On Integer Programming and Convolution

Klaus Jansen   Lars Rohwedder

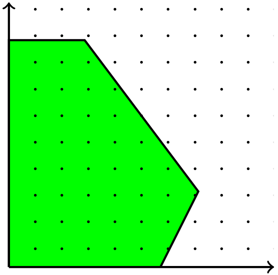
Department of Computer Science
University of Kiel

Berkeley 2020

Christian-Albrechts-Universität zu Kiel

Standard Form

$$\max\ c^T x$$
$$Ax = b$$
$$x \in \mathbb{Z}^n_{\geq 0}$$



where $A \in \mathbb{Z}^{m \times n}$, $b \in \mathbb{Z}^m, c \in \mathbb{Z}^n$.

Considered case

m (#constraints) is a fixed constant, entries of A are small ($\leq \Delta$).

Applications

Knapsack and scheduling problems, configuration IPs,...

Papadimitrou 1981

IP can be solved in time $(m(\Delta + \|b\|_\infty))^{O(m^2)}$.

Eisenbrand & Weismantel 2018

IP can be solved in time $n \cdot O(m\Delta)^{2m} \cdot \|b\|_\infty^2$.

Pseudo-polynomial Algorithms

Papadimitrou 1981

IP can be solved in time $(m(\Delta + \|b\|_\infty))^{O(m^2)}$.

Eisenbrand & Weismantel 2018

IP can be solved in time $n \cdot O(m\Delta)^{2m} \cdot \|b\|_\infty^2$.

This talk

IP can be solved in time $O(m\Delta)^{2m} \cdot \log(\|b\|_\infty) + O(nm)$.

Papadimitrou 1981

IP can be solved in time $(m(\Delta + \|b\|_\infty))^{O(m^2)}$.

Eisenbrand & Weismantel 2018

IP can be solved in time $n \cdot O(m\Delta)^{2m} \cdot \|b\|_\infty^2$.

This talk

IP can be solved in time $O(m\Delta)^{2m} \cdot \log(\|b\|_\infty) + O(nm)$.
Moreover, for every m and $\delta > 0$ improving the exponent to $2m - \delta$ is equivalent to finding a truly subquadratic algorithm for $(\min, +)$-convolution.

Feasibility problem

Our algorithm: $O(m\Delta)^m \cdot \log(\Delta) \cdot \log(\Delta + \|b\|_\infty) + O(nm)$.
Improving exponent to $m - \delta$ would contradict the Strong Exponential Time Hypothesis (SETH).

Previous best result (Eisenbrand, Weismantel 2018):
$n \cdot O(m\Delta)^m \cdot \|b\|_\infty$.

Knapsack problems with small weights

	Running time	Previous
Unbounded Knapsack	$O(\Delta^2)$	$O(nC), O(n\Delta^2)$
Unbounded Subset-Sum	$O(\Delta \log^2(\Delta))$	$O(C \log(C))$

(Δ = maximum weight; C = capacity)

Knapsack problems with small weights

	Running time	Previous
Unbounded Knapsack	$O(\Delta^2)$	$O(nC), O(n\Delta^2)$
Unbounded Subset-Sum	$O(\Delta \log^2(\Delta))$	$O(C \log(C))$

(Δ = maximum weight; C = capacity)

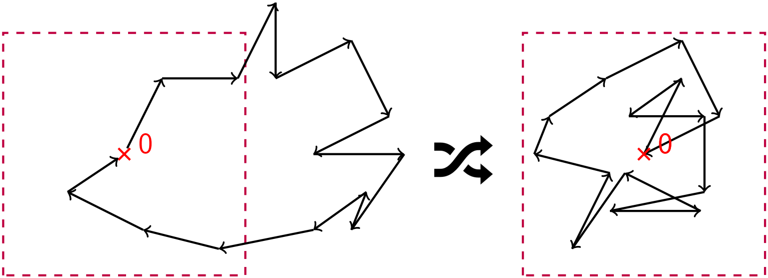
Scheduling on identical machines $P||C_{max}$

Previous EPTAS	$2^{O(1/\epsilon \log^4(1/\epsilon))} + O(N \log N)$
New EPTAS	$2^{O(1/\epsilon \log^2(1/\epsilon))} + O(N)$

(N = number of jobs, M = number of machines with $M \leq N$)

Let $\|\cdot\|$ be a norm in $\mathbb{R}^m$ and let $v^{(1)}, \dots, v^{(t)} \in \mathbb{R}^m$ such that $\|v^{(i)}\| \leq 1$ for all i and $v^{(1)} + \dots + v^{(t)} = 0$. Then there exists a permutation $\pi \in S_t$ such that for all $j \in \{1, \dots, t\}$

$$\|\sum_{i=1}^{j} v^{(\pi(i))}\| \leq m.$$



Steinitz for IP

Consider an optimal solution x^* of (IP) and the sequence of column vectors

$$\underbrace{A_1, \ldots, A_1}_{x_1^* \text{ times}}, \underbrace{A_2, \ldots, A_2}_{x_2^* \text{ times}}, \ldots$$

$$\begin{aligned} \max\ & c^T x \\ Ax &= b \qquad \text{(IP)} \\ x &\in \mathbb{Z}^n_{\geq 0} \end{aligned}$$

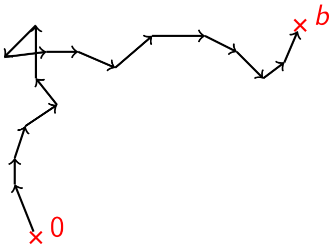
Recall that $\|A_i\|_\infty \leq \Delta$.

Consider an optimal solution x^* of (IP) and the sequence of column vectors

$$\underbrace{A_1, \dots, A_1}_{x_1^* \text{ times}}, \underbrace{A_2, \dots, A_2}_{x_2^* \text{ times}}, \dots$$

$$\begin{aligned} \max\ & c^T x \\ Ax &= b \qquad \text{(IP)} \\ x &\in \mathbb{Z}^n_{\geq 0} \end{aligned}$$

Recall that $\|A_i\|_\infty \leq \Delta$.

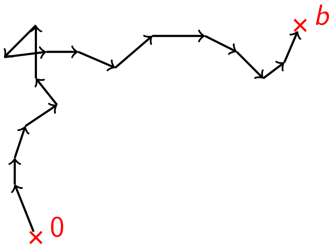


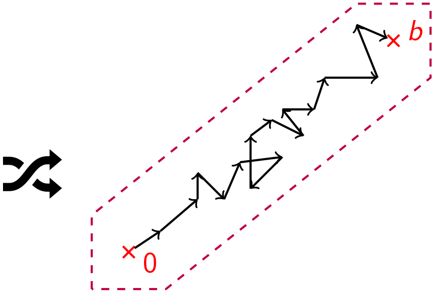
Consider an optimal solution x^* of (IP) and the sequence of column vectors

$$\underbrace{A_1, \ldots, A_1}_{x_1^* \text{ times}}, \underbrace{A_2, \ldots, A_2}_{x_2^* \text{ times}}, \ldots$$

$$\max c^T x$$
$$Ax = b \qquad \text{(IP)}$$
$$x \in \mathbb{Z}^n_{\geq 0}$$

Recall that $\|A_i\|_\infty \leq \Delta$.





More formally,

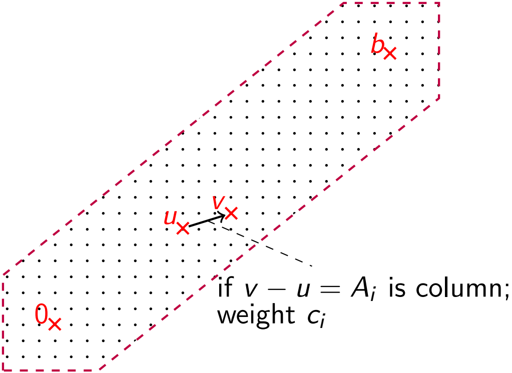
Corollary

Let $v^{(1)}, \ldots, v^{(t)}$ denote columns of A with $\sum_{i=1}^{t} v^{(i)} = b$. Then there exists a permutation $\pi \in S_t$ such that for all $j \in \{1, \ldots, t\}$

$$\left\| \sum_{i=1}^{j} v^{(\pi(i))} - j \cdot b/t \right\|_\infty \leq 2m\Delta.$$

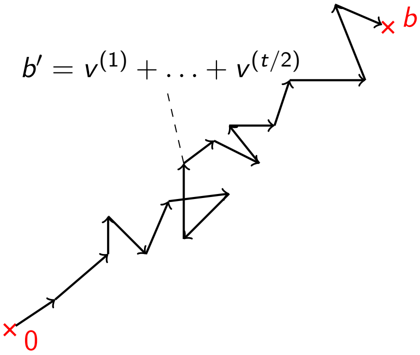
This follows easily from the Steinitz Lemma: Insert vectors $\frac{v^{(i)} - b/t}{2\Delta}$, $i \in \{1, \ldots, t\}$, in the Steinitz Lemma. Note that $\|\frac{v^{(i)} - b/t}{2\Delta}\|_\infty \leq 1$.

Eisenbrand & Weismantel

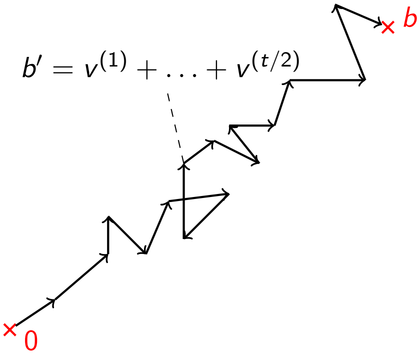


- Every $0 - b$ path gives a feasible solution
- Longest path is optimal solution
- $O(m\Delta)^m \cdot \|b\|_\infty$ vertices
- $n \cdot O(m\Delta)^m \cdot \|b\|_\infty$ edges
- Running time: $n \cdot O(m\Delta)^{2m} \cdot \|b\|_\infty^2$

Observation: There is an optimal solution of bounded norm, i.e., $\|x\|_1 \leq O(m\Delta)^m \cdot \|b\|_\infty$.



Let $v^{(1)} + \ldots + v^{(t)} = b$ be columns corresponding to an optimal solution of (IP).



Let $v^{(1)} + \ldots + v^{(t)} = b$ be columns corresponding to an optimal solution of (IP).

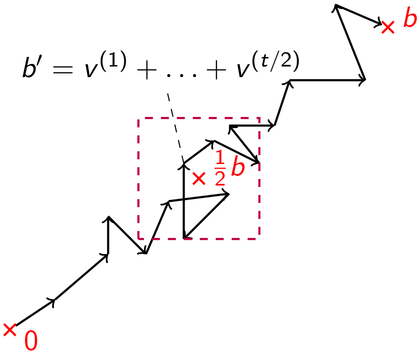
Equivalent:

$v^{(1)} + \ldots + v^{(t/2)}$ is optimal for

$$\{\max c^T x, Ax = b', x \in \mathbb{Z}^n_{\geq 0}\}$$

and $v^{(t/2+1)} + \ldots + v^{(t)}$ is for

$$\{\max c^T x, Ax = b - b', x \in \mathbb{Z}^n_{\geq 0}\}.$$



Let $v^{(1)} + \ldots + v^{(t)} = b$ be columns corresponding to an optimal solution of (IP).

Equivalent:
$v^{(1)} + \ldots + v^{(t/2)}$ is optimal for

$$\{\max c^T x, Ax = b', x \in \mathbb{Z}^n_{\geq 0}\}$$

and $v^{(t/2+1)} + \ldots + v^{(t)}$ is for

$$\{\max c^T x, Ax = b - b', x \in \mathbb{Z}^n_{\geq 0}\}.$$

If ordered via Steinitz Lemma, b' and $b - b'$ are not far from $\frac{1}{2}b$.
Also, t cut in half in subproblems.

Assume w.l.o.g. there is an optimal solution x with $\|x\|_1 = 2^K$, where $K \in \log(O(m\Delta)^m \cdot \|b\|_\infty) = O(m \log(m\Delta) + \log(\|b\|_\infty))$

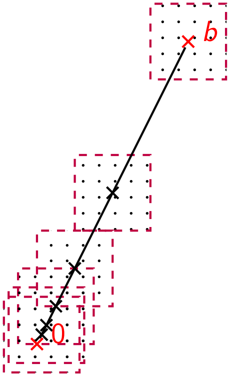
Assume w.l.o.g. there is an optimal solution x with $\|x\|_1 = 2^K$, where $K \in \log(O(m\Delta)^m \cdot \|b\|_\infty) = O(m \log(m\Delta) + \log(\|b\|_\infty))$

Solve for every $i = K, K-1, \ldots, 0$ and every b' with

$$\left\| b' - \frac{1}{2^i} b \right\|_\infty \leq 4m\Delta$$

the problem

$$\max c^T x$$
$$Ax = b'$$
$$\|x\|_1 = 2^{K-i}$$
$$x \in \mathbb{Z}^n_{\geq} 0.$$



Solution for original problem at $i = 0$ and $b' = b$.

Iterative calculation

Let $i < K$ and b' with $\|b' - 1/2^i \cdot b\|_\infty \leq 4m\Delta$.

Let $v^{(1)}, \dots, v^{(2^{K-i})}$ correspond to a solution of

$$\max\{c^T x, Ax = b', \|x\|_1 = 2^{K-i}, x \in \mathbb{Z}^n_{\geq 0}\},$$

ordered via Steinitz Lemma.

Let $i < K$ and b' with $\|b' - 1/2^i \cdot b\|_\infty \leq 4m\Delta$.

Let $v^{(1)}, \ldots, v^{(2^{K-i})}$ correspond to a solution of

$$\max\{c^T x, Ax = b', \|x\|_1 = 2^{K-i}, x \in \mathbb{Z}^n_{\geq 0}\},$$

ordered via Steinitz Lemma. Set $b'' := v^{(1)} + \ldots + v^{(2^{K-i-1})}$.

$$\left\| b'' - \frac{1}{2^{i+1}} b \right\|_\infty \leq \underbrace{\left\| b'' - \frac{1}{2} b' \right\|_\infty}_{\leq 2m\Delta} + \underbrace{\left\| \frac{1}{2} b' - \frac{1}{2^{i+1}} b \right\|_\infty}_{\leq 1/2 \cdot 4m\Delta} \leq 4m\Delta$$

Similarly,

$$\left\| (b' - b'') - \frac{1}{2^{i+1}} b \right\|_\infty \leq 4m\Delta.$$

Iterative calculation

Let $i < K$ and b' with $\|b' - 1/2^i \cdot b\|_\infty \leq 4m\Delta$.

Let $v^{(1)}, \dots, v^{(2^{K-i})}$ correspond to a solution of

$$\max\{c^T x, Ax = b', \|x\|_1 = 2^{K-i}, x \in \mathbb{Z}^n_{\geq 0}\},$$

ordered via Steinitz Lemma. Set $b'' := v^{(1)} + \dots + v^{(2^{K-i-1})}$.

$$\left\| b'' - \frac{1}{2^{i+1}} b \right\|_\infty \leq \underbrace{\left\| b'' - \frac{1}{2} b' \right\|_\infty}_{\leq 2m\Delta} + \underbrace{\left\| \frac{1}{2} b' - \frac{1}{2^{i+1}} b \right\|_\infty}_{\leq 1/2 \cdot 4m\Delta} \leq 4m\Delta$$

Similarly,

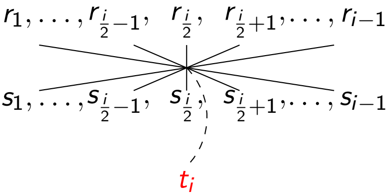
$$\left\| (b' - b'') - \frac{1}{2^{i+1}} b \right\|_\infty \leq 4m\Delta.$$

Guess b'' ($O(m\Delta)^m$ candidates), look up solutions for $(i+1, b'')$ and $(i+1, b' - b'')$, and take the best.

(MAX, +)-CONVOLUTION

Input: $r_1, \ldots, r_N \in \mathbb{R}$,
$s_1, \ldots, s_N \in \mathbb{R}$

Output: $t_1, \ldots, t_N \in \mathbb{R}$ with
$t_i = \max_j [r_j + s_{i-j}]$



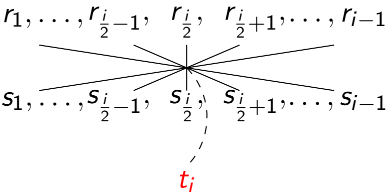
For $m = 1$, merging solutions directly corresponds to solving (MAX, +)-CONVOLUTION of size $N = O(\Delta)$.
For general m, we can cast the problem to an instance of (MAX, +)-CONVOLUTION of size $N = O(m\Delta)^m$.

Merging solutions

(max, +)-convolution

Input: $r_1, \ldots, r_N \in \mathbb{R}$,
$s_1, \ldots, s_N \in \mathbb{R}$

Output: $t_1, \ldots, t_N \in \mathbb{R}$ with
$t_i = \max_j [r_j + s_{i-j}]$



For $m = 1$, merging solutions directly corresponds to solving (max, +)-convolution of size $N = O(\Delta)$.
For general m, we can cast the problem to an instance of (max, +)-convolution of size $N = O(m\Delta)^m$.

$T(N)$ time algorithm for (min, +)-convolution $\Rightarrow$
$T(O(m\Delta)^m) \cdot O(m \log(m\Delta) + \log(\|b\|_\infty)) + O(nm)$ for IP.

With $T(n) = O(n^2/\log(n))$: $O(m\Delta)^{2m} \cdot \log(\|b\|_\infty) + O(nm)$.


ONE WAY
ONE WAY

Theorem

If there is an $m \in \mathbb{N}$ and $\delta > 0$ for which an Algorithm exists that solves IPs with m constraints in time $O(m(\Delta + \|b\|_\infty))^{2m-\delta}$, then $(\textsc{min}\ ,+)$-CONVOLUTION can be solved in time $O(N^{2-\delta'})$.

Theorem (Cygan et al. 2017)

1. There exists a $\delta > 0$ and an $O(N^{2-\delta})$ time algorithm for $(\textsc{min}\ ,+)$-CONVOLUTION

if and only if

2. There exists a $\delta > 0$ and an $O(C^{2-\delta})$ time algorithm for UNBOUNDED KNAPSACK.

Unbounded Knapsack

$$\max \sum_{i=1}^{N} p_i x_i$$

$$\sum_{i=1}^{N} w_i x_i \leq C$$

$$x_1, \dots, x_N \in \mathbb{Z}_{\geq 0}$$

$$\max \sum_{i=1}^{N} p_i x_i + 0 \cdot y$$

$$\sum_{i=1}^{N} w_i x_i + 1 \cdot y = C$$

$$x_1, \ldots, x_N, y \in \mathbb{Z}_{\geq 0}$$

$m = 1$

Assume there is a $O(m(\underbrace{\Delta + \|b\|_\infty}_{=O(C)}))^{2m-\delta} = O(C^{2-\delta})$.

$$\max \sum_{i=1}^{N} p_i x_i + 0 \cdot y$$

$$\sum_{i=1}^{N} w_i x_i + 1 \cdot y = C$$

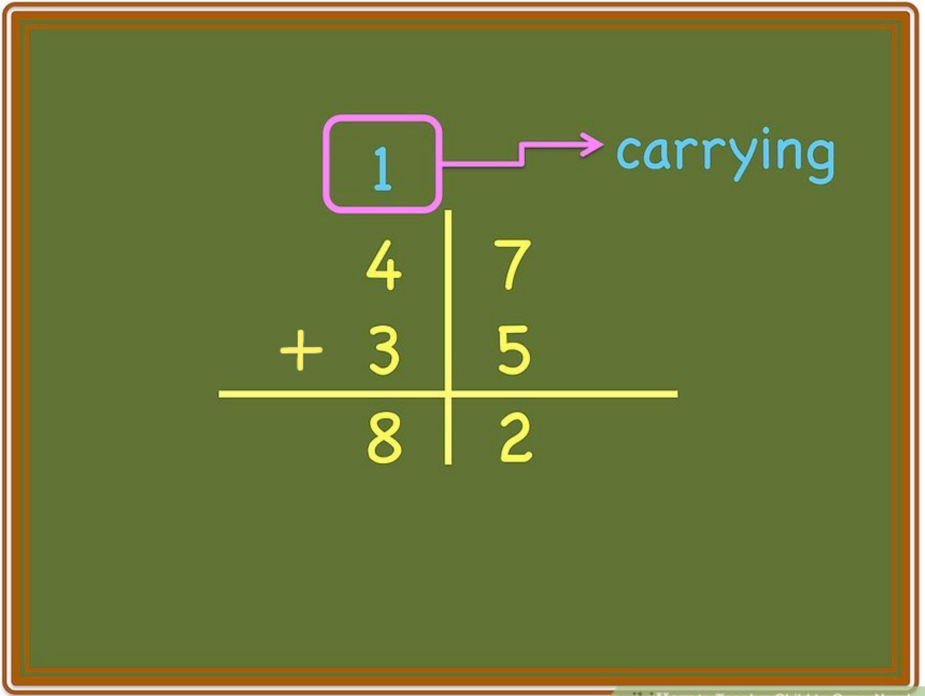
$$x_1, \ldots, x_N, y \in \mathbb{Z}_{\geq 0}$$

$m = 1$

Assume there is a $O(m(\underbrace{\Delta + \|b\|_\infty}_{=O(C)}))^{2m-\delta} = O(C^{2-\delta})$.

$m > 1$

Reduce Δ by introducing additional equalities.


1
carrying
4 7
+ 3 5
8 2

Set $\Delta = \lceil C^{1/m} \rceil$. Write

$$C = C^{(0)} + \Delta \cdot C^{(1)} + \Delta^2 \cdot C^{(2)} + \cdots + \Delta^{m-1} \cdot C^{(m-1)},$$
$$w_i = w_i^{(0)} + \Delta \cdot w_i^{(1)} + \Delta^2 \cdot w_i^{(2)} + \cdots + \Delta^{m-1} \cdot w_i^{(m-1)},$$

with each number smaller than Δ.

$$\sum_{i=1}^{N} w_i x_i = C \quad \Leftrightarrow \quad \begin{aligned} \sum_{i=1}^{N} w_i^{(0)} x_i - \Delta \cdot y_0 &= C^{(0)} \\ \sum_{i=1}^{N} w_i^{(1)} x_i + y_0 - \Delta \cdot y_1 &= C^{(1)} \\ \sum_{i=1}^{N} w_i^{(2)} x_i + y_1 - \Delta \cdot y_2 &= C^{(2)} \\ &\vdots \end{aligned}$$

Putting together the pieces

Christian-Albrechts-Universität zu Kiel

- Suppose for some fixed m there exists an algorithm that solves IPs with m constraints in $O(m(\Delta + \|b\|_\infty))^{2m-\delta}$.
- Construction shows Unbounded Knapsack can be solved via IP with m constraints and biggest entry $\Delta = \lceil C^{1/m} \rceil$.
- Running time:

$$
\begin{aligned}
O(m(\Delta + \|b\|_\infty))^{2m-\delta} &= O(m\lceil C^{1/m} \rceil)^{2m-\delta} \\
&= O(m)^{2m-\delta} \cdot (C^{1/m})^{2m-\delta} = f(m) \cdot C^{2-\frac{\delta}{m}}.
\end{aligned}
$$

$\Rightarrow$ Unbounded Knapsack can be solved in subquadratic time.

$\Rightarrow$ (min, +)-convolution can be solved in subquadratic time.

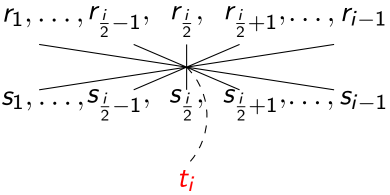
Feasibility of IP

Christian-Albrechts-Universität zu Kiel

Boolean-convolution

Input: $r_1, \ldots, r_N \in \{0,1\}$,
$s_1, \ldots, s_N \in \{0,1\}$

Output: $t_1, \ldots, t_N \in \{0,1\}$
with
$t_i = \bigvee_j [r_j \wedge s_{i-j}]$



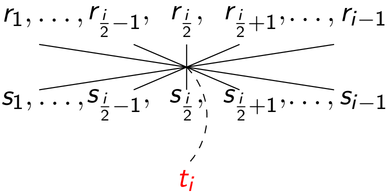
Boolean Convolution can be computed in time $T(N) = O(N \log N)$ time.

Feasibility of IP

Boolean-convolution

Input: $r_1, \dots, r_N \in \{0,1\}$,
$s_1, \dots, s_N \in \{0,1\}$

Output: $t_1, \dots, t_N \in \{0,1\}$
with
$t_i = \bigvee_j [r_j \wedge s_{i-j}]$



Boolean Convolution can be computed in time
$T(N) = O(N \log N)$ time.

$\Rightarrow$ Feasibility of IP in time

$$T(O(m\Delta)^m) \cdot (m \log(m\Delta) + \log(\|b\|_\infty)) + O(nm)$$
$$= O(m\Delta)^m \cdot \log(\Delta) \cdot \log(\Delta + \|b\|_\infty) + O(nm).$$

k-SUM

Input: $T \in \mathbb{N}_0$ and $Z_1, \dots, Z_k \subset \mathbb{N}_0$ where $|Z_1| + |Z_2| + \ldots + |Z_k| = n \in \mathbb{N}$.

Output: $z_1 \in Z_1, z_2 \in Z_2 \dots, z_k \in Z_k$ such that $z_1 + z_2 + \ldots + z_k = T$.

Theorem (Abboud et al. 2017)

If SETH holds, then for every $\delta > 0$ there exists a $\gamma > 0$ such that k-SUM cannot be solved in time $O(T^{1-\delta} n^{\gamma k})$.

Theorem

If the SETH holds, for every fixed m there does not exist an algorithm that solves feasibility of IPs with m constraints in time $n^{f(m)} \cdot (\Delta + \|b\|_\infty)^{m-\delta}$.

Theorem (Eisenbrand, Weismantel 2018)

Let $\max\{c^T x : Ax = b, x \in \mathbb{Z}^n_{\geq 0}\}$ be feasible and bounded and x^* be an optimal vertex solution of the LP relaxation. Then there is an optimal solution z^* of IP with $\|z^* - x^*\|_\infty \leq m(2m\Delta + 1)^m$.

Theorem (Eisenbrand, Weismantel 2018)

Let $\max\{c^T x : Ax = b, x \in \mathbb{Z}^n_{\geq 0}\}$ be feasible and bounded and x^* be an optimal vertex solution of the LP relaxation. Then there is an optimal solution z^* of IP with $\|z^* - x^*\|_\infty \leq m(2m\Delta + 1)^m$.

Reduction of right-hand side

This implies $z^*_i \geq \ell_i := \max\{0, \lceil x^*_i \rceil - m(2m\Delta + 1)^m\}$. Therefore, we get an equivalent IP $\max\{c^T y : Ay = b', y \in \mathbb{Z}^n_{\geq 0}\}$ with $b'_j = \max\{b_j - a_j^T \ell, 0\}$.

Consequence: $\|b'\|_\infty \leq O(m\Delta)^{m+1}$

Theorem (Eisenbrand, Weismantel 2018)

Optimality and Feasibility of the IP can be done in time $n \cdot O(m\Delta)^{4m+2} + LP$ and $n \cdot O(m\Delta)^{2m+1} + LP$, respectively.

Theorem (Eisenbrand, Weismantel 2018)

Optimality and Feasibility of the IP can be done in time $n \cdot O(m\Delta)^{4m+2} + LP$ and $n \cdot O(m\Delta)^{2m+1} + LP$, respectively.

Using our new result for the IP we obtain:

Theorem

Optimality and Feasibility of the IP can be done in time $O(m\Delta)^{2m} + O(nm) + LP$ and $O(m\Delta)^{m} \cdot \log^2(\Delta) + O(nm) + LP$, respectively.

Application Knapsack

Christian-Albrechts-Universität zu Kiel

UNBOUNDED KNAPSACK
with equality constraint is an IP with $m = 1$ constraint:

$$\max\{\sum_{i=1}^{n} p_i x_i : \sum_{i=1}^{n} w_i x_i = C, x \in \mathbb{Z}^n_{\geq 0}\}.$$

An optimal fractional LP solution can be computed in $O(\Delta)$ and $O(1)$ time for UNBOUNDED KNAPSACK and UNBOUNDED SUBSET-SUM.

UNBOUNDED KNAPSACK
with equality constraint is an IP with $m = 1$ constraint:

$$\max\{\sum_{i=1}^{n} p_i x_i : \sum_{i=1}^{n} w_i x_i = C, x \in \mathbb{Z}^n_{\geq 0}\}.$$

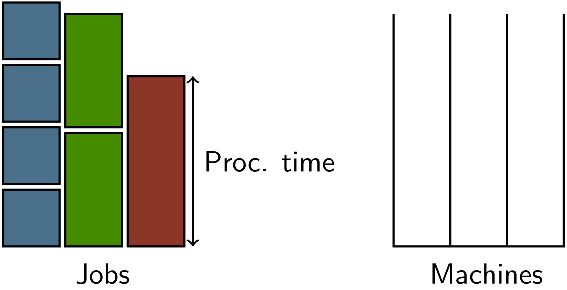
An optimal fractional LP solution can be computed in $O(\Delta)$ and $O(1)$ time for UNBOUNDED KNAPSACK and UNBOUNDED SUBSET-SUM.
Using the proximity results we get:

	Running time	**Previous**
UNBOUNDED KNAPSACK	$O(\Delta^2)$	$O(nC), O(n\Delta^2)$
UNBOUNDED SUBSET-SUM	$O(\Delta \log^2(\Delta))$	$O(C \log(C))$

Scheduling on identical machines

Input: N jobs with processing times $p_j \in \mathbb{N}$ and $M \leq N$ machines.

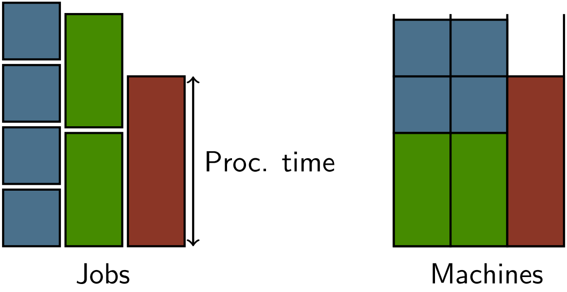
Output: A schedule $\alpha : \{1, \ldots, N\} \rightarrow \{1, \ldots, M\}$ which minimizes the maximum load $L_i = \sum_{j:\alpha(j)=i} p_j$ over all machines $i = 1, \ldots, M$.



Scheduling on identical machines

Input: N jobs with processing times $p_j \in \mathbb{N}$ and $M \leq N$ machines.

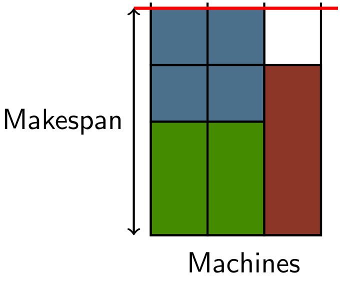
Output: A schedule $\alpha : \{1, \ldots, N\} \to \{1, \ldots, M\}$ which minimizes the maximum load $L_i = \sum_{j:\alpha(j)=i} p_j$ over all machines $i = 1, \ldots, M$.



Scheduling on identical machines

Input: N jobs with processing times $p_j \in \mathbb{N}$ and $M \leq N$ machines.

Output: A schedule $\alpha : \{1, \dots, N\} \to \{1, \dots, M\}$ which minimizes the maximum load $L_i = \sum_{j:\alpha(j)=i} p_j$ over all machines $i = 1, \dots, M$.



Configuration IP

$$\begin{array}{ll} \sum_{C \in \mathcal{C}} x_C = M & \\ \sum_{C \in \mathcal{C}} C_i x_C = N_i & \forall i \in \{1, \ldots, m-1\} \\ x_C \in \mathbb{Z}_{\geq 0} & \forall C \in \mathcal{C} \end{array}$$

has $m = O(1/\epsilon \log(1/\epsilon))$ constraints and $n = |\mathcal{C}| = 2^{O(1/\epsilon)}$ many variables. The value $\Delta \leq 1/\epsilon$ and $\|b\|_\infty \leq N$.

Previous best result: $2^{O(1/\epsilon \log^4(1/\epsilon))} + O(N \log N)$.

Configuration IP

$$\begin{array}{ll} \sum_{C \in \mathcal{C}} x_C = M & \\ \sum_{C \in \mathcal{C}} C_i x_C = N_i & \forall i \in \{1, \ldots, m-1\} \\ x_C \in \mathbb{Z}_{\geq 0} & \forall C \in \mathcal{C} \end{array}$$

has $m = O(1/\epsilon \log(1/\epsilon))$ constraints and $n = |\mathcal{C}| = 2^{O(1/\epsilon)}$ many variables. The value $\Delta \leq 1/\epsilon$ and $\|b\|_\infty \leq N$.

Previous best result: $2^{O(1/\epsilon \log^4(1/\epsilon))} + O(N \log N)$.

New result: Including the rounding in time $O(N + 1/\epsilon \log(1/\epsilon))$, the total running time for the ILP is:

$$O(m\Delta)^m \cdot \log(\Delta) \cdot \log(\Delta + \|b\|_\infty) + O(nm) + O(N + 1/\epsilon \log(1/\epsilon))$$
$$\leq 2^{O(1/\epsilon \log^2(1/\epsilon))} \log(N) + O(N) \leq 2^{O(1/\epsilon \log^2(1/\epsilon))} + O(N).$$

Conclusion

- Improved pseudo-polynomial algorithm for IP with fixed number of constraints
- Equivalence to $(\text{MIN}, +)$-CONVOLUTION w.r.t. improvements
- Lower bound for feasibility IP under SETH
- Use of proximity to reduce running time
- Application in knapsack and scheduling

Open Question

Can we solve the following IP in time $(m\Delta)^{O(m)} \cdot \log(\|b\|_\infty) + O(nm)$?

$$\begin{aligned} \max\ c^T x \\ Ax = b \\ x \leq u \\ x \in \mathbb{Z}^n_{\geq 0} \end{aligned}$$

Best algorithm known: $n \cdot m^{O(m)} \cdot \Delta^{O(m^2)}$.