# Spatio-Temporal Compressive Sensing


**Yin Zhang**

*The University of Texas at Austin*

yzhang@cs.utexas.edu

Joint work with

**Matthew Roughan** *University of Adelaide*

**Walter Willinger** *AT&T Labs–Research*

**Lili Qiu** *Univ. of Texas at Austin*



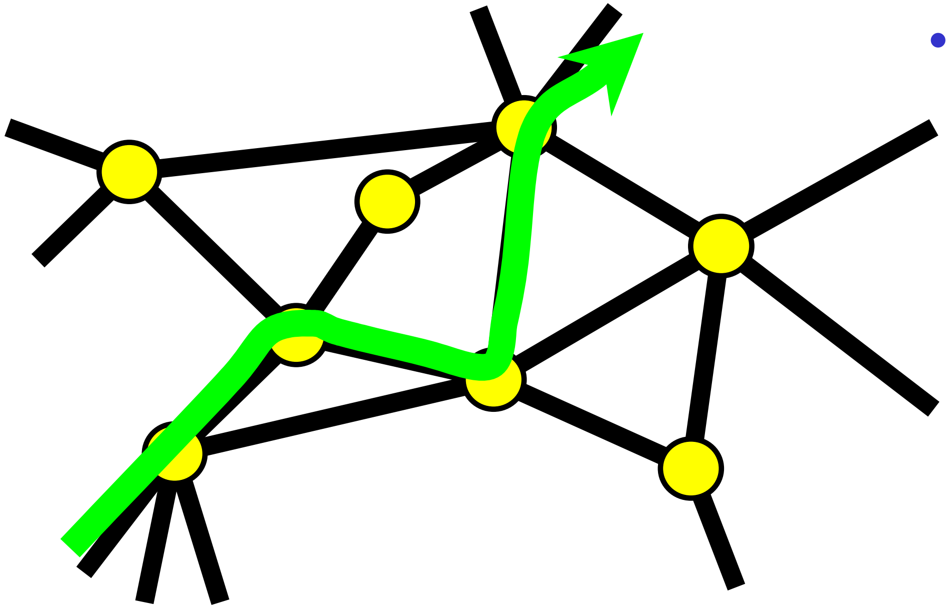
**ACM SIGCOMM 2009**
**August 20, 2009**

Q: How to fill in missing values in a matrix?

- Traffic matrix
- Delay matrix
- Social proximity matrix

# Internet Traffic Matrices

- Traffic Matrix (TM)
  - Gives traffic volumes between origins and destinations
- Essential for many networking tasks
  - what-if analysis, traffic engineering, anomaly detection

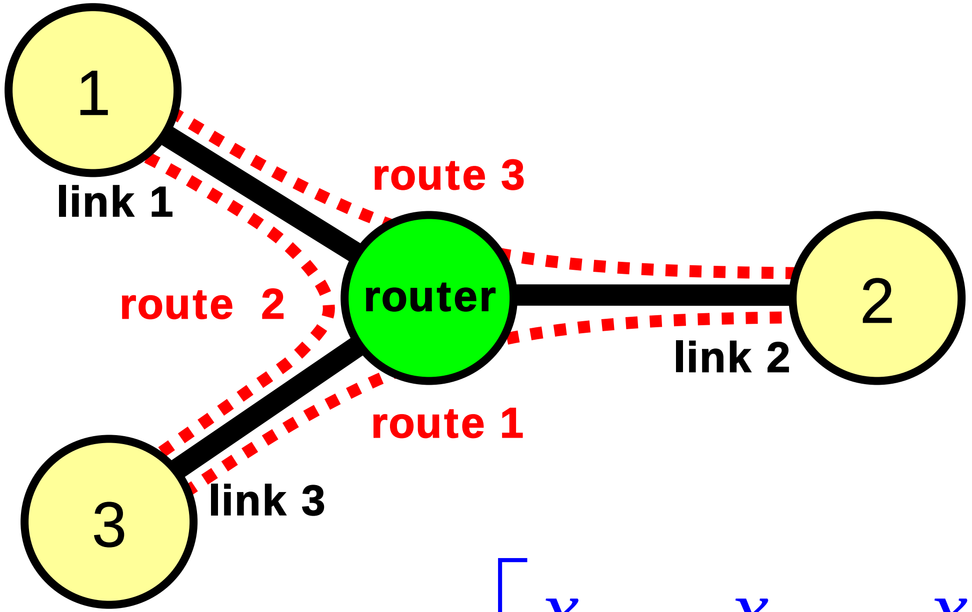


- Lots of prior research
  - Measurement, e.g. [FGLR+01, VE03]
  - Inference, e.g. [MTSB+02, ZRDG03, ZRLD03, ZRLD05, SLTP+06, ZGWX06]
  - Anomaly detection, e.g. [LCD04, ZGRG05, RSRD07]

# Missing Values: Why Bother?

- Missing values are common in TM measurements
  - Direct measurement is infeasible/expensive
  - Measurement and data collection are unreliable
  - Anomalies/outliers hide non-anomaly-related traffic
  - Future traffic has not yet appeared
- The need for missing value interpolation
  - Many networking tasks are sensitive to missing values
  - Need non-anomaly-related traffic for diagnosis
  - Need predicted TMs in what-if analysis, traffic engineering, capacity planning, etc.

# The Problem



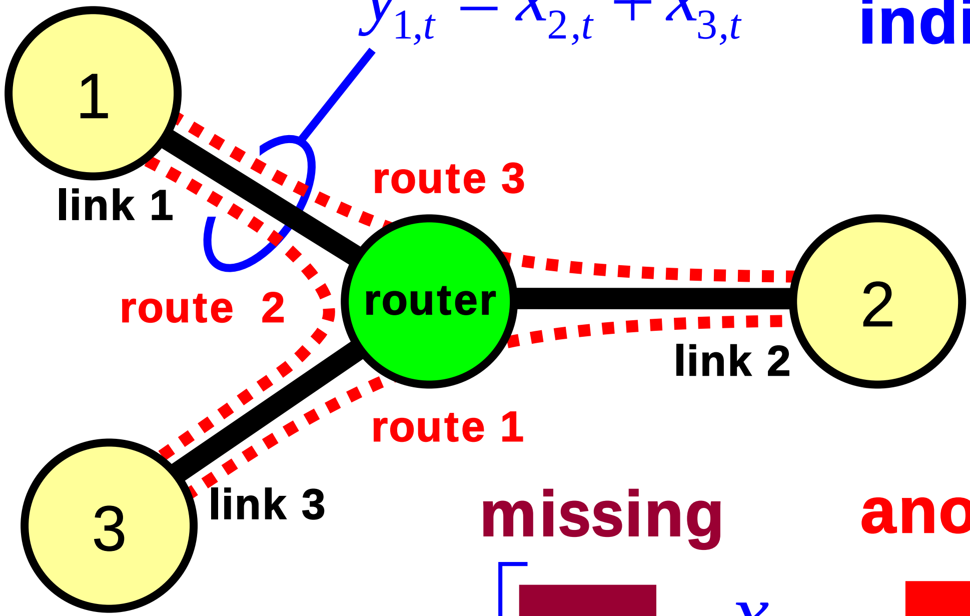

$$X = \begin{bmatrix} x_{1,1} & x_{1,2} & x_{1,3} & x_{1,4} & x_{1,5} & x_{1,6} \\ x_{2,1} & x_{2,2} & x_{2,3} & x_{1,4} & x_{2,5} & x_{2,6} \\ x_{3,1} & x_{3,2} & x_{3,3} & x_{1,4} & x_{3,5} & x_{3,6} \end{bmatrix}$$

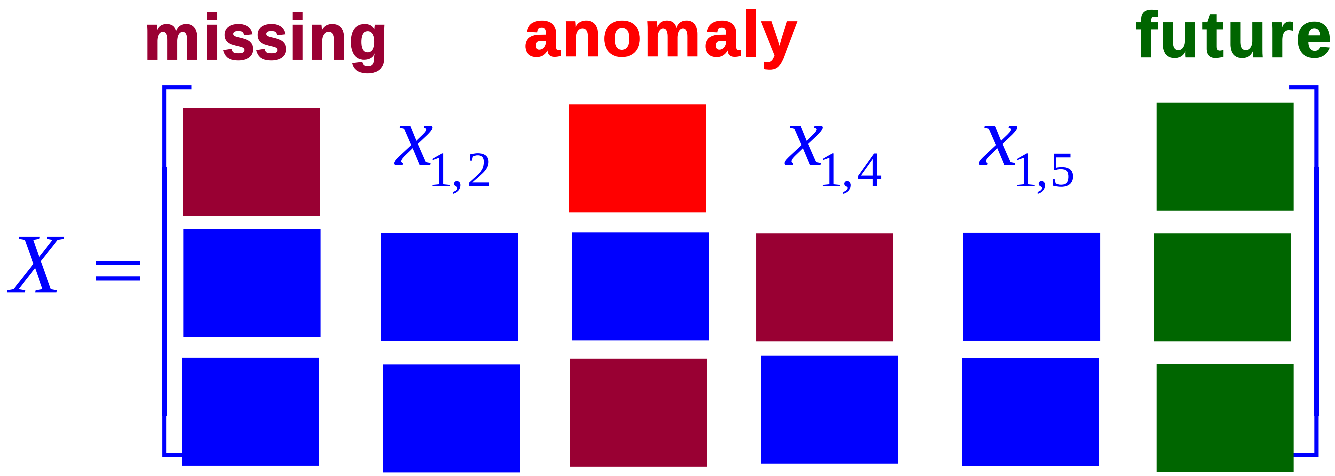
$X_{r,t}$ : traffic volume on route $r$ at time $t$

# The Problem

$$y_{1,t} = x_{2,t} + x_{3,t}$$

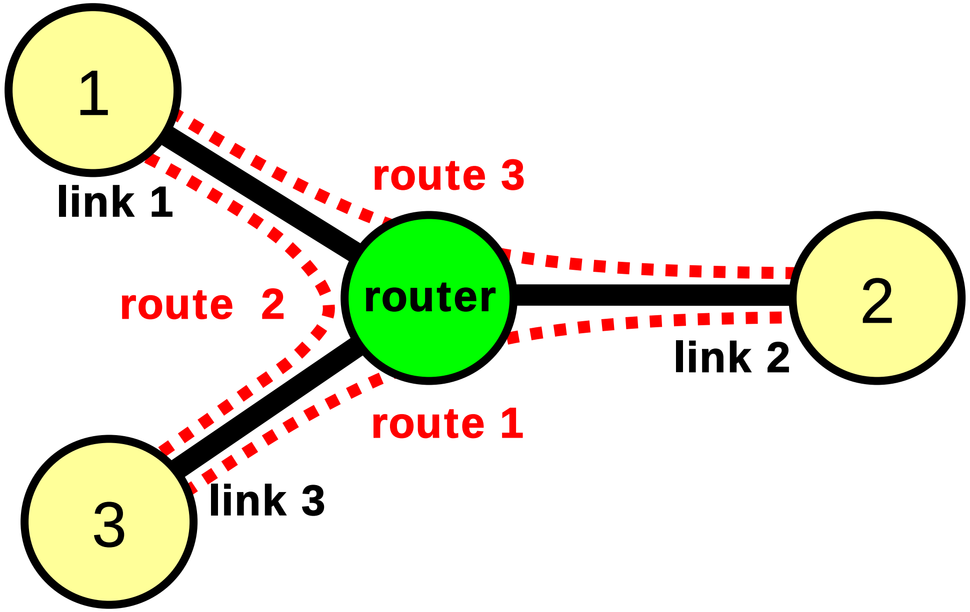
**indirect**: only measure at links




Interpolation: fill in missing values from *incomplete* and/or *indirect* measurements

# The Problem




E.g., link loads only: **AX=Y**

- A: routing matrix;
  Y: link load matrix

E.g., direct measurements only: **M.*X=M.*D**

- M(r,t)=1 ⇔ X(r,t) exists;
  D: direct measurements

$$\mathcal{A}(X)=B$$

Challenge: In real networks, the problem is massively underconstrained!

# Spatio-Temporal Compressive Sensing

- Idea 1: Exploit low-rank nature of TMs
  - Observation: TMs are low-rank [LPCD+04, LCD04]:

$$\mathbf{X}_{nxm} \approx \mathbf{L}_{nxr} * \mathbf{R}_{mxr}^{T} \quad (r \ll n,m)$$

- Idea 2: Exploit spatio-temporal properties
  - Observation: TM rows or columns close to each other (in some sense) are often close in value
- Idea 3: Exploit local structures in TMs
  - Observation: TMs have both global & local structures

# Spatio-Temporal Compressive Sensing

- Idea 1: Exploit low-rank nature of TMs
  - Technique: Compressive Sensing

- Idea 2: Exploit spatio-temporal properties
  - Technique: Sparsity Regularized Matrix Factorization (SRMF)

- Idea 3: Exploit local structures in TMs
  - Technique: Combine global and local interpolation

# Compressive Sensing

- Basic approach: find $X=LR^T$ s.t. $A(LR^T)=B$
  - (m+n)*r unknowns (instead of m*n)
- Challenges
  - $A(LR^T)=B$ may have many solutions → which to pick?
  - $A(LR^T)=B$ may have zero solution, e.g. when X is approximately low-rank, or there is noise
- Solution: Sparsity Regularized SVD (SRSVD)
  - minimize $|\mathcal{A}(LR^T) - B|^2$ *// fitting error*

    $+ \lambda (|L|^2+|R|^2)$ *// regularization*
  - Similar to SVD but can handle missing values and indirect measurements

# Sparsity Regularized Matrix Factorization

- Motivation
  - The theoretical conditions for compressive sensing to perform well may not hold on real-world TMs

- Sparsity Regularized Matrix Factorization
  - minimize $|\mathcal{A}(LR^T) - B|^2$ *// fitting error*

    $+ \lambda (|L|^2 + |R|^2)$ *// regularization*

    $+ |S(LR^T)|^2$ *// spatial constraint*

    $+ |(LR^T)T^T|^2$ *// temporal constraint*

  - S and T capture spatio-temporal properties of TMs
  - Can be solved efficiently via alternating least-squares

# Spatio-Temporal Constraints

- Temporal constraint matrix T
  - Captures temporal smoothness
  - Simple choices suffice, e.g.:

$$T = \begin{bmatrix} 1 & -1 & 0 & \cdots \\ 0 & 1 & -1 & \ddots \\ 0 & 0 & 1 & \ddots \\ \vdots & \ddots & \ddots & \ddots \end{bmatrix}$$

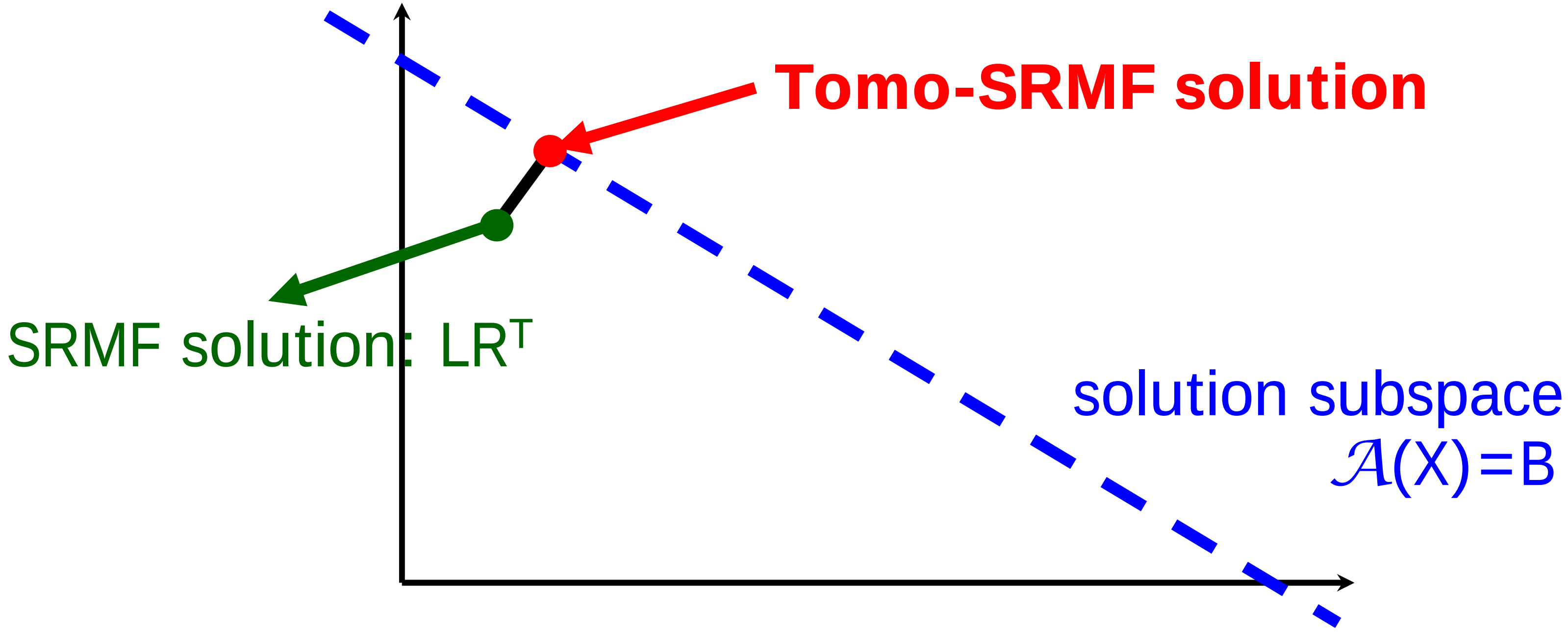
- Spatial constraint matrix S
  - Captures which rows of X are close to each other
  - Challenge: TM rows are ordered arbitrarily
  - Our solution: use a initial estimate of X to approximate similarity between rows of X

# Combining Global and Local Methods

- Local correlation among individual elements may be stronger than among TM rows/columns
  - S and T in SRMF are chosen to capture global correlation among entire TM rows or columns
- SRMF+KNN: combine SRMF with local interpolation
  - Switch to K-Nearest-Neighbors if a missing element is temporally close to observed elements

# Generalizing Previous Methods

- Tomo-SRMF: find a solution that is close to $LR^T$ yet satisfies $\mathcal{A}(X)=B$




Tomo-SRMF generalizes the tomo-gravity method for inferring TM from link loads

# Applications

- Inference (a.k.a. tomography)
  - Can combine both direct and indirect measurements for TM inference
- Prediction
  - What-if analysis, traffic engineering, capacity planning all require predicted traffic matrix
- Anomaly Detection
  - Project TM onto a low-dimensional, spatially & temporally smooth subspace ($LR^T$) → normal traffic

Spatio-temporal compressive sensing provides a unified approach for many applications

# Evaluation Methodology

- Data sets

| Network | Date | Duration | Resolution | Size |
|---|---|---|---|---|
| Abilene | 03/2003 | 1 week | 10 min. | 121x1008 |
| Commercial ISP | 10/2006 | 3 weeks | 1 hour | 400x504 |
| GEANT | 04/2005 | 1 week | 15 min. | 529x672 |

- Metrics
  - Normalized Mean Absolute Error for missing values

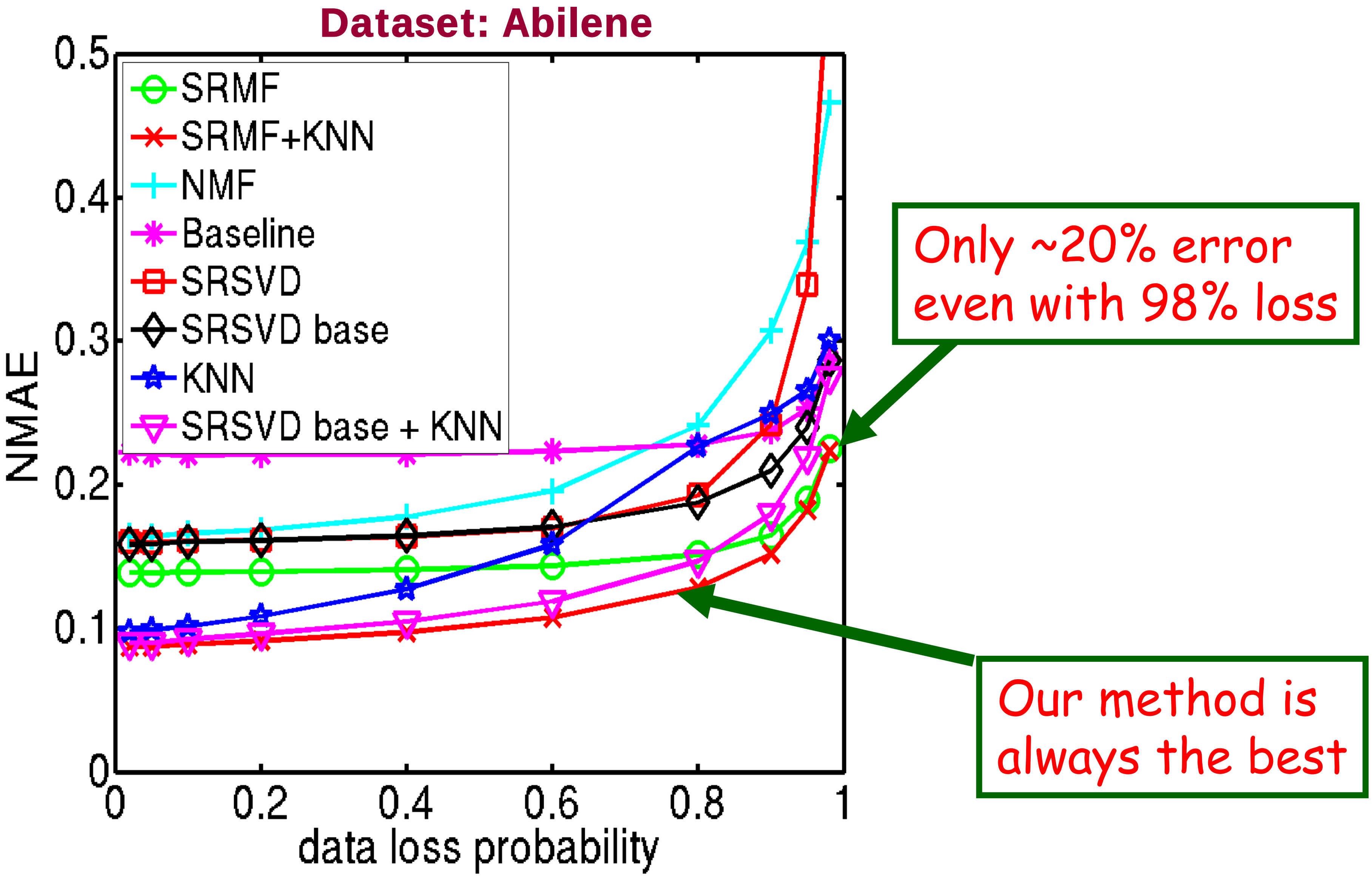
$$NMAE = \frac{\sum_{i,j:M(i,j)=0} |X(i,j) - X_{est}(i,j)|}{\sum_{i,j:M(i,j)=0} |X(i,j)|}$$

  - Other metrics yield qualitatively similar results.

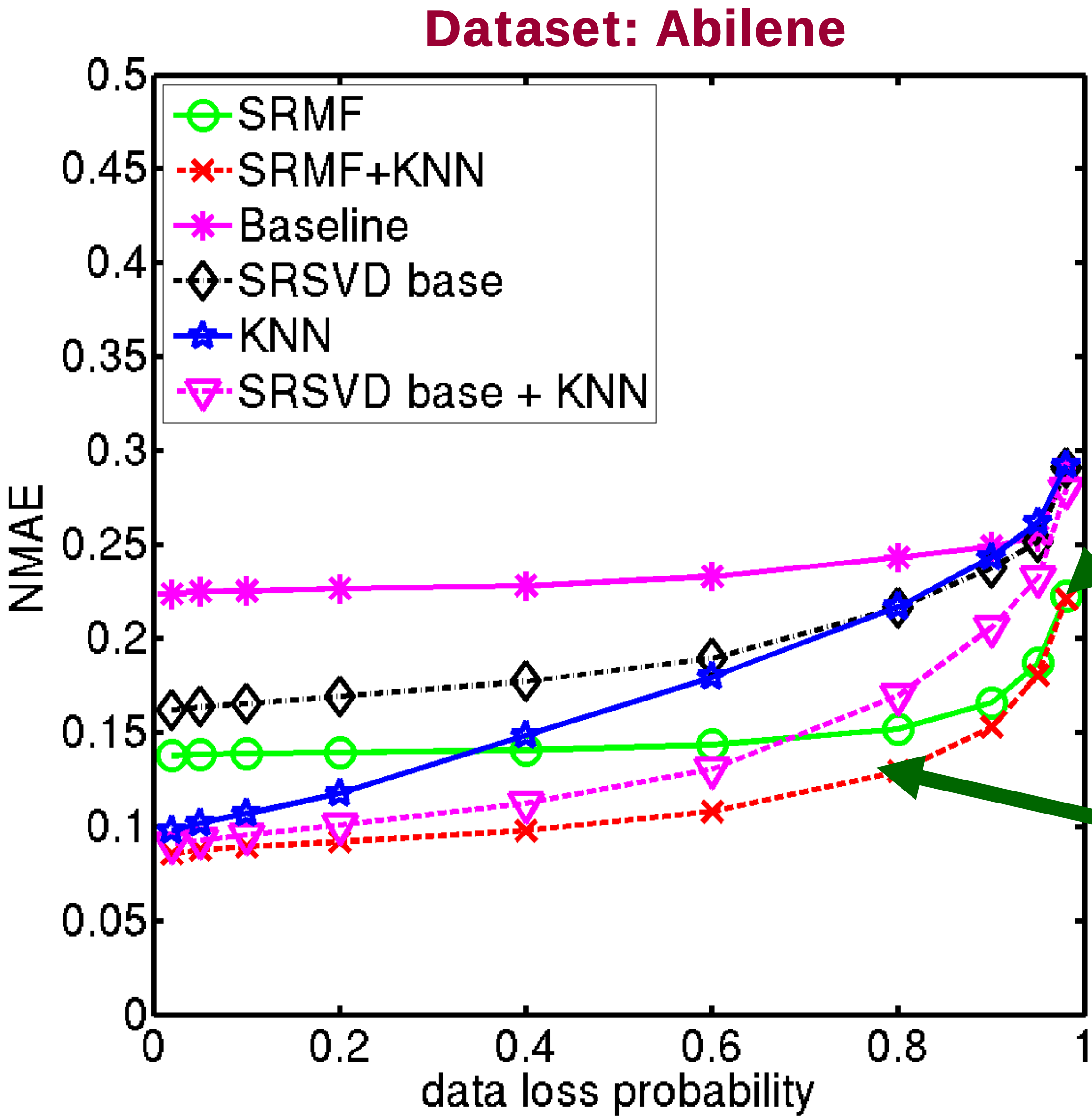
# Algorithms Compared

| Algorithm | Description |
|---|---|
| Baseline | Baseline estimate via rank-2 approximation |
| SRSVD | Sparsity Regularized SVD |
| SRSVD-base | SRSVD with baseline removal |
| NMF | Nonnegative Matrix Factorization |
| KNN | K-Nearest-Neighbors |
| SRSVD-base+KNN | Hybrid of SRSVD-base and KNN |
| SRMF | Sparsity Regularized Matrix Factorization |
| SRMF+KNN | Hybrid of SRMF and KNN |
| Tomo-SRMF | Generalization of tomo-gravity |

# Interpolation: Random Loss



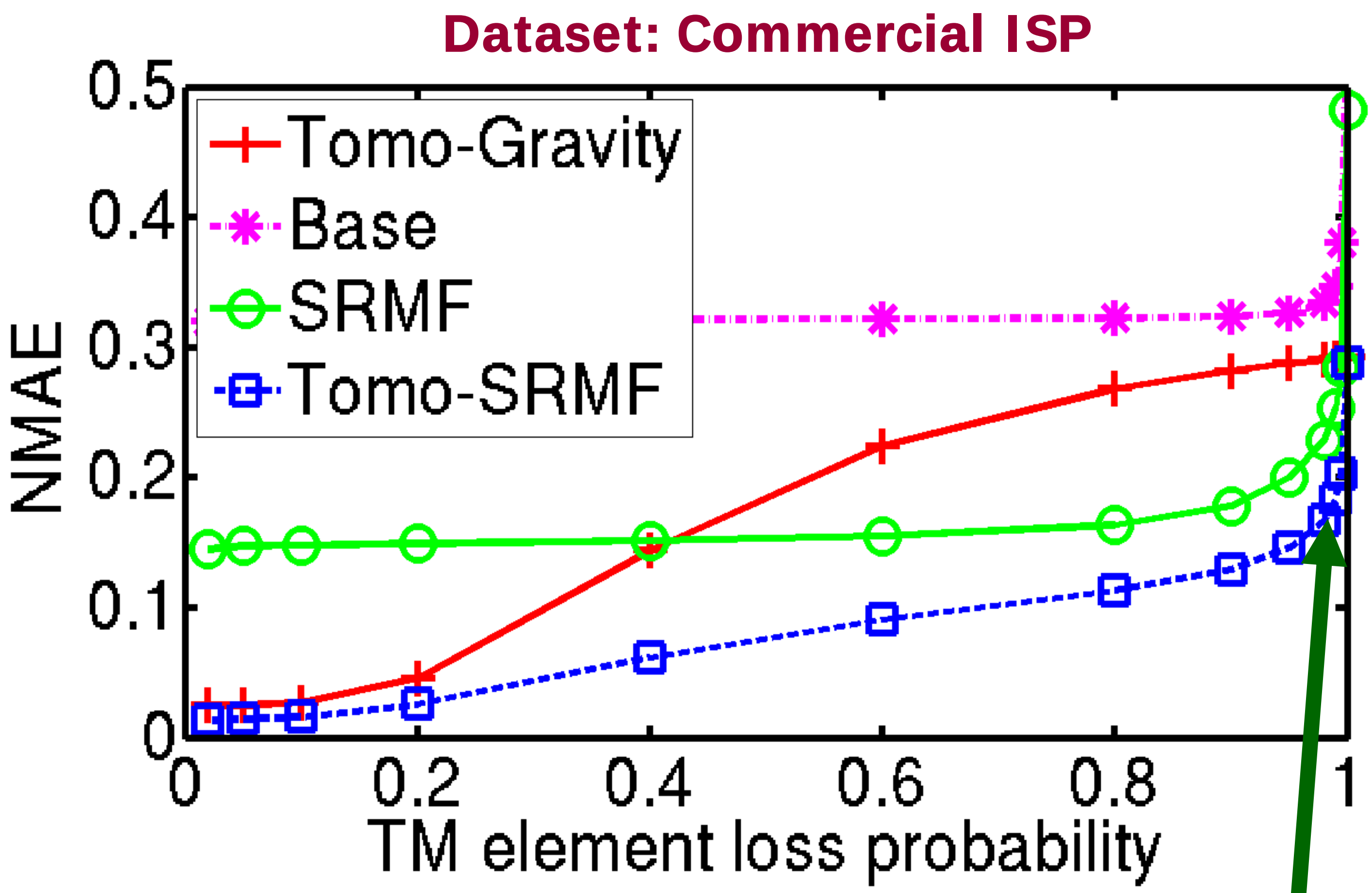
# Interpolation: Structured Loss




Only ~20% error even with 98% loss

Our method is always the best; sometimes dramatically better

# Tomography Performance




Can halve the error of Tomo-Gravity by measuring only 2% elements!

# Other Results

- Prediction
  - Taking periodicity into account helps prediction
  - Our method consistently outperforms other methods
    - Smooth, low-rank approximation improves prediction

- Anomaly detection
  - Generalizes many previous methods
    - E.g., PCA, anomography, time domain methods
  - Yet offers more
    - Can handle missing values, indirect measurements
    - Less sensitive to contamination in normal subspace
    - No need to specify exact # of dimensions for normal subspace
  - Preliminary results also show better accuracy

# Conclusion

- Spatio-temporal compressive sensing
  - Advances ideas from compressive sensing
  - Uses the first truly spatio-temporal model of TMs
  - Exploits both global and local structures of TMs
- General and flexible
  - Generalizes previous methods yet can do much more
  - Provides a unified approach to TM estimation, prediction, anomaly detection, etc.
- Highly effective
  - Accurate: works even with 90+% values missing
  - Robust: copes easily with highly structured loss
  - Fast: a few seconds on TMs we tested

# Lots of Future Work

- Other types of network matrices
  - Delay matrices, social proximity matrices
- Better choices of S and T
  - Tailor to both applications and datasets
- Extension to higher dimensions
  - E.g., 3D: source, destination, time
- Theoretical foundation
  - When and why our approach works so well?

# Thank you!

# Alternating Least Squares

- Goal: minimize $|\mathcal{A}(LR^T) - B|^2 + \lambda (|L|^2 + |R|^2)$

- Step 1: fix L and optimize R
  - A standard least-squares problem

- Step 2: fix R and optimize L
  - A standard least-squares problem

- Step 3: goto Step 1 unless MaxIter is reached