High-Multiplicity Scheduling on Identical Machines

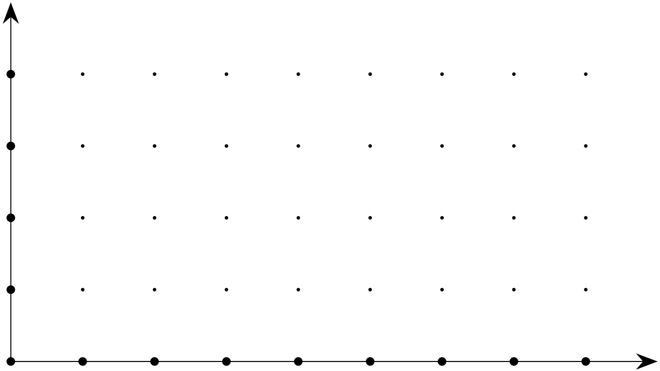
Kai Kahler

joint work with Klaus Jansen and Esther Zwanger

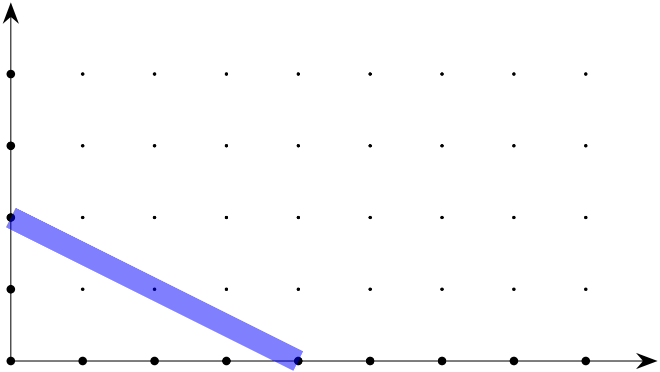
Kiel University

May 16, 2024

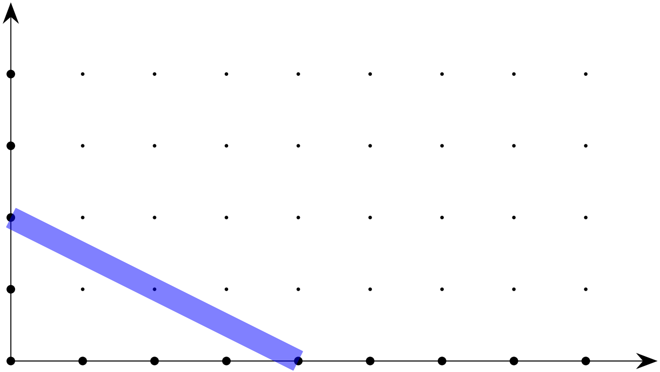
ConeAndPolytopeIntersection



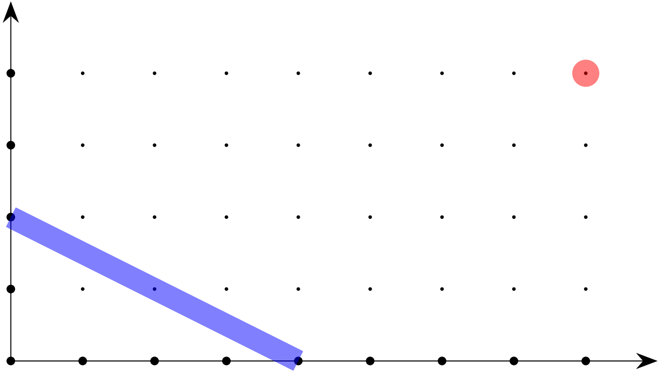
ConeAndPolytopeIntersection



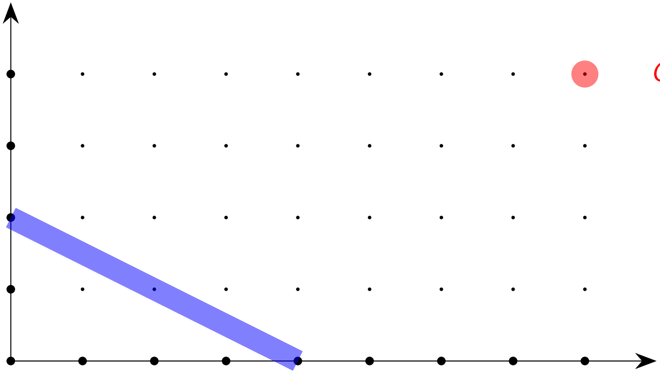
ConeAndPolytopeIntersection



$$P = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2_{\geq 0} \mid \begin{pmatrix} \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2 \right\}$$



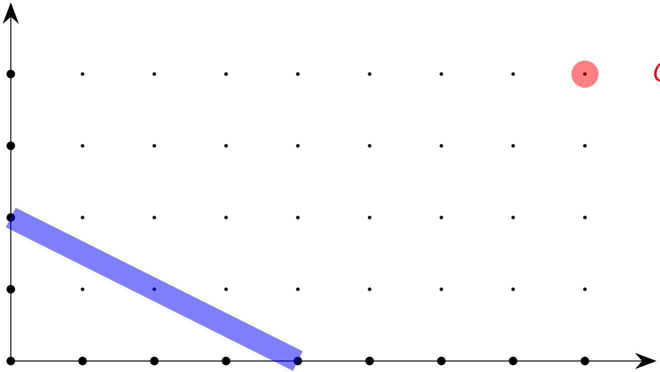
$$P = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2_{\geq 0} \mid \begin{pmatrix} \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2 \right\}$$



$$Q = \left\{ \begin{pmatrix} 8 \\ 4 \end{pmatrix} \right\}$$

$$P = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2_{\geq 0} \mid \begin{pmatrix} \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2 \right\}$$

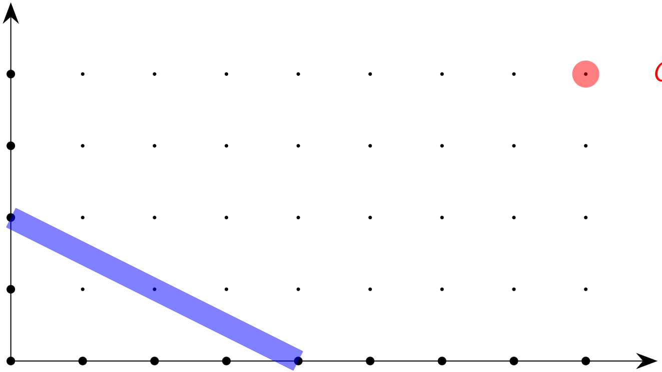
ConeAndPolytopeIntersection



$Q = \left\{ \begin{pmatrix} 8 \\ 4 \end{pmatrix} \right\}$

$m = 4$

$P = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2_{\geq 0} \mid \begin{pmatrix} \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2 \right\}$



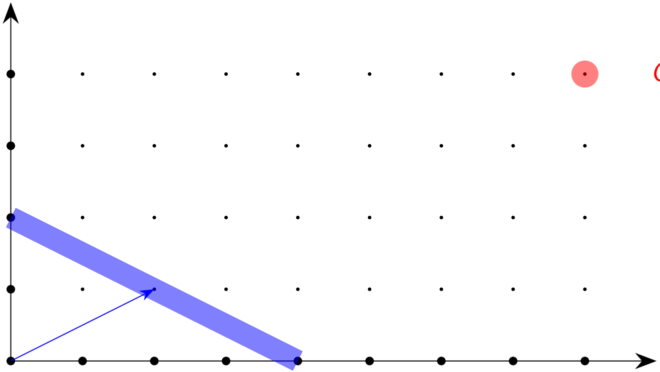
$$Q = \left\{ \begin{pmatrix} 8 \\ 4 \end{pmatrix} \right\}$$

$$m = 4$$

$$P = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2_{\geq 0} \mid \begin{pmatrix} \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2 \right\}$$

$$P \cap \mathbb{Z}^2 = \left\{ \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \end{pmatrix} \right\}$$

ConeAndPolytopeIntersection



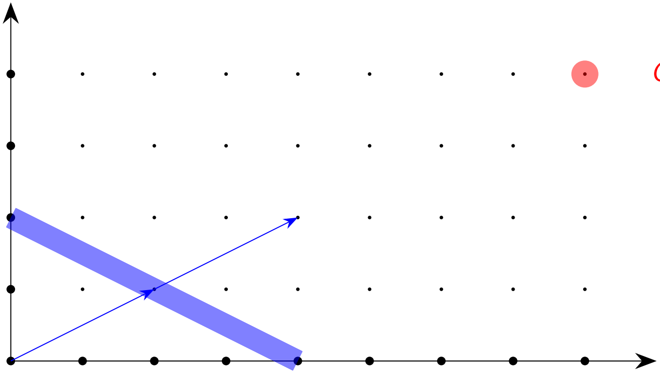
$$Q = \left\{ \begin{pmatrix} 8 \\ 4 \end{pmatrix} \right\}$$

$$m = 4$$

$$P = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2_{\geq 0} \mid \begin{pmatrix} \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2 \right\}$$

$$P \cap \mathbb{Z}^2 = \left\{ \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \end{pmatrix} \right\}$$

ConeAndPolytopeIntersection



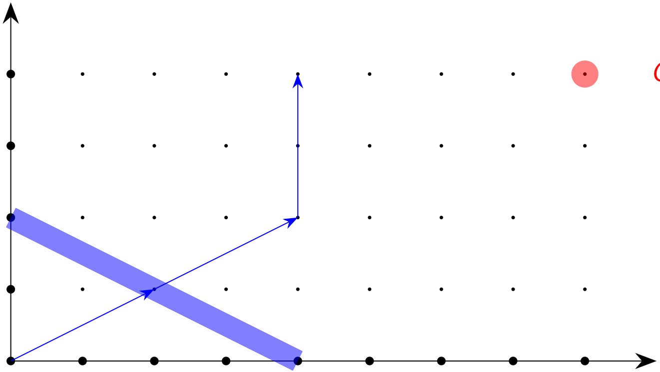
$Q = \left\{ \begin{pmatrix} 8 \\ 4 \end{pmatrix} \right\}$

$m = 4$

$P = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2_{\geq 0} \mid \begin{pmatrix} \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2 \right\}$

$P \cap \mathbb{Z}^2 = \left\{ \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \end{pmatrix} \right\}$

ConeAndPolytopeIntersection

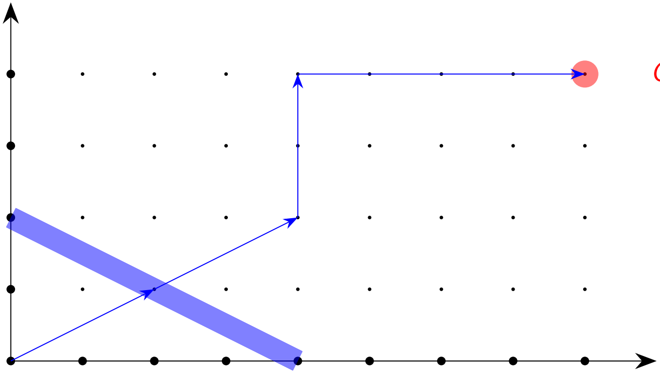


$Q = \left\{ \begin{pmatrix} 8 \\ 4 \end{pmatrix} \right\}$

$m = 4$

$$P = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2_{\geq 0} \mid \begin{pmatrix} \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2 \right\}$$

$$P \cap \mathbb{Z}^2 = \left\{ \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \end{pmatrix} \right\}$$

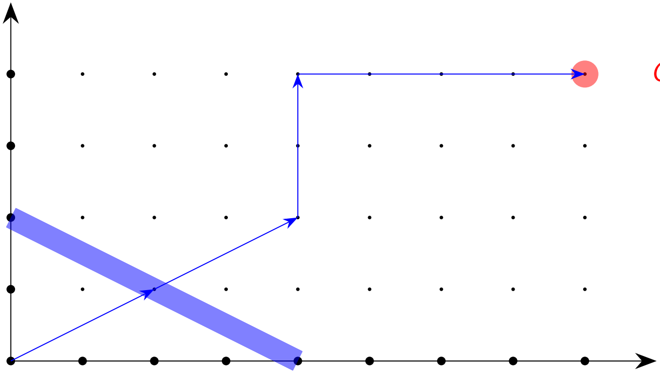


$Q = \left\{ \begin{pmatrix} 8 \\ 4 \end{pmatrix} \right\}$

$m = 4$

$P = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2_{\geq 0} \mid \begin{pmatrix} \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2 \right\}$

$P \cap \mathbb{Z}^2 = \left\{ \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \end{pmatrix} \right\}$



$Q = \left\{ \begin{pmatrix} 8 \\ 4 \end{pmatrix} \right\}$

$m = 4$

$P = \left\{ \begin{pmatrix} x \\ y \end{pmatrix} \in \mathbb{R}^2_{\geq 0} \mid \begin{pmatrix} \frac{1}{2} & 1 \end{pmatrix} \begin{pmatrix} x \\ y \end{pmatrix} = 2 \right\}$

$P \cap \mathbb{Z}^2 = \left\{ \begin{pmatrix} 2 \\ 1 \end{pmatrix}, \begin{pmatrix} 0 \\ 2 \end{pmatrix}, \begin{pmatrix} 4 \\ 0 \end{pmatrix} \right\}$

$2 \cdot \begin{pmatrix} 2 \\ 1 \end{pmatrix} + 1 \cdot \begin{pmatrix} 0 \\ 2 \end{pmatrix} + 1 \cdot \begin{pmatrix} 4 \\ 0 \end{pmatrix} \in \text{int.cone}(P \cap \mathbb{Z}^2) \cap Q$

The ConeAndPolytopeIntersection Problem

Input:

- Polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(P)}x = b^{(P)}\}$
- Polytope $Q = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(Q)}x = b^{(Q)}\}$
- Number $m \in \mathbb{N}_{>0}$

The ConeAndPolytopeIntersection Problem

Input:

- Polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(P)}x = b^{(P)}\}$
- Polytope $Q = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(Q)}x = b^{(Q)}\}$
- Number $m \in \mathbb{N}_{>0}$

Task: Decide whether there is a point $y \in \text{int.cone}(P \cap \mathbb{Z}^N) \cap Q$ such that $y = \sum_{i=1}^{m} y_i$ with $y_i \in P \cap \mathbb{Z}^N$ for all i.

High-Multiplicity $P||C_{\max}$

Input:

- Processing times $p \in \mathbb{N}_{>0}^{d}$

High-Multiplicity $P||C_{\max}$

Input:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$

High-Multiplicity $P||C_{\max}$

Input:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$

High-Multiplicity $P||C_{\max}$

Input:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$
- Makespan threshold T

High-Multiplicity $P||C_{\max}$

Input:

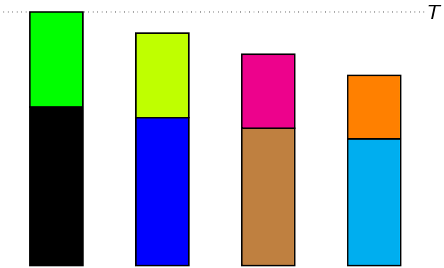
- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$
- Makespan threshold T

Task: Decide whether there exists a *schedule* with makespan $\leq T$.

High-Multiplicity $P||C_{\max}$

Input:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$
- Makespan threshold T

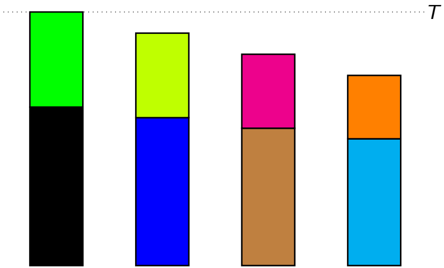


Task: Decide whether there exists a *schedule* with makespan $\leq T$.

High-Multiplicity $P||C_{\max}$

Input:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$
- Makespan threshold T



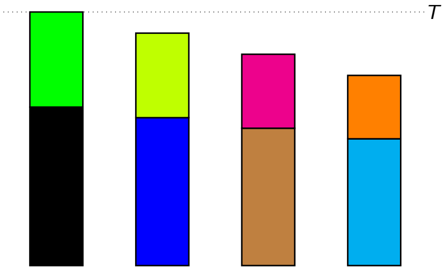
Task: Decide whether there exists a *schedule* with makespan $\leq T$.

Encoding Length: $\langle I \rangle \leq O(d(\log(p_{\max} + n_{\max})) + \log(m) + \log(T))$.

High-Multiplicity $P||C_{\max}$

Input:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$
- Makespan threshold T



Task: Decide whether there exists a *schedule* with makespan $\leq T$.

Encoding Length: $\langle I \rangle \leq O(d(\log(p_{\max} + n_{\max})) + \log(m) + \log(T))$.

Parameter: Number of distinct processing times d.

Configurations

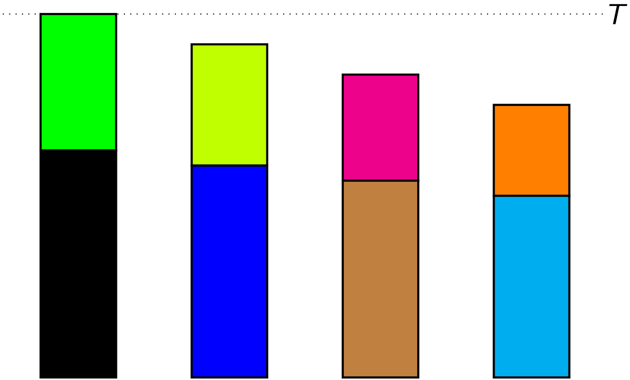
Configuration: Set of jobs (in form of a vector) that can be scheduled together on a machine (in time T).

Configurations

Configuration: Set of jobs (in form of a vector) that can be scheduled together on a machine (in time T).
Set of configurations: $C = \{c \in \mathbb{N}^d \mid p^T c \leq T\}$

Configurations

Configuration: Set of jobs (in form of a vector) that can be scheduled together on a machine (in time T).
Set of configurations: $C = \{c \in \mathbb{N}^d \mid p^T c \leq T\}$



$P||C_{\max}$ via ConeAndPolytopeIntersection

ConeAndPolytopeIntersection:

- Polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(P)}x = b^{(P)}\}$
- Polytope $Q = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(Q)}x = b^{(Q)}\}$
- Number $m \in \mathbb{N}_{>0}$

$P||C_{\max}$ via ConeAndPolytopeIntersection

ConeAndPolytopeIntersection:

- Polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(P)}x = b^{(P)}\}$
- Polytope $Q = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(Q)}x = b^{(Q)}\}$
- Number $m \in \mathbb{N}_{>0}$

$P||C_{\max}$:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$
- Makespan threshold T

$P||C_{\max}$ via ConeAndPolytopeIntersection

ConeAndPolytopeIntersection:

- Polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(P)}x = b^{(P)}\}$
- Polytope $Q = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(Q)}x = b^{(Q)}\}$
- Number $m \in \mathbb{N}_{>0}$

$P||C_{\max}$:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$
- Makespan threshold T

$$P := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid \begin{pmatrix} p^T & 1 \end{pmatrix} \begin{pmatrix} c \\ s \end{pmatrix} = T \right\},$$

$P||C_{\max}$ via ConeAndPolytopeIntersection

ConeAndPolytopeIntersection:

- Polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(P)}x = b^{(P)}\}$
- Polytope $Q = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(Q)}x = b^{(Q)}\}$
- Number $m \in \mathbb{N}_{>0}$

$P||C_{\max}$:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$
- Makespan threshold T

$$P := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid \begin{pmatrix} p^T & 1 \end{pmatrix} \begin{pmatrix} c \\ s \end{pmatrix} = T \right\}, \quad Q := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid c = n \right\},$$

$P||C_{\max}$ via ConeAndPolytopeIntersection

ConeAndPolytopeIntersection:

- Polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(P)}x = b^{(P)}\}$
- Polytope $Q = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(Q)}x = b^{(Q)}\}$
- Number $m \in \mathbb{N}_{>0}$

$P||C_{\max}$:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$
- Makespan threshold T

$$P := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid \begin{pmatrix} p^T & 1 \end{pmatrix} \begin{pmatrix} c \\ s \end{pmatrix} = T \right\}, \quad Q := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid c = n \right\}, \quad m := m$$

$P||C_{\max}$ via ConeAndPolytopeIntersection

ConeAndPolytopeIntersection:

- Polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(P)}x = b^{(P)}\}$
- Polytope $Q = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(Q)}x = b^{(Q)}\}$
- Number $m \in \mathbb{N}_{>0}$

$P||C_{\max}$:

- Processing times $p \in \mathbb{N}^d_{>0}$
- Multiplicities $n \in \mathbb{N}^d_{>0}$
- Number of machines $m \in \mathbb{N}_{>0}$
- Makespan threshold T

$$P := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid \begin{pmatrix} p^T & 1 \end{pmatrix} \begin{pmatrix} c \\ s \end{pmatrix} = T \right\}, \quad Q := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid c = n \right\}, \quad m := m$$

$P||C_{\max}$-instance is feasible

$$\Longleftrightarrow$$

$\exists y \in \text{int.cone}(P \cap \mathbb{Z}^{d+1}) \cap Q$ such that $y = \sum_{i=1}^{m} y_i$ with $y_i \in P \cap \mathbb{Z}^{d+1}$ for all i

Previous Results for ConeAndPolytopeIntersection

- $\langle P \rangle^{2^{O(N)}} (\langle Q \rangle \log(m))^{O(1)}$-time algorithm (Goemans & Rothvoss, 2014)

Previous Results for ConeAndPolytopeIntersection

- $\langle P \rangle^{2^{O(N)}} (\langle Q \rangle \log(m))^{O(1)}$-time algorithm (Goemans & Rothvoss, 2014)
- $|V|^{2^{O(N)}} (\langle P \rangle \langle Q \rangle \log(m))^{O(1)}$-time algorithm (Jansen & Klein, 2017)
  - $|V|$ = number of vertices of $\text{conv}(P \cap \mathbb{Z}^N)$

Previous Results for ConeAndPolytopeIntersection

- $\langle P \rangle^{2^{O(N)}}(\langle Q \rangle \log(m))^{O(1)}$-time algorithm (Goemans & Rothvoss, 2014)
- $|V|^{2^{O(N)}}(\langle P \rangle \langle Q \rangle \log(m))^{O(1)}$-time algorithm (Jansen & Klein, 2017)
  - $|V| =$ number of vertices of $\text{conv}(P \cap \mathbb{Z}^N)$
- Double-exponential dependency on N needed (for general polytopes) under ETH (Kowalik et al., 2024)

Previous Results for ConeAndPolytopeIntersection

- $\langle P \rangle^{2^{O(N)}} (\langle Q \rangle \log(m))^{O(1)}$-time algorithm (Goemans & Rothvoss, 2014)
- $|V|^{2^{O(N)}} (\langle P \rangle \langle Q \rangle \log(m))^{O(1)}$-time algorithm (Jansen & Klein, 2017)
  - $|V| =$ number of vertices of $\mathrm{conv}(P \cap \mathbb{Z}^N)$
- Double-exponential dependency on N needed (for general polytopes) under ETH (Kowalik et al., 2024)

$$P := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid \begin{pmatrix} p^T & 1 \end{pmatrix} \begin{pmatrix} c \\ s \end{pmatrix} = T \right\}, \quad Q := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid c = n \right\}, \quad m := m$$

Previous Results for ConeAndPolytopeIntersection

- $\langle P \rangle^{2^{O(N)}} (\langle Q \rangle \log(m))^{O(1)}$-time algorithm (Goemans & Rothvoss, 2014)
- $|V|^{2^{O(N)}} (\langle P \rangle \langle Q \rangle \log(m))^{O(1)}$-time algorithm (Jansen & Klein, 2017)
  - $|V| =$ number of vertices of $\mathrm{conv}(P \cap \mathbb{Z}^N)$
- Double-exponential dependency on N needed (for general polytopes) under ETH (Kowalik et al., 2024)

$$P := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid \begin{pmatrix} p^T & 1 \end{pmatrix} \begin{pmatrix} c \\ s \end{pmatrix} = T \right\}, \quad Q := \left\{ \begin{pmatrix} c \\ s \end{pmatrix} \in \mathbb{R}^{d+1}_{\geq 0} \mid c = n \right\}, \quad m := m$$

$$\implies \log(T)^{2^{O(d)}} \langle I \rangle^{O(1)}\text{-time algorithm for } P||C_{\max}$$

Our Contribution

Theorem

ConeAndPolytopeIntersection *can be solved in time*

$$(\log(\Delta))^{2^{O(K)}} (\langle P\rangle\langle Q\rangle \log(m))^{O(1)}$$

where $K := \max\{N, M^{(P)}, M^{(Q)}\}$ *and* $\Delta = \max\{\|A^{(P)}\|_\infty, \|A^{(Q)}\|_\infty\}$.

Our Contribution

Theorem

ConeAndPolytopeIntersection *can be solved in time*

$$(\log(\Delta))^{2^{O(K)}}(\langle P\rangle\langle Q\rangle\log(m))^{O(1)} = 2^{2^{O(K)}}\Delta^{o(1)}(\langle P\rangle\langle Q\rangle\log(m))^{O(1)},$$

where $K := \max\{N, M^{(P)}, M^{(Q)}\}$ *and* $\Delta = \max\{\|A^{(P)}\|_\infty, \|A^{(Q)}\|_\infty\}$.

Our Contribution

Theorem

ConeAndPolytopeIntersection can be solved in time

$$(\log(\Delta))^{2^{O(K)}}(\langle P\rangle\langle Q\rangle\log(m))^{O(1)} = 2^{2^{O(K)}}\Delta^{o(1)}(\langle P\rangle\langle Q\rangle\log(m))^{O(1)},$$

where $K := \max\{N, M^{(P)}, M^{(Q)}\}$ *and* $\Delta = \max\{\left\|A^{(P)}\right\|_\infty, \left\|A^{(Q)}\right\|_\infty\}$.

Theorem

$P||C_{\max}$ *can be solved in time* $(\log(p_{\max}))^{2^{O(d)}}\langle I\rangle^{O(1)} = 2^{2^{O(d)}}p_{\max}^{o(1)}\langle I\rangle^{O(1)}$.

Outline

Balancing for $P||C_{\max}$

Theorem (Govzmann et al.)

$P||C_{\max}$ has a kernel where the number of jobs of a specific type on a specific machine is bounded by $2p_{\max}$, i.e., the load of every machine is bounded by $2p_{\max}^2 d$.

Balancing for $P||C_{\max}$

Theorem (Govzmann et al.)

$P||C_{\max}$ has a kernel where the number of jobs of a specific type on a specific machine is bounded by $2p_{\max}$, i.e., the load of every machine is bounded by $2p_{\max}^2 d$.

$$\implies \log(T)^{2^{O(d)}} \langle I \rangle^{O(1)} \leq \log(2p_{\max}^2 d)^{2^{O(d)}} \langle I \rangle^{O(1)} = \log(p_{\max})^{2^{O(d)}} \langle I \rangle^{O(1)}$$

Balancing for $P||C_{\max}$

Theorem (Govzmann et al.)

$P||C_{\max}$ has a kernel where the number of jobs of a specific type on a specific machine is bounded by $2p_{\max}$, i.e., the load of every machine is bounded by $2p_{\max}^2 d$.

$$\implies \log(T)^{2^{O(d)}} \langle I \rangle^{O(1)} \leq \log(2p_{\max}^2 d)^{2^{O(d)}} \langle I \rangle^{O(1)} = \log(p_{\max})^{2^{O(d)}} \langle I \rangle^{O(1)}$$

Rough idea:

- Show that there is an optimal schedule that uses only similar ($\|c_i - c_j\|_\infty \leq p_{\max}$) configurations.

Balancing for $P||C_{\max}$

Theorem (Govzmann et al.)

$P||C_{\max}$ has a kernel where the number of jobs of a specific type on a specific machine is bounded by $2p_{\max}$, i.e., the load of every machine is bounded by $2p_{\max}^2 d$.

$$\implies \log(T)^{2^{O(d)}} \langle I \rangle^{O(1)} \leq \log(2p_{\max}^2 d)^{2^{O(d)}} \langle I \rangle^{O(1)} = \log(p_{\max})^{2^{O(d)}} \langle I \rangle^{O(1)}$$

Rough idea:

- Show that there is an optimal schedule that uses only similar ($\|c_i - c_j\|_\infty \leq p_{\max}$) configurations.
- Then such an optimal schedule is quite similar to the fractional schedule.

Balancing for $P||C_{\max}$

Theorem (Govzmann et al.)

$P||C_{\max}$ has a kernel where the number of jobs of a specific type on a specific machine is bounded by $2p_{\max}$, i.e., the load of every machine is bounded by $2p_{\max}^2 d$.

$$\implies \log(T)^{2^{O(d)}} \langle I \rangle^{O(1)} \leq \log(2p_{\max}^2 d)^{2^{O(d)}} \langle I \rangle^{O(1)} = \log(p_{\max})^{2^{O(d)}} \langle I \rangle^{O(1)}$$

Rough idea:

- Show that there is an optimal schedule that uses only similar ($\|c_i - c_j\|_\infty \leq p_{\max}$) configurations.
- Then such an optimal schedule is quite similar to the fractional schedule.
- Pre-schedule many of the jobs like in the fractional schedule.

Outline

ILP Representation

Lemma

Given $P = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(P)}x = b^{(P)}\}$ *and* $Q = \{x \in \mathbb{R}^N_{\geq 0} \mid A^{(Q)}x = b^{(Q)}\}$ *and* $m \in \mathbb{N}$, *the points* $y \in \text{int.cone}(P \cap \mathbb{Z}^N) \cap Q$ *have a one-to-one correspondence to the solutions of the following ILP, projected to block* $m+1$ *and the other blocks correspond to vectors* $y^{(1)}, \ldots, y^{(m)} \in P \cap \mathbb{Z}^N$ *such that* $y = y^{(1)} + \ldots + y^{(m)}$:

$$\begin{pmatrix} \begin{matrix} 0 \\ I \end{matrix} & \begin{matrix} 0 \\ I \end{matrix} & \begin{matrix} \cdots \\ \cdots \end{matrix} & \begin{matrix} 0 \\ I \end{matrix} & \begin{matrix} A^{(Q)} \\ -I \end{matrix} \\ A^{(P)} & & & & \\ & A^{(P)} & & & \\ & & \ddots & & \\ & & & A^{(P)} & \\ & & & & 0 \end{pmatrix} x = \begin{pmatrix} b^{(Q)} \\ \mathbf{0} \\ b^{(P)} \\ b^{(P)} \\ \vdots \\ b^{(P)} \\ \mathbf{0} \end{pmatrix}, \quad x \in \mathbb{N}^{(m+1)N}$$

Proximity

Lemma

A convexified relaxation of the ILP can be solved efficiently.

Proximity

Lemma

A convexified relaxation of the ILP can be solved efficiently.

Theorem (Cslovjecsek et al., 2020)

For every solution x^* of the convexified relaxation, there exists a solution z^* of the corresponding integer program such that $\|x^* - z^*\|_1$ is small (and in particular does not depend on $\|b\|_\infty$).

Idea

- Compute a solution x^* of the convexified relaxation.
  - This corresponds to a "fractional solution" of the ConeAndPolytopeIntersection problem.

Idea

- Compute a solution x^* of the convexified relaxation.
  - This corresponds to a "fractional solution" of the $\textsc{ConeAndPolytopeIntersection}$ problem.
- Then there exists a solution z^* of the ILP close to x^*.
  - This corresponds to an actual solution of the $\textsc{ConeAndPolytopeIntersection}$ problem.

Idea

- Compute a solution x^* of the convexified relaxation.
  - This corresponds to a "fractional solution" of the ConeAndPolytopeIntersection problem.
- Then there exists a solution z^* of the ILP close to x^*.
  - This corresponds to an actual solution of the ConeAndPolytopeIntersection problem.

$\implies$ We can restrict our search for a solution of the ConeAndPolytopeIntersection problem.

Idea

- Compute a solution x^* of the convexified relaxation.
  - This corresponds to a "fractional solution" of the ConeAndPolytopeIntersection problem.
- Then there exists a solution z^* of the ILP close to x^*.
  - This corresponds to an actual solution of the ConeAndPolytopeIntersection problem.

$\implies$ We can restrict our search for a solution of the ConeAndPolytopeIntersection problem.

$\implies$ We only need to consider points close to x^*.

Idea

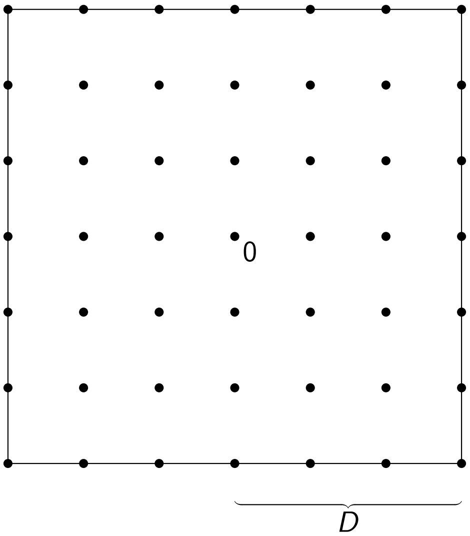
- Compute a solution x^* of the convexified relaxation.
  - This corresponds to a "fractional solution" of the $\textsc{ConeAndPolytopeIntersection}$ problem.
- Then there exists a solution z^* of the ILP close to x^*.
  - This corresponds to an actual solution of the $\textsc{ConeAndPolytopeIntersection}$ problem.

$\Longrightarrow$ We can restrict our search for a solution of the $\textsc{ConeAndPolytopeIntersection}$ problem.

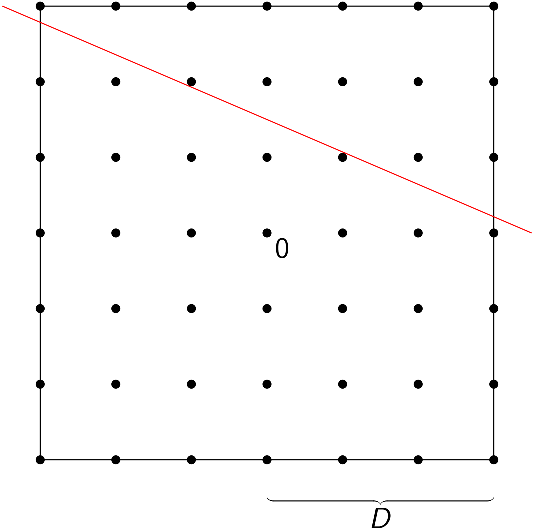
$\Longrightarrow$ We only need to consider points close to x^*.

$\Longrightarrow$ We can assume that the points in P are bounded by some D (depending on the proximity bound).

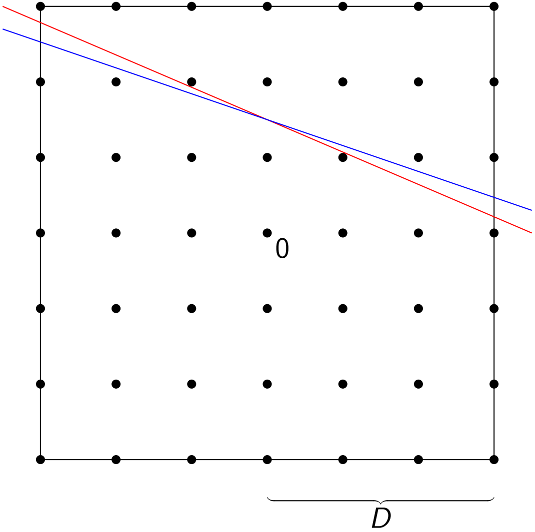
Frank & Tardos, 1987



Frank & Tardos, 1987



Frank & Tardos, 1987



Algorithm

1. Compute a solution x^* of the convexified relaxation.

Algorithm

1. Compute a solution x^* of the convexified relaxation.
2. Restrict P to the points near x^*.

Algorithm

1. Compute a solution x^* of the convexified relaxation.
2. Restrict P to the points near x^*.
3. Modify the description of P (and reduce $\langle P \rangle$) with the technique by Frank and Tardos.

Algorithm

1. Compute a solution x^* of the convexified relaxation.
2. Restrict P to the points near x^*.
3. Modify the description of P (and reduce $\langle P \rangle$) with the technique by Frank and Tardos.
4. Apply the $\langle P \rangle^{2^{O(N)}}(\langle Q \rangle \log(m))^{O(1)}$-time algorithm by Goemans and Rothvoss to m, Q and the modified P.

Algorithm

1. Compute a solution x^* of the convexified relaxation.
2. Restrict P to the points near x^*.
3. Modify the description of P (and reduce $\langle P \rangle$) with the technique by Frank and Tardos.
4. Apply the $\langle P \rangle^{2^{O(N)}} (\langle Q \rangle \log(m))^{O(1)}$-time algorithm by Goemans and Rothvoss to m, Q and the modified P.

$$\implies (\log(\Delta))^{2^{O(K)}} (\langle P \rangle \langle Q \rangle \log(m))^{O(1)} - \text{time algorithm},$$

where $K := \max\{N, M^{(P)}, M^{(Q)}\}$ and $\Delta = \max\{\left\|A^{(P)}\right\|_\infty, \left\|A^{(Q)}\right\|_\infty\}$.

Outline

1. Problem-Specific Preprocessing
2. Block-ILPs and Proximity
3. A Bound for the Vertices of the Integer Hull

Vertices of the Integer Hull

Theorem

The number of vertices of the integer hull of a polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid Ax = b\}$ *is bounded by* $N^M O(M \log(M\Delta))^N$*, where* $A \in \mathbb{Z}^{M \times N}$ *and* $\Delta = \|A\|_\infty$*.*

Vertices of the Integer Hull

Theorem

The number of vertices of the integer hull of a polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid Ax = b\}$ *is bounded by* $N^M O(M \log(M\Delta))^N$*, where* $A \in \mathbb{Z}^{M \times N}$ *and* $\Delta = \|A\|_\infty$.

With the $|V|^{2^{O(N)}}(\langle Q\rangle \log(m))^{O(1)}$-time algorithm by Jansen and Klein, this yields:

$$|V|^{2^{O(N)}}(\langle P\rangle\langle Q\rangle \log(m))^{O(1)}$$

Vertices of the Integer Hull

Theorem

The number of vertices of the integer hull of a polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid Ax = b\}$ *is bounded by* $N^M O(M \log(M\Delta))^N$, *where* $A \in \mathbb{Z}^{M \times N}$ *and* $\Delta = \|A\|_\infty$.

With the $|V|^{2^{O(N)}}(\langle Q \rangle \log(m))^{O(1)}$-time algorithm by Jansen and Klein, this yields:

$$|V|^{2^{O(N)}}(\langle P \rangle \langle Q \rangle \log(m))^{O(1)} \leq \left(\log\left(\left\|A^{(P)}\right\|_\infty\right)\right)^{2^{O(\max\{N, M^{(Q)}\})}} (\langle P \rangle \langle Q \rangle \log(m))^{O(1)}$$

Vertices of the Integer Hull

Theorem

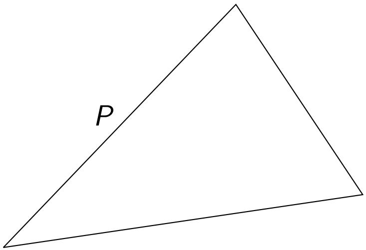
The number of vertices of the integer hull of a polytope $P = \{x \in \mathbb{R}^N_{\geq 0} \mid Ax = b\}$ *is bounded by* $N^M O(M \log(M\Delta))^N$, *where* $A \in \mathbb{Z}^{M \times N}$ *and* $\Delta = \|A\|_\infty$.

With the $|V|^{2^{O(N)}}(\langle Q \rangle \log(m))^{O(1)}$-time algorithm by Jansen and Klein, this yields:

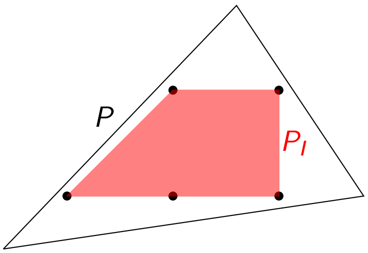
$$|V|^{2^{O(N)}}(\langle P \rangle \langle Q \rangle \log(m))^{O(1)} \leq \left(\log\left(\left\|A^{(P)}\right\|_\infty\right)\right)^{2^{O(\max\{N, M^{(Q)}\})}} (\langle P \rangle \langle Q \rangle \log(m))^{O(1)}$$

Proof is quite similar to the one of the vertex bound by Berndt et al., 2021.

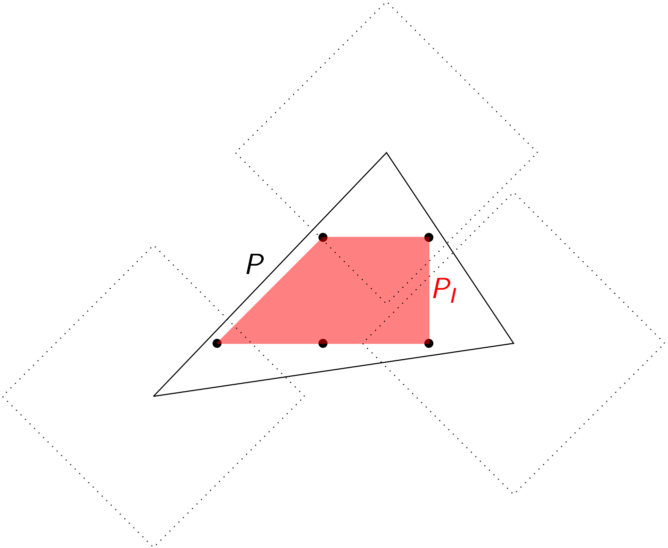
Proof of the Vertex Bound



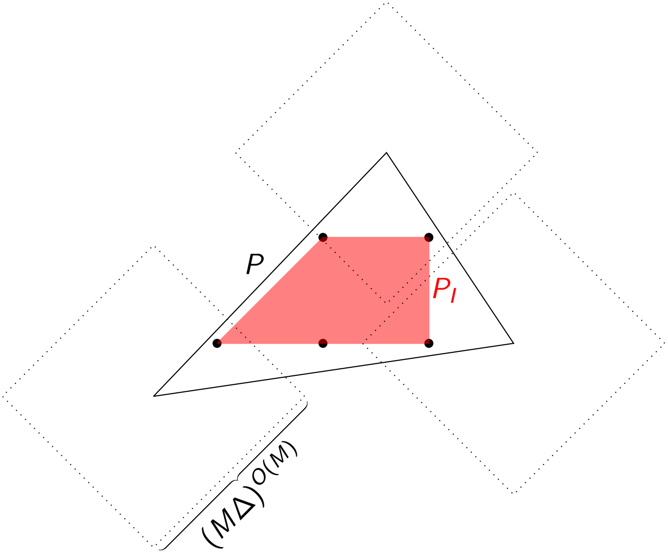
Proof of the Vertex Bound



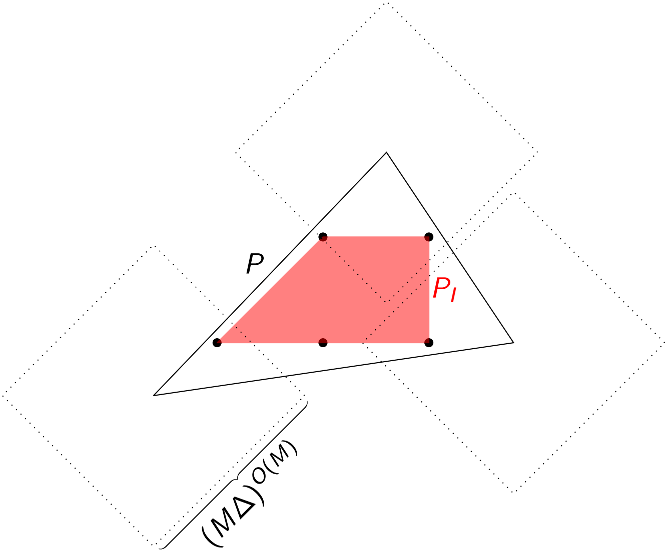
Proof of the Vertex Bound



Proof of the Vertex Bound

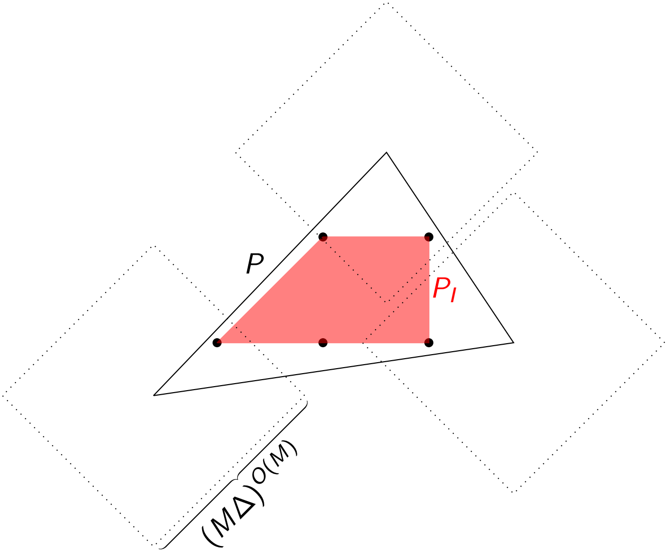


Proof of the Vertex Bound



$\#$ vertices of P_I

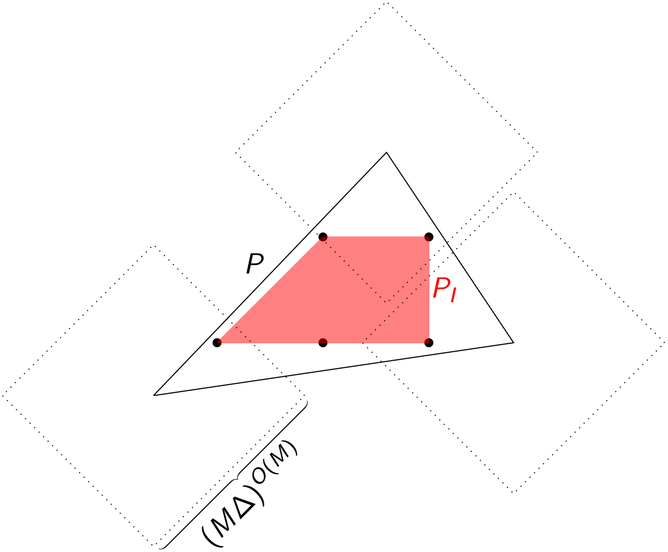
Proof of the Vertex Bound



$\#$ vertices of P_I

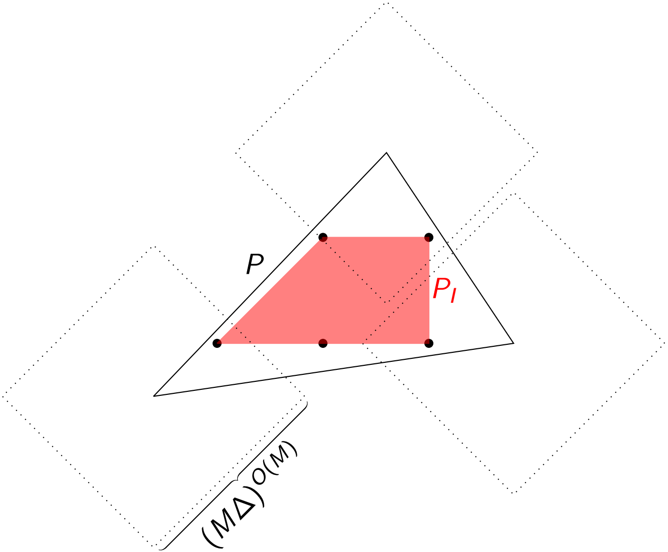
$\leq \#$ vertices of $P \cdot \#$ grid cells

Proof of the Vertex Bound



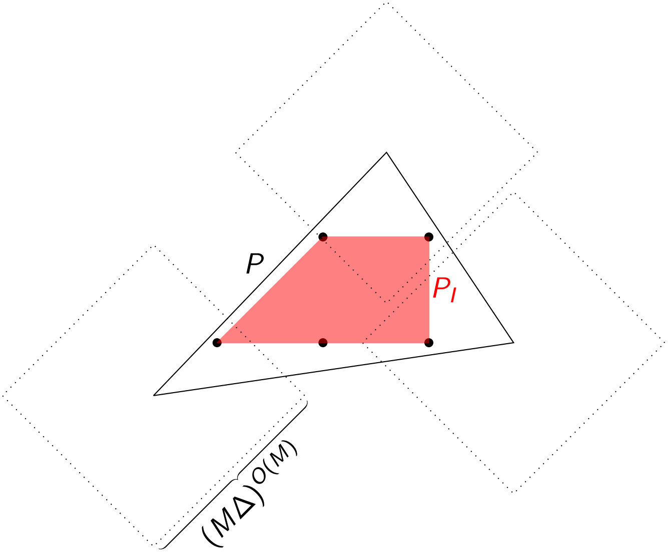
$$\begin{aligned} &\# \text{ vertices of } P_I \\ \leq\ &\# \text{ vertices of } P \cdot \# \text{ grid cells} \\ \leq\ &\# \text{ BFS} \cdot \# \text{ grid cells} \end{aligned}$$

Proof of the Vertex Bound



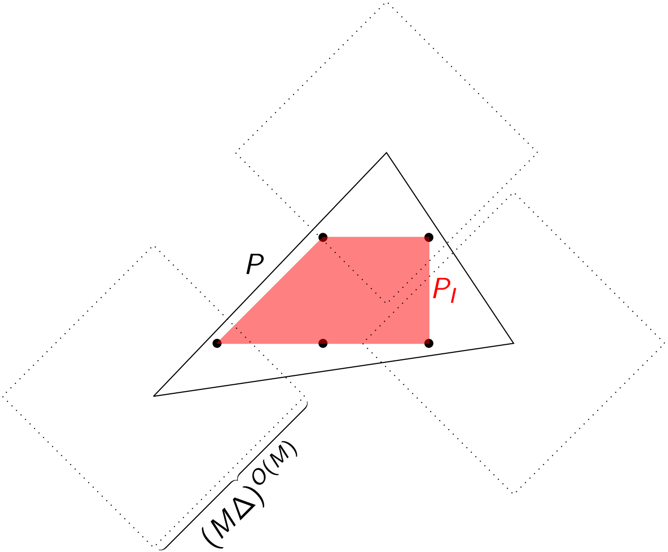
$$\begin{aligned} &\#\text{ vertices of } P_I \\ \leq\ &\#\text{ vertices of } P \cdot \#\text{ grid cells} \\ \leq\ &\#\text{ BFS} \cdot \#\text{ grid cells} \\ \leq\ &\binom{N}{M} \cdot (\log(\text{proximity bound}))^N \end{aligned}$$

Proof of the Vertex Bound



$$
\begin{aligned}
&\#\text{ vertices of } P_I \\
\leq\ &\#\text{ vertices of } P \cdot \#\text{ grid cells} \\
\leq\ &\#\text{ BFS} \cdot \#\text{ grid cells} \\
\leq\ &\binom{N}{M} \cdot (\log(\text{proximity bound}))^N \\
\leq\ &\binom{N}{M} \cdot (\log((M\Delta)^{O(M)}))^N
\end{aligned}
$$

Proof of the Vertex Bound



$$
\begin{aligned}
&\#\text{ vertices of } P_I \\
\le\ &\#\text{ vertices of } P \cdot \#\text{ grid cells} \\
\le\ &\#\text{ BFS} \cdot \#\text{ grid cells} \\
\le\ &\binom{N}{M} \cdot (\log(\text{proximity bound}))^N \\
\le\ &\binom{N}{M} \cdot (\log((M\Delta)^{O(M)}))^N \\
\le\ &N^M O(M \log(M\Delta))^N
\end{aligned}
$$

Conclusion

Theorem

ConeAndPolytopeIntersection *can be solved in time*

$$(\log(\Delta))^{2^{O(K)}}(\langle P\rangle\langle Q\rangle\log(m))^{O(1)} = 2^{2^{O(K)}}\Delta^{o(1)}(\langle P\rangle\langle Q\rangle\log(m))^{O(1)},$$

where $K := \max\{N, M^{(P)}, M^{(Q)}\}$ *and* $\Delta = \max\{\|A^{(P)}\|_\infty, \|A^{(Q)}\|_\infty\}$.

Conclusion

Theorem

ConeAndPolytopeIntersection *can be solved in time*

$$(\log(\Delta))^{2^{O(K)}}(\langle P\rangle\langle Q\rangle\log(m))^{O(1)} = 2^{2^{O(K)}}\Delta^{o(1)}(\langle P\rangle\langle Q\rangle\log(m))^{O(1)},$$

where $K := \max\{N, M^{(P)}, M^{(Q)}\}$ *and* $\Delta = \max\{\|A^{(P)}\|_\infty, \|A^{(Q)}\|_\infty\}$.

Theorem

$P||C_{\max}$ *can be solved in time* $(\log(p_{\max}))^{2^{O(d)}}\langle I\rangle^{O(1)} = 2^{2^{O(d)}}p_{\max}^{o(1)}\langle I\rangle^{O(1)}$.

Conclusion

Theorem

ConeAndPolytopeIntersection *can be solved in time*

$$(\log(\Delta))^{2^{O(K)}} (\langle P\rangle\langle Q\rangle \log(m))^{O(1)} = 2^{2^{O(K)}} \Delta^{o(1)} (\langle P\rangle\langle Q\rangle \log(m))^{O(1)},$$

where $K := \max\{N, M^{(P)}, M^{(Q)}\}$ *and* $\Delta = \max\{\|A^{(P)}\|_\infty, \|A^{(Q)}\|_\infty\}$.

Theorem

$P||C_{\max}$ *can be solved in time* $(\log(p_{\max}))^{2^{O(d)}} \langle I\rangle^{O(1)} = 2^{2^{O(d)}} p_{\max}^{o(1)} \langle I\rangle^{O(1)}$.

Try applying this to your favorite scheduling problem!

Running Times

Problem	Running Time
$P\|\|\{C_{\max}, C_{\min}, C_{\text{envy}}\}$	$(\log(p_{\max}))^{2^{O(d)}} \langle I \rangle^{O(1)}$
$P\|\text{class}\|\{C_{\max}, C_{\min}, C_{\text{envy}}\}$	$(\log(\max\{\|\|n\|\|_1, p_{\max}\}))^{2^{O(d)}} \langle I \rangle^{O(1)}$
$P\|\text{cap}\|\{C_{\max}, C_{\min}, C_{\text{envy}}\}$	$(\log(p_{\max}))^{2^{O(d)}} \langle I \rangle^{O(1)}$
$P\|\text{vec}\|\{C_{\max}, C_{\min}, C_{\text{envy}}\}$	$(\log(p_{\max}))^{2^{O(d+M)}} \langle I \rangle^{O(1)}$
$P\|s_j\|\{C_{\max}, C_{\min}, C_{\text{envy}}\}$	$(\log(\max\{\|\|n\|\|_1, p_{\max}, s_{\max}\}))^{2^{O(d)}} \langle I \rangle^{O(1)}$
$P\|\|\sum w_j U_j$	$(\log(\max\{p_{\max}, w_{\max}\}))^{2^{O(d)}} \langle I \rangle^{O(1)}$
$P\|\text{vec}\|\sum w_j U_j$	$(\log(\max\{p_{\max}, w_{\max}\}))^{2^{O(d+d_{\#}M)}} \langle I \rangle^{O(1)}$
MSWBP	$(\log(\max\{p_{\max}, w_{\max} \min\{B, \|\|n\|\|_1\}\}))^{2^{O(d+m)}} \langle I \rangle^{O(1)}$
Uniform n-fold ILPs	$(\log(\max\{\|\|C\|\|_\infty, \|\|B\|\|_\infty, \|\|c\|\|_\infty\}))^{2^{O(r+s+t)}} \langle I \rangle^{O(1)}$