the blue-red matching problem: approximations, exact solutions, and applications

aris pagourtzis (ntu athens)

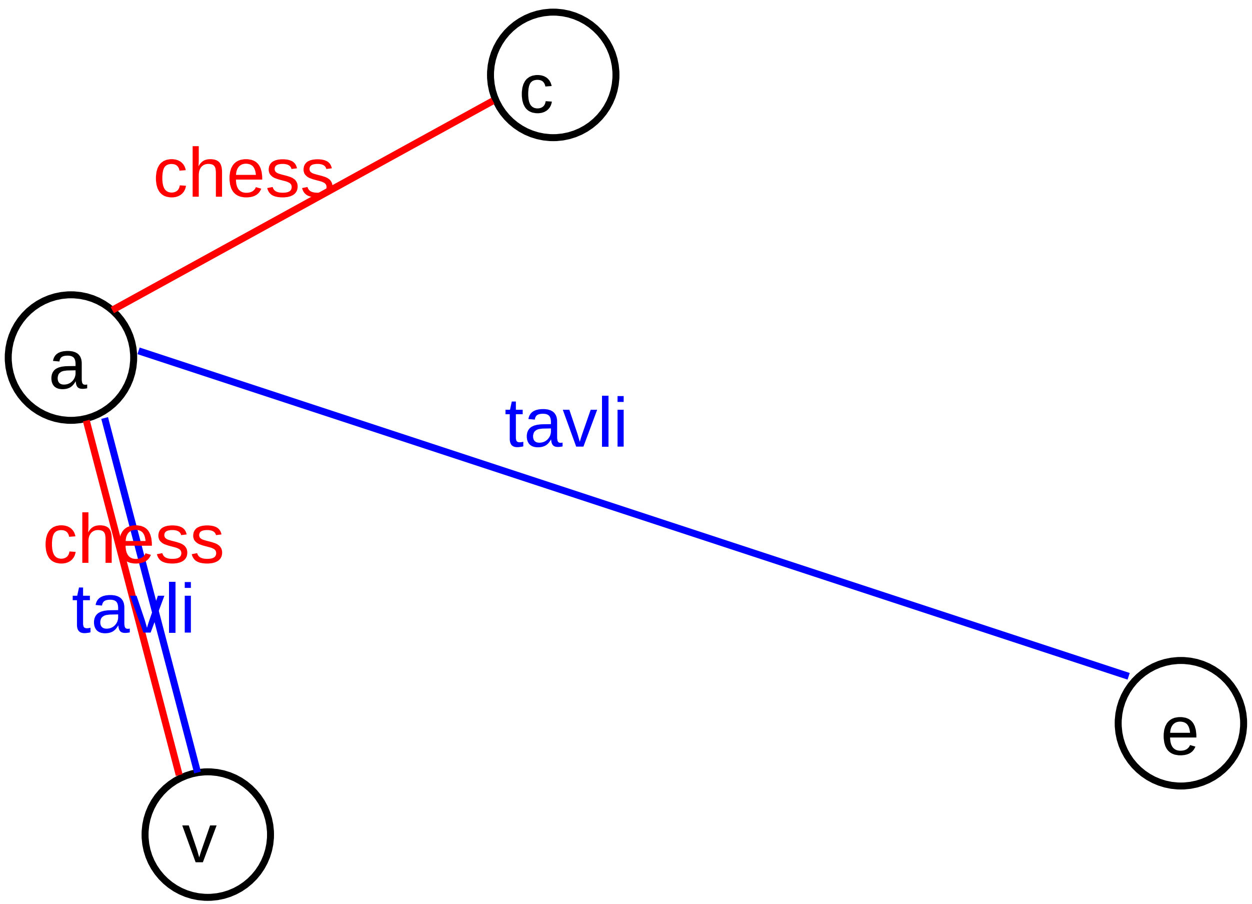
joint work with

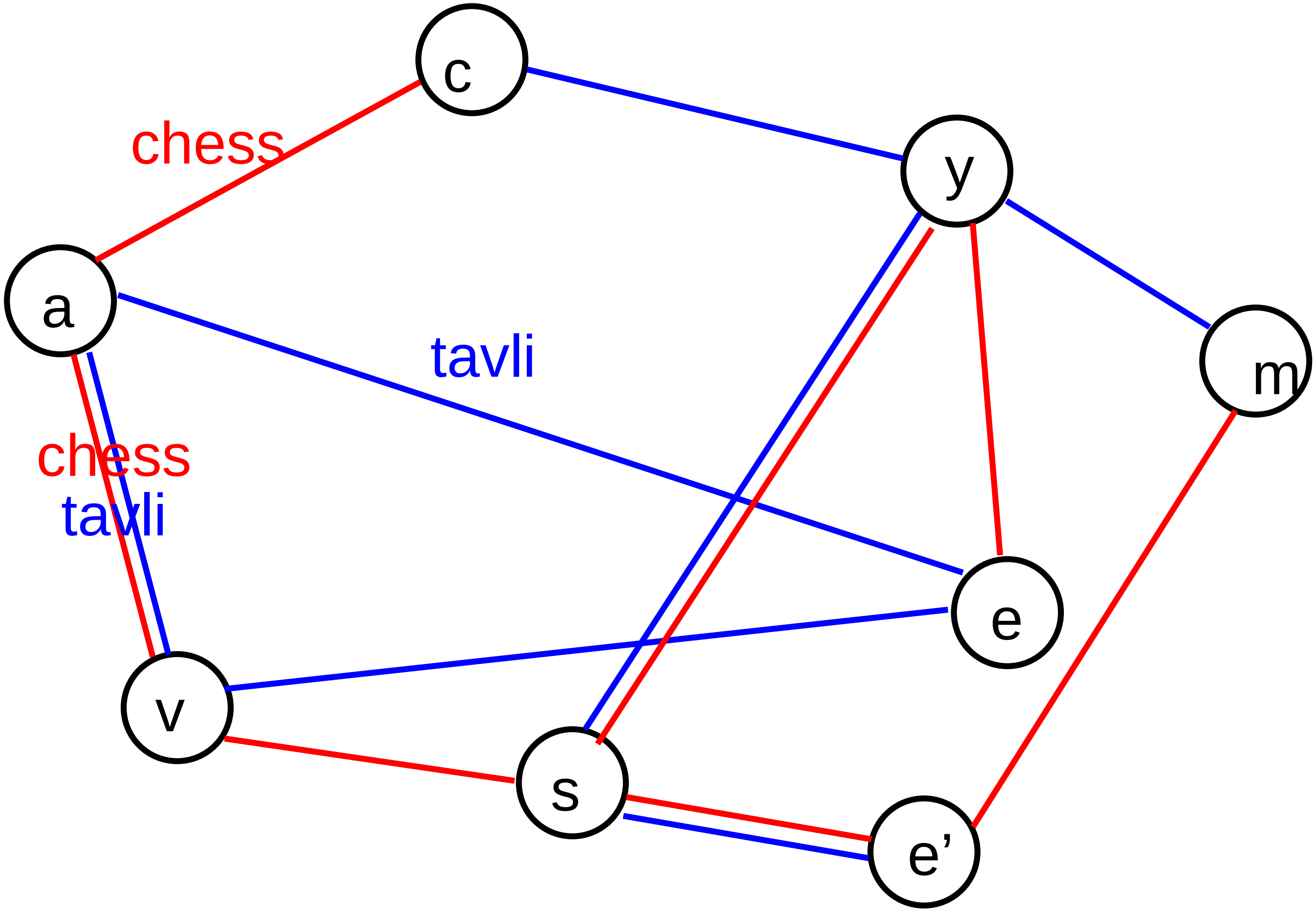
christos nomikos (u ioannina)
stathis zachos (ntu athens)

theory_tea@aueb

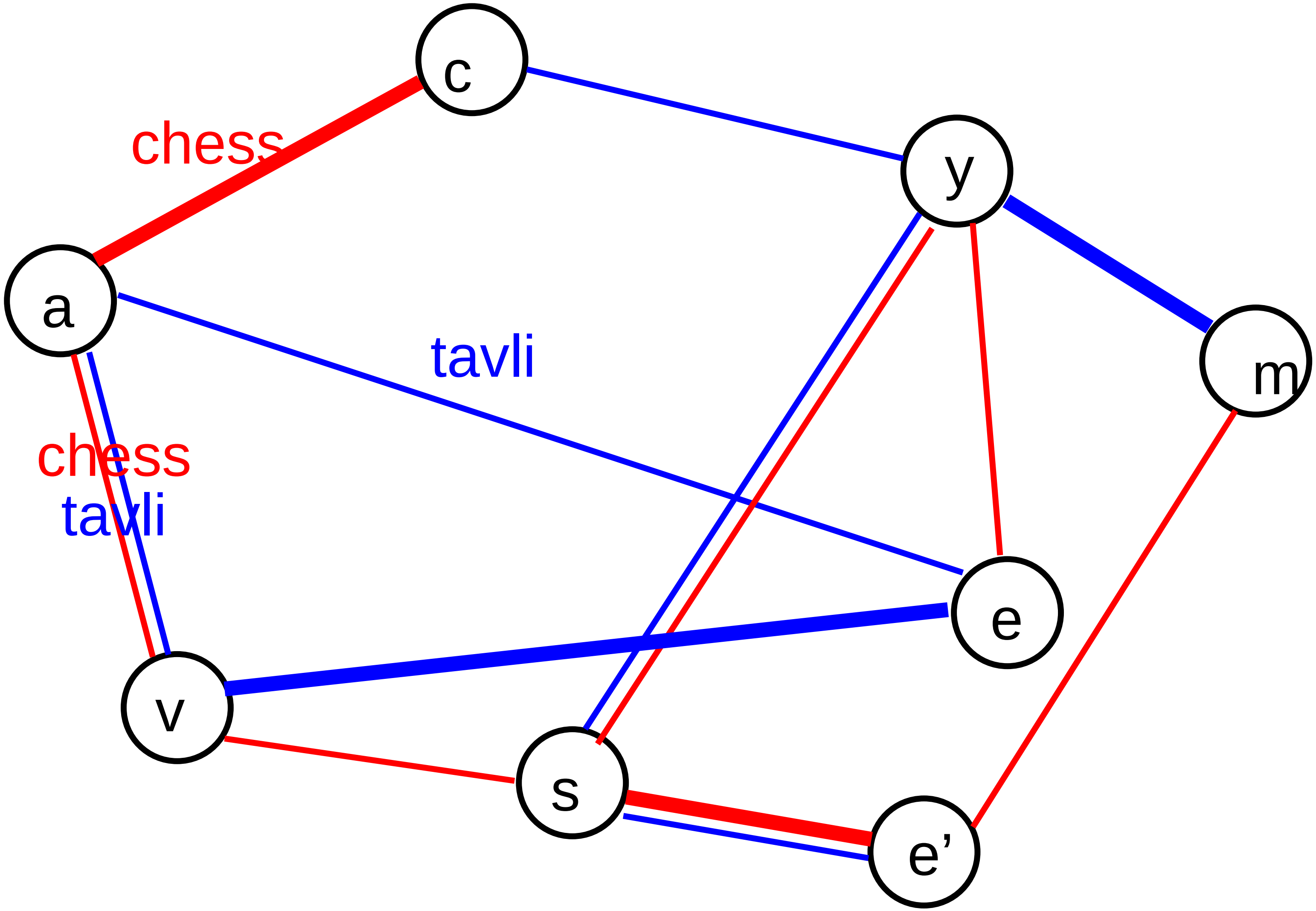
18 dec 2012

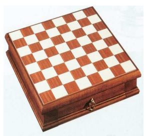
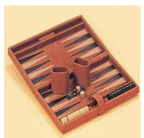
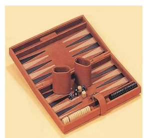
chess or tavli ?

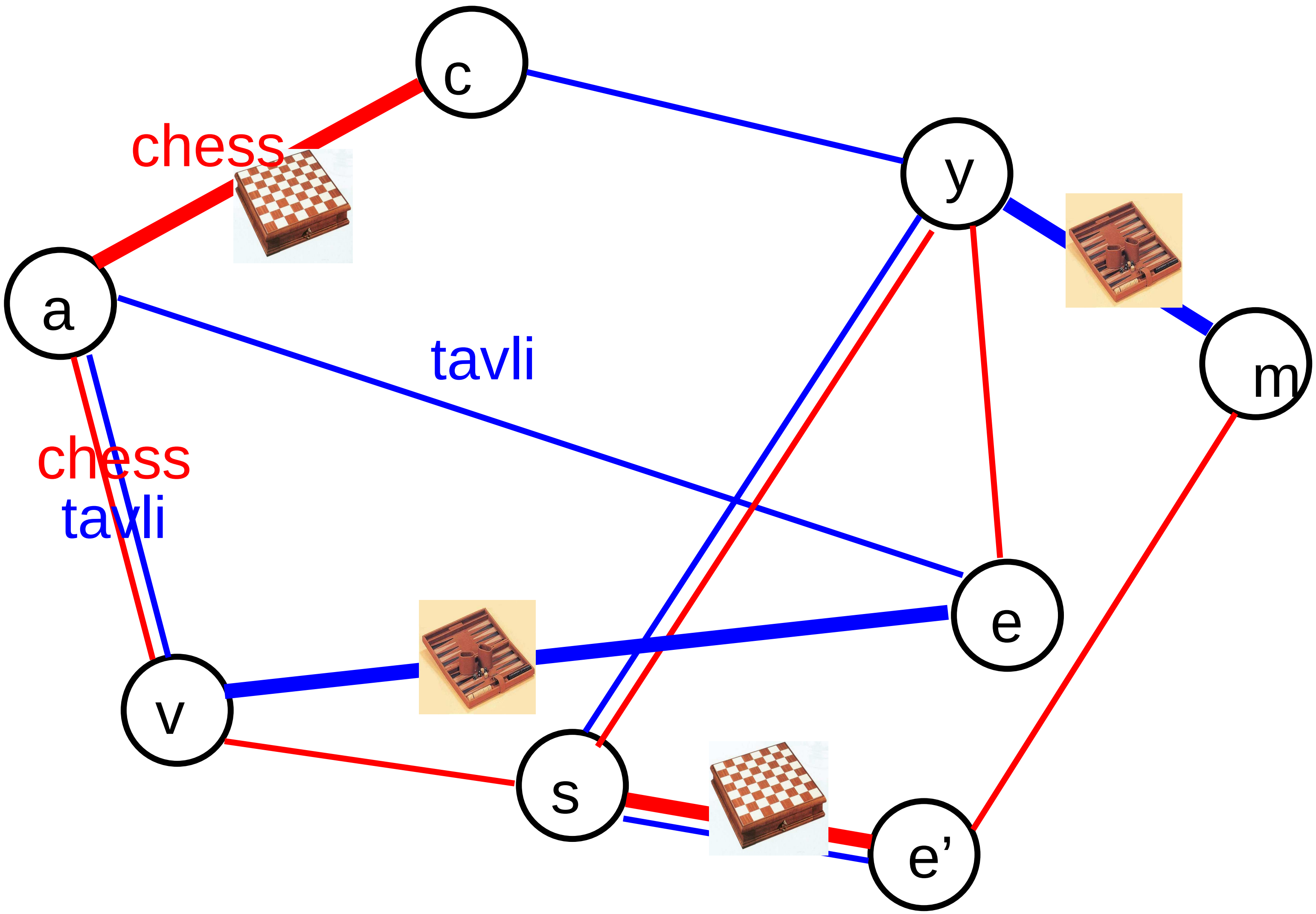



c
y
chess
a
tavli
m
chess
tavli
e
v
s
e'





c
chess
y
a
tavli
m
chess
tavli
e
v
s
e'

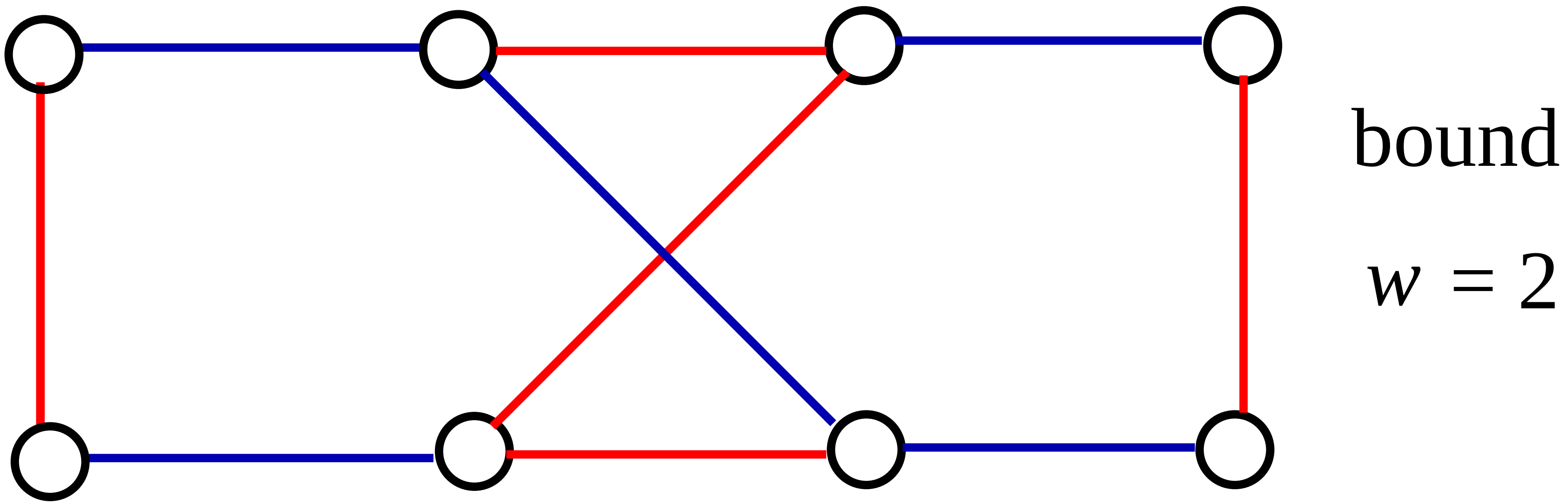

c
chess
y
a
tavli
m
chess
tavli
e
v
s
e'

Outline

- Blue-Red Matching: definition and hardness
- Approximation algorithms.
- Randomized algorithm.
- Application to optical networking.
- Open questions.

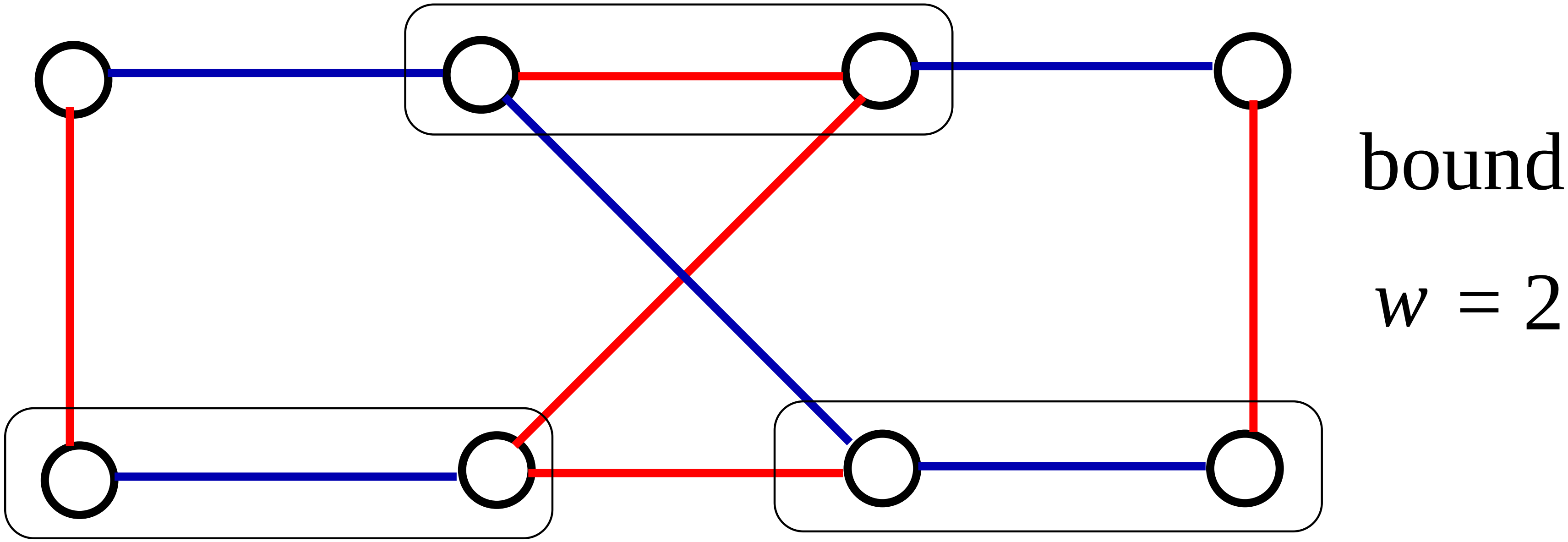
The Blue-Red Matching problem

Definition. *[NPZ, MFCS'07] Given a (multi)graph with red and blue edges, and an integer w, find a maximum matching consisting of at most w edges of each color.*



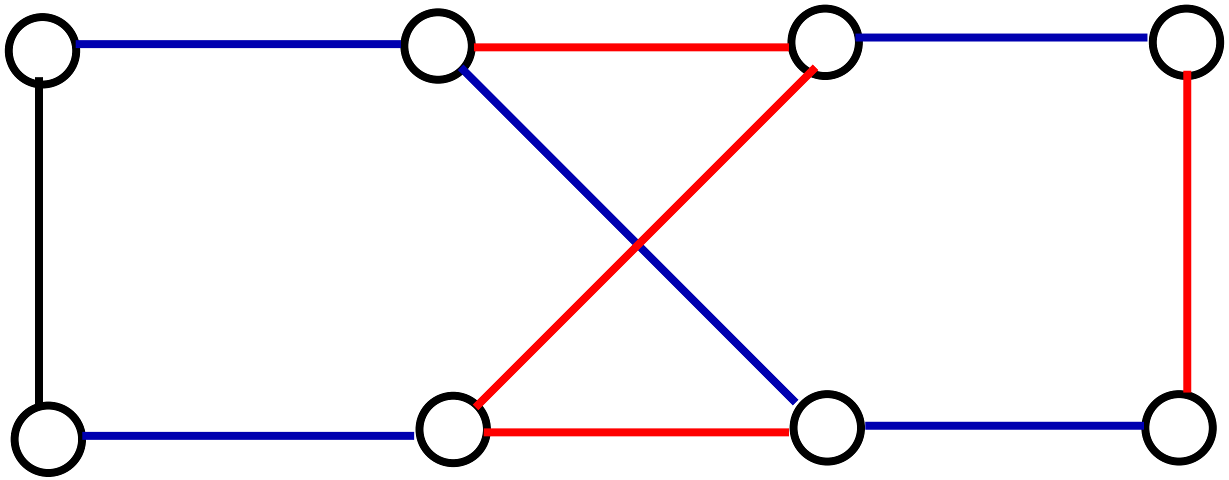
The Blue-Red Matching problem

Definition. *[NPZ, MFCS'07] Given a (multi)graph with red and blue edges, and an integer w, find a maximum matching consisting of at most w edges of each color.*



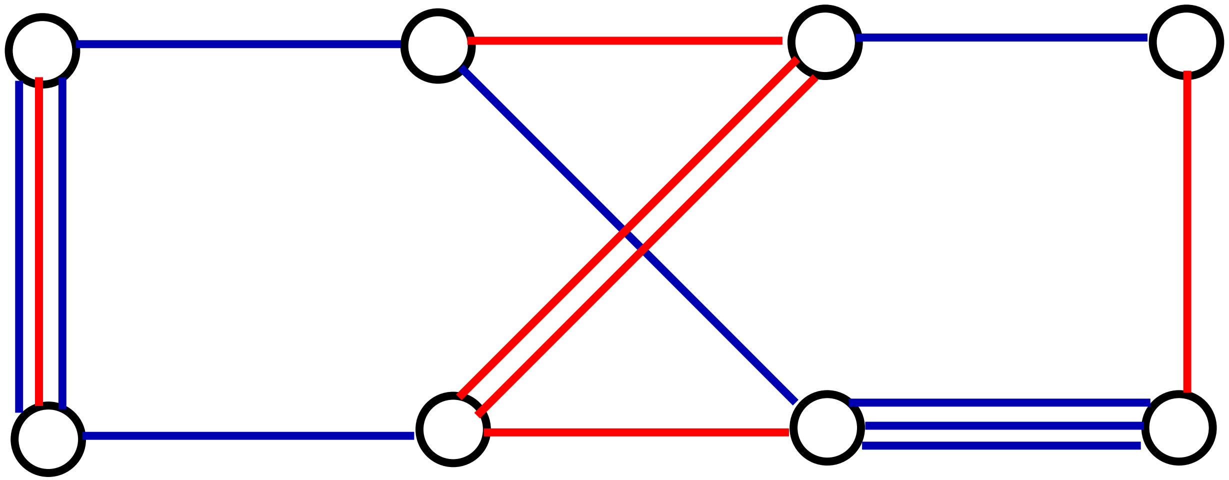
A useful generalization

It makes sense to consider a third type of uncolored edges that can be colored either blue or red.



A useful generalization

It makes sense to consider a third type of uncolored edges that can be colored either blue or red.



Remark. BRM in multigraphs can be reduced to BRM in simple graphs with red, blue, and uncolored edges.

Hardness of Blue-Red Matching

Blue-Red Matching (BRM) is at least as hard as Exact Matching, [Papadimitriou and Yannakakis 1982]:

Given a graph with some red edges, and a positive integer k, is there a perfect matching with exactly k red edges?

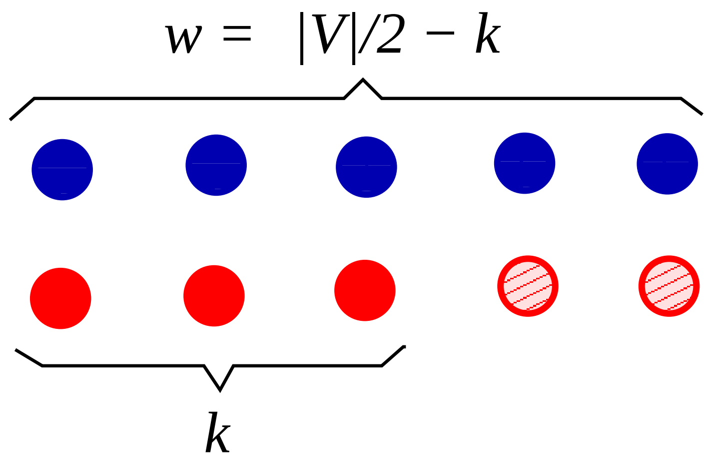
Remark. Exact Matching admits an RNC algorithm [Mulmuley, U.Vazirani, V.Vazirani, 1987], however it is still an open question whether it can be solved in polynomial time.

Reduction of EXACT MATCHING to BRM(D)

- Paint uncolored edges blue.

- Set $w = |V|/2 - k$ and add $r = w - k$ new red edges (assuming $k < |V|/4$, the othe case is similar).

- Ask for a w-blue-red matching of cardinality $2w$.



Corollary. *A poly-time algorithm for* BRM *would answer a long-standing open question in the affirmative.*

A simple approximation algorithm for BRM

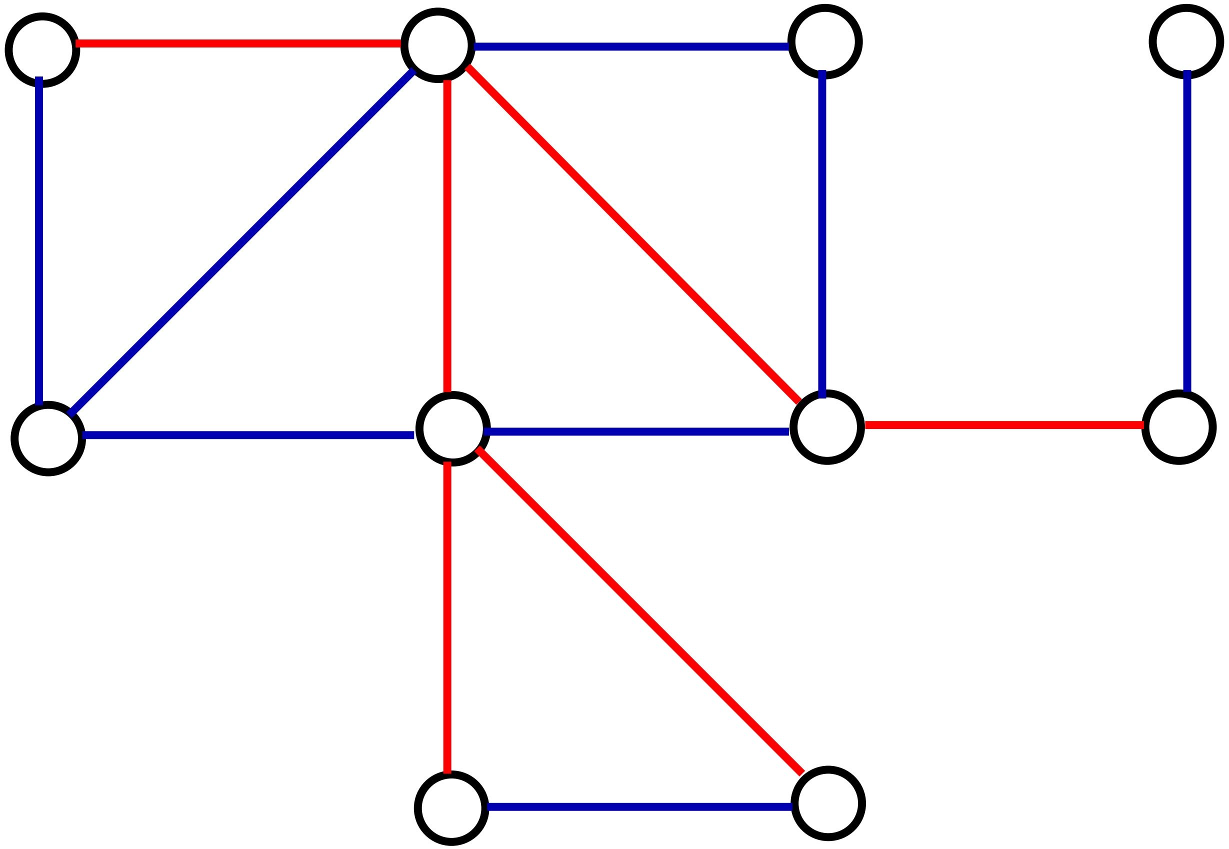
Proposition. *The greedy heuristic achieves a* $\frac{1}{2}$*-approximation ratio.*

Reasoning: each greedily chosen edge may block at most two edges that are present in an optimal solution.

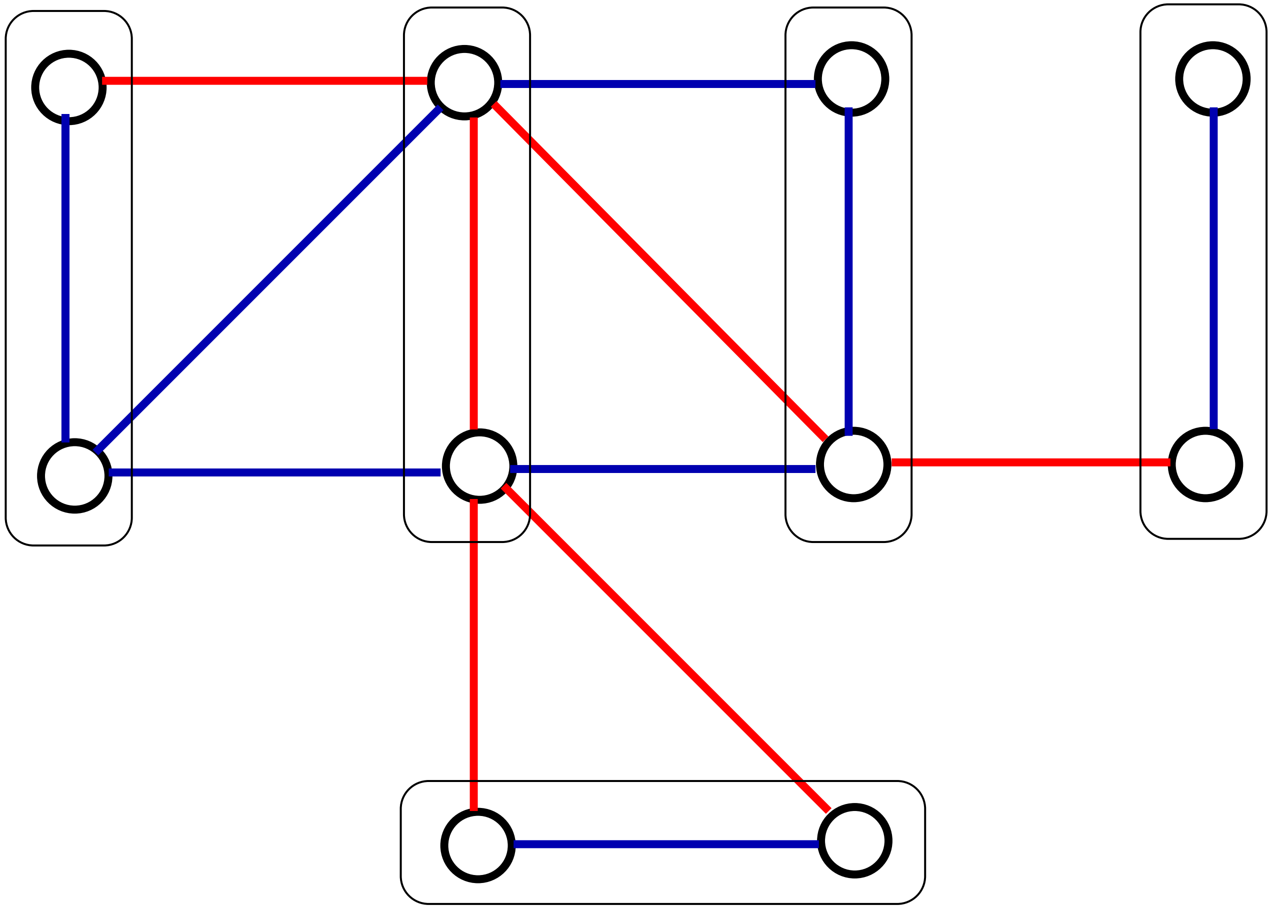
An asymp. $\frac{3}{4}$-approximation for BRM

- Compute a maximum matching M.
- If both the number of blue and the number of red edges in M are $\leq w$ or $\geq w$, then **stop**: M is, or can be immediately converted to, a maximum w-blue-red matching.
- Otherwise: (w.l.o.g. assume that $\#$ blue edges $> w$, $\#$ red edges $< w$)
  - Compute a maximum matching M_r on the red subgraph.
  - Superimpose M_r over M, thus obtaining a graph of chains and cycles that alternate between M and M_r.
  - *Balancing*: Use components of the above graph in order to replace unnecessary blue edges in M by red edges in M_r.

An asymp. $\frac{3}{4}$-approximation for BRM (ctd.)

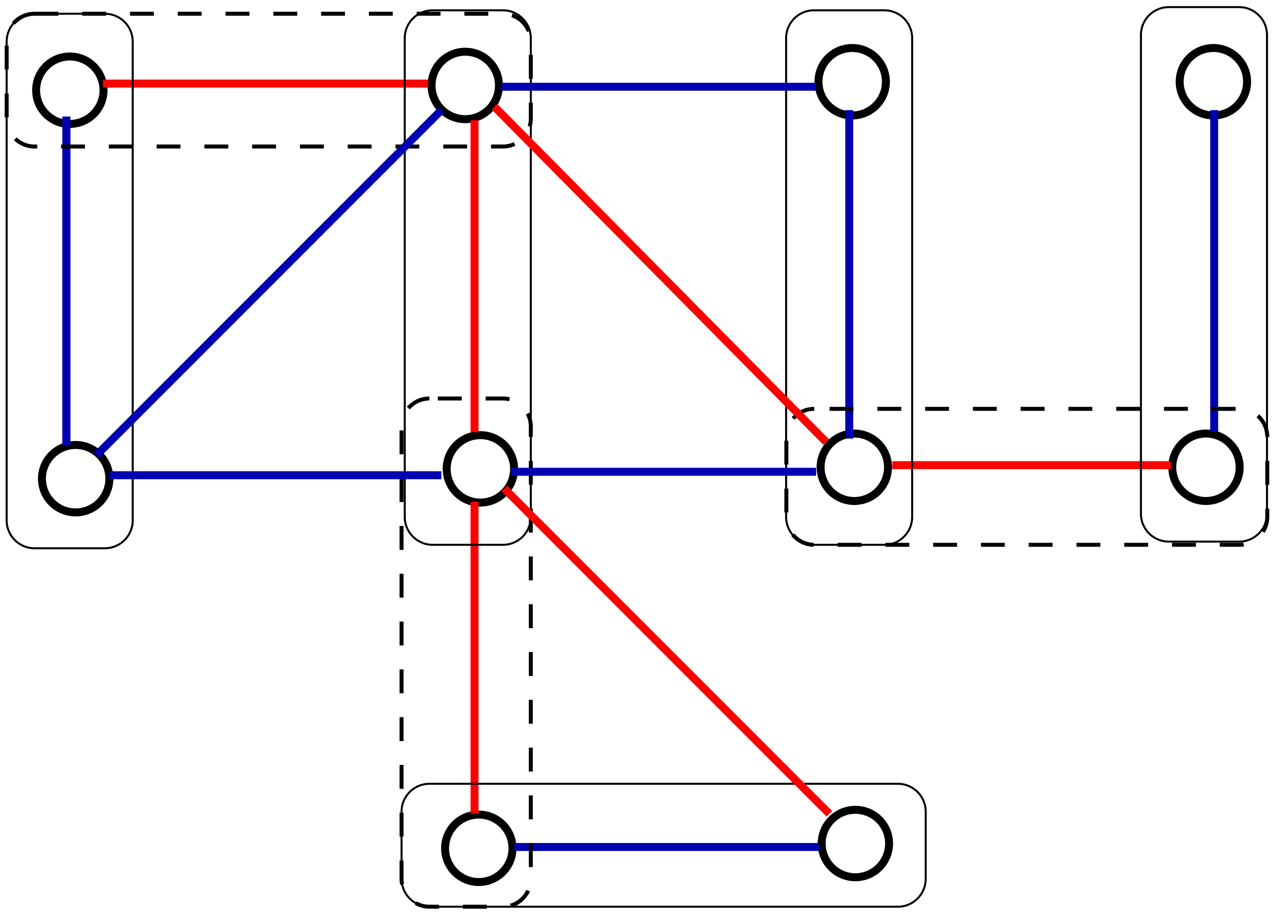


An asymp. $\frac{3}{4}$-approximation for BRM (ctd.)



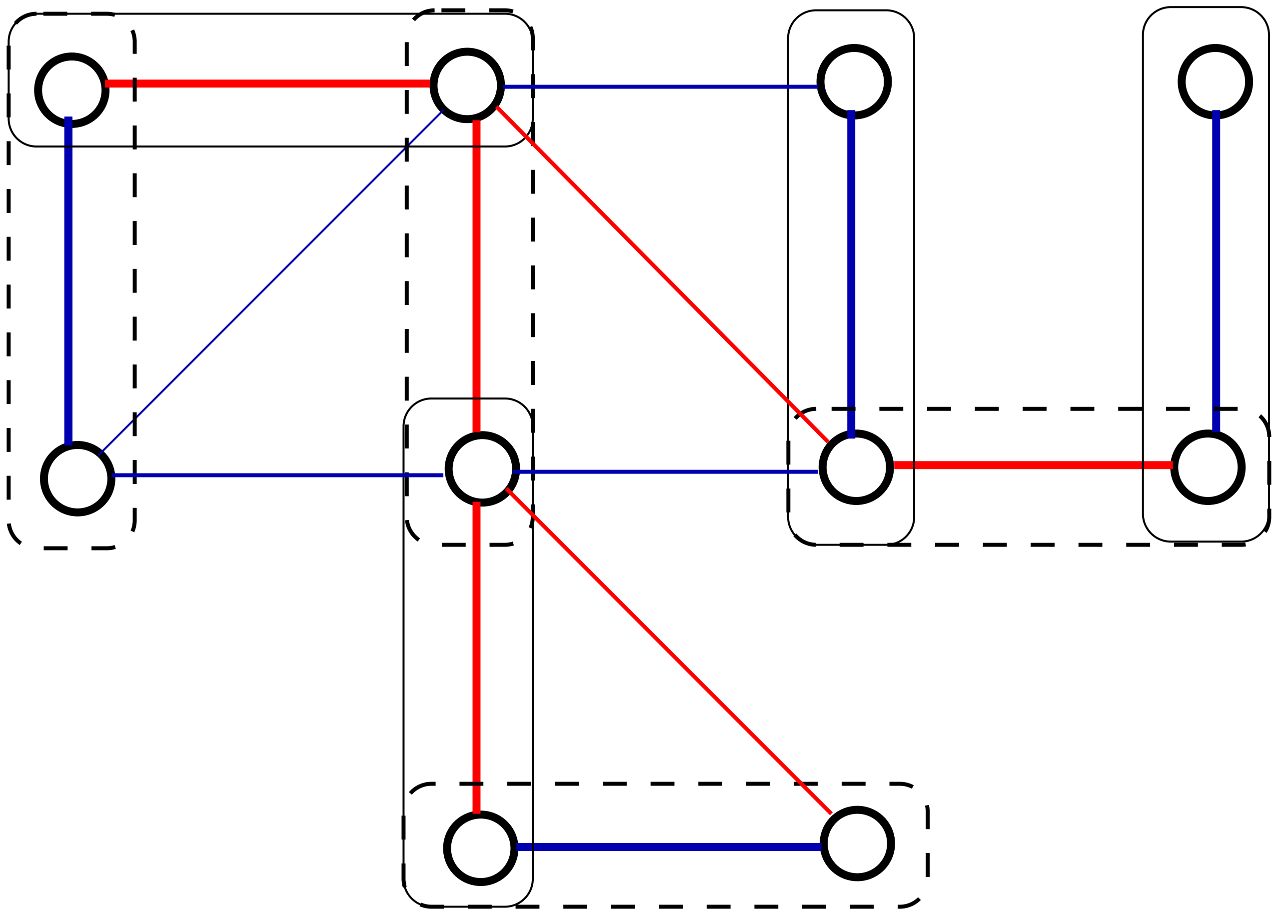
Compute a maximum matching M.

An asymp. $\frac{3}{4}$-approximation for BRM (ctd.)



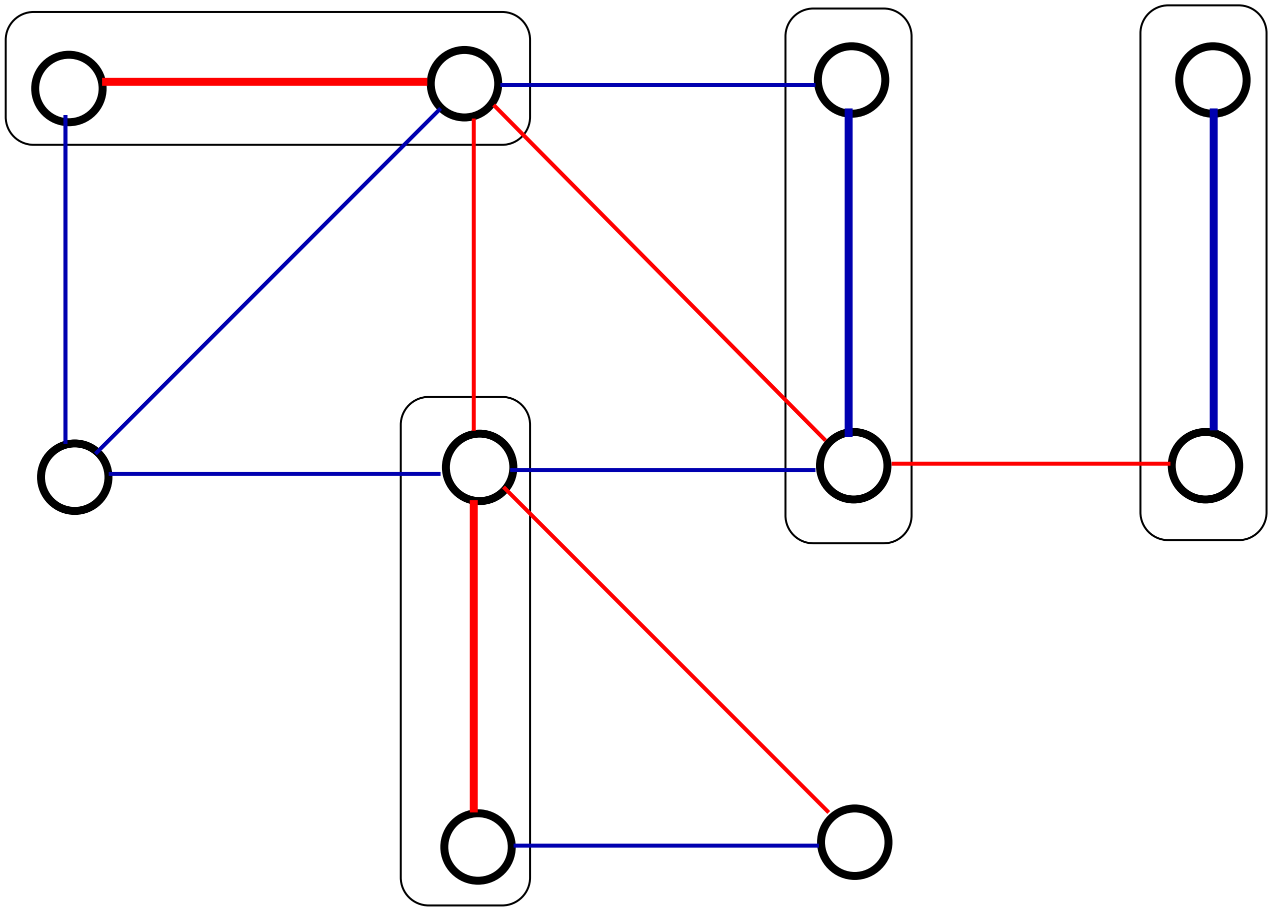
Compute a maximum red matching M_r; superimpose M_r over M.

An asymp. $\frac{3}{4}$-approximation for BRM (ctd.)



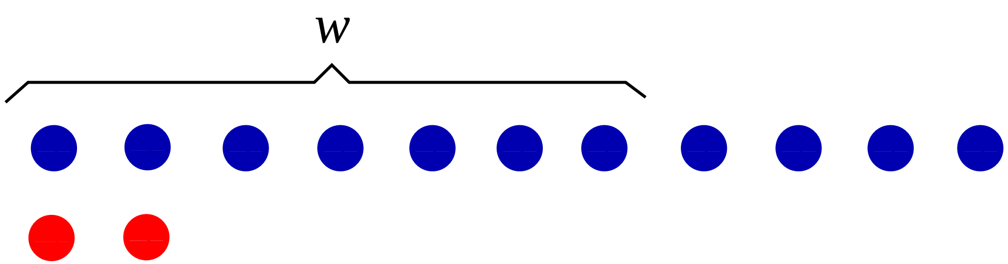
In each red-increasing component 'swap' edges.

An asymp. $\frac{3}{4}$-approximation for BRM (ctd.)



An asymp. $\frac{3}{4}$-approximation for BRM (ctd.)

Theorem. *The cardinality of the resulting matching is at least* $\frac{3}{4}\mu_{opt} - \frac{1}{2}$, *where* μ_{opt} *is the cardinality of a maximum* w*-blue-red matching.*

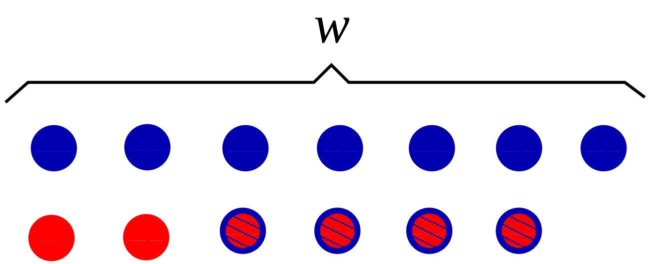


Proof sketch:

- If the algorithm enters *Balancing*, the solution must contain at least w blue edges, that is, at least half the optimum is guaranteed.

An asymp. $\frac{3}{4}$-approximation for BRM (ctd.)

Theorem. *The cardinality of the resulting matching is at least* $\frac{3}{4}\mu_{opt} - \frac{1}{2}$*, where* μ_{opt} *is the cardinality of a maximum* w*-blue-red matching.*

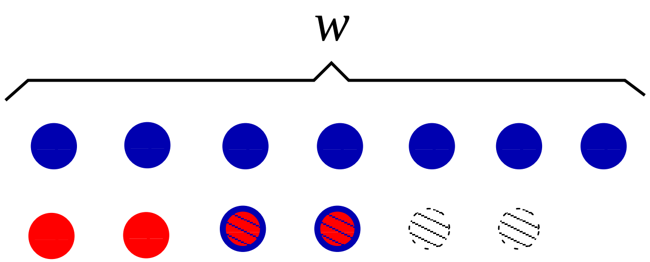


Proof sketch:

- If the algorithm enters *Balancing*, the solution must contain at least w blue edges, that is, at least half the optimum is guaranteed.
- If we could replace surplus blue edges by red ones we would obtain an optimal solution.

An asymp. $\frac{3}{4}$-approximation for BRM (ctd.)

Theorem. *The cardinality of the resulting matching is at least* $\frac{3}{4}\mu_{opt} - \frac{1}{2}$*, where* μ_{opt} *is the cardinality of a maximum* w*-blue-red matching.*



Proof sketch:

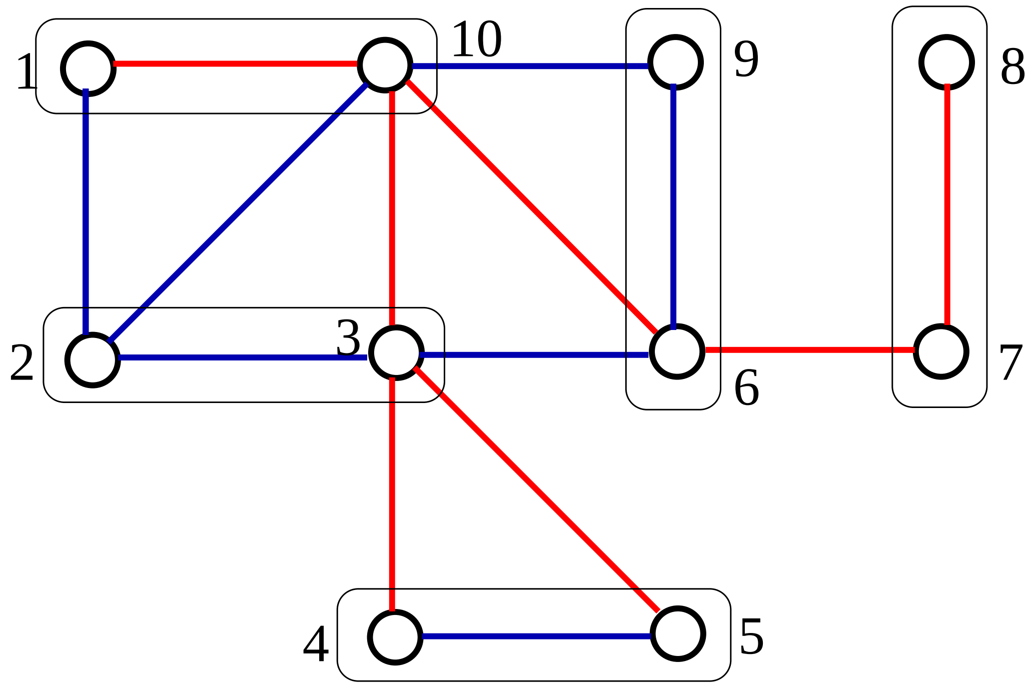
- If the algorithm enters *Balancing*, the solution must contain at least w blue edges, that is, at least half the optimum is guaranteed.
- If we could replace surplus blue edges by an equal number of red ones we would obtain an optimal solution.
- We achieve at least half of that. □

The randomized algorithm

- Add black edges to obtain a complete graph.
- For all $1 \leq p, q \leq |V|/2$ check whether there exists a (p, q)-perfect matching in the graph.

  This can be done by adapting techniques from [MVV 87] based on Pfaffian computations, used to show that $\textsc{Matching}$ *and* $\textsc{Exact Matching}$ *are in* RNC*: more details soon.*
- Among those p, q that pass the test, select a pair that maximizes $\min(w, p) + \min(w, q)$
- Compute and output the corresponding blue-red matching by checking Pfaffians of A_{ij} submatrices

The Pfaffian of A



$\pi_M = \{1, 10, 2, 3, 4, 5, 6, 9, 7, 8\}$ is the *canonical permutation* corresponding to the above matching M.

$value(\pi_M)$ is the product of entries of A corresponding to the edges of M, therefore: $value(\pi_M) = \Pi_{(i,j)\in M} a_{ij} = 2^{W(M)} x^3 y^2$

The Pfaffian of A (ctd.)

- In general: $value(\pi_M) = 2^{W(M)} x^p y^q$

  where $W(M)$ is the sum of weights of edges in M and M is a (p,q)-perfect matching.

- Since A is skew-symmetric, the Pfaffian of A is equal to:

$$\mathcal{PF}(A) = \sum_{M \in \mathcal{M}} sign(\pi_M) \cdot value(\pi_M)$$

- Therefore, $\mathcal{PF}(A)$ is a polynomial in two variables:

  $$\mathcal{PF}(A) = \sum_{p=0}^{|V|/2} \sum_{q=0}^{|V|/2} c_{pq} x^p y^q$$

  and c_{pq} is a sum of terms of the form $\pm 2^{W(M_{pq})}$ where M_{pq} ranges over all (p,q)-perfect matchings.

Checking (p, q)-perfect matching existence

- assign to each edge (i, j) a weight w_{ij} randomly selected from $\{1, \ldots, n^4\}$
- construct Tutte matrix A:

$$a_{ij} = \begin{cases} 0 & \text{if } i = j \\ 2^{w_{ij}} & \text{if } i < j \text{ and } e_{ij} \in E_{black} \\ {\color{blue}x}2^{w_{ij}} & \text{if } i < j \text{ and } e_{ij} \in E_{blue} \\ {\color{red}y}2^{w_{ij}} & \text{if } i < j \text{ and } e_{ij} \in E_{red} \\ -a_{ji} & \text{if } i > j \end{cases}$$

Checking (p, q)-perfect matching existence

$$a_{ij} = \begin{cases} 0 & \text{if } i = j \\ 2^{w_{ij}} & \text{if } i < j \text{ and } e_{ij} \in E_{black} \\ x2^{w_{ij}} & \text{if } i < j \text{ and } e_{ij} \in E_{blue} \\ y2^{w_{ij}} & \text{if } i < j \text{ and } e_{ij} \in E_{red} \\ -a_{ji} & \text{if } i > j \end{cases}$$

- compute the Pfaffian of A, $\mathcal{PF}(A)$; if the coefficient c_{pq} of $x^p y^q$ is nonzero, then at least one (p, q)-perfect matching exists: c_{pq} is a sum of terms of the form $\pm 2^{W(M_{pq})}$ where M_{pq} ranges over all (p, q)-perfect matchings.
- but what if the coefficient is zero? possible term cancellation?

Avoiding term cancellation

Lemma. *Let p, q be integers, with $0 \leq p, q \leq \frac{|V|}{2}$ and suppose that there exists a unique minimum weight (p, q)-perfect matching M^*_{pq}. Then the coefficient c_{pq} of $\mathcal{PF}(A)$ is nonzero. Furthermore, $W(M^*_{pq})$ is the maximum power of 2 that divides c_{pq}.*

Proof. The term corresponding to M^*_{pq} cannot be cancelled since all other terms are even multiples of it. □

Why it works: the Isolating Lemma

- Uniqueness of minimum weight perfect matchings is proven by using

  The Isolating Lemma [MVV 87]

  Let $B = \{b_1, b_2, \ldots, b_k\}$ *be a set of elements, let* $\mathcal{S} = \{S_1, S_2, \ldots, S_\ell\}$ *be a collection of subsets of* B. *If we choose integer weights* $w_1, w_2, \ldots w_k$ *for the elements of* B *at random from the set* $\{1, 2, \ldots, m\}$, *and define the weight of set* S_j *to be* $\sum_{b_i \in S_j} w_i$ *then the probability that the minimum weight subset in* $\mathcal{S}$ *is unique is at least* $1 - \frac{k}{m}$.

- Modification of weight assignment and of the Isolating Lemma is needed if there are uncolored edges in the graph.

Computing a min. weight (p, q)-perfect matching

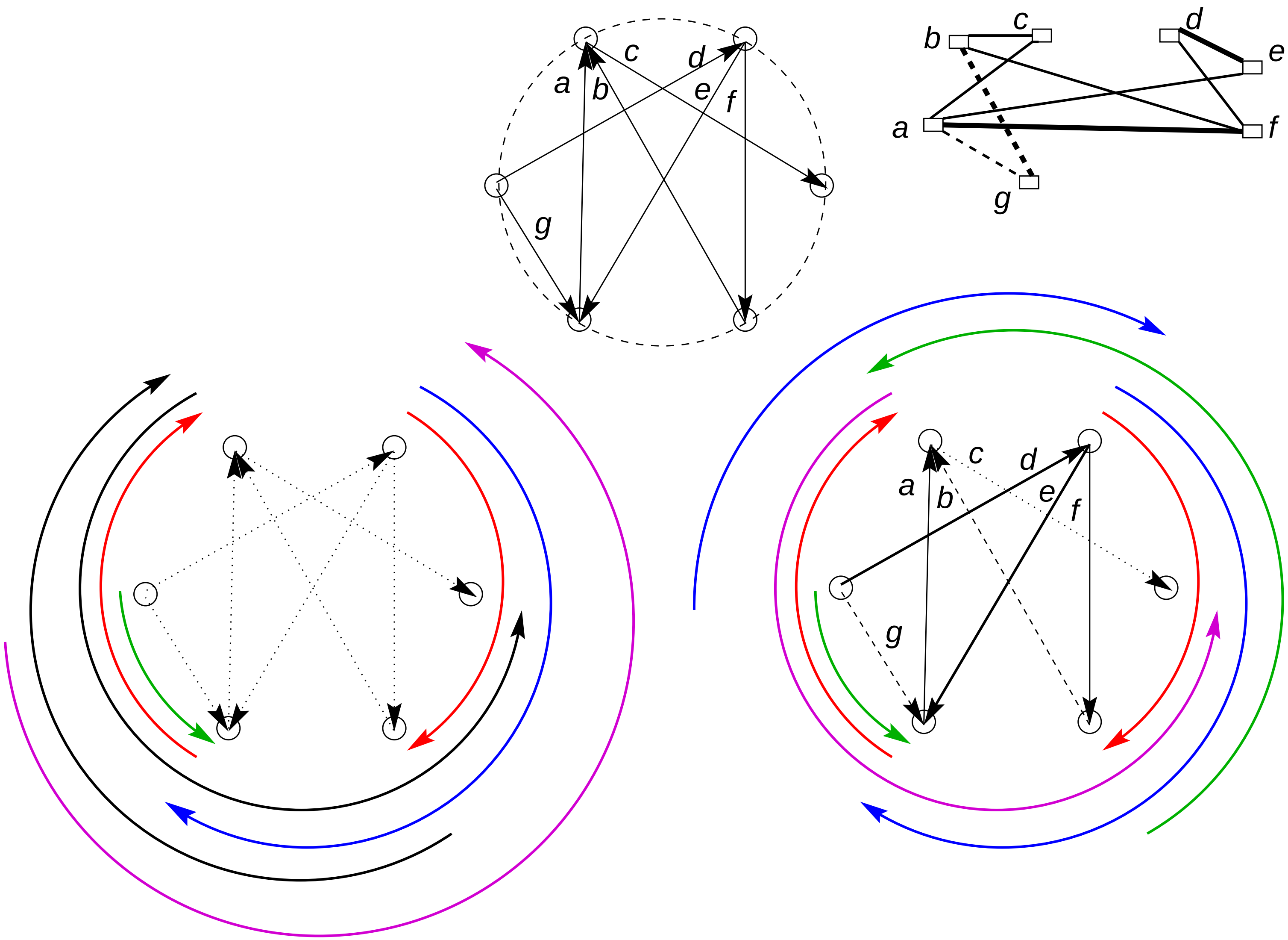
- $\mathcal{PF}(A)$ can be computed by interpolation [Horowitz, Sahni, 1975], using an algorithm for computing arithmetic Pfaffians [Galbiati, Maffioli, 1994 & Mahajan, Subramanya, Vinay, 2004].
- Once we know $W(M^*_{pq})$ we can construct M^*_{pq} by computing appropriate Pfaffians of submatrices A_{ij}, for all i, j.

Application to optical networking

- DirMaxRWA [NPZ, INFOCOM'03]:
  Given are a directed symmetric graph G, a set of requests (pairs of nodes) R on G, and an integer w (bound on the number of available wavelengths).

- The goal is to find a routing and wavelength assignment to an as large as possible set of requests $R' \subseteq R$ such that any two requests routed via edge-intersecting paths receive different wavelengths and only wavelengths from $\{1, \ldots, w\}$ are used.

Application to optical networking (ctd.)



Application to optical networking (ctd.)

- 'Avoid-an-edge' technique leads to solution that misses at most w requests per direction compared to an optimal solution. Their number is bounded by the cardinality of a maximum w-blue-red matching.
- We give a direct relation beteween the approximation ratios of $\textsc{DirMaxRWA}$ and BRM.
- Our results for BRM imply that $\textsc{DirMaxRWA}$ in rings admits a randomized approximation algorithm with ratio $\frac{2}{3}$ and a deterministic approximation algorithm with (asympt.) ratio $\frac{7}{11}$.
- The $\frac{2}{3}$ ratio for $\textsc{DirMaxRWA}$ is best possible via edge avoidance technique. Was shortly after beaten by an 0.708-approximation algorithm [Caragiannis, 2007].

Generalizations – recent results

- Different bounds ($w_1 \neq w_2$): randomized algorithm works fine.

  Deterministically: apx. ratio of our algorithm falls to $\frac{1}{2}$. Improvement to $\frac{2}{3}$ [Stamoulis, 2009].

  Even better: 2-Budgeted Matching: PTAS (Blue-Red Matching is a special case) [Grandoni, Zenklusen, 2010].

- More color classes: randomized algorithm is polynomial-time for fixed number of colors (but exponent increases).

  k-Budgeted Matching: PTAS for $k = 1$ [Berger, Bonifaci, Grandoni, Schaefer, 2008], no PTAS known for $k > 2$.

  Various related results (next slide).

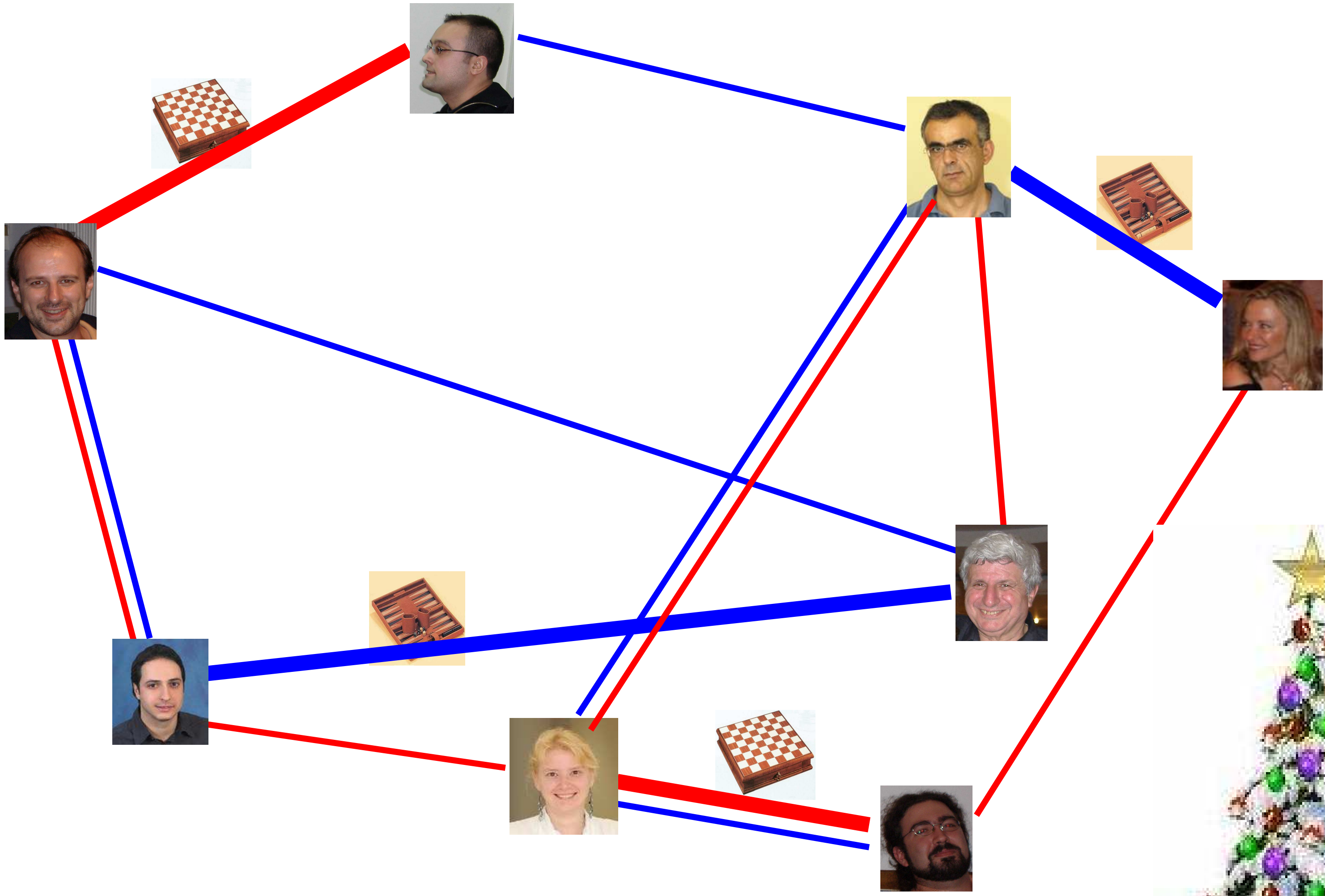
Bounded/Budgeted/Labeled (Weighted) Matching

- LABELED MATCHING [Monnot, 2005]: edges have labels, maximum matching is sought with min/max number of different labels; APX-hardness results, constant approximation algorithms.
- Multi-budgeted optimization [Grandoni, Zenklusen, 2010]: independence systems, length bounds.

  k-BUDGETED MATCHING: PTAS for $k = 1$ [BBGS'08] and $k = 2$ [GZ'10]. No PTAS known for $k > 2$.
- BOUNDED-COLOR MATCHING Bipartite graphs [Mastrolilli, Stamoulis, ISCO'12]: constant bi-criteria approximations (allowing bound violation), based on iterative rounding of LPs.

Open questions

- Deterministic algorithm?
- Different/faster randomized algorithm?
- Deterministic/randomized approximation schemes for k-BUDGETED MATCHING for $k > 2$ colors? Restricting to cardinality constraints?
- Game-theoretic considerations.




Καλές γιορτές!