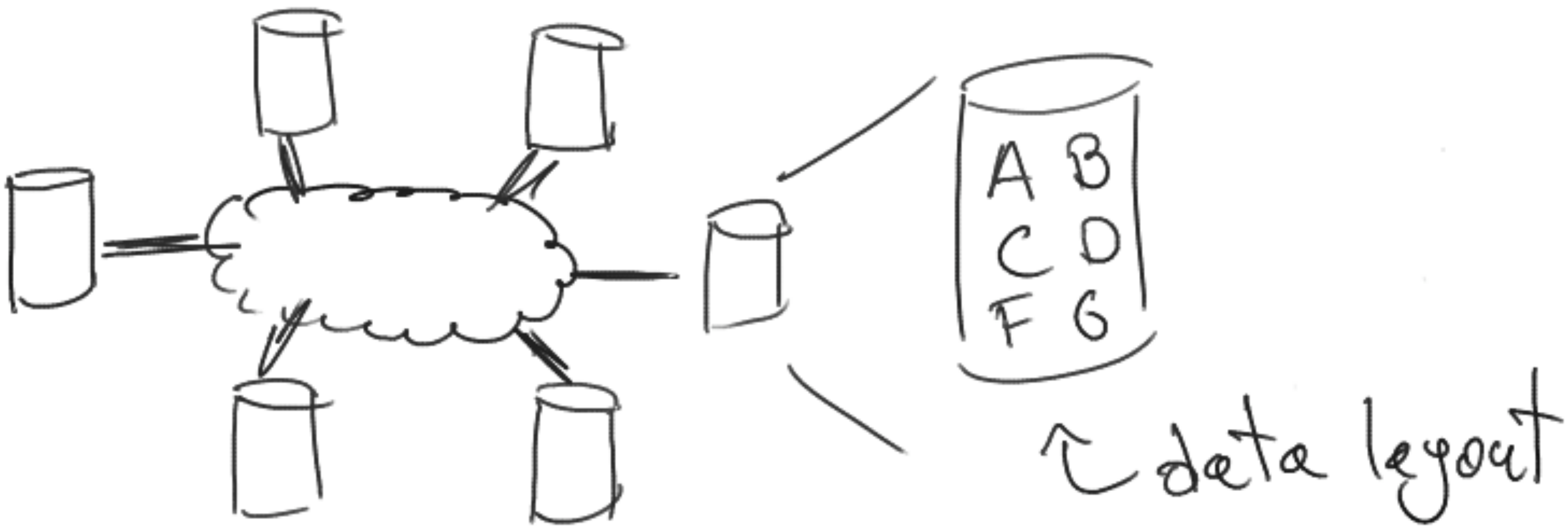
Combinatorial Algorithms for Data Migration

Rajiv Gandhi & Julian Mestre

The Data Migration Problem



- # of requests per item
- disk load

may change over time

More formally...

Given $G = (V, E)$ $\nearrow$ V : disks
$\searrow$ E : transfers

We want to schedule E

$\equiv$ break into matchings $M_1, \ldots, M_k$

Minimize

$\rightarrow \max_e \phi(e)$ [EDGE COLORING]

$\rightarrow \sum_e \phi(e)$

$\rightarrow \sum_v \max_{e \in \delta(v)} \phi(e)$

[SUM EDGE/VERTEX COMP. TIME]

where $\phi : E \rightarrow \mathbb{N}$ • $e \in M_{\phi(e)}$

Previous Results

Edge coloring: $\min \max_e \phi(e)$

- NP-hard
- can get $\Delta+1$ (gen. graphs) [V 64]
- can get Δ (for bipartite graphs)

Previous Results

$$\min \sum_e \phi(e)$$

- APX-hard in bipartite graphs [M]
- 2-approx [BHKSS 00]
- 1.79-approx (for bipartite) [HKS 03]

↳ Our result: $\sqrt{2}$-approx

Previous Results

$$\min \sum_{v} \max_{e \in \delta(v)} \phi(e)$$

- NP-hard (for general graphs)
- 3-approx (LP rounding) [K03]

↳ Our result: Primal-Dual 3-approx

Algorithm for $\min \sum \phi(e)$

Find matchings $M_1, \ldots, M_\Delta$ s.t.

$\bigcup_{i \leq b} M_i$ is a maximal b-matching

$$\boxed{\text{Thm: this is a } \sqrt{2}\text{-approx}}$$

How do we find $M_1, \dots M_\Delta$?

1) Find matching M_Δ hitting all degre Δ vertices

2) Remove M_Δ, repeat for $M_{\Delta-1} \dots M_1$

we can always do ①
for bipartite graphs, but...

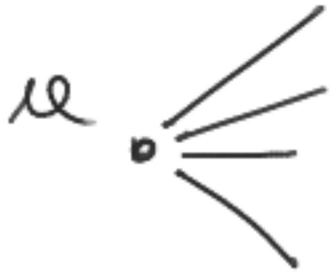
Algorithm for $\min \sum \phi(e)$

Find matchings $M_1, \ldots, M_\Delta$ s.t.

$\bigcup_{i \leq b} M_i$ is a maximal b-matching

$\boxed{\text{Thm: this is a } \sqrt{2}\text{-approx}}$

Lowerbound



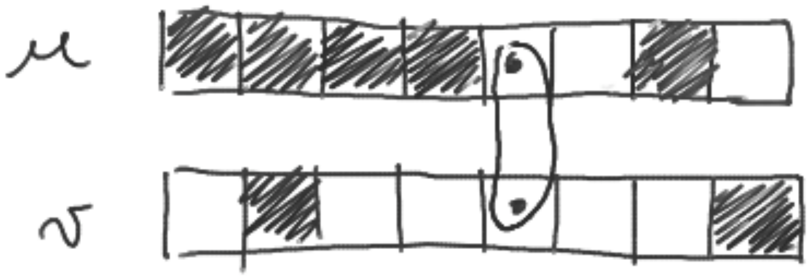
$S \subseteq \delta(u)$

Any solution must spend on S

$$\geq 1 + 2 + \cdots + |S| = \frac{|S|(|S|+1)}{2}$$

Analysis

$\forall (u,v)$ • either u or v is full

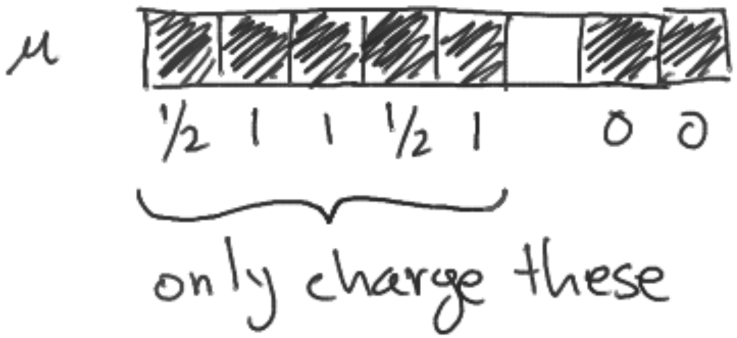


blame the full endpoint

If both endpoints are full

$\Rightarrow$ they share the cost

Analysis



Let μ be s-full

- αs : pay 1
- $(1-\alpha)s$: pay ½

$$\text{we need} \leq \frac{1}{2}\sum_{i=1}^{s} i + \frac{1}{2}\sum_{i=(1-\alpha)s+1}^{s} i$$

$$\text{BUDGET} = \rho\sum_{i=1}^{s} i + \rho\sum_{i=1}^{\alpha s} i \Rightarrow \sqrt{2}\text{-approx}$$

A few comments

- Almost tight: 1.375 example
- Downside: only unweighted
- Integrality gap of $10/9$

Integrality Gap?

$$\max \sum_{S} \frac{|S|(|S|+1)}{2} \, y_S$$

$$\sum_{e \in S} y_S \leq 1 \quad \forall e$$

$$y_S \geqslant 0 \quad \forall S \subseteq \delta(u)$$

?

Algorithm for $\min \sum_u \max_{e \in \delta(u)} \phi(e)$

I) Every vertex is unlabeled

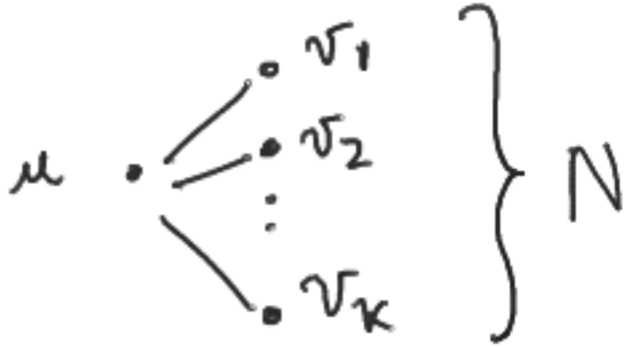
Find u with most unlabeled neighbors N

for all $v \in N$ set $\ell(v) = |N|$

II) Sort edges: $\min(\ell(u), \ell(v))$, $\max(\ell(u), \ell(v))$

III) Greedy Schedule

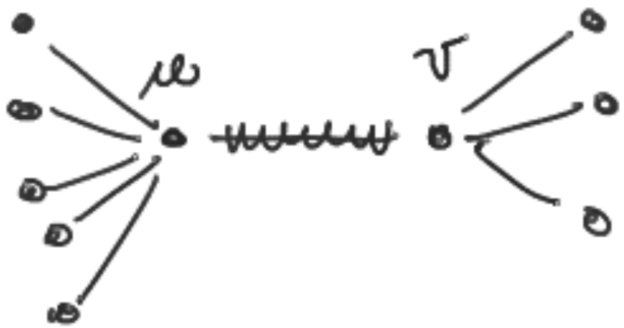
Lowerbound



$$\sum_i \text{COMP TIME } v_i \geqslant \frac{|N|(|N|+1)}{2} \geqslant \frac{1}{2}\sum \ell(v_i)$$

$$\boxed{\text{Lemma: } \sum_v \ell(v) \leqslant 2\ \text{OPT}}$$

Analysis



How many edges were scheduled before (u,v)?

$\boxed{\text{Lemma: } \forall u \text{ finishes by } \ell(u) + \deg(u)}$

Analysis

$\boxed{\text{Lemma: } \sum_v \ell(v) \leq 2\,\mathrm{OPT}}$

$\boxed{\text{Lemma: } \forall u \text{ finishes by } \ell(u) + \deg(u)}$

$\boxed{\text{Thm: Alg is a 3-approx}}$

A few comments

- → Greedy Schedule
- → Dual update takes $O(m)$ time
- → Can handle weights
- → Integrality Gap $4/3$
- → Extends to Arb. proc times

Future Work

- → For min sum edge coloring Greedy is the best for general graphs
- → Almost b-maximal
- → Bipartite edge coloring in $O(m \log n)$
- → PD for scheduling problems

Thanks for
your attention !