix} \qquad A = \begin{bmatrix} \Phi\Psi_R^{-1} & -\Phi\Psi_I^{-1} \\ \Phi\Psi_I^{-1} & \Phi\Psi_R^{-1} \end{bmatrix} \qquad X' = \begin{bmatrix} X_R \\ X_I \end{bmatrix}$$

Decode y′ = AX′

$$\underset{2M\times 1}{\begin{bmatrix} \mathbf{y}_\mathbf{R} \\ \mathbf{y}_\mathbf{I} \end{bmatrix}} = \underset{2M\times 2N}{\begin{bmatrix} \Phi\Psi_R^{-1} & -\Phi\Psi_I^{-1} \\ \Phi\Psi_I^{-1} & \Phi\Psi_R^{-1} \end{bmatrix}} \underset{2N\times 1}{\begin{bmatrix} \mathbf{X}_\mathbf{R} \\ \mathbf{X}_\mathbf{I} \end{bmatrix}}$$