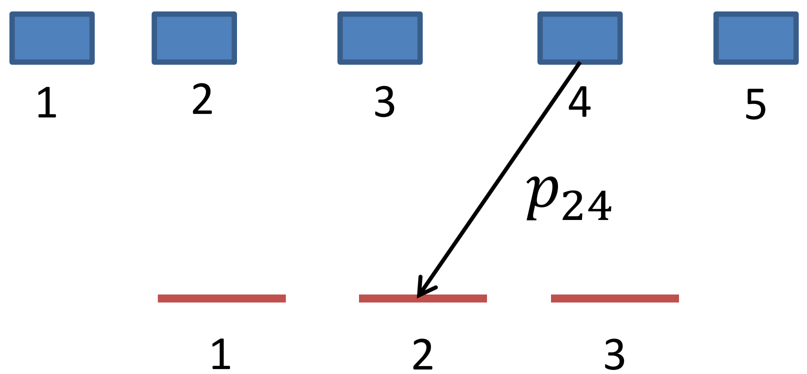
Minimizing total Completion Time on Unrelated Machines

w/ Aravind Srinivasan (Maryland)
Ola Svensson (EPFL)

Unrelated machine setting

n jobs: indexed by j

m machines: indexed by i



n=5 Jobs

m=3 machines

Job j has size p_{ij} on machine i

Unrelated setting: p_{ij} are arbitrary

Classic variants

Makespan: Minimize max. load $Min\ \ Max_i\ (\underbrace{\sum_j p_{ij} x_{ij}}_{\text{Load on i}})$

x_{ij}: 1 if job j assigned to machine i

2-approx (Lenstra,Shmoys, Tardos)

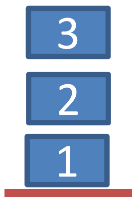
NP-Hard to beat 1.5

Lots of work, several variants

Total Weighted Completion Time

Min $\sum_j w_j C_j$

C_j: Completion time of j



Total completion time of these jobs

$w_1 p_1 + w_2(p_1 + p_2) + w_3(p_1 + p_2 + p_3)$

On any machine i, Smith rule: Decreasing order of $\frac{w_j}{p_{ij}}$

Only issue: Which machine to assign jobs

Objective: $\sum_i \sum_j w_j \left(\sum_{j' \preccurlyeq_i j} p_{ij'}\, x_{ij'}\, x_{ij} \right)$

Smith ordering $\prec_i$: $j' \prec_i j$ if $w_{j'}/p_{ij'} \geq w_j/p_{ij}$

Break ties arbitrarily to get total ordering

3/2: Convex programming

[Skutella, Sethuraman Squillante, Chudak late 90's]

Thm: 3/2 – 10^{-7} approximation

Convex Program: Integrality gap of 3/2

New SDP formulation

Independent Randomized Rounding

New dependent rounding w/ strict negative correlation

Convex Programming

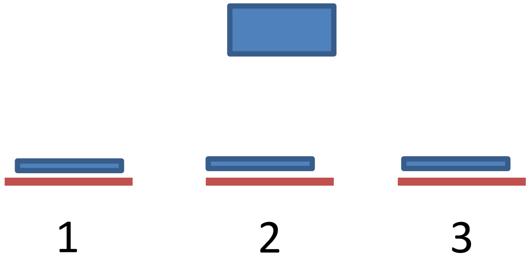
Objective (machine i): $\sum_j w_j \ (\sum_{j' \preccurlyeq_i j} p_{ij'} \ x_{ij'} \, x_{ij} \ )$

Suppose $w_i = \ p_{ij}$ (same Smith ratio): $p_{i1}x_{i1}\,(p_{i1}x_{i1}) + p_{i2}x_{i2}\,(p_{i1}x_{i1} + p_{i2}x_{i2}) + p_{i3}x_{i3}\,(p_{i1}x_{i1} + p_{i2}x_{i2} + p_{i3}x_{i3}) +$ …

$= \frac{1}{2}\ p_{ij}^2 x_{ij}^2 \ + \frac{1}{2}\ \left(\sum_j p_{ij} x_{ij}\right)^2$

Convex Program: Min $\sum_i \ (Expression\ i)$

s.t. $\sum_i x_{ij} = 1$ for all j.



Bad example: 1 job of size 1, m machines

Convex Program: Sets x_{i1} = 1/m $\left(m \ \cdot \frac{1}{m^2} = \frac{1}{m}\right)$

Convex Programming: Fix 1

$\frac{1}{2}\, p_{ij}^2 x_j^2 \;+ \frac{1}{2}\; \left(\sum_j p_{ij} x_j\right)^2$

Expression' i: $= \frac{1}{2}\, p_{ij}^2 x_{ij} \;+ \frac{1}{2}\; \left(\sum_j p_{ij} x_{ij}\right)^2$ [$x_{ij}^2 = x_{ij}$ valid]

Convex Program: Min $\sum_i \;(Expression'\, i)$

s.t. $\sum_i x_{ij} = 1$ for all j.

Bad example: 1 job of size 1, m machines

Convex Program: puts x_{i1} = 1/m

Objective = 1/2 + 1/(2m) [Still integrality gap of 2]

Fix 2: Reducing gap to 3/2

Expression' i: $= \frac{1}{2} p_{ij}^2 x_{ij} + \frac{1}{2} \left(\sum_j p_{ij} x_{ij}\right)^2$ [$x_{ij}^2 = x_{ij}$ valid]

Convex Program: Min $\sum_i (Expression' \ i)$

s.t. $\sum_i x_{ij} = 1$ for all j.

Add constraint: $OPT \geq \sum_i \sum_j p_{ij}^2 x_{ij}$ (i.e. $\sum_i \sum_j w_{ij} p_{ij} x_{ij}$)

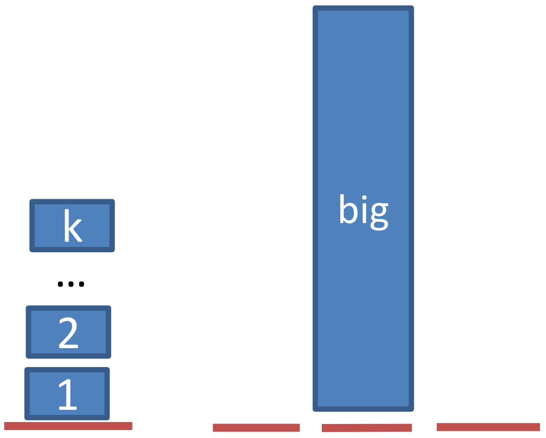
Somewhat adhoc fix

Surprisingly, integrality gap becomes 3/2

Another integrality gap example

k jobs: Size 1 each, only on machine 1

1 job: Size k^2 on any machine 2,…,k+1



Optimum: $\frac{k(k+1)}{2} + k^2 \approx \frac{3}{2}k^2$

Convex Program: ($OPT > \frac{L}{2} + \frac{Q}{2}, \quad OPT \geq L$)

Quadratic term (Q): $\approx \frac{k^2}{2} + \frac{1}{2}k^2$

Linear term (L) : $k + k^2$

New SDP

Write natural SDP (vectors v_{ij}, $x_{ij} = |v_{ij}|^2$)
Captures correlations and integrality more effectively

Add $v_{ij} \cdot v_{ij} = v_0 \cdot v_{ij}$ (like $x_{ij} = x_{ij}^2$)
($v_{ij} \cdot v_{ij}$' gives joint probability of j and j' on i)

Key: Linear and quadratic terms combined more systematically

E.g. For any subset of jobs $S \subset J$
OPT $\geq L(S) +$ ½ $L(S^c) +$ ½ $Q(S^c)$

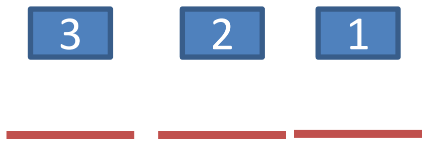
Previously: $OPT \geq L(J)$ and $OPT \geq$ ½ $L(J) +$ ½ $Q(J)$

The Rounding Issue

Given the x_{ij} , how to use these?

Randomized rounding stuck at 3/2.

k identical jobs (size 1) on k machines. Clearly, OPT = 1



Suppose $x_{ij} = 1/k$

Randomized rounding may assign > 1 jobs to machine

Pr[c jobs on a machine] := $p_c \approx \frac{1}{e}\left(\frac{1}{c!}\right)$

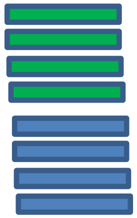
$$E\left[\frac{c(c+1)}{2} \quad p_c\right] = \frac{3}{2}$$

Suggests the following

1) If few jobs, do matching type rounding
2) If many jobs, randomized rounding ok.

Need a refinement of this (do matching for each job "class")

Consider machine 1
Class 1: size 1, weight 1
Class 2: size M, wt 1/M
Class 3: …

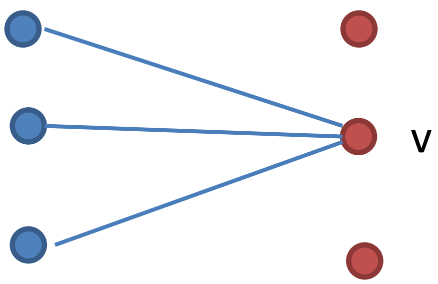


If k jobs fractionally to extent 1/k each.
Need to do matching in each class

And show this basically works.

Dependent rounding theorems

Gandhi et al. (Randomized pipage): Can find assignment so that get nice negative correlation at nodes



(e.g. $\Pr[\, x_{ij} x_{ij'}] \leq \Pr[x_{ij}]\Pr[x_{ij'}]$)
Only $\lceil f_v \rceil$ or $\lfloor f_v \rfloor$ edges at v

Our theorem: Machine -> Groups

(i) Strict negative correlation within groups, and
(ii) Negative correlation across groups.

[Randomized Pipage on paths of length 4, carefully chosen]

Questions!