A Bayesian method for matching two similar graphs without seeds

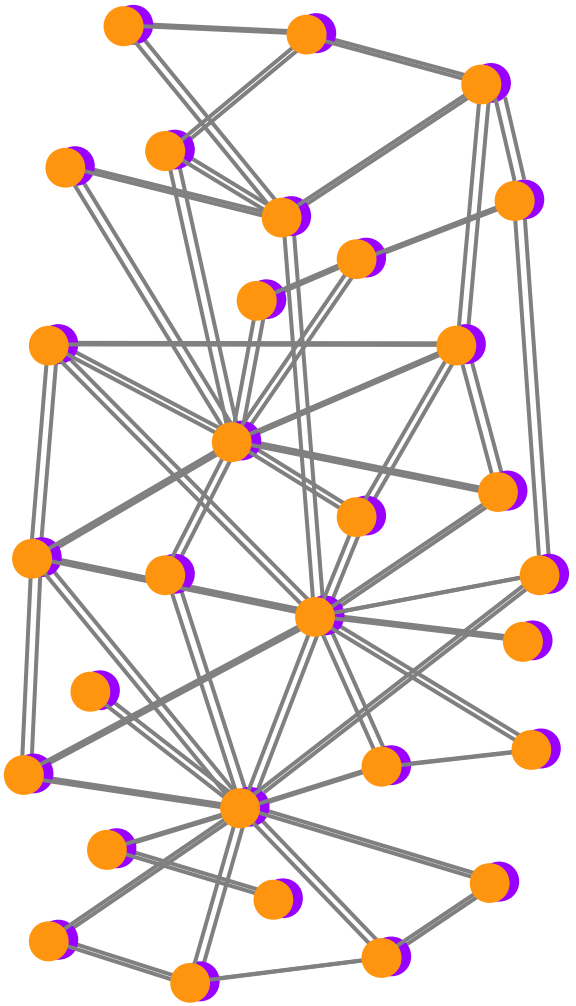
Pedram Pedarsani (EPFL)
Matthias Grossglauser (EPFL)
Daniel R. Figueiredo (UFRJ)

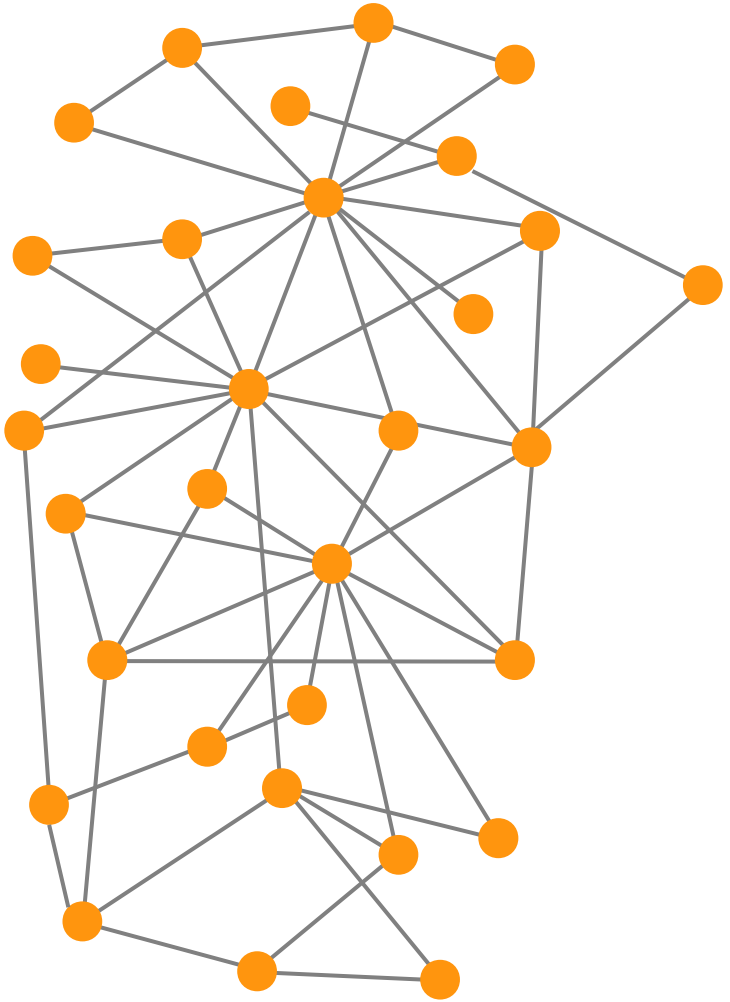
IEEE Allerton Conference 2013

Adapted by Daniel R. Figueiredo

Approximate Graph Matching

- Match nodes two *structurally related* graphs







Can we match the nodes?

Fundamental Questions

- When is approximate graph matching feasible?



- Assume graph model (structure)
- Consider model for graph similarity
- Provide conditions for finding correct matching

- How to match nodes of two graphs in practice?

- Polynomial time algorithm to find correct matching
- Settle for *mostly correct* matching

Applications

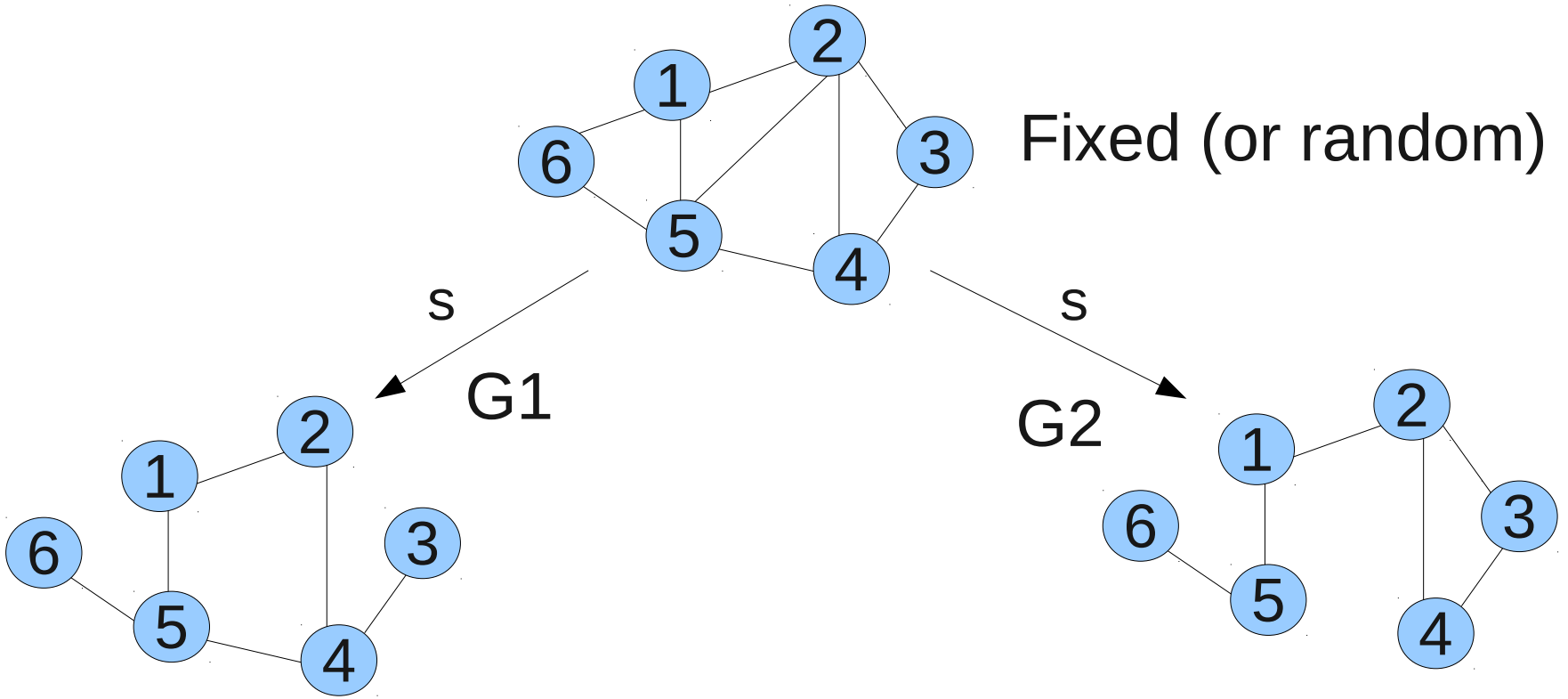
- Computer vision: object recognition
  - match part of segmented images
- Biology: identifying genes or protein functions
  - match regulatory gene or protein interaction networks
- Social networks: breeching privacy
  - identifying nodes using network structure

Many applications require matching similar structures

Edge Sampling Model

- Model for graph similarity
- Consider fixed graph G
  - could be realization of G(n,p)
- Sample every edge from G with probability s, iid.
- G1 ~ G(s) and G2 ~ G(s)
  - G1 and G2 are two independent samples from the **same** sampled G
- Structural correlations between G1 and G2 controlled by parameter s
  - s=1 : isomorphism problem, s=0 no structure!
  - preserves nodes, randomness only on edges

Edge Sampling Example





- **Problem:** Match nodes of G1 and G2
- **Q1:** When is it possible?
- **Q2:** How to do it?

Theoretical Formulation

- Consider a mapping π between nodes in G1 and G2
  - n! possible mappings
- Consider an error function of a mapping
  - $\Delta(\pi)$: number of edges that appear in G1 but not G2 (and vice-versa)
- Let π_0 be correct mapping. Conditions such that



$$P\{\pi_0 \text{ unique min of } \Delta(\pi)\} \to 1$$

- Adversary can then correctly match using just structure of graphs
  - inspect all mappings, choose one with lowest error

Theoretical Result

- nps: E[degree] of G1,2
- threshold for aug(G)=1
- Thm [**PG'12**]: For G(n,p; s) matching if

$$nps\frac{s^2}{2-s} = 8\log n + \omega(1)$$

then correct permutation minimizes $\Delta(\ )$, aas.

Penalty for difference G1-G2

"growing slowly"

- Two pieces of bad news
  - surprisingly weak condition: avg degree of $G_{1,2}$ growing faster than log n is sufficient
  - decrease with s only quadratically

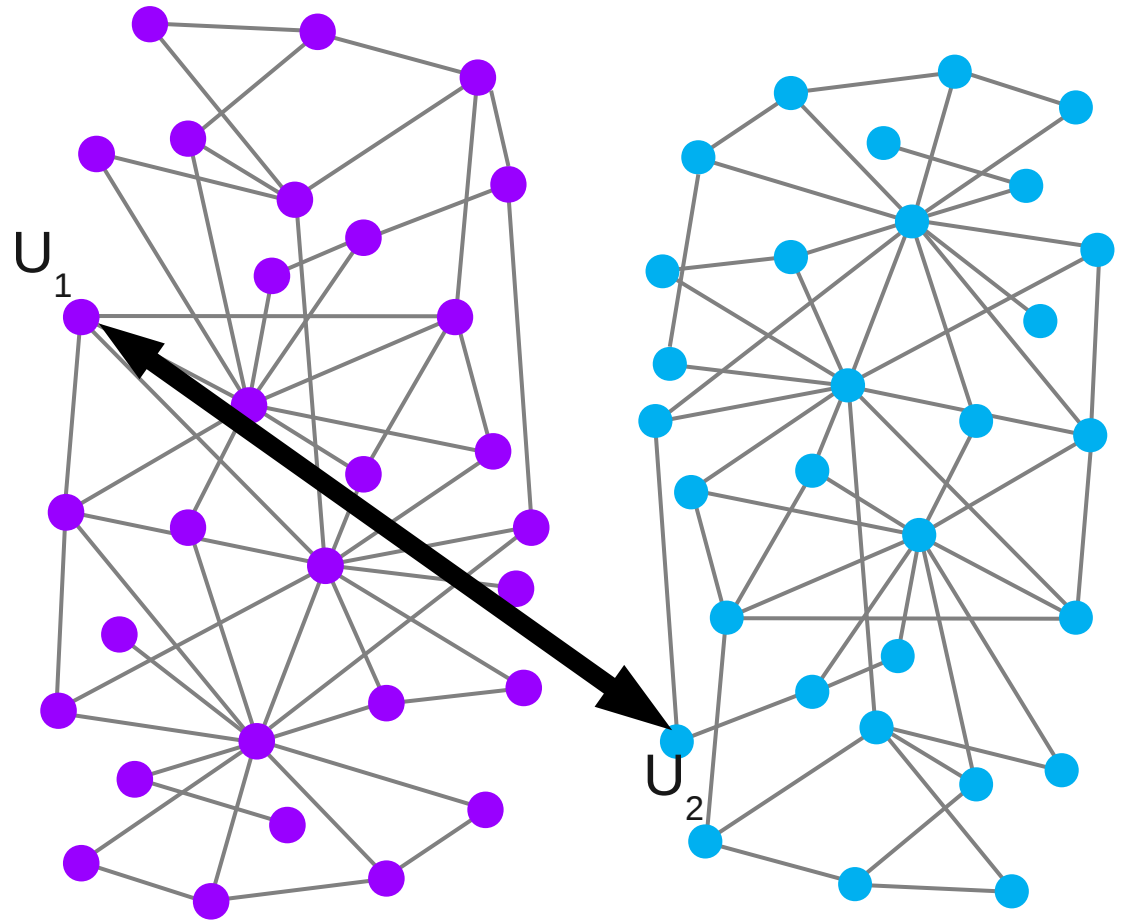
But in Practice?

- Previous result is theoretical
  - unconstrained computational power (n! mappings)
  - does not help us find the right mapping



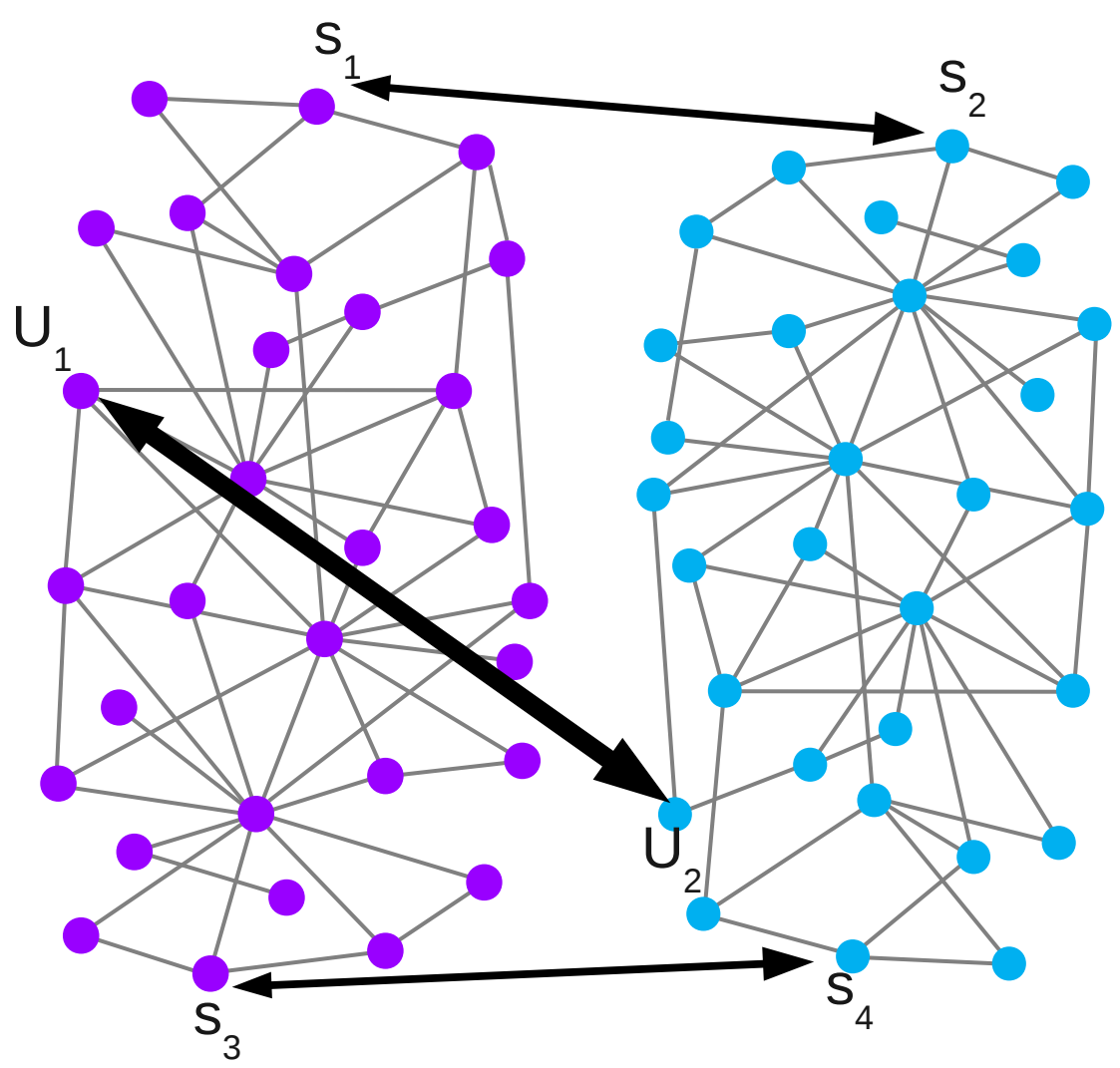
- Idea: Bayesian framework based on fingerprint of nodes
- Compute confidence of pairwise matchings
- Reduce to maximum weighted bipartite matching problem
- Iterative and incremental algorithm (produce evidence on the run)

Using Structural Evidence



- $P[U_1 = U_2]$: two nodes chosen at random
  - 1/n if no other information
- What if degree $D_1 = 100$, $D_2 = 97$?
- What if degree $D_1 = 100$, $D_2 = 2$?
- Use degree as structural evidence

Distances as Evidence



- Suppose s_1 is mapped to s_2
  - (s_1, s_2): anchor pair
- X_{11}: distance between U1 and s_1
- X_{21} : distances between U2 and s_2
- Will consider multiple anchor pairs
- Anchor pair match can be wrong!

Evidence Probability

- Consider fingerprints of nodes U1 and U2
  - $F_{U1} = (D_1, X_{11}, X_{12}, ..., X_{1s})$ ← s anchor pairs (distances)
  - $F_{U2} = (D_2, X_{21}, X_{22}, ..., X_{2s})$
  - $X_{\{1,2\}i}$, distance from U1, U2 and anchor i
- Prob. of observing these fingerprints
  - $P[F_{U1}, F_{U2} \mid U1 = U2]$ ← U1 = U2: nodes correspond to one another
- Assume conditional independence between evidence pairs
  - $= P[D_1, D_2 \mid U1=U2]\ P[X_{11}, X_{21} \mid U1=U2] ... P[X_{1s}, X_{2s} \mid U1=U2]$

Evidence Probability



- How to calculate $P[D_1, D_2 \mid U1=U2]$ or $P[X_{11}, X_{21} \mid U1=U2]$?
- Need a sampling model and prior distribut.

- Consider a fixed but hidden G
  - assume we know degree, distance distribution
- Edge sampling model to generate G1 and G2
  - each edge in G sampled iid with prob s
- Can now compute $P[D_1, D_2 \mid U1=U2]$
  - $P[D_1, D_2 \mid U1=U2, D]$ is a product of binomials with parameters D, s and values D_1 and D_2
  - uncondition D by using prior of G

Match Probability

- Same reasoning for $P[F_{U1}, F_{U2} \mid U1 != U2]$
  - when nodes U1 and U2 are do not correspond
- Using both and prior $P[U1 = U2] = 1/n$
- Apply Bayes rule to obtain

$$P[U1 = U2 \mid FP1, FP2]$$

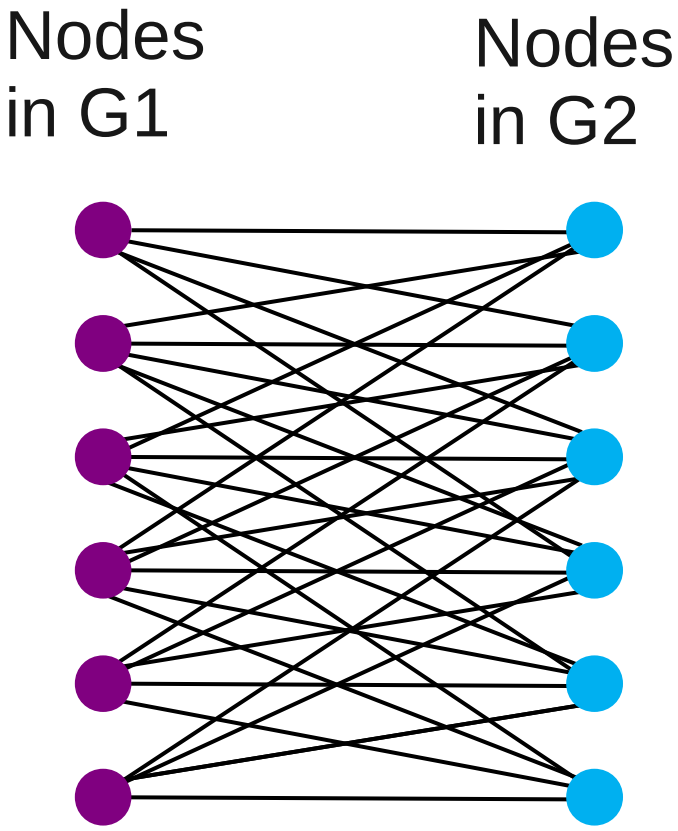
Prob of match given fingerprints!

- M_i : indicator for anchor pair i correctly mapped
  - $P[M_i = 1]$: prob of anchor pair i correctly mapped

$$P[U1 = U2 \mid FP1, FP2, M_1, \ldots, M_s]$$

  - use priors to marginalize out M_i

Weighted Bipartite Matching



- Complete bipartite graph
- Weight of edge (U1, U2) = log P[U1 = U2 | FP1, FP2]
- Assuming independence, P[all matched pairs | all evidence] =

  Π P[matched pair | evidence pair]
  - maximum weight matching = log (matching with highest probability)
- compute maximum weight matching
  - Hungarian algorithm $O(n^3)$

The Algorithm

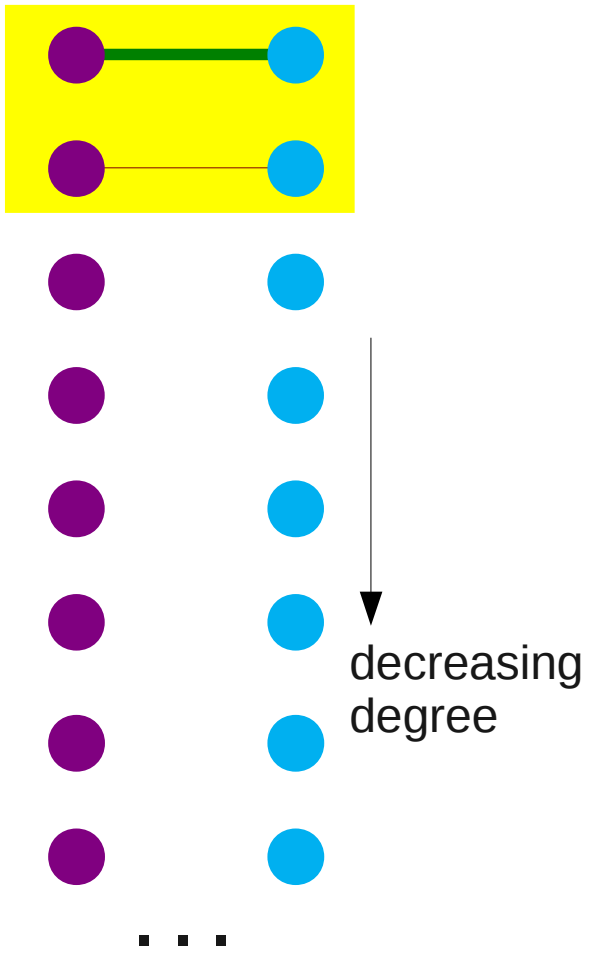
- Idea: generate and use evidence on the run
  - allows matching to change
- Algorithm proceed in phases
  - in phase i, consider 2^i nodes to match
  - bipartite graph has only 2^i nodes
- Candidate nodes in phase i are the highest degree nodes of each graph
- Use half of matched nodes as anchors for next phase
  - best half: matches with highest edge weight
- In phase i>1, we use 2^{i-2} seeds as evidence
  - edge weight from phase i-1 used as prior for correct matched seed in phase i

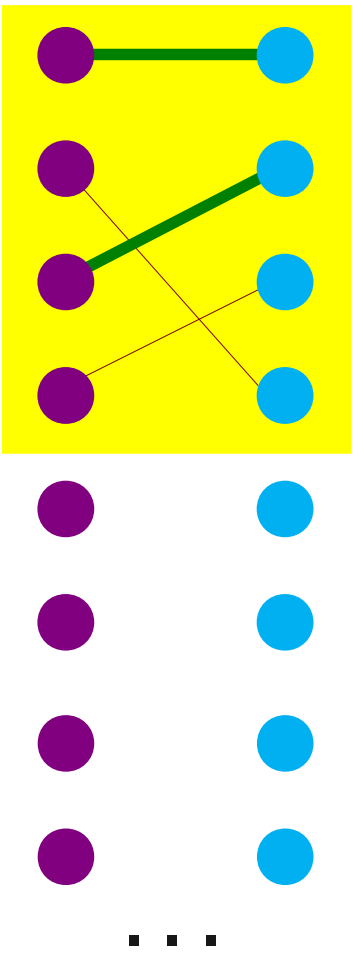
Illustration of Algorithm

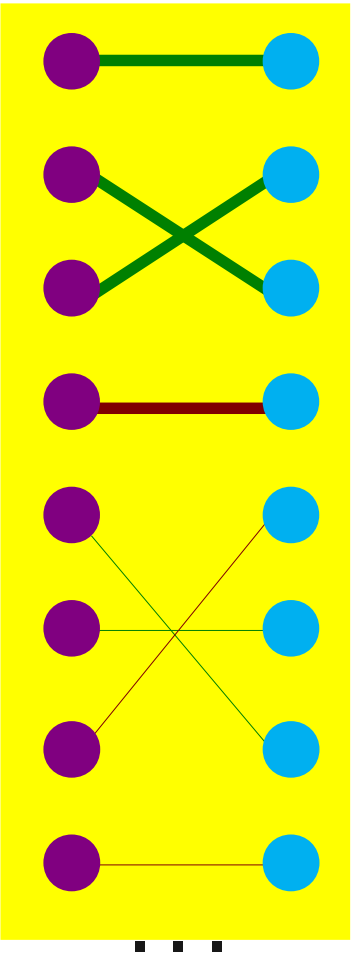
Phase 1:
2 candidates
0 seeds used
1 seed prod.

Phase 2:
4 candidates
1 seed used
2 seeds prod.

Phase 3:
8 candidates
2 seeds used
4 seeds prod.







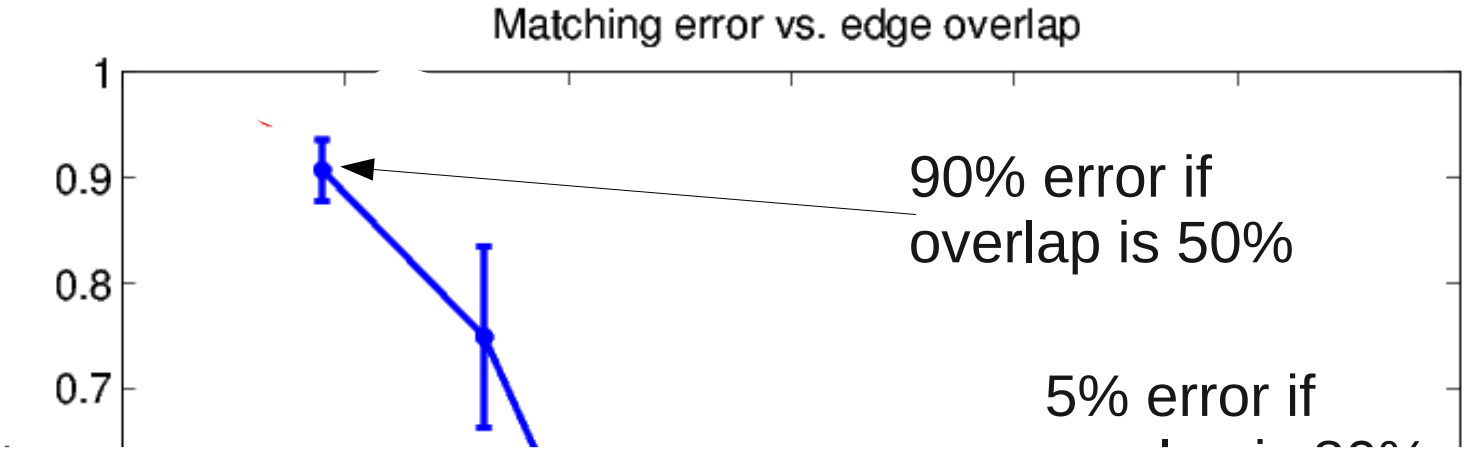
. . .

Green: correct
Red: incorrect
Thick: highest weight

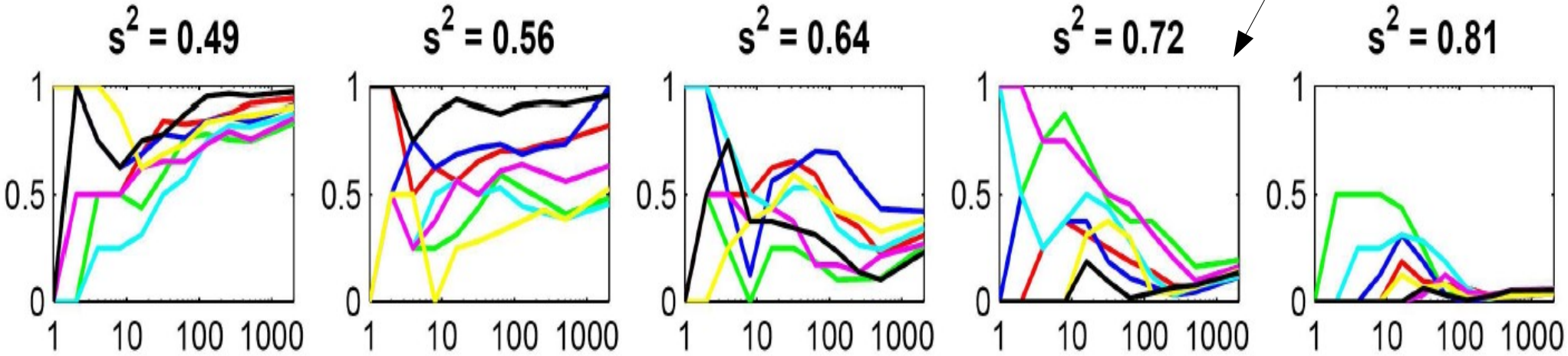
Evaluation

- Email exchange network among EPFL users
  - Social network, week timescale
- Experiment 1:
  - accumulate network for 5 weeks (2024 nodes, 25K edges)
  - edge sample network twice for different s values
- Experiment 2:
  - accumulate network for 10 weeks (considering only nodes that appear in all weeks)
  - time shifted accumulation gives second network, overlap of 9,8,..., 1 week
  - No explicit edge sampling, s estimated from dataset based on overlapped edges

Evaluation: Experiment 1



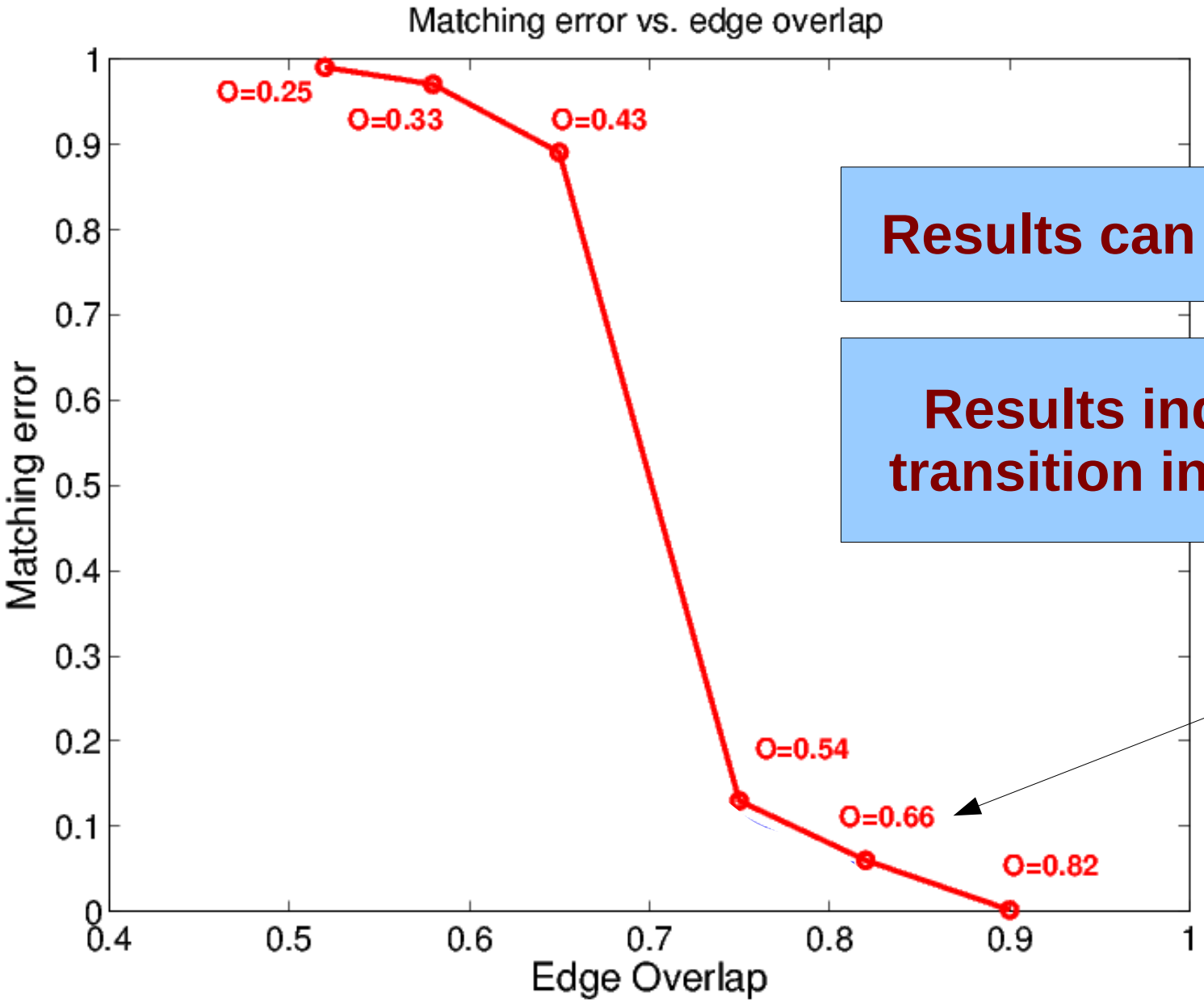
Run time performance for different samples



Edge Overlap

Expected fraction of edges that appear in both G1 and G2

Evaluation: Experiment 2



Results can be very good!

Results indicate sharp transition in edge overlap

Time overlap

Expected fraction of edges that appear in both G1 and G2

Conclusions

- Network privacy seems hard
  - in theory and practice!
  - two networks matched using just structure (no *other* side information)
  - conditions on avg. degree and edge overlap not unrealistic
- Principled graph matching algorithm
  - sampling model allows for Bayesian formulation and bipartite matching
  - incremental and iterative approach: generate and use more evidence with uncertainty
  - performance is good if above threshold

Thank You



- Questions or comments?



- contact: daniel@land.ufrj.br

Collaborators:

Matthias Grossglauser
Pedram Pedarsani

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE