Decentralized Prediction of End-to-End Network Performance Classes

Yongjun Liao, Wei Du, Pierre Geurts, Guy Leduc

Research Unit in Networking(RUN), University of Liège, Belgium

December 08, 2011

End-to-End Network Performance

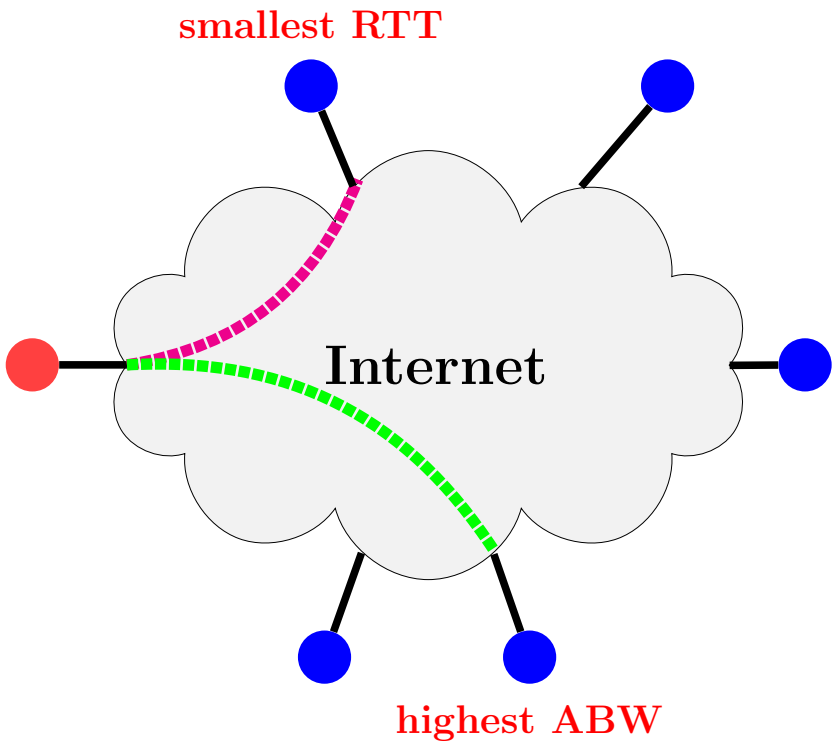
Metrics

- round-trip time (RTT)
- available bandwidth (ABW)
- packet loss rate (PLR)

Network Performance Matters!

- peer-to-peer downloading
- overlay routing
- content distribution network
- Internet games

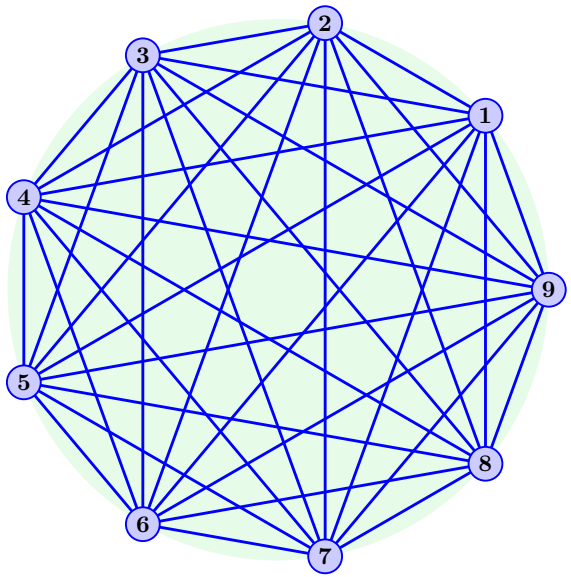
Peer Selection



Université de Liège

Acquisition on Large-Scale Networks

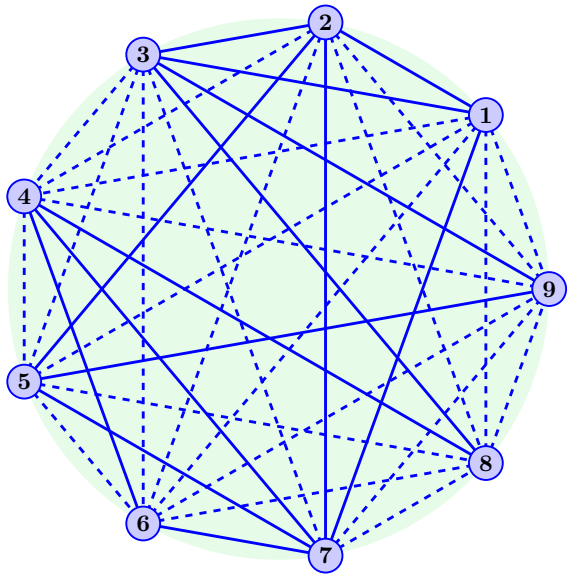
full-mesh active measurements



n nodes $\Rightarrow$ $o(n^2)$ measurements

accurate but expensive

network performance prediction



n nodes $\Rightarrow$ measurements $\ll o(n^2)$

less accurate but cheap

Université de Liège

Network Performance Prediction

Challenges

Network Performance Prediction

Challenges

- Networks are dynamic.
  - Churn: nodes join and leave frequently.
  - Metric values vary over time.

Network Performance Prediction

Challenges

- Networks are dynamic.
  - Churn: nodes join and leave frequently.
  - Metric values vary over time.
- Metrics differ largely.
  - RTT: symmetric; ABW: asymmetric.
  - RTT: the smaller the better; ABW: the larger the better.
  - RTT and ABW are measured with different methodologies.

Network Performance Prediction

Challenges

- Networks are dynamic.
  - Churn: nodes join and leave frequently.
  - Metric values vary over time.
- Metrics differ largely.
  - RTT: symmetric; ABW: asymmetric.
  - RTT: the smaller the better; ABW: the larger the better.
  - RTT and ABW are measured with different methodologies.
- Decentralized processing is prefered.
  - no landmarks or central servers
  - no infrastructure

Related Work on Network Performance Prediction

Round-Trip Time

- Euclidean Embedding
  - GNP: Ng et al. INFOCOM 2002
  - Vivaldi: Dabek et al. SIGCOMM 2004
- Matrix Factorization
  - IDES: Mao et al. IMC 2004
  - DMF: Liao et al. Networking 2010

Available Bandwidth

- SEQUOIA: Rama et al. SIGMETRICS 2009
- iPlane: Madhyastha et al. USENIX OSDI 2006

Our Contributions

Our Contributions

1. Class-based Performance Representation

Represent network performance by discrete-valued classes, instead of real-valued quantities.

Our Contributions

1. Class-based Performance Representation

Represent network performance by discrete-valued classes, instead of real-valued quantities.

2. Formulation as Matrix Completion

Treat the prediction problem as a matrix completion problem.

Our Contributions

1. Class-based Performance Representation

Represent network performance by discrete-valued classes, instead of real-valued quantities.

2. Formulation as Matrix Completion

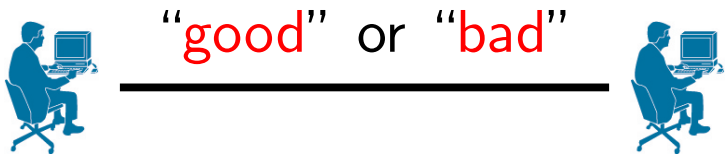
Treat the prediction problem as a matrix completion problem.

3. Decentralized Prediction Algorithm

DMFSGD: a decentralized matrix facotrization algorithm based on stochastic gradient descent.

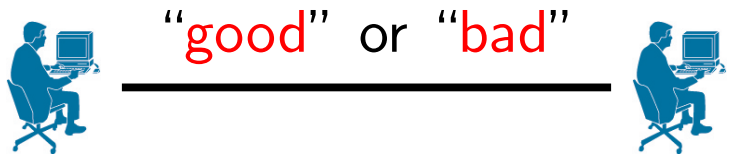
Class-based Performance Representation

Binary Classification



Class-based Performance Representation

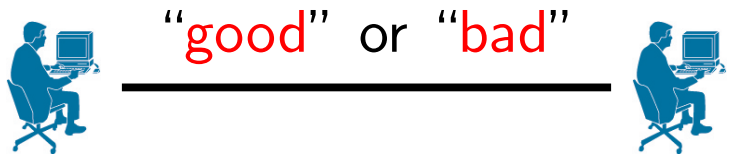
Binary Classification



- Class reflects the QoS experience of end users.

Class-based Performance Representation

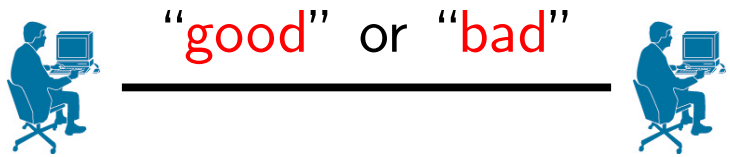
Binary Classification



- Class reflects the QoS experience of end users.
- Class unifies different metrics.

Class-based Performance Representation

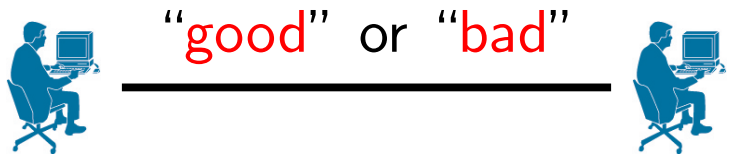
Binary Classification



- Class reflects the QoS experience of end users.
- Class unifies different metrics.
- Class information is often sufficient.
  - Streaming applications care if ABW is high enough.
  - P2P applications care if RTT is small enough.

Class-based Performance Representation

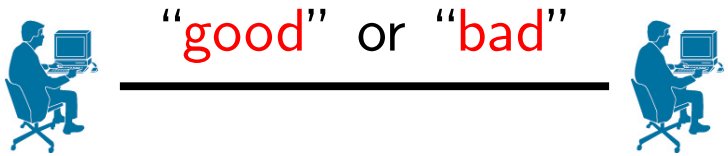
Binary Classification



- Class reflects the QoS experience of end users.
- Class unifies different metrics.
- Class information is often sufficient.
  - Streaming applications care if ABW is high enough.
  - P2P applications care if RTT is small enough.
- Class measurements are cheap.
  - Classes are rough.
  - Classes are more stable.

Université de Liège

Measure Performance Classes

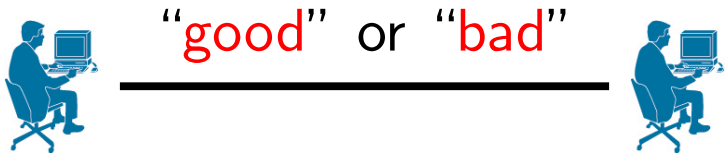


Thresholding

Measure if the metric value is larger or smaller than a threshold τ.

- If RTT $<$ 100ms, performance is "good".
- If ABW $>$ 100Mbps, performance is "good".

Measure Performance Classes



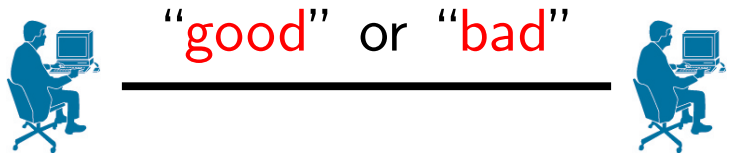
Thresholding

Measure if the metric value is larger or smaller than a threshold τ.

- If RTT $<$ 100ms, performance is "good".
- If ABW $>$ 100Mbps, performance is "good".

Measuring classes is much cheaper!

Measure Performance Classes



Thresholding

Measure if the metric value is larger or smaller than a threshold τ.

- If RTT $<$ 100ms, performance is "good".
- If ABW $>$ 100Mbps, performance is "good".

Measuring classes is much cheaper!

Threshold τ

defined according to requirements of applications

- Google TV requires 10Mbps for HD contents.

Our Contributions

1. Class-based Performance Representation

Represent network performance by discrete-valued classes, instead of real-valued quantities.

2. Formulation as Matrix Completion

Treat the prediction problem as a matrix completion problem.

3. Decentralized Prediction Algorithm

DMFSGD: a decentralized matrix facotrization algorithm based on stochastic gradient descent.

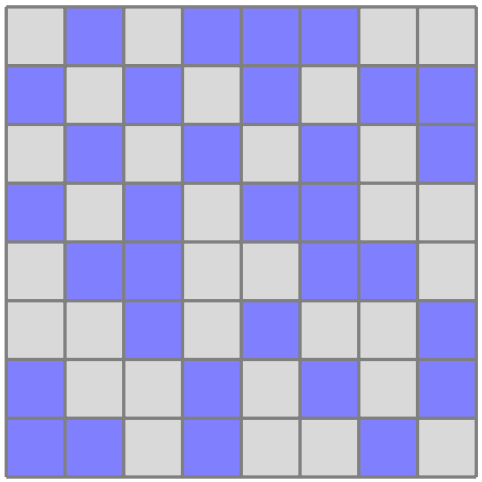
Université de Liège

Matrix Completion for Network Performance Prediction

Matrix Completion

Predict the unknown entries from a few known entries.

X



Université de Liège

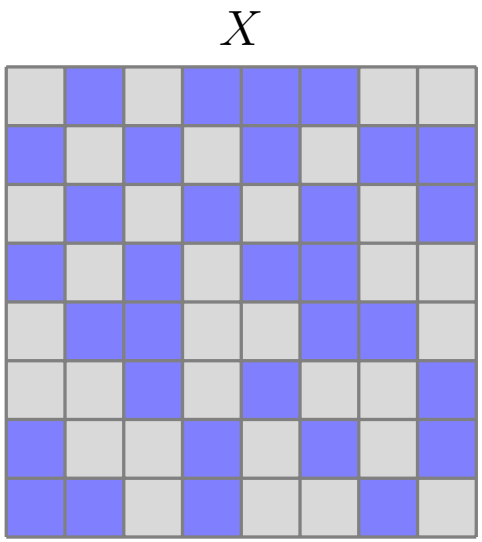
Matrix Completion for Network Performance Prediction

Matrix Completion

Predict the unknown entries from a few known entries.

Why is it possible?

- Matrix entries are correlated.



Université de Liège

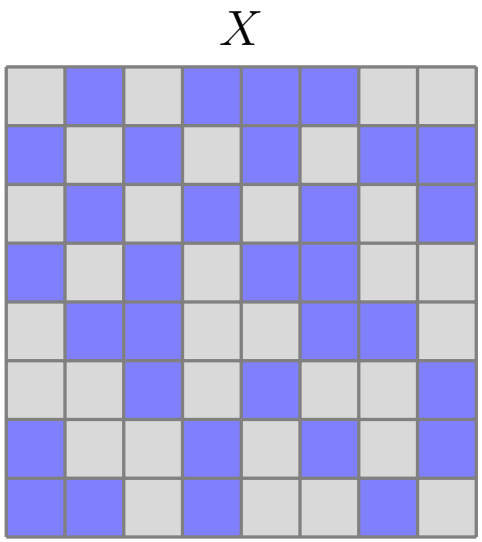
Matrix Completion for Network Performance Prediction

Matrix Completion

Predict the unknown entries from a few known entries.

Why is it possible?

- Matrix entries are correlated.
- The correlations induce low rank.



Université de Liège

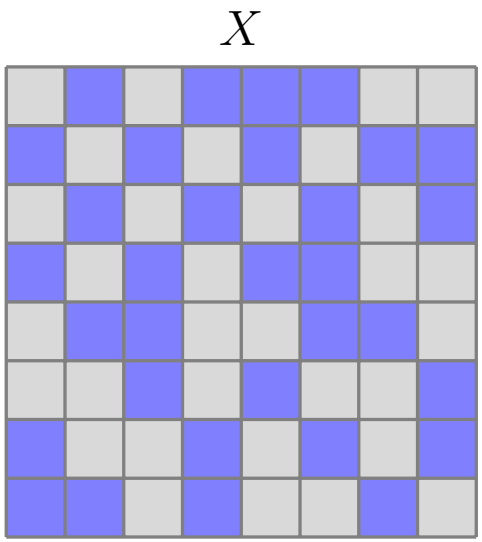
Matrix Completion for Network Performance Prediction

Matrix Completion

Predict the unknown entries from a few known entries.

Why is it possible?

- Matrix entries are correlated.
- The correlations induce low rank.
- $n \times n$ matrix of rank $r < n$
  - only r linearly independent columns or rows



Matrix Completion for Network Performance Prediction

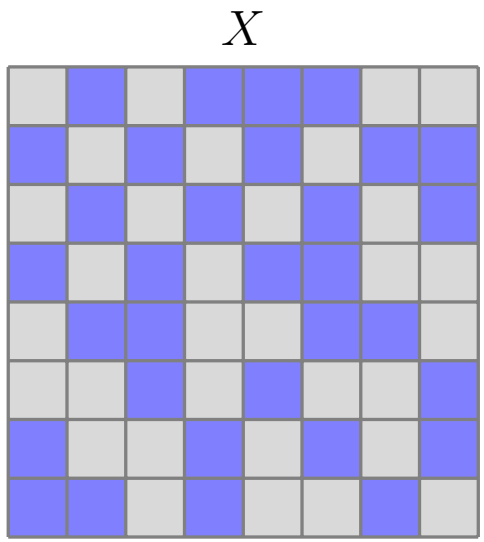
Matrix Completion

Predict the unknown entries from a few known entries.

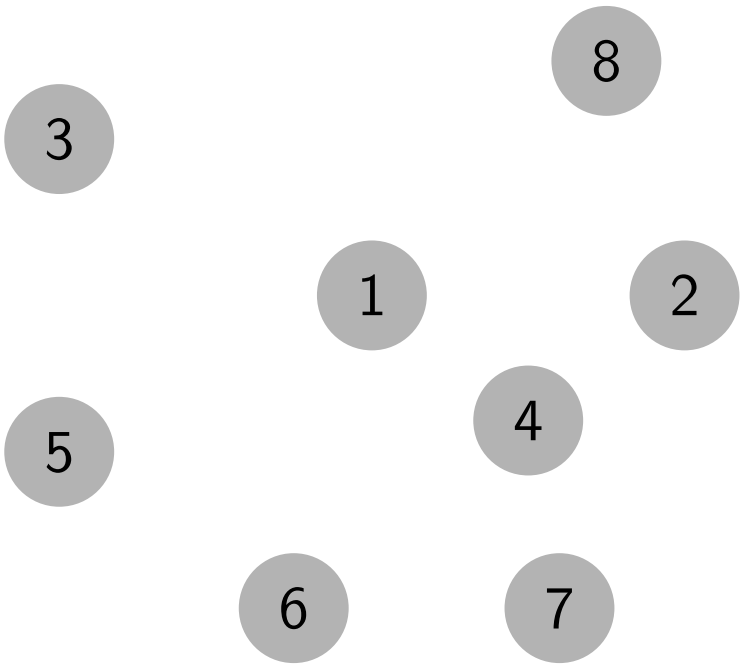
Why is it possible?

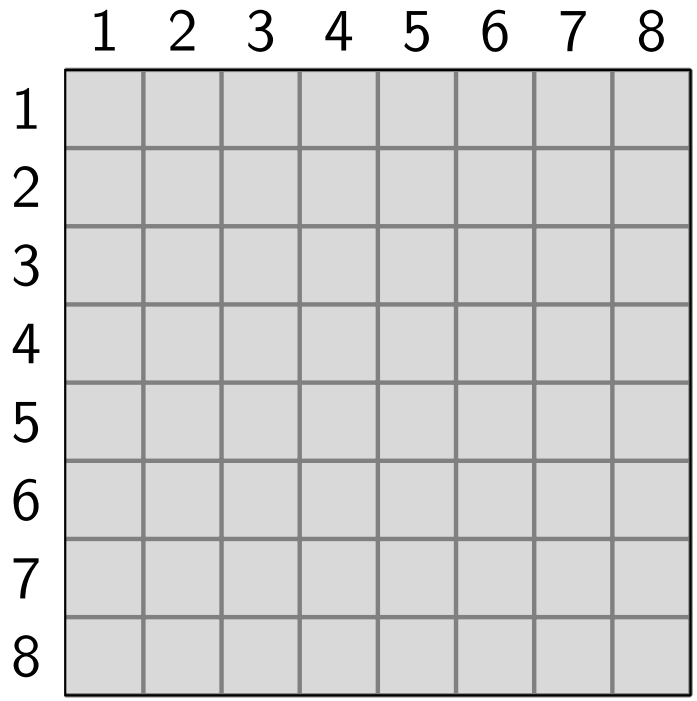
- Matrix entries are correlated.
- The correlations induce low rank.
- $n \times n$ matrix of rank $r < n$
  - only r linearly independent columns or rows

You don't need all $n \times n$ entries!



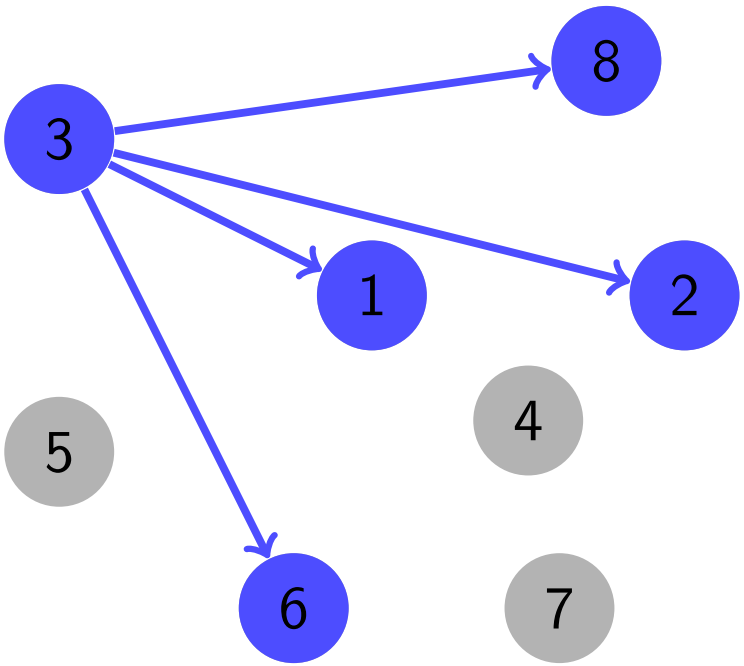
Matrix Completion for Network Performance Prediction

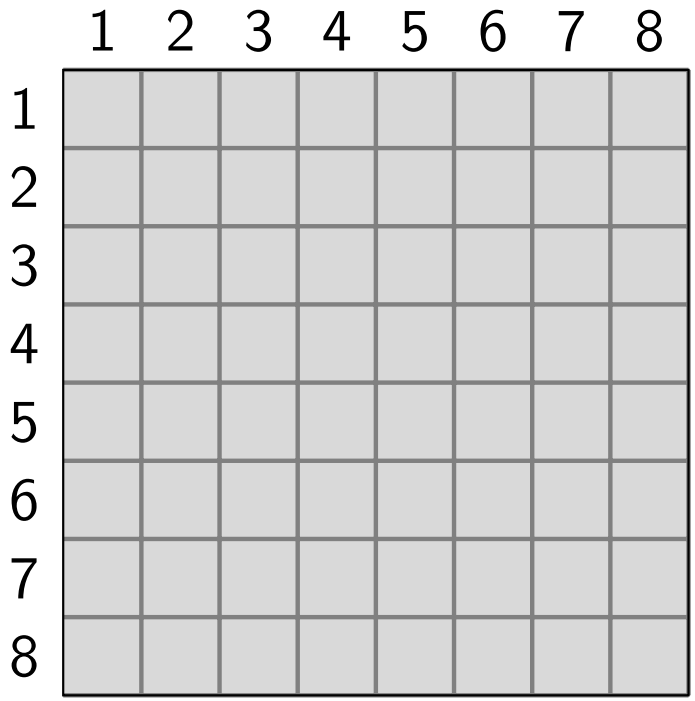




Université de Liège

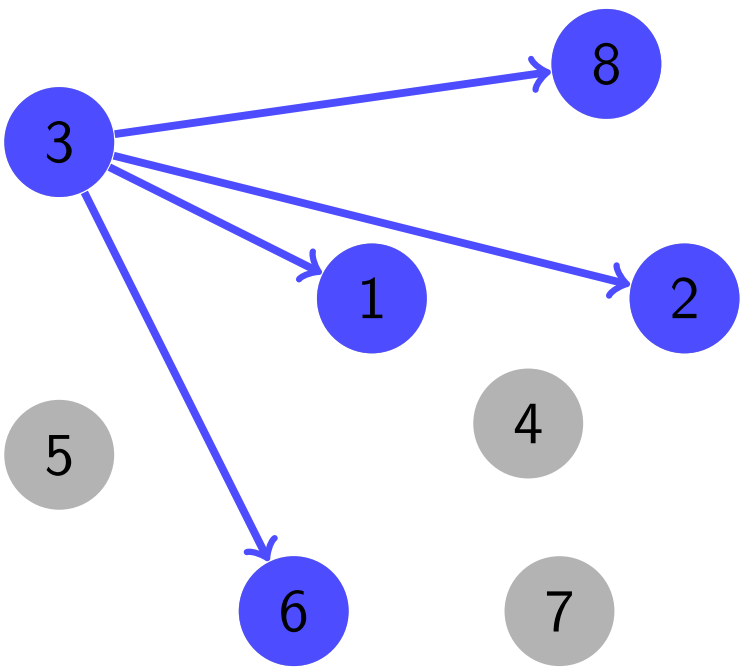
Matrix Completion for Network Performance Prediction

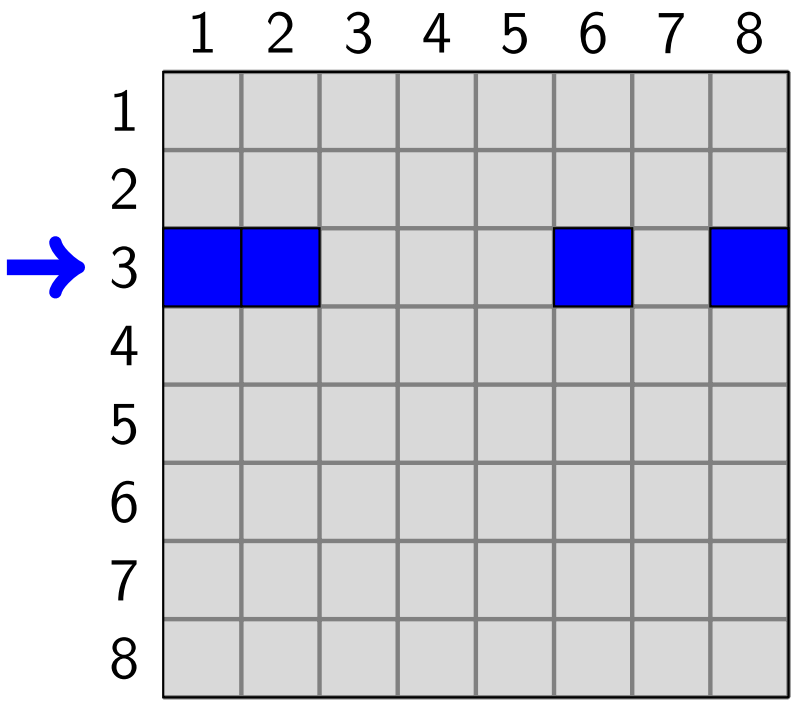




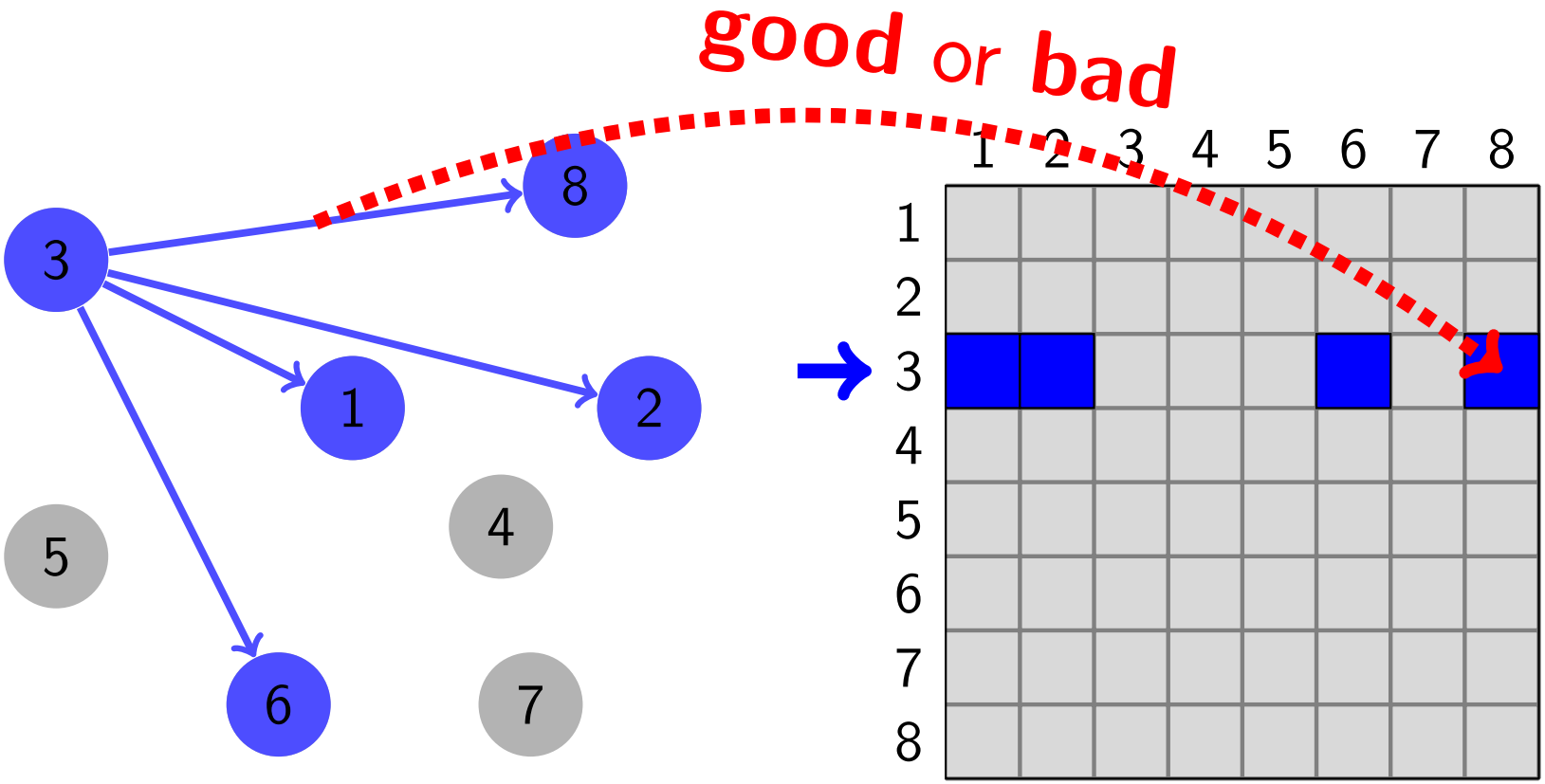
Université de Liège

Matrix Completion for Network Performance Prediction

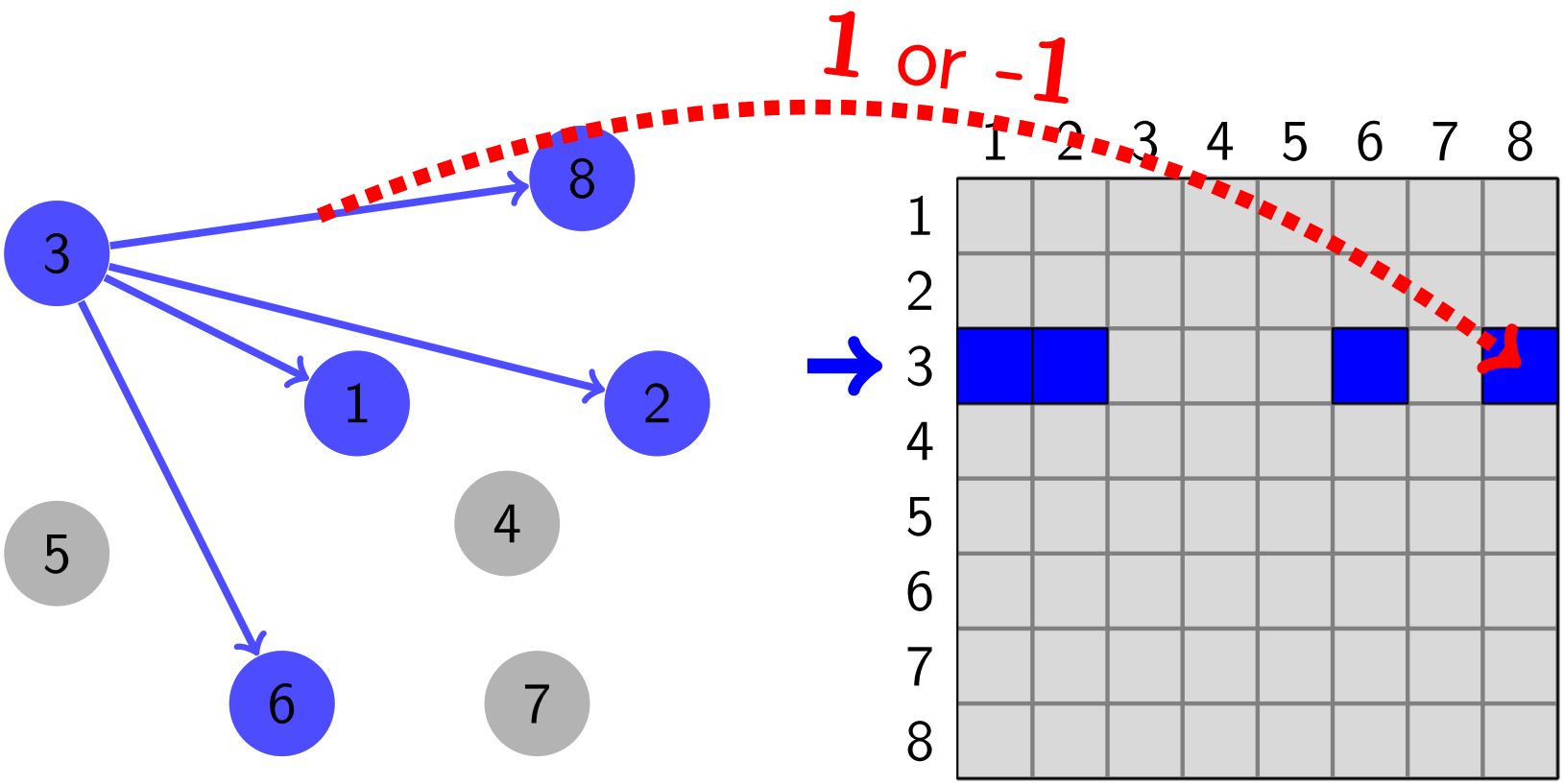




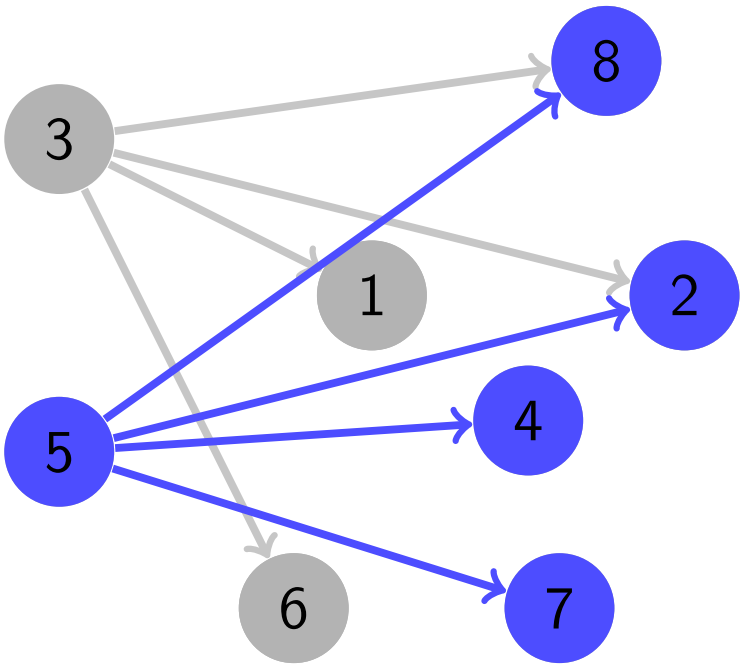
Matrix Completion for Network Performance Prediction

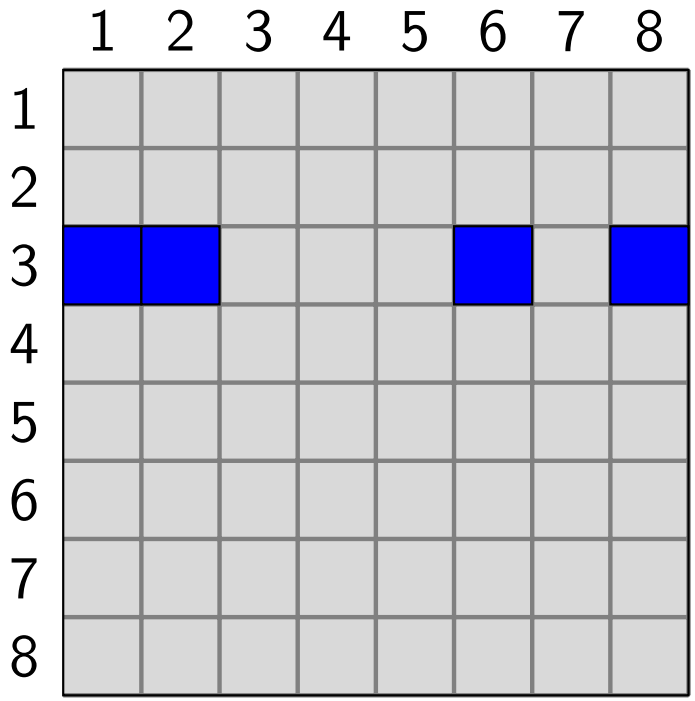


Matrix Completion for Network Performance Prediction



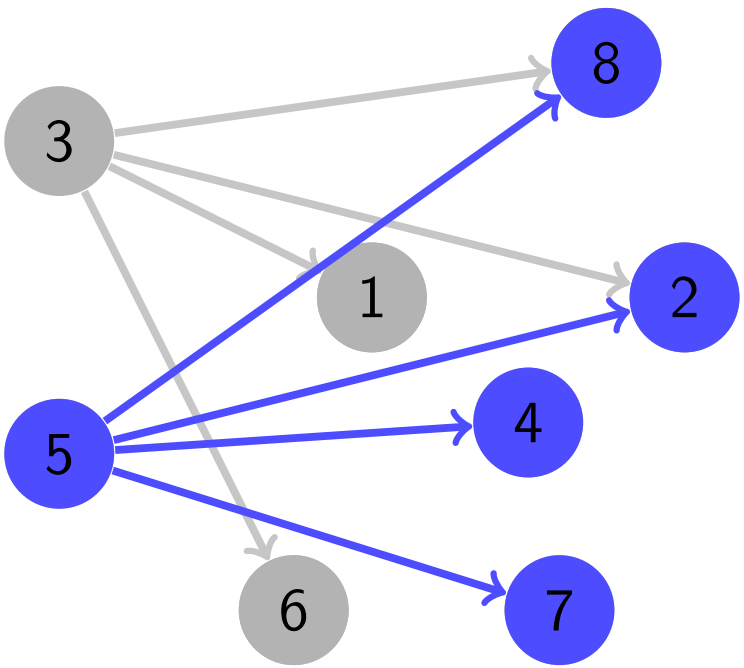
Matrix Completion for Network Performance Prediction

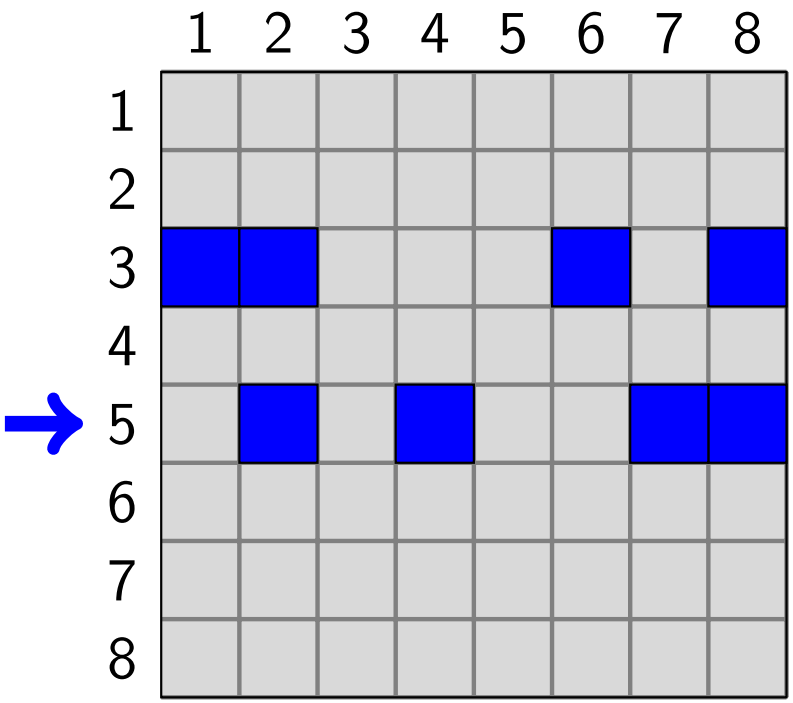




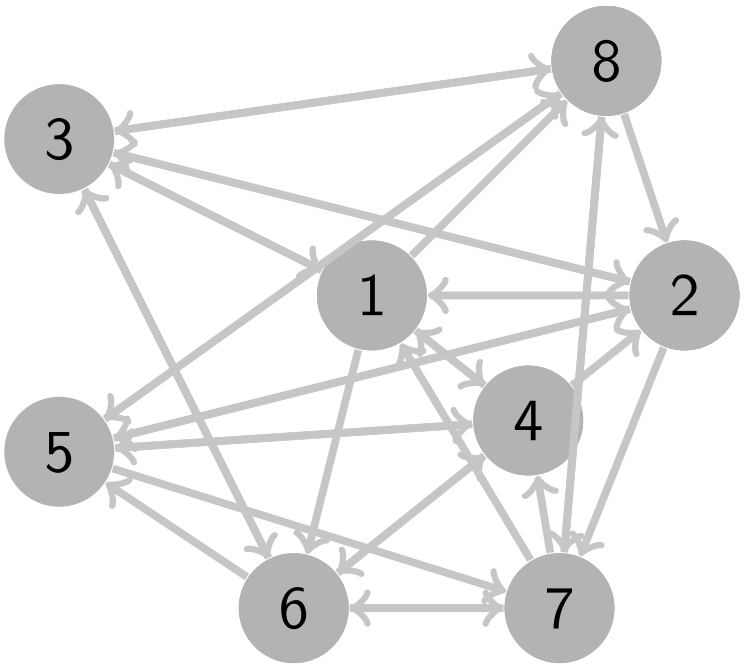
Université de Liège

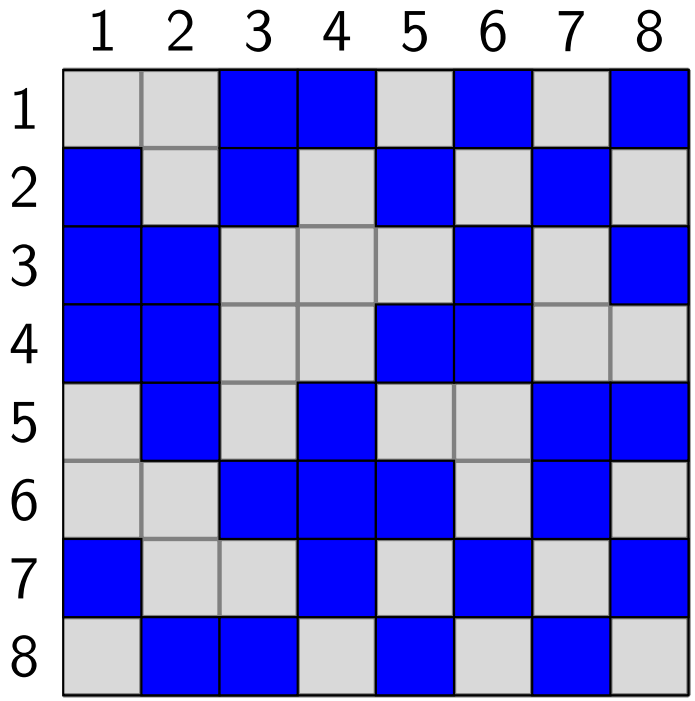
Matrix Completion for Network Performance Prediction





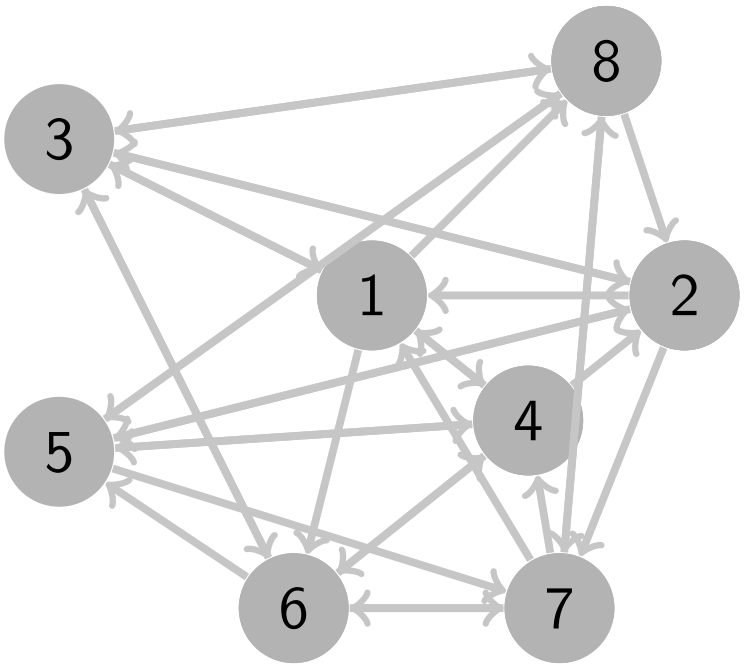
Matrix Completion for Network Performance Prediction

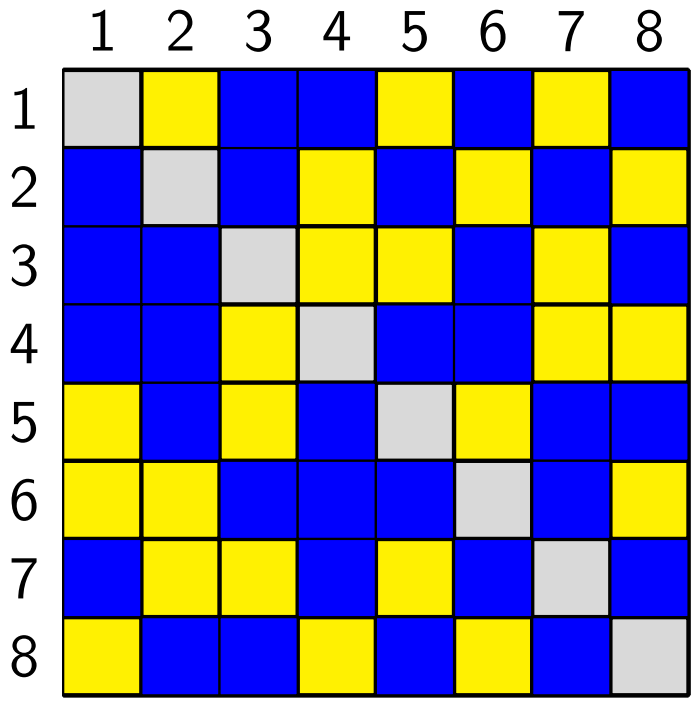




Université de Liège

Matrix Completion for Network Performance Prediction



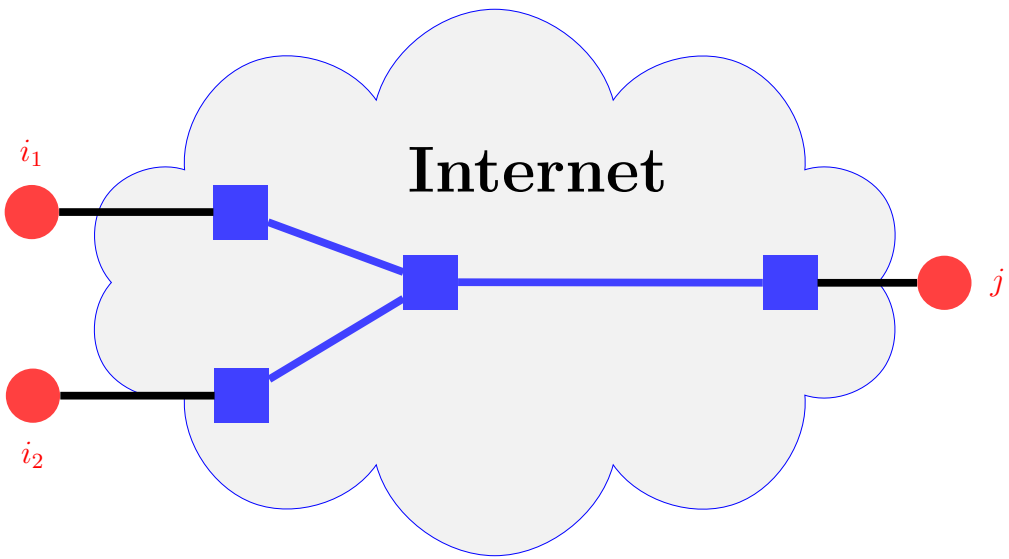


Université de Liège

Why is Matrix Completion Possible

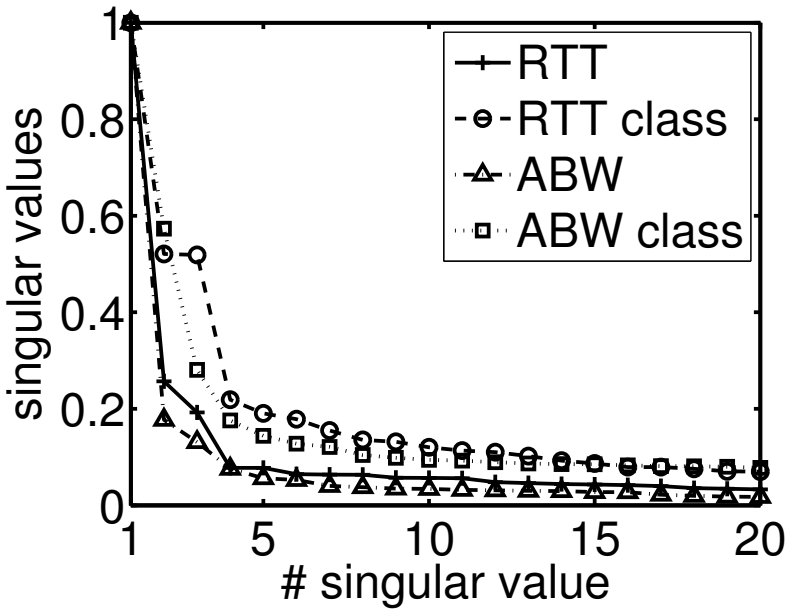
Correlations across network performance

- network topology
- routing algorithms
- redundancies among network paths
- ...



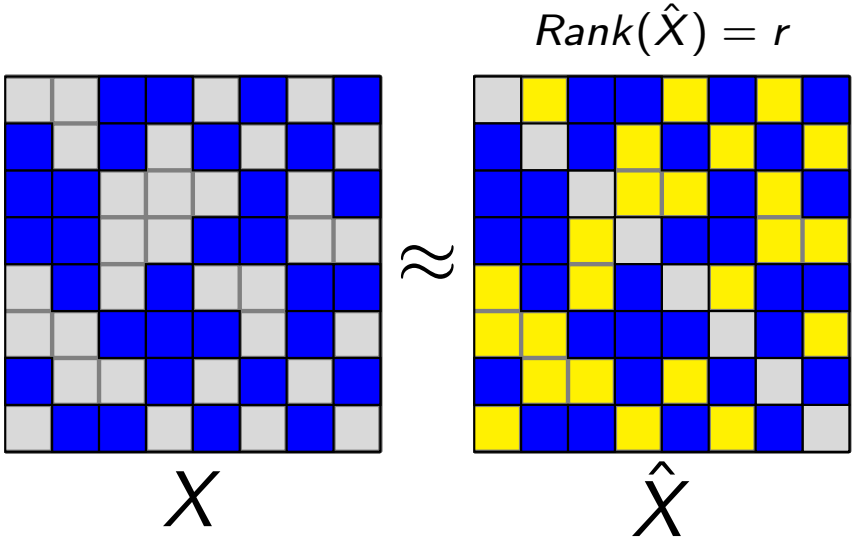
Why is Matrix Completion Possible

Low-rank of performance matrices

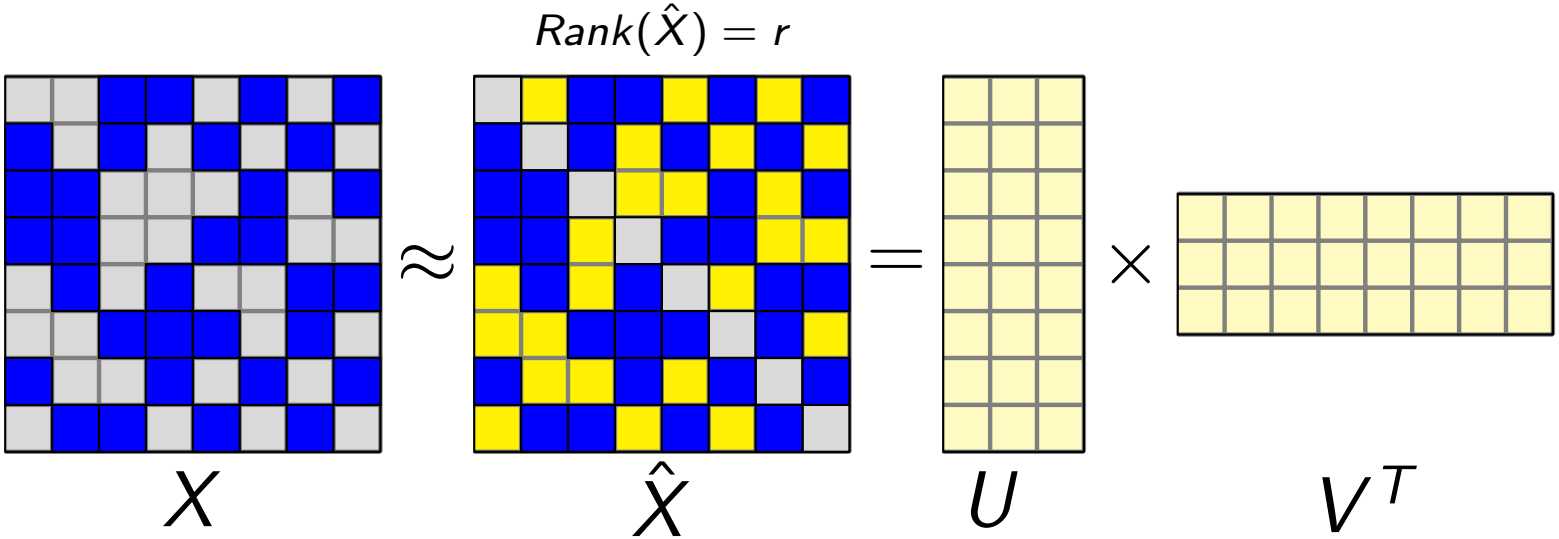


- RTT matrix: 2255×2255
- ABW matrix: 201×201
- Class matrices are obtained by thresholding.

Low-Rank Matrix Factorization



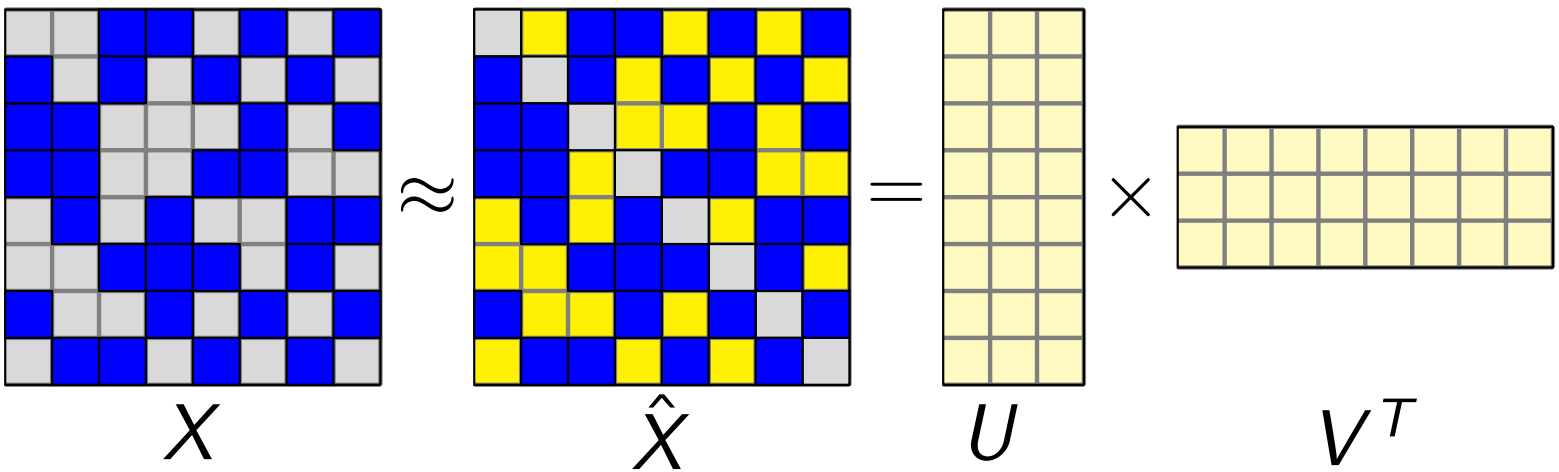
Low-Rank Matrix Factorization



Low-Rank Matrix Factorization

Look for (U, V), instead of $\hat{X}$

$Rank(\hat{X}) = r$



Our Contributions

1. Class-based Performance Representation

Represent network performance by discrete-valued classes, instead of real-valued quantities.

2. Formulation as Matrix Completion

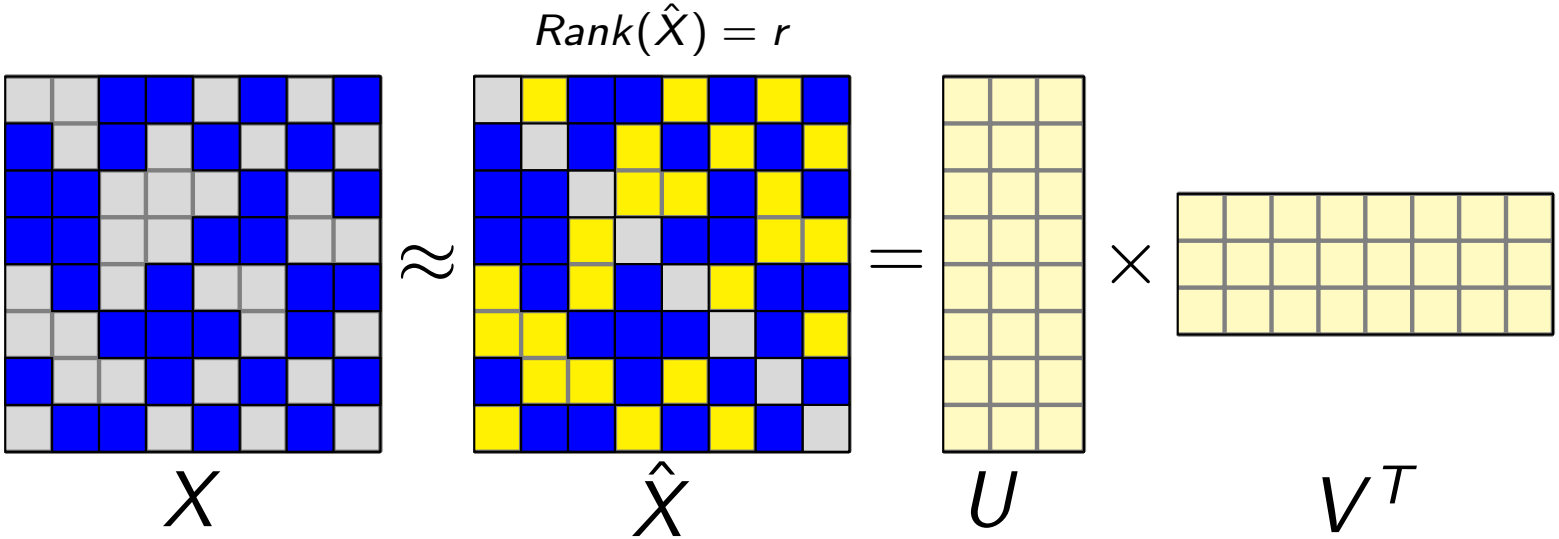
Treat the prediction problem as a matrix completion problem.

3. Decentralized Prediction Algorithm

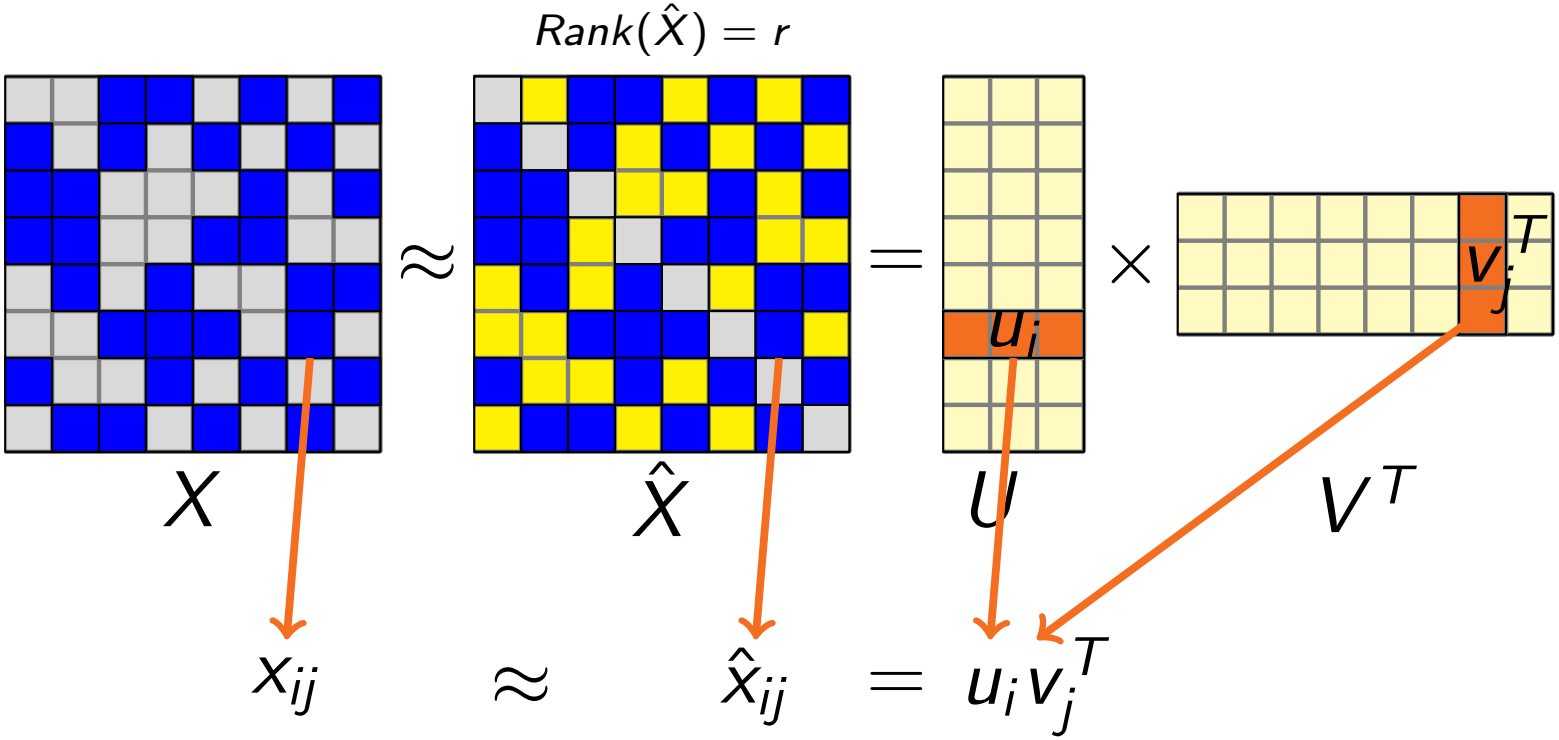
DMFSGD: a decentralized matrix facotrization algorithm based on stochastic gradient descent.

Université de Liège

Stochastic Gradient Descent



Stochastic Gradient Descent



Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

Repeated SGD updates

Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.

Repeated SGD updates

Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.
- U, V: row vectors u_i, v_i are stored at node i.

Repeated SGD updates

Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.
- U, V: row vectors u_i, v_i are stored at node i.

Repeated SGD updates

- When x_{ij} is available, update so that $x_{ij} \approx u_i v_j^T$.

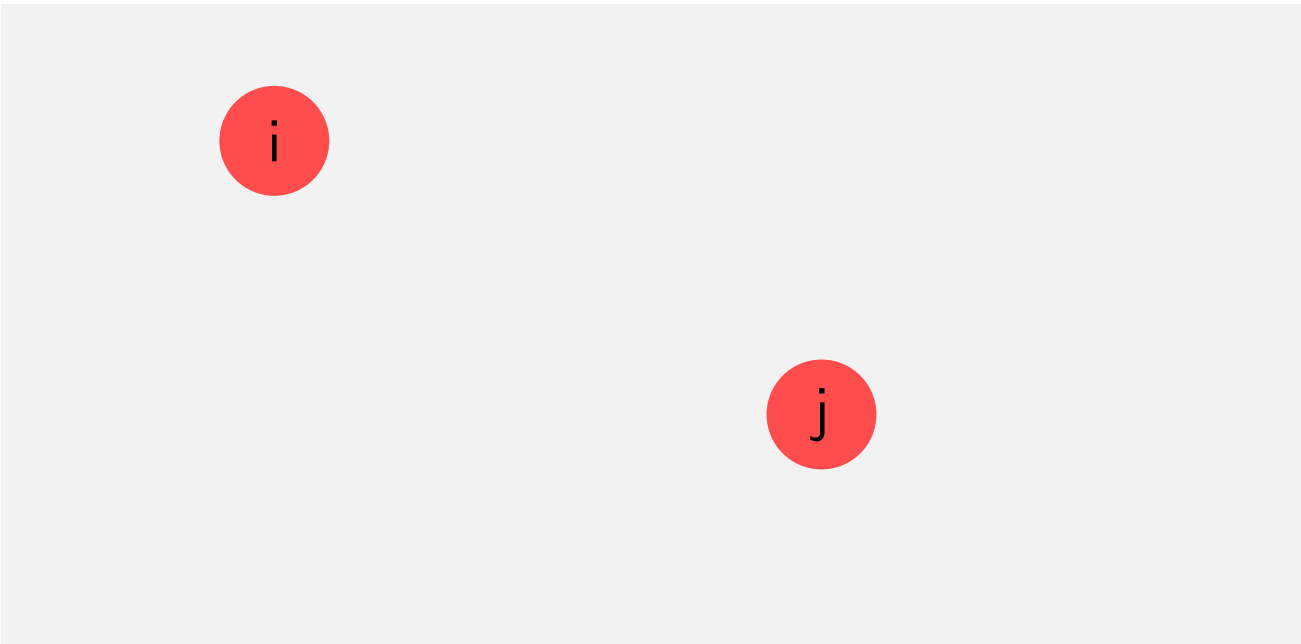
Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.
- U, V: row vectors u_i, v_i are stored at node i.

Repeated SGD updates

- When x_{ij} is available, update so that $x_{ij} \approx u_i v_j^T$.



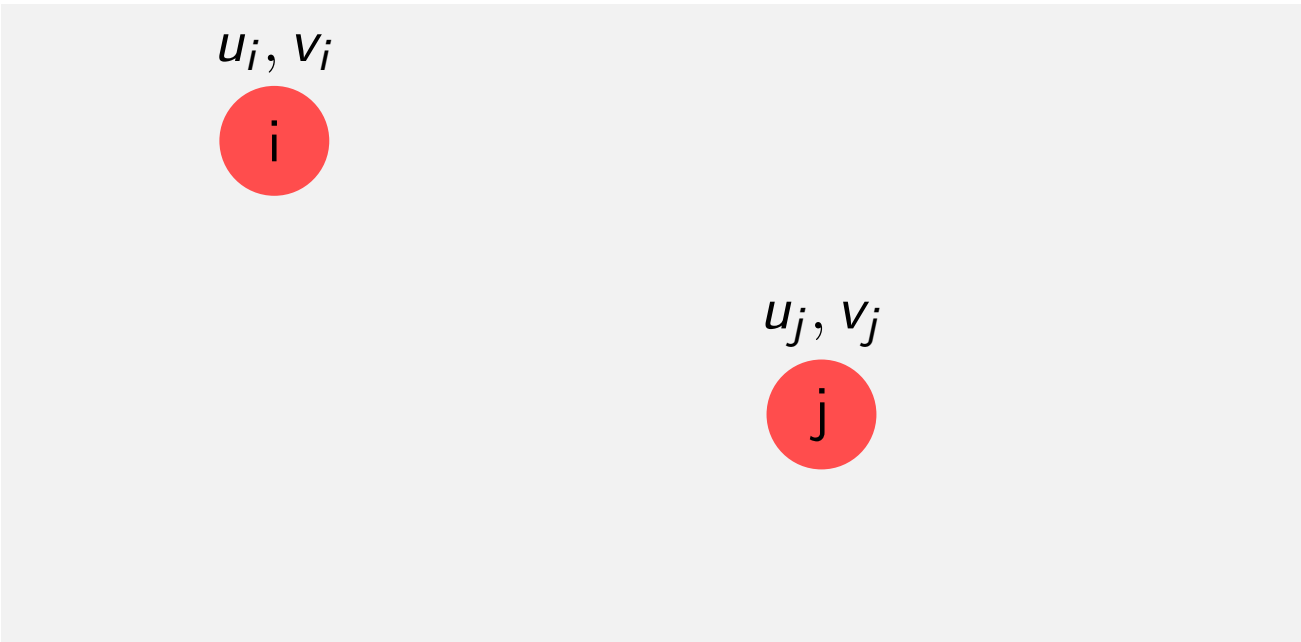
Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.
- U, V: row vectors u_i, v_i are stored at node i.

Repeated SGD updates

- When x_{ij} is available, update so that $x_{ij} \approx u_i v_j^T$.



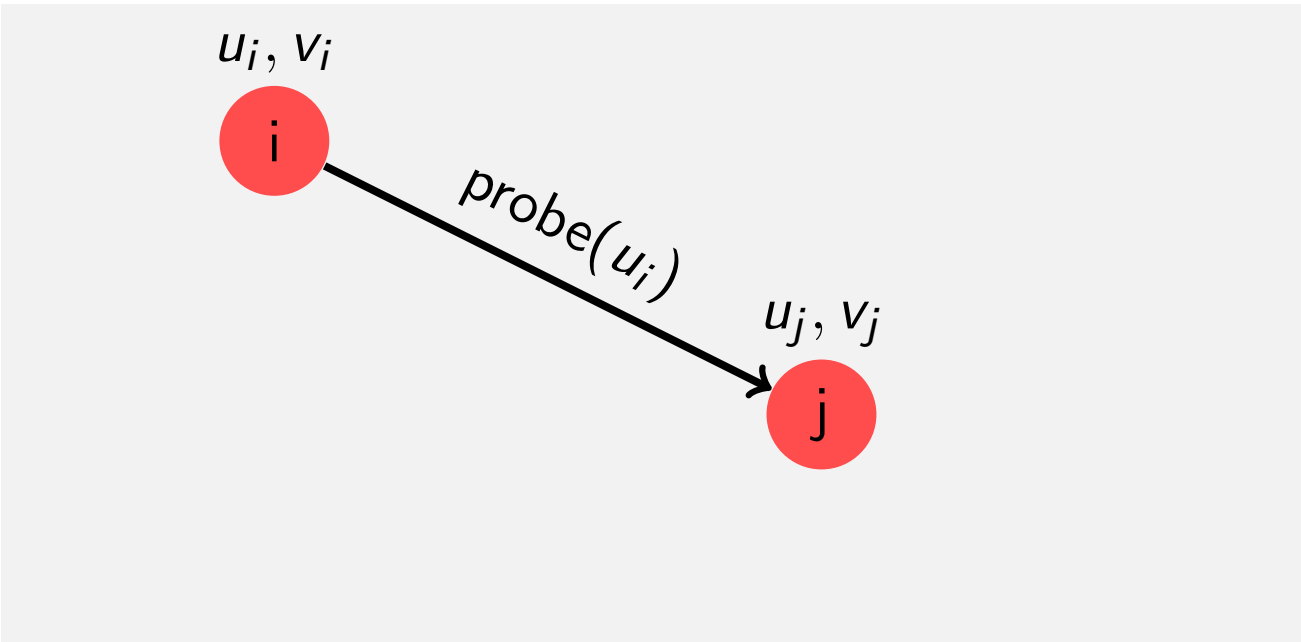
Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.
- U, V: row vectors u_i, v_i are stored at node i.

Repeated SGD updates

- When x_{ij} is available, update so that $x_{ij} \approx u_i v_j^T$.



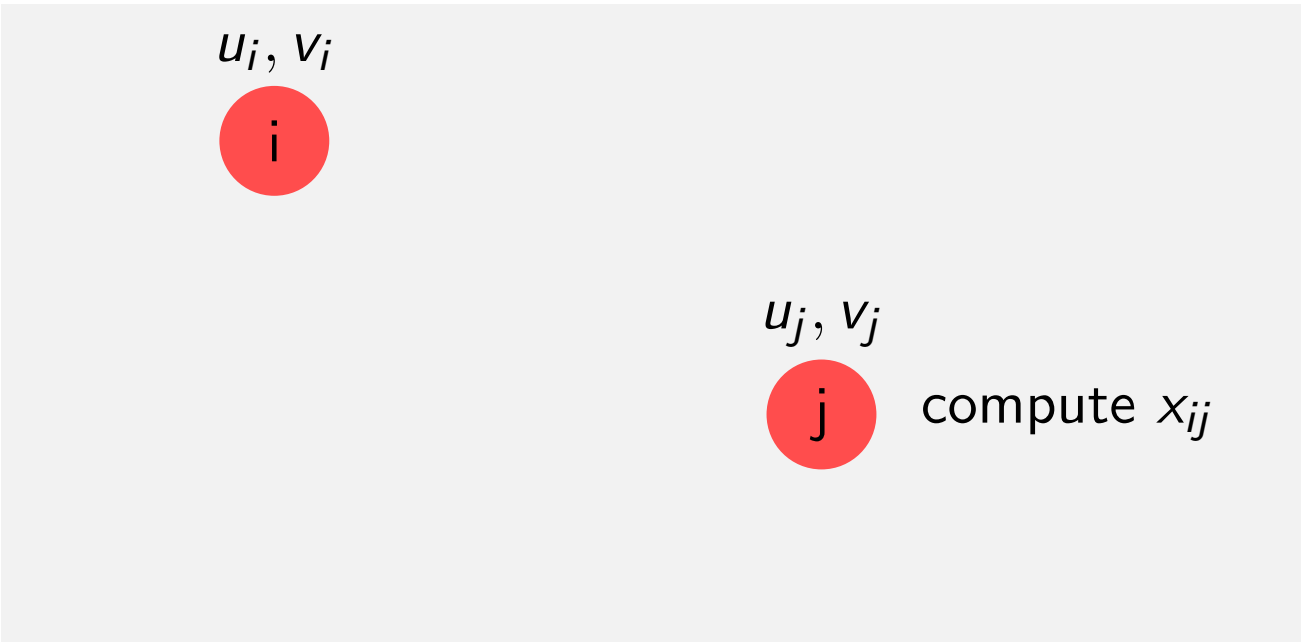
Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.
- U, V: row vectors u_i, v_i are stored at node i.

Repeated SGD updates

- When x_{ij} is available, update so that $x_{ij} \approx u_i v_j^T$.



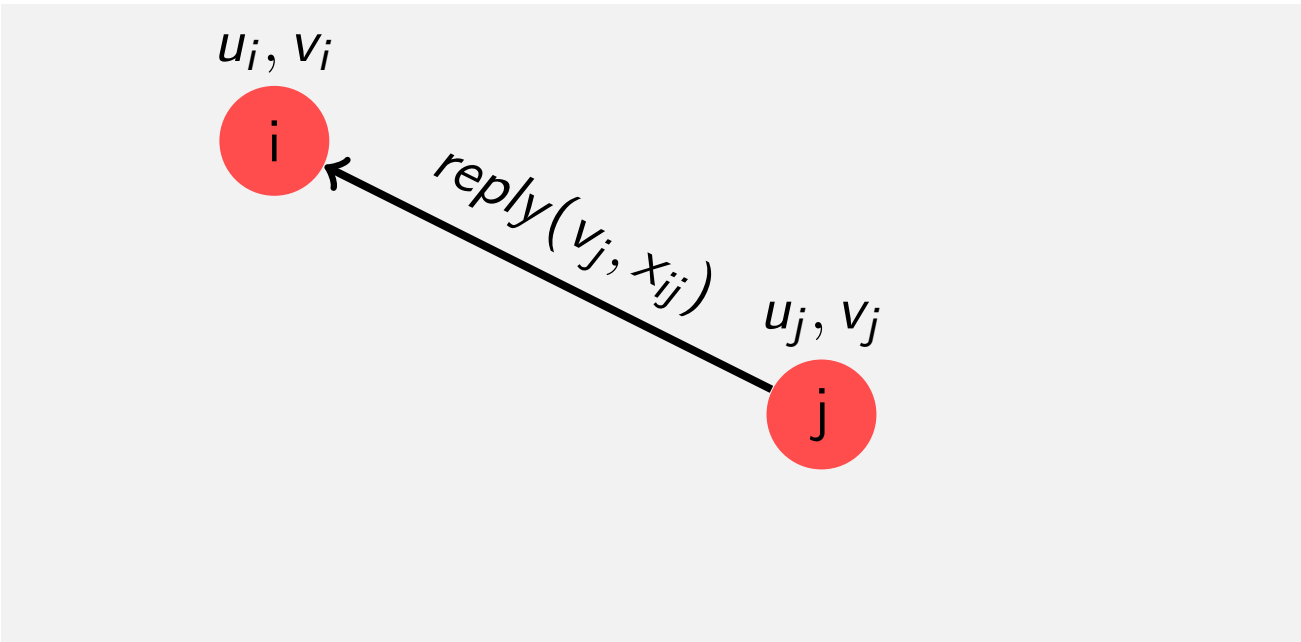
Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.
- U, V: row vectors u_i, v_i are stored at node i.

Repeated SGD updates

- When x_{ij} is available, update so that $x_{ij} \approx u_i v_j^T$.



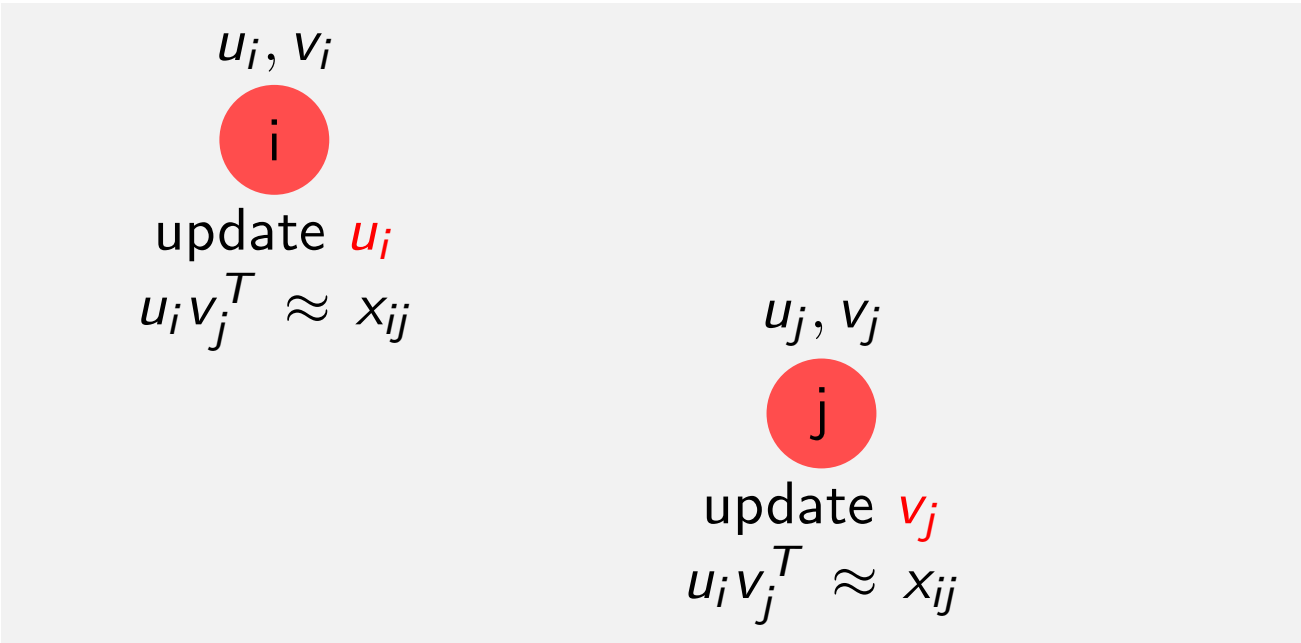
Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.
- U, V: row vectors u_i, v_i are stored at node i.

Repeated SGD updates

- When x_{ij} is available, update so that $x_{ij} \approx u_i v_j^T$.



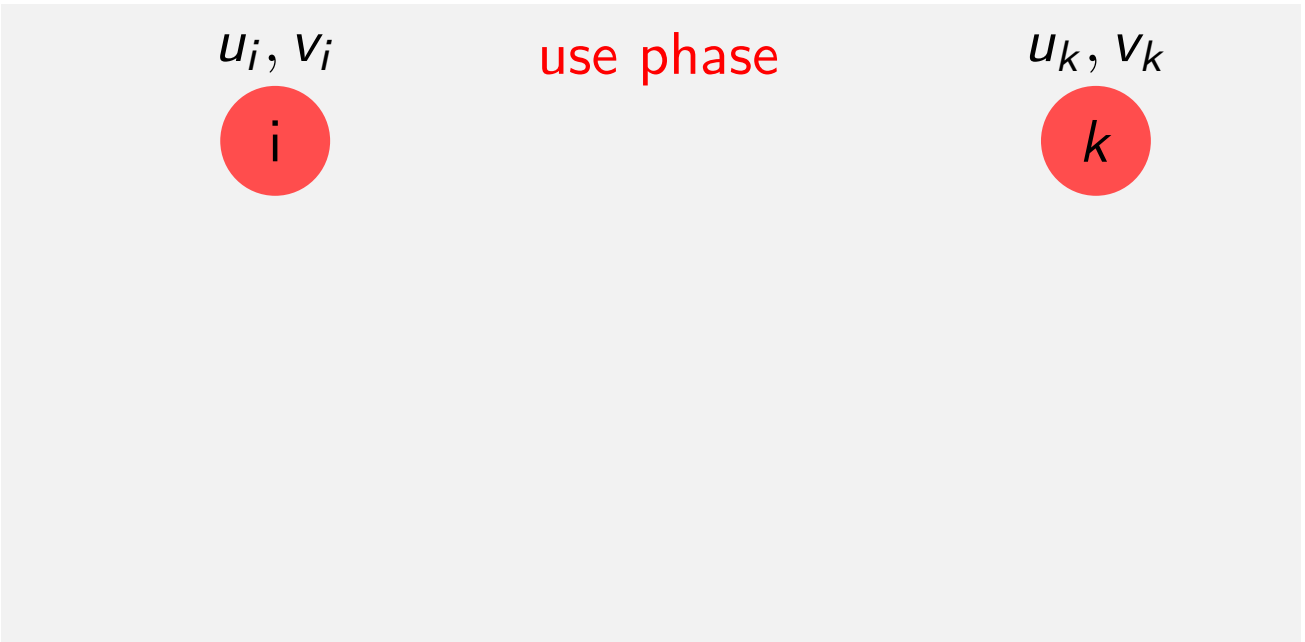
Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.
- U, V: row vectors u_i, v_i are stored at node i.

Repeated SGD updates

- When x_{ij} is available, update so that $x_{ij} \approx u_i v_j^T$.



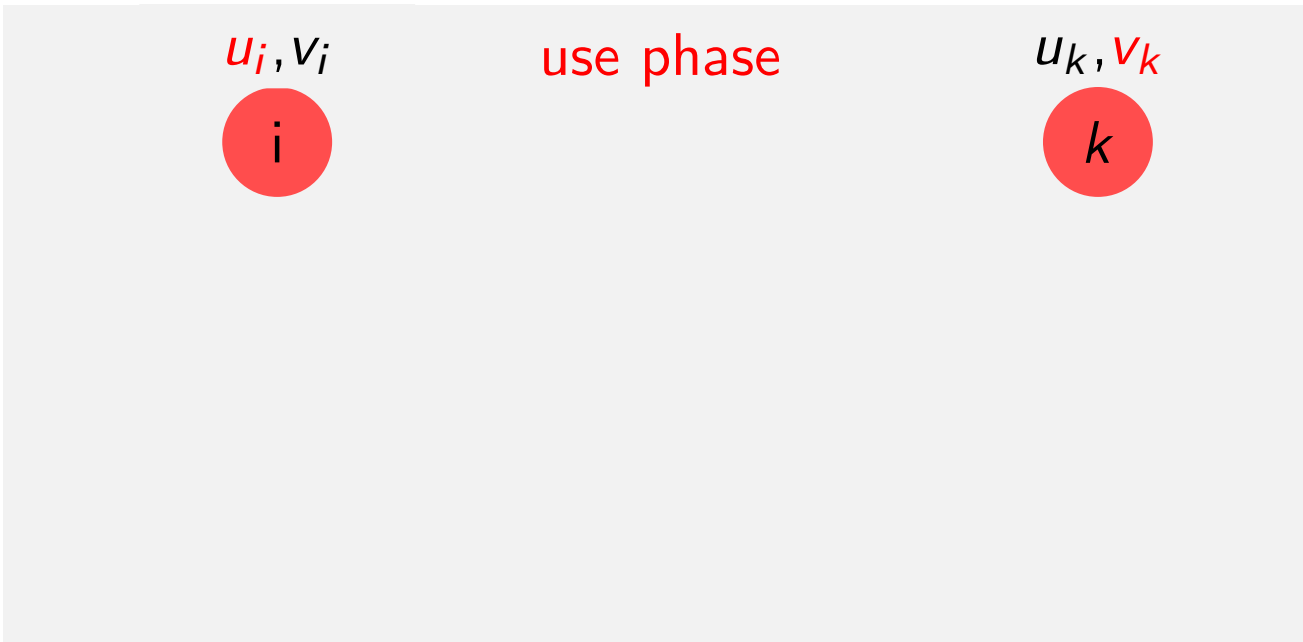
Decentralized Matrix Factorization by Stochastic Gradient Descent

No construction of matrices

- X: measurement x_{ij} is probed by node i.
- U, V: row vectors u_i, v_i are stored at node i.

Repeated SGD updates

- When x_{ij} is available, update so that $x_{ij} \approx u_i v_j^T$.



Decentralized Matrix Factorization by Stochastic Gradient Descent

DMFSGD

All nodes employ the same processing.

- Each node selects k neighbors to communicate with.
- Each node collaborates with one neighbor at each time.

Advantages

- easy to implement
- computationally lightweight
- suitable for large-scale dynamic measurements
- adaptable for various metrics

Experiments and Evaluations

Datasets

	Harvard	Meridian	HP-S3
nodes	226	2500	231
metric	RTT	RTT	ABW
dynamic	Yes	No	No
source	Ledlie et al. NSDI 2007	Wong et al. SIGCOMM 2005	Ramasubramanian et al. SIGMETRICS 2009

*Harvard dataset is collected from Azureus (now Vuze) and contains dynamic measurements with time-stamps.

Experiments and Evaluations

Datasets

	Harvard	Meridian	HP-S3
nodes	226	2500	231
metric	RTT	RTT	ABW
dynamic	Yes	No	No
source	Ledlie et al. NSDI 2007	Wong et al. SIGCOMM 2005	Ramasubramanian et al. SIGMETRICS 2009

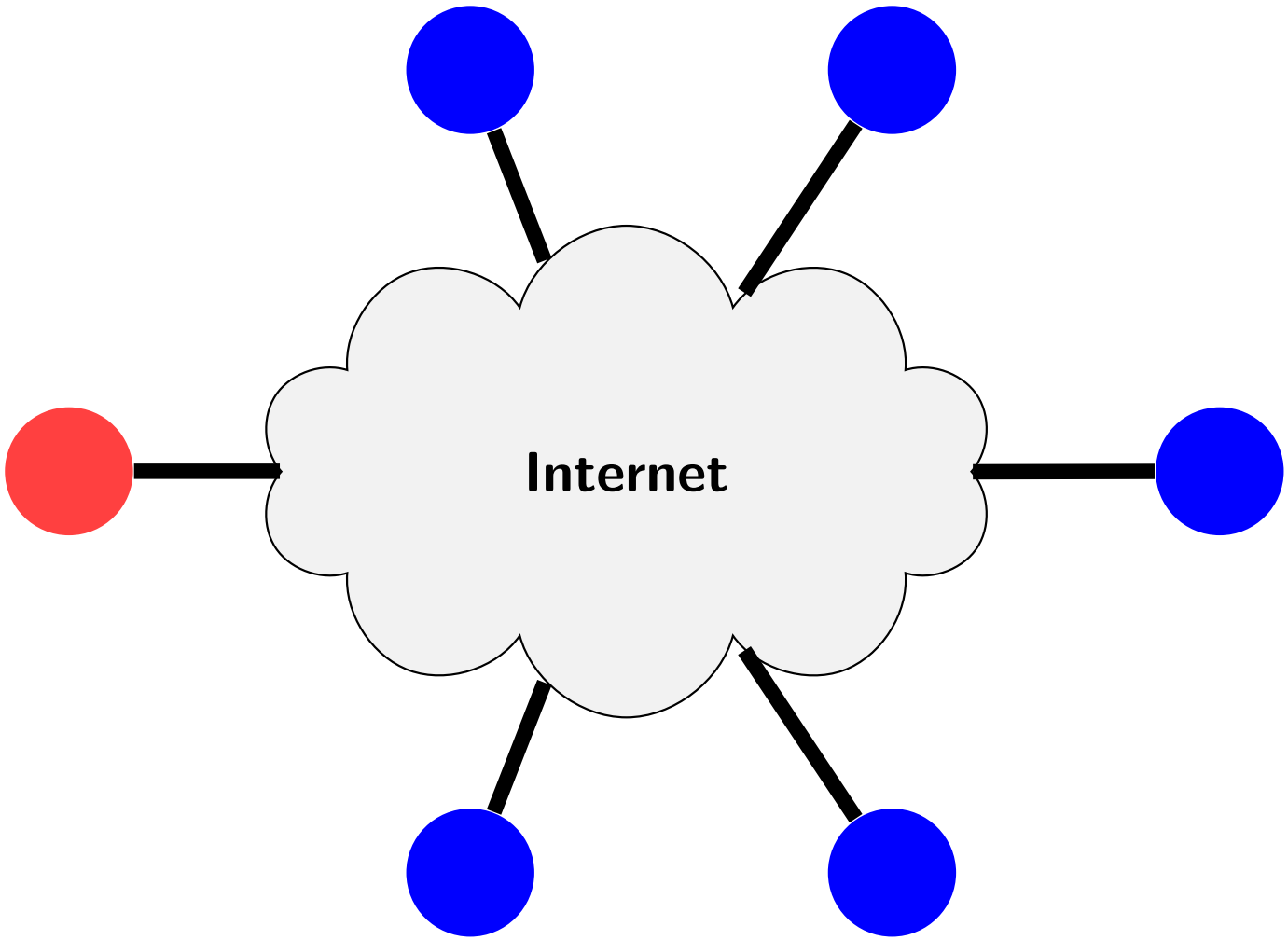
*Harvard dataset is collected from Azureus (now Vuze) and contains dynamic measurements with time-stamps.

$$\text{Accuracy} = \frac{\#\text{ of correct prediction}}{\#\text{ of data}}$$

	Harvard	Meridian	HP-S3
Accuracy	89.4%	85.4%	87.3%

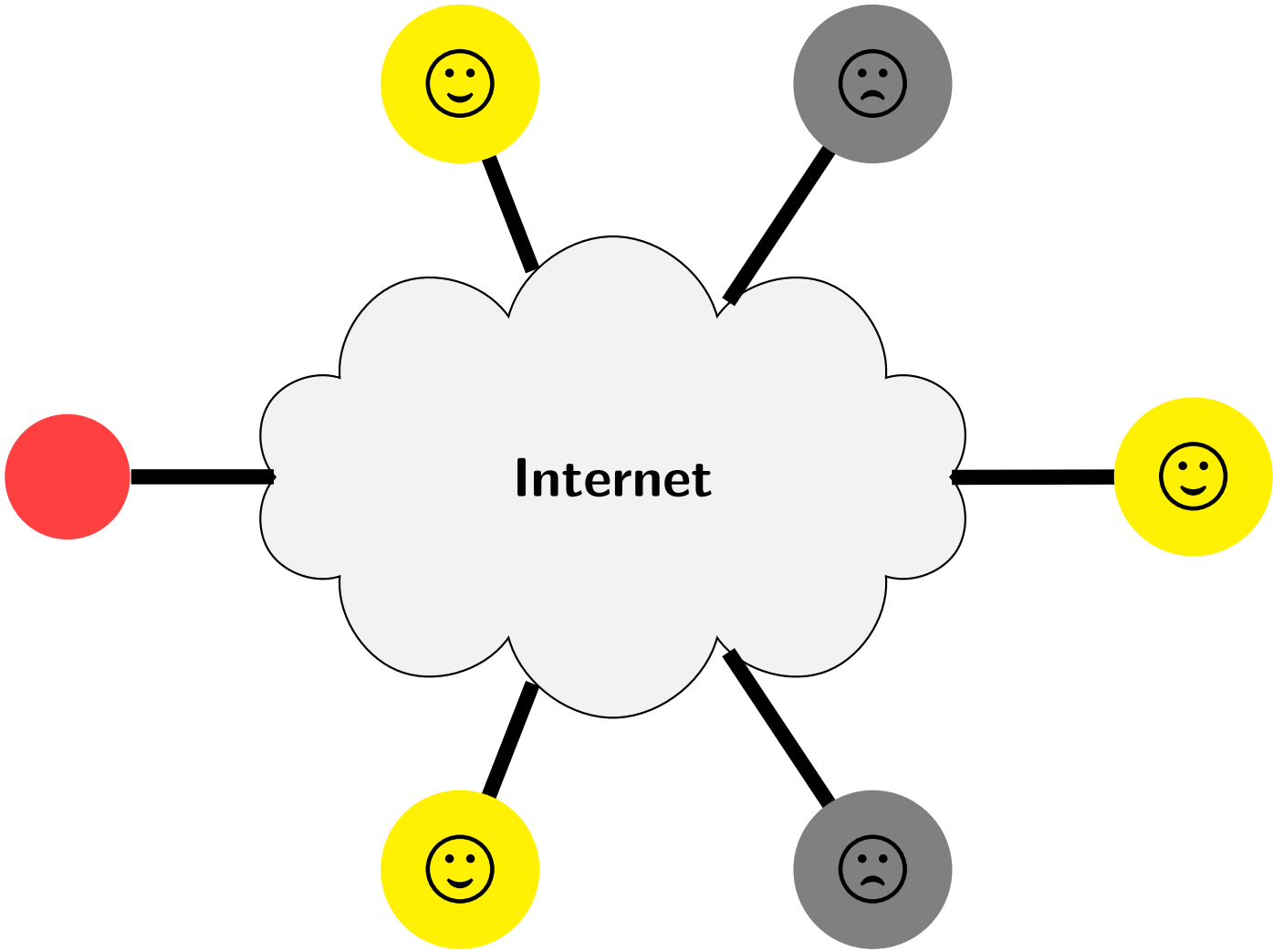
Peer Selection

Find a "good" peer, not the "best" peer!



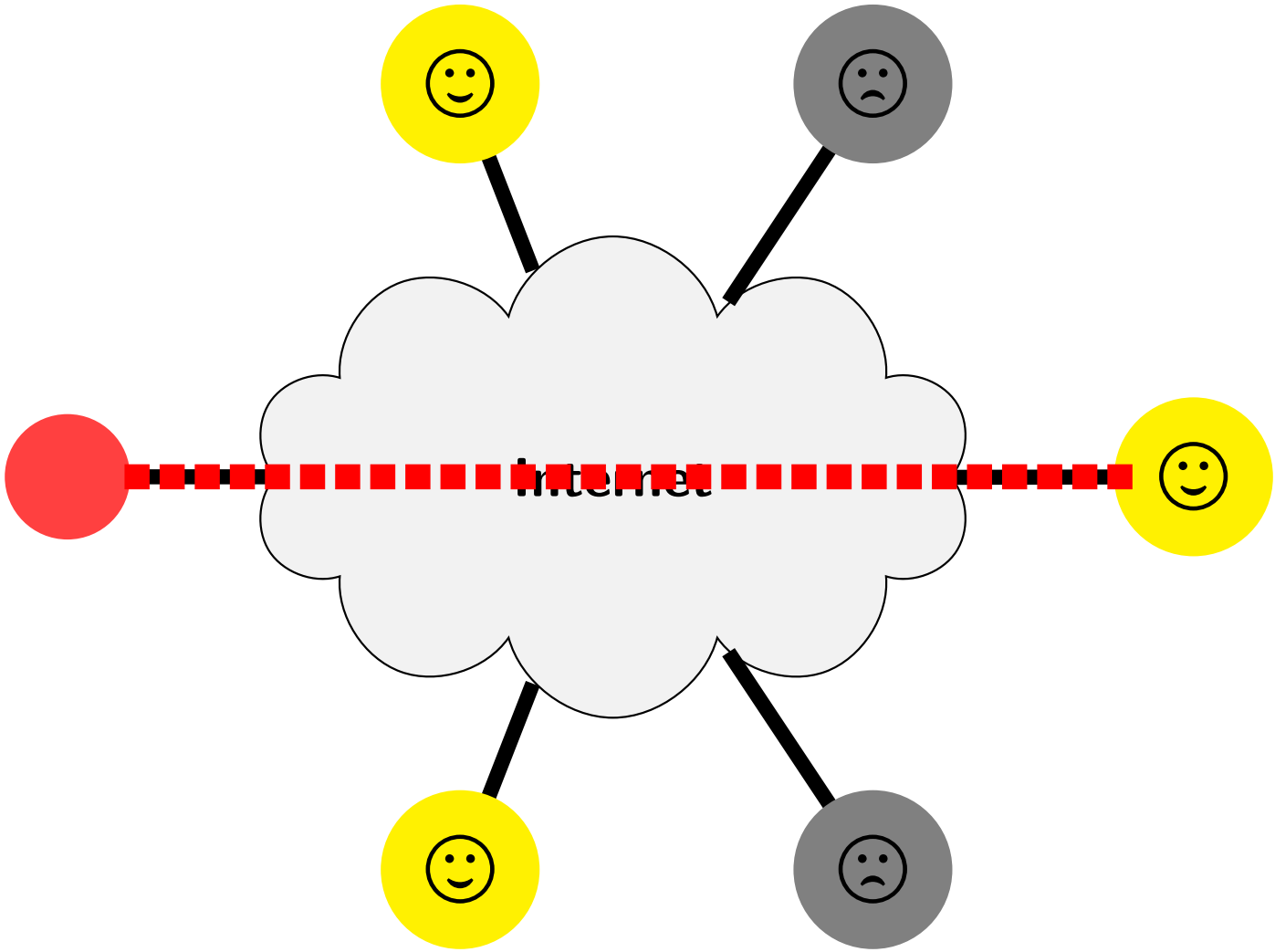
Peer Selection

Some peers are predicted as "good" and some "bad".



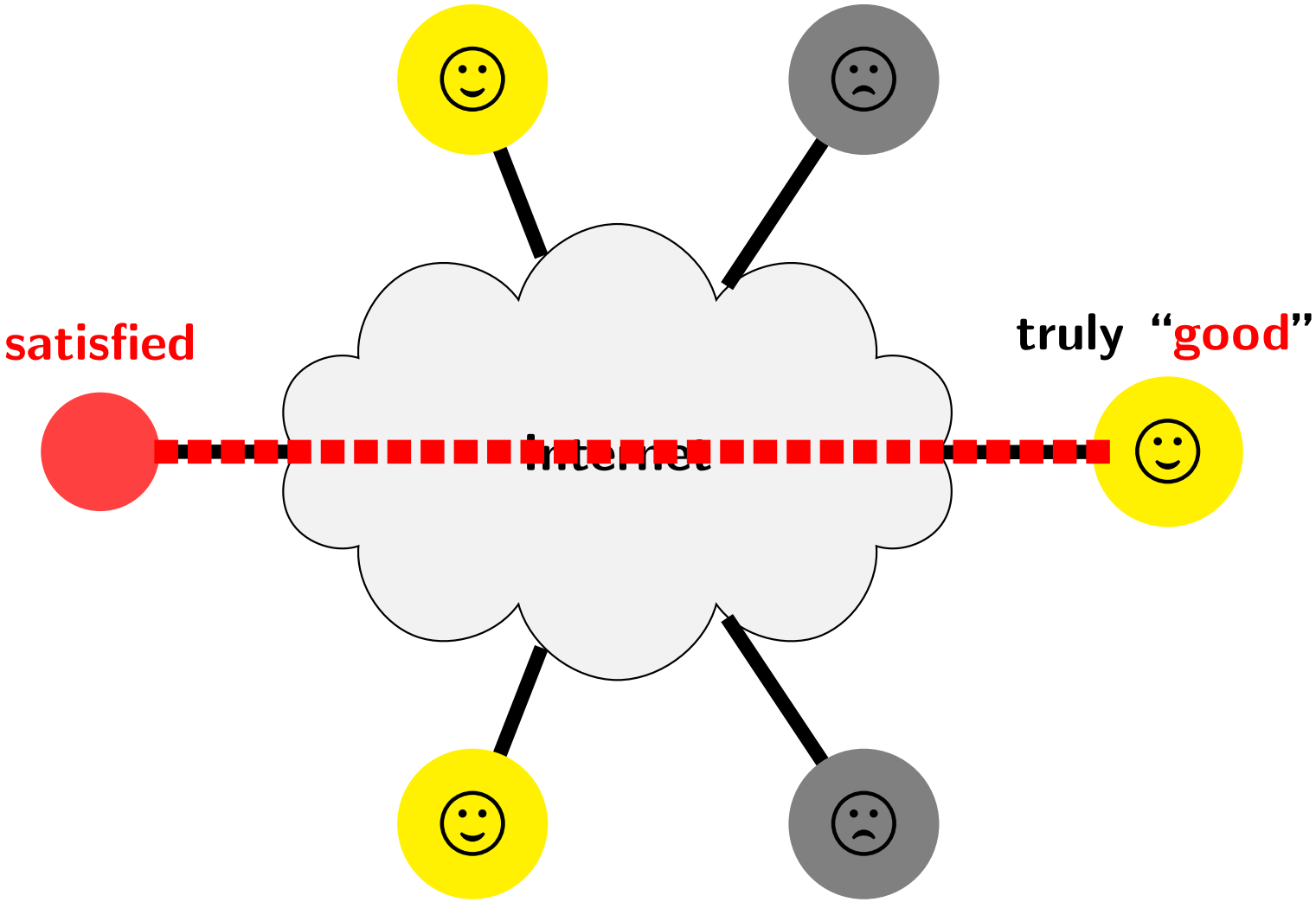
Peer Selection

Select a peer that is predicted as "good".



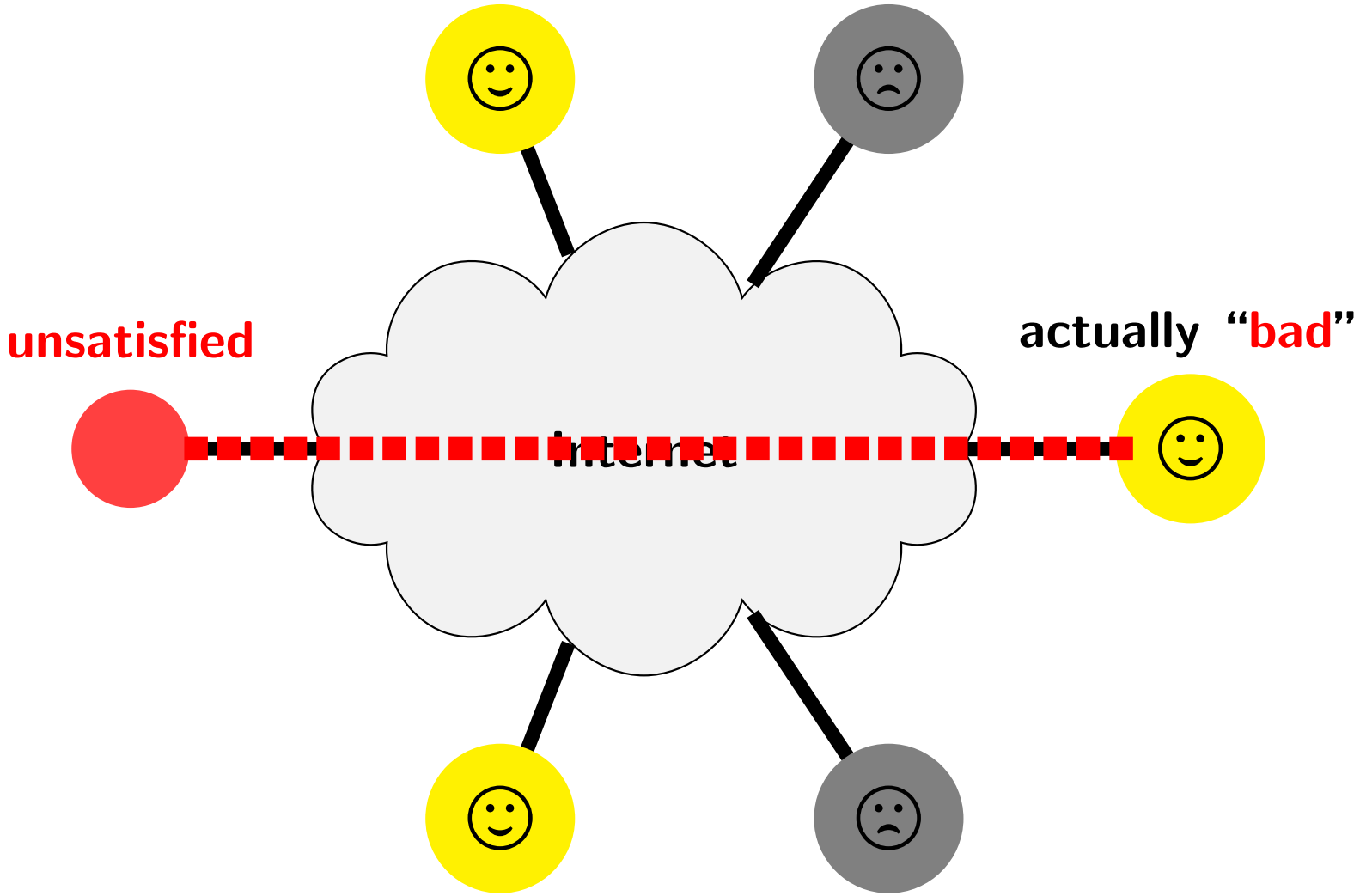
Peer Selection

The node is satisfied if the selected peer is truly "good".



Peer Selection

The node is unsatisfied if the selected peer is actually "bad".



Peer Selection

Evaluation

Count the number of unsatisfied nodes.

Peer Selection

Evaluation

Count the number of unsatisfied nodes.

Methods that are compared with

classification: class-based prediction

Peer Selection

Evaluation

Count the number of unsatisfied nodes.

Methods that are compared with

classification: class-based prediction

- random peer selection

Peer Selection

Evaluation

Count the number of unsatisfied nodes.

Methods that are compared with

classification: class-based prediction

- random peer selection
- regression: value-based prediction
  - Predict values of some metric by our DMFSGD algorithm.
  - Select the predicted best peer for each node.
  - Check if the selected peers are truly "good".

Peer Selection

Evaluation

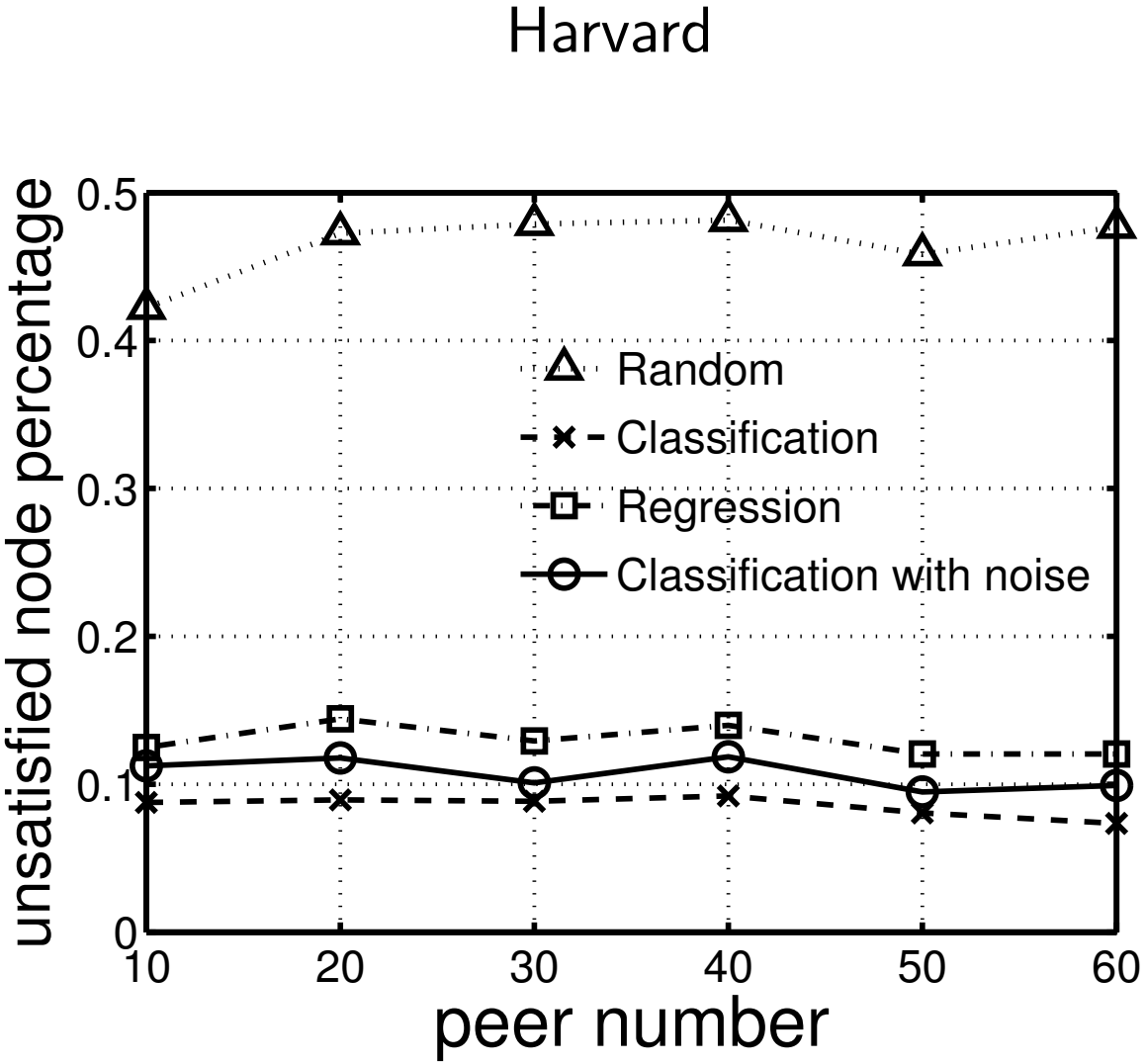
Count the number of unsatisfied nodes.

Methods that are compared with

classification: class-based prediction

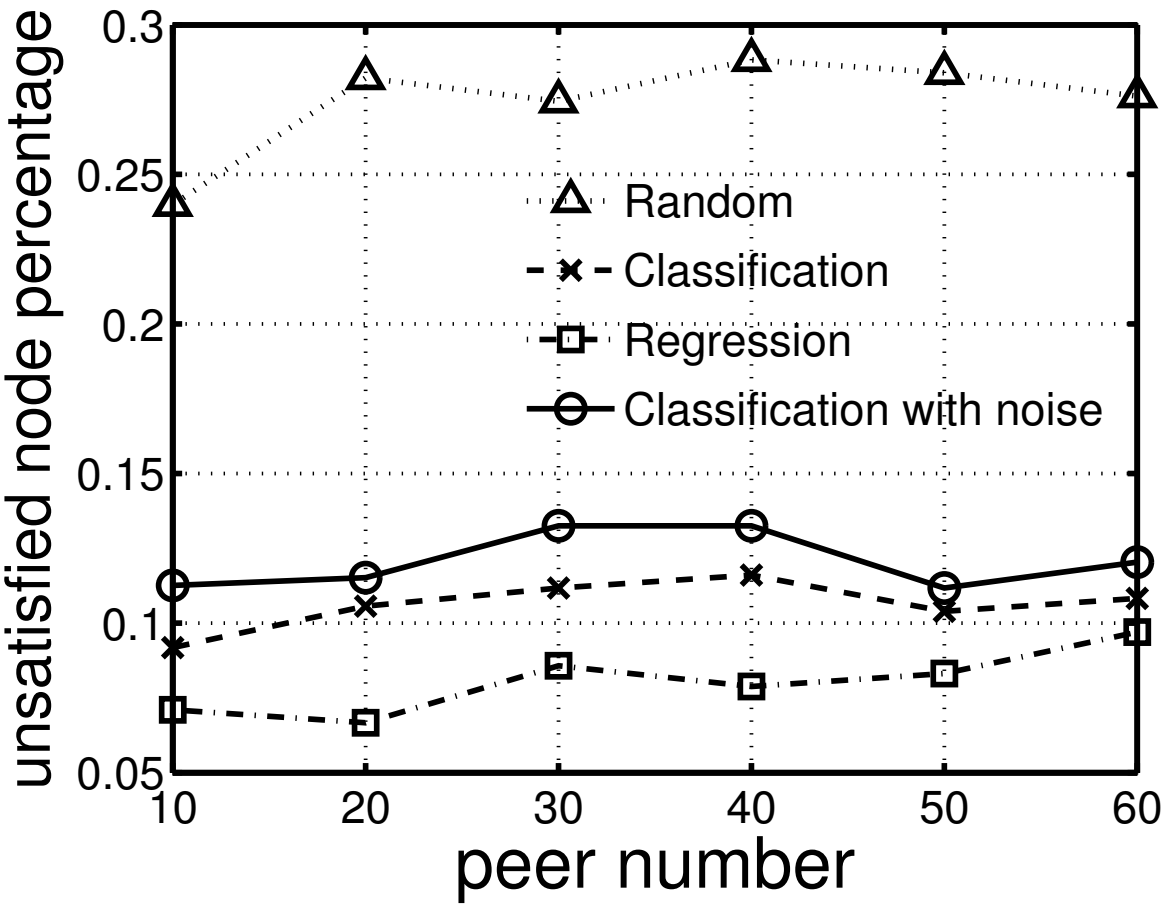
- random peer selection
- regression: value-based prediction
  - Predict values of some metric by our DMFSGD algorithm.
  - Select the predicted best peer for each node.
  - Check if the selected peers are truly "good".
- classification with noise
  - Overall 15% erroneous labels were simulated.

Peer Selection



Peer Selection

HP-S3



Conclusions and Future Work

Decentralized Prediction of Network Performance Classes

- **Class**-based Performance Representation
- Formulation as **Matrix Completion**
- **DMFSGD**: a **decentralized matrix factorization** algorithm by stochastic gradient descent
  - accurate and scalable
  - generic to deal with various metrics
  - robust against erroneous measurements
  - usable on real Internet applications

Future Work

- Multiclass classification

Université de Liège

Acknowledgement

We wish to thank

- Dr. Ramasubramanian for providing the HP-S3 dataset;
- our shepherd Augustin Chaintreau;
- the anonymous reviewers;
- project FP7-Fire ECODE.

Thank you for listening. Any questions?

Low-Rank Matrix Factorization

$$(U, V) = \arg\min L(X, U, V, W, \lambda)$$
$$= \arg\min \sum_{i,j=1}^{n} w_{ij} l(x_{ij}, u_i v_j^T) + \lambda \sum_{i=1}^{n} u_i u_i^T + \lambda \sum_{i=1}^{n} v_i v_i^T$$

- $(U, V) = \{(u_i, v_i),\ i = 1, \ldots, n\}$
- $w_{ij} = 1$ if x_{ij} is known and 0 otherwise
- l: loss function that penalizes the difference between x and $\hat{x}$
  - square loss function: $l(x, \hat{x}) = (x - \hat{x})^2$;
  - hinge loss function: $l(x, \hat{x}) = \max(0, 1 - x\hat{x})$;
  - logistic loss function: $l(x, \hat{x}) = \ln(1 + e^{-x\hat{x}})$.
- λ: regularization coefficient

Université de Liège

Parameter Sensitivity

parameter	tested values
learning rate η	0.001, 0.01, 0.1, 1
regularization coefficient λ	0.001, 0.01, 0.1, 1
rank r	3, 10, 20, 100
loss function l	hinge, logistic
neighbor number k	5, 10, 30, 50 (Harvard and HP-S3) 16, 32, 64, 128 (Meridian)
classification threshold τ	10%, 25%, 50%, 75%, 90% (portion of good-performing)

*chosen value

Insensitive because the inputs are either 1 or -1.

Université de Liège

Robustness Against Erroneous Measurement

Source of Erroneous Measurements

- inaccurate measurement techniques
- network anomaly

Errors Type

1. Flip near τ
2. Underestimation bias
3. Flip randomly
4. Good-to-Bad