Julius-Maximilians-
UNIVERSITÄT
WÜRZBURG

Approximationsalgorithmen

8. Vorlesung: Scheduling Jobs on Parallel Machines
via Parametrized Pruning

– Folien von J. Spoerhase –

Referenzen: [Vazirani, §17] und [Williamson & Shmoys, §2.3]

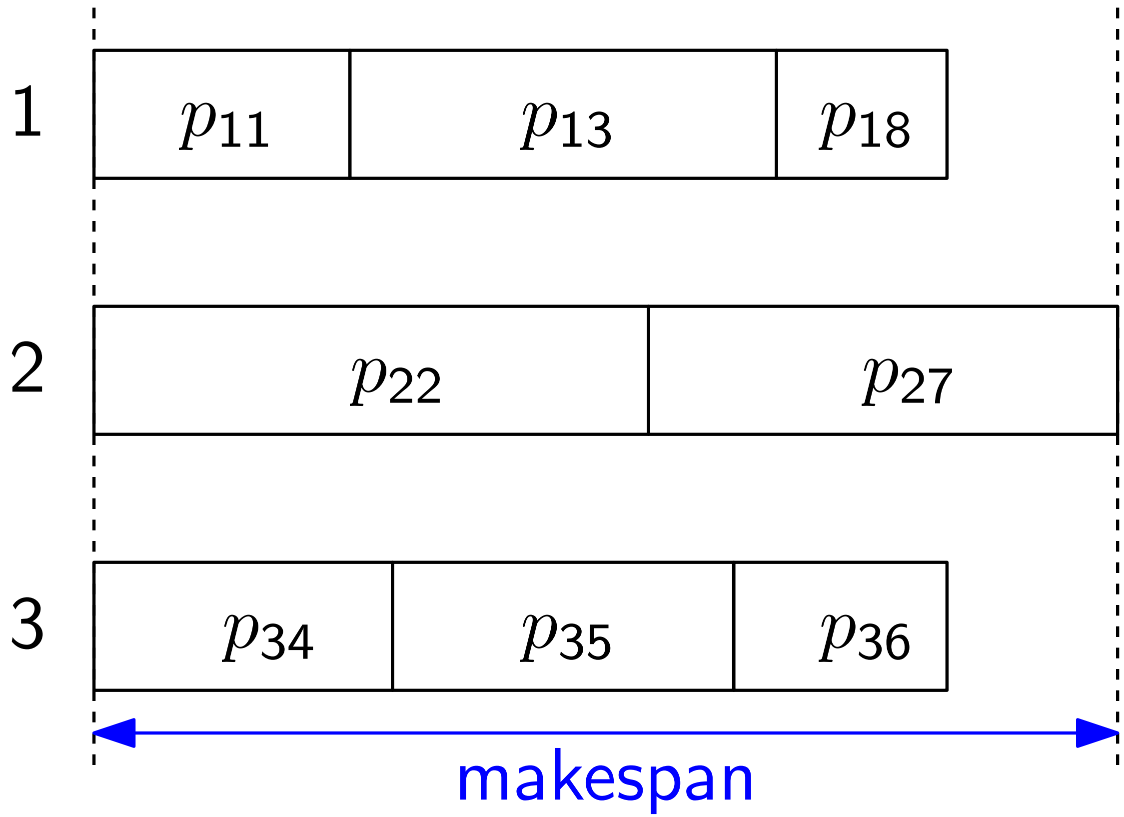
[Lenstra, Shmoys, Tardos, *Mathematical Programming*, 1990]

Alexander Wolff Lehrstuhl für Informatik I WS 2018/19

Scheduling on Parallel Machines

Given: A set J of **jobs**, a set M of **machines**, and for each $j \in J$ and $i \in M$ the **processing time** $p_{ij} \in \mathbb{N}^+$ of j on i.

Find: A **schedule** $\sigma\colon J \to M$ of the jobs on the machines that minimizes the total time to completion (**makespan**), i.e., minimizes the maximum time a machine is in use.



$J = \{1, 2, \ldots, 8\}$

$M = \{1, 2, 3\}$

$(p_{ij})_{i \in M, j \in J}$

A natural ILP

$$\begin{aligned} \text{Minimize} \quad & t \\ \text{subject to} \quad & \sum_{i \in M} x_{ij} = 1, && j \in J \\ & \sum_{j \in J} x_{ij} p_{ij} \leq t, && i \in M \\ & x_{ij} \in \{0, 1\}, && i \in M, j \in J \end{aligned}$$

Task: Show that the integrality gap of this ILP is unbounded!

Solution:

m machines and one job with processing time m

$\Rightarrow$ OPT $= m$ and $\text{OPT}_{\text{frac}} = 1$.

Parametrized Pruning

Strengthen the ILP $\to$ implicit (non-linear) constraint:
If $p_{ij} > t$ then set $x_{ij} = 0$.

Introduce a parameter $T \in \mathbb{N}$ to estimate lower bound on OPT.

Define $S_T := \{ (i,j) \colon i \in M,\, j \in J,\, p_{ij} \le T \}$.

Define the "pruned" relaxation LP(T):

LP(T) has no objective function; we just need to decide whether a feasible solution exists.

$$\sum_{i \colon (i,j) \in S_T} x_{ij} = 1, \qquad j \in J$$

$$\sum_{j \colon (i,j) \in S_T} x_{ij} p_{ij} \le T, \qquad i \in M$$

$$x_{ij} \ge 0, \qquad (i,j) \in S_T$$

But why does this LP solve the problem with the integrality gap?

Properties of Extreme-Point Solutions

Use binary search to find the smallest T so that LP(T) has a solution. Let T^* be this value of T.

What are the bounds for our search?

Note: $T^* \leq$ OPT

Idea: Round an extreme-point solution of LP(T^*) to a schedule whose makespan is $\leq 2T^*$

LP(T)

$$\sum_{i\colon (i,j)\in S_T} x_{ij} = 1, \qquad j \in J$$

$$\sum_{j\colon (i,j)\in S_T} x_{ij}p_{ij} \leq T, \qquad i \in M$$

$$x_{ij} \geq 0, \qquad (i,j) \in S_T$$

Lemma 1.
Each extreme-point solution to LP(T) has at most $m+n$ positive variables where $m = |M|, n = |J|$.

Lemma 2.
Any extreme-point solution to LP(T) must set at least $n - m$ jobs integrally.

Extreme-Point Solutions of LP(T)

Def. bipartite graph $G = (J \cup M, E)$ s.t. $(j, i) \in E \Leftrightarrow x_{ij} \neq 0$.

Jobs can be assigned *integrally* or *fractionally*.
$(\exists i \in M : 0 < x_{ij} < 1)$

Let $F \subseteq J$ be the set of fractionally assigned jobs.
Let $H := G[F \cup M]$.

Note: (i, j) is an edge in $H \Leftrightarrow 0 < x_{ij} < 1$

A matching in H is called F*-perfect* if it matches every vertex in F.

Key step: Show that H always has an F-perfect matching.

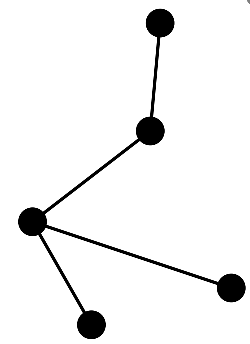
Why is this useful ....?

Algorithm

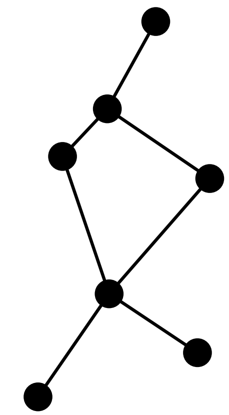
- Assign job j to machine i such that i is the machine minimizing p_{ij}. Let α be the makespan of this schedule.
- By a binary search in the interval $[\frac{\alpha}{m}, \alpha]$, find the smallest value of $T \in \mathbb{Z}^+$ for which LP(T) has a feasible solution and let this value be T^*.
- Find an extreme point solution, say $\mathbf{x}$, to LP(T^*).
- Assign all integrally set jobs to machines as in $\mathbf{x}$.
- Construct the graph H and find a perfect matching P in it (see Lemma 4 later).
- Assign the fractional jobs to machines using P.

Thm. This algorithm is a 2-approximation (assuming that we have an F-perfect matching).

Pseudo-Trees and -Forests



A connected graph with vertex set V is called a **pseudo-tree** if it has at most $|V|$ edges.



A pseudo-tree is a tree or a tree plus a single edge.

A collection of disjoint pseudo-trees is called a **pseudo-forest**.

Lem. 3. The bipartite graph $G = (J \cup M, E)$ is a pseudo-forest.

Extreme-point solutions have $\leq n + m$ non-zero variables (L1).
Each component of G corresponds to extreme-point solution.

Lem. 4. The graph H has an F-perfect matching.

H is a pseudo-forest, too. The leaves in H are machines.
After picking leaf edges exhaustively, only even cycles remain.

Scheduling on Parallel Machines

Thm. There is an LP-based 2-approximation algorithm for the problem of scheduling jobs on unrelated parallel machines.

Is this tight? Yes!

Instance I_m:

- $m^2 - m + 1$ jobs to be scheduled on m machines.
- job j_1 has a processing time of m on all machines,
- all other jobs have unit processing time on each machine.

Optimum: one machine with j_1, and all others spread evenly.

Algorithm:

- LP(T) has no feasible solutions for any $T < m$.
- Extreme-pt. solution: Assign $1/m$ of j_1 and $m-1$ other jobs to each machine. $\Rightarrow$ Makespan $2m-1$.

Scheduling on Parallel Machines

Thm. There is an LP-based 2-approximation algorithm for the problem of scheduling jobs on unrelated parallel machines. The approximation factor is tight.