A Faster Exponential Time Algorithm for Bin Packing with Constant Number of Bins

To appear at SODA 2021

Jesper Nederlof

Jakub Pawlewicz

Céline Swennenhuis

Karol Węgrzycki

UU

Warsaw

TU/e

Warsaw/Saarbrücken

Started with this research at “Parameterized Algorithms Retreat of the University of Warsaw” in February 2020

Department of Mathematics and Computer Science, Eindhoven University of Technology
Bin Packing

Given:

- $n$ items
- $w(j)$ weight of item $j$
- $w(X) = \sum_{j \in X} w(j)$
- $m$ bins with capacity $c$

Goal: distribute items over bins
Bin Packing

- Dynamic Programming
  - $O(c^m \cdot n)$

- Björklund, Husfeldt and Koivisto (SICOMP 2009)
  - $O(2^n \cdot n)$

Open Question
Can do in $O(1.99999^n)$ time?

Key idea:
DP in $O(2^{(1-\varepsilon_m)n})$ time
or
Björklund et al. in $O(2^{(1-\varepsilon_m)n})$ time

Our result:
Can do in $O(2^{(1-\varepsilon_m)n})$ time with $\varepsilon_m > 0$ that depends on $m$.

Before our work, only known for $m = 2,3$
(Lente et al.)
Dynamic Program

\[ T(j, c_1, c_2, \ldots, c_m) = \bigvee_{i \in [m]} T(j-1, \ldots, c_i - w(j), \ldots) \]

Items 1, ..., j
leftover capacities

\[ O(c^m \cdot n) \]

\[ \mathcal{W} := \{w(X): X \subseteq [n]\} \]
Runtime: \( O(|\mathcal{W}|^m \cdot n) \)

\( \mathcal{W} \leq 2^{\frac{(1-\gamma)n}{m}} \) , runtime \( O(2^{(1-\gamma)n}) \)

\( \{2, 2, 2, \ldots, 2\} \rightarrow \mathcal{W} = \{2, 4, \ldots, 2n\} \)

\( \{1, 2, \ldots, 2^{n-1}\} \rightarrow \mathcal{W} = \{1, 2, 3, \ldots, 2^n - 1\} \)
Algorithm by Björklund et al.

**Theorem:** For $X \subseteq [n]$, we can compute in time $O(|\downarrow X| \cdot n)$ whether $X$ can be distributed over $m$ bins.
Algorithm by Björklund et al.

**Theorem:** For $X \subseteq [n]$, we can compute in time $O(|\downarrow X| \cdot n)$ whether $X$ can be distributed over $m$ bins.

If $X = [n]$, then in time $O(2^n \cdot n)!$
Algorithm by Björklund et al.

**Theorem:** For $X \subseteq [n]$, we can compute in time $O(|\downarrow X| \cdot n)$ whether $X$ can be distributed over $m$ bins.

If $X = [n]$, then in time $O(2^n \cdot n)$!

**Theorem:** For $\mathcal{B} \subseteq 2^n$, we can compute in time $O(|\downarrow \mathcal{B}| \cdot n)$, whether $X$ can be distributed over $m$ bins for all $X \in \mathcal{B}$.
Algorithm by Björklund et al.

$$\mathcal{B} := \left\{ X : |X| = \frac{n}{2}, w(X) = k \cdot c \right\}$$

Compute in time $O(\downarrow \mathcal{B} \cdot n)$ which $X$ can be put in $k$ bins

Compute in time $O(\uparrow \mathcal{B} \cdot n)$ which $[n] \setminus X$ can be put in $m - k$ bins
Algorithm by Björklund et al.

\[ B = \left| \left\{ X : |X| = \frac{n}{2}, \omega(X) = n \right\} \right| = \binom{n}{n/2} \]

\[ \{2,2,2, \ldots, 2\} \]

\[ B \leq 1 \]

\[ B := \left\{ X : |X| = \frac{n}{2}, w(X) = k \cdot c \right\} \]

\[ \{1,2,4, \ldots, 2^{n-1}\} \]
Algorithm by Björklund et al.

If $|\mathcal{B}| \leq 2^{(1-\varepsilon)n}$ for $\varepsilon > 0$
Then $|\downarrow \mathcal{B}| + |\uparrow \mathcal{B}| \leq 2^{(1-\delta\varepsilon)n}$ for $\delta\varepsilon > 0$

So if $|\mathcal{B}| \leq 2^{(1-\varepsilon)n}$, we can do faster than $2^n$!

$\mathcal{B} := \left\{ X : |X| = \frac{n}{2}, w(X) = k \cdot c \right\}$
Additive Combinatorics

**Theorem:** For all $\delta > 0$ there exists $\epsilon > 0$ s.t.

$$\text{if } |\mathcal{W}| \geq 2^{\delta n}, \text{ then } |\mathcal{B}| \leq 2^{(1-\epsilon)n}.$$ 

Take $\delta = \frac{(1-\gamma)}{m}$, then either

$$|\mathcal{W}| \leq 2^{\delta n} \quad \text{or} \quad |\mathcal{B}| \leq 2^{(1-\epsilon)n} \quad (\text{or both})$$
Conclusion

Main result:
Bin Packing in $O\left(2^{(1-\varepsilon_m)n}\right)$ time with $\varepsilon_m > 0$ that depends on $m$.

Key idea:
Tradeoff between $|\mathcal{W}|$ and $|\mathcal{B}|$. 

Thanks for your attention!