Makespan Scheduling of Unit Jobs with Precedence Constraints in $O(1.995^n)$ time

Jesper Nederlof
Céline Swennenhuis
Karol Węgrzycki

UU  TU/e  Saarbrücken

Department of Mathematics and Computer Science, Eindhoven University of Technology
Given:

- \( n \) jobs of length 1
- \( m \) (identical, parallel) machines
- A precedence graph \( G \)
- \( T \in \mathbb{N} \)

Q: Is there a schedule of makespan \( T \)?

**Problem:**

\[ P|prec, p_j = 1|C_{\text{max}} \]
\( P|\text{prec, } p_j = 1|C_{\text{max}} \)

- NP-complete
- For \( m = 2 \): polynomial time algorithms [FKN 1969, CG 1972, G 1982, GT 1985]
- For \( m \geq 3 \): complexity unknown! [Garey and Johnson 1961]

- XNLP-hard when parameterized by \( m \) \((+\text{width}(G))\) (i.e. W[t]-hard for all t) [BGNS 2021]
**Definitions**

**Precendence Constraints Graph $G$**

$G$ ⇒ partial order:

- $i < j$ if $(i, j) \in G$

**Definition:** Let $A$ be a set of jobs.

- $\text{pred}[A] = \{x \mid \exists a \in A \text{ s.t. } x \preceq a\}$
- $\text{succ}[A] = \{x \mid \exists a \in A \text{ s.t. } x \succeq a\}$
- $\text{comp}(A) = \{x \mid \exists a \in A \text{ s.t. } x \preceq a \text{ or } x \succeq a\}$
- $\text{sinks}(A) = \max\{A\}$

Let $A = \{c, d\}$, then:

- $\text{pred}[A] = \{a, c, d\}$
- $\text{succ}[A] = \{c, d, e, f\}$
- $\text{comp}(A) = \{a, c, d, e, f\}$
- $\text{sinks}(\{a, c, d\}) = \{c, d\}$
**Definitions**

**Def:** An *antichain* is a set $A$ whose elements are pairwise incomparable.

Ex. of antichains in $G$

- $\checkmark \{b, c, d\}$
- $\checkmark \{b, c\}$
- $\checkmark \{d, f\}$

Corresponding $\text{pred}[A]$

- $\Rightarrow \{a, b, c, d\}$
- $\Rightarrow \{a, b, c\}$
- $\Rightarrow \{a, b, c, d, f\}$

No antichain:

- $\times \{b, f\}$
- $\times \{a, e\}$

Precendence Constraints Graph $G$
There is a one-to-one relation between:

- \( A = \) antichain of graph \( G \)
- \( \text{pred}[A] \)
- \( \text{sinks}(X) \)
- \( X = \) possible set of jobs to schedule

**Def:** An antichain is a set \( A \) whose elements are pairwise incomparable.
DP Algorithm:

\[ T(A, t) = \begin{cases} 
  \text{true,} & \text{if pred}[A] \text{ can be processed in } t \text{ time}, \\
  \text{false,} & \text{otherwise}. 
\end{cases} \]

To compute \( T(A, t) \):
- try all \( \binom{n}{m} \) possible jobs at time \( t \)
- Runs in \( O^*(\#AC \cdot \binom{n}{m}) \) time
- \( \#AC \leq 2^n \) \( \Rightarrow O^*(2^n \cdot \binom{n}{m}) \) time algorithm
- No \( O^*(2^n) \) time algorithm known!

Makespan Scheduling of Unit Jobs with Precedence Constraints in \( O(1.995n) \) time
Our result:

\[ P|\text{prec}, p_j = 1|C_{\text{max}} \] can be solved in \( O(1.995^n) \) time.

Proof: Combination of three algorithms
$P|\text{prec}, p_j = 1|C_{\max}$

- **DP Algorithm**
  - Dynamic Programming
  - $O^*\left(\#AC \cdot \binom{n}{m}\right)$ time

- **Fast Subset Convolution Algorithm**
  - Non-trivial application of Björklund, Husfeldt and Koivisto (SICOMP 2009)
  - $O^*(\#AC + 2^n - m)$ time

- **FPT Algorithm**
  - Parameterized by vertex cover of comparability graph
  - $O^*(169|C|)$ time

---

**Fast enough when**

$|C| \geq \frac{n}{7.5}$

⇒ largest antichain

$< \left(1 - \frac{1}{7.5}\right)n$

Dilworth’s Theorem

$\#AC \leq 1.94^n$

**Fast enough when**

$|C| < \frac{n}{7.5}$
What is vertex cover of comparability graph?

Let $G = (V, A)$ be input. Then $G^{\text{comp}} := (V, E)$ where $(v, w) \in E$ if $v$ and $w$ are comparable.

- $G^{\text{comp}}$ is undirected transitive closure of $G$.

Let $C := \text{smallest vertex cover of } G^{\text{comp}}$

**Claim:** $V \setminus C$ is an antichain.
Zero-Adjusted Schedule (D&W)

Assumption: $n = m \cdot T$

Makespan Scheduling of Unit Jobs with Precedence Constraints in $O(1.995n)$ time
Zero-Adjusted Schedule (D&W)

Assumption: $n = m \cdot T$

$= \text{sink}(G)$

Let $z \in [1, T]$ be the first moment with a sink.

**D&W:** W.m.a. Each job $x$ after $z$ is a sink or a successor of a job at time $z$.

$\Rightarrow n^{O(m \cdot h)}$ time algorithm, $h = \mid\text{longest chain}\mid$
Sink-Adjusted Schedule

Assumption: $n = m \cdot T$

$= \text{sink}(G)$

$z \in [1, T]$ is a sink moment if there are both sinks and non-sinks at time $z$. 

Makespan Scheduling of Unit Jobs with Precedence Constraints in $O(1.995n)$ time
Sink-Adjusted Schedule

Assumption: \( n = m \cdot T \)

\( = \text{sink}(G) \)

\( z \in [1, T] \) is a \textit{sink moment} if there are both sinks and non-sinks at time \( z \).
Sink-Adjusted Schedule

Assumption: \( n = m \cdot T \)

\[ = \text{sink}(G) \]

\[ z \in [1, T] \] is a sink moment if there are both sinks and non-sinks at time \( z \).
Sink-Adjusted Schedule

Assumption: $n = m \cdot T$

$V \setminus (\text{succ}[H_1] \cup \text{sinks})$

$z \in [1, T]$ is a sink moment if there are both sinks and non-sinks at time $z$. 

No sinks in
Sink-Adjusted Schedule

How to use this...?

Claim: if we know $\bigcup_i H_i$, we can derive $H_1, H_2, \ldots, H_l$
Middle-Adjusted Schedule

Assumption: $n = m \cdot T$

<table>
<thead>
<tr>
<th>$C^L$</th>
<th>$L$</th>
<th>$C^R$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\text{pred}[C^L]$</td>
<td></td>
<td>$\text{succ}[C^R]$</td>
</tr>
</tbody>
</table>

$|C^L| \approx \frac{|C|}{2}$

$|C^R| \approx \frac{|C|}{2}$
Middle-Adjusted Schedule

Assumption: $n = m \cdot T$

Undecided $U := V \setminus (\text{pred}[C^L] \cup \text{succ}[C^R])$

- $U$ is antichain
- $U$ are sinks in $L$
- $U$ are sources in $R$
Middle-Adjusted Schedule

Assumption: $n = m \cdot T$

Undecided $U \equiv V \setminus (\text{pred}[C^L] \cup \text{succ}[C^R])$

- $U$ is antichain
- $U$ are sinks in $L$
Middle-Adjusted Schedule

Assumption: $n = m \cdot T$

- Assume sink-adjusted
- Guess jobs in $H$ from $C^L$
- Reconstruct $H_1, H_2, ...$
- Find out which jobs from $U$ can be processed at which slots

Undecided $U \equiv V \setminus (\text{pred}[C^L] \cup \text{succ}[C^R])$

- $U$ is antichain
- $U$ are sinks in $L$

Makespan Scheduling of Unit Jobs with Precedence Constraints in $O(1.995n)$ time
Middle-Adjusted Schedule

Assumption: \( n = m \cdot T \)

Undecided \( U \) := \( V \setminus (\text{pred}[C^L] \cup \text{succ}[C^R]) \)

- \( U \) is antichain
- \( U \) are sinks in \( L \)
- \( U \) are sources in \( R \)
Sink-Adjusted Schedule (D&W)

Makespan Scheduling of Unit Jobs with Precedence Constraints in $O(1.995n)$ time
Sink-Adjusted Schedule (D&W)

Makespan Scheduling of Unit Jobs with Precedence Constraints in $O(1.995^n)$ time
Middle-Adjusted Schedule

Assumption: $n = m \cdot T$

<table>
<thead>
<tr>
<th>Machines</th>
<th>time</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>2</td>
</tr>
</tbody>
</table>

$|C^L| \approx \frac{|C|}{2}$

Guesses: $2^{|C|} \cdot 2^{|C|} \Rightarrow O^*(16^{|C|})$ time. **Not the full story.**
Conclusion

Main result:

\[ P|\text{prec}, p_j = 1|C_{\text{max}} \text{ in } O(1.995^n) \text{ time.} \]

Key idea’s:

- Tradeoff between \#AC’s and \(|C|\).
- \(O^*(169|C|)\) time algorithm

Future Research:

\[ P3|\text{prec}, p_j = 1|C_{\text{max}} \text{ in } 2^{o(n)} \text{ time?} \]
Extra ideas for the general case

\[ |C| \leq \frac{n}{7.5} \]

\[ |C| > \frac{n}{7.5} \quad \text{Claim 5.7} \quad \#AC \leq O(1.9445^n) \]

\[ \mathcal{O}^* \left( 169|C| \right) \text{ algorithm (Theorem 1.2)} \]

\[ m \leq \frac{n}{279} \]

\[ m > \frac{n}{279} \]

\[ \mathcal{O}^*(\#AC \cdot \binom{n}{m}) \text{ algorithm (Theorem 5.6)} \]

\[ \mathcal{O}^*(\#AC + 2^{n-m}) \text{ algorithm (Theorem 5.5)} \]