Complexity and Inapproximability Results for Parallel Task Scheduling and Strip Packing

Sören Henning, Klaus Jansen, *Malin Rau*, Lars Schmarje

Kiel University

Faculty of Engineering

Department of Computer Science

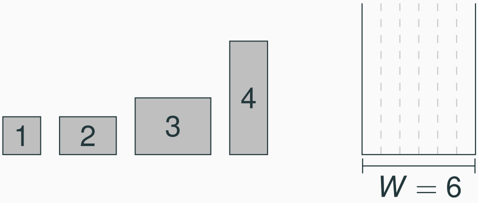
Strip Packing

Given:

- Strip with width $W \in \mathbb{N}$ and infinite height
- Set of n items I with width $w_i \in \mathbb{N}_{\leq W}$ height $h_i \in \mathbb{N}$

Objective:

Find a *feasible* packing into the strip with minimal height

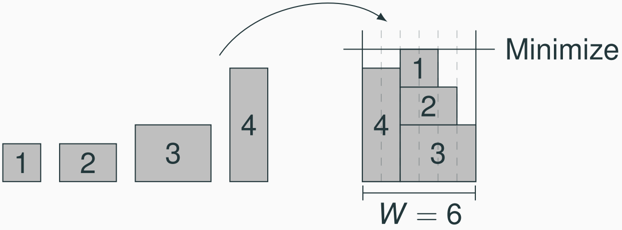


Given:

- Strip with width $W \in \mathbb{N}$ and infinite height
- Set of n items I with width $w_i \in \mathbb{N}_{\leq W}$ height $h_i \in \mathbb{N}$

Objective:

Find a *feasible* packing into the strip with minimal height



Known Results

Polynomial Time

[1] Baker, Coffman, Rivest, 1980

[2] Coffman, Garey, Johnson, Tarjan, 1980

[3] Steinberg, 1997

[4] Schiermeyer, 1994

[5] Harren, van Stee, 2009

[6] Harren, Jansen, Prädel, van Stee, 2014

Pseudo-Polynomial Time

[7] A. Adamaszek, T. Kociumaka, M. Pilipczuk, M. Pilipczuk, 2016

[8] K. Jansen, R. Thöle, 2010

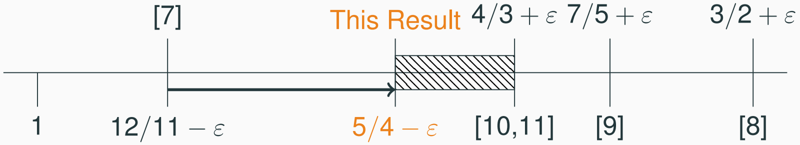
[9] G. Nadiradze, A. Wiese, 2016

[10] W. Gálvez, F. Grandoni, S. Ingala, A. Khan, 2016

[11] K. Jansen, M. Rau, 2016

Known Results

Pseudo-Polynomial Time



[7] A. Adamaszek, T. Kociumaka, M. Pilipczuk, M. Pilipczuk, 2016

[8] K. Jansen, R. Thöle, 2010

[9] G. Nadiradze, A. Wiese, 2016

[10] W. Gálvez, F. Grandoni, S. Ingala, A. Khan, 2016

[11] K. Jansen, M. Rau, 2016

Scheduling Parallel Tasks

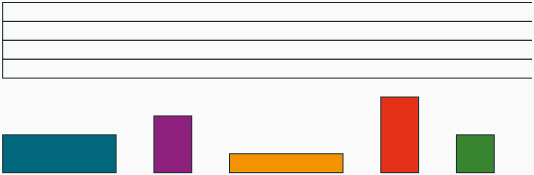
Given:

- $m \in \mathbb{N}$ parallel machines
- Set of n jobs J with width processing time $p_j \in \mathbb{N}_{\leq w}$
  machine requirement $q_j \in \mathbb{N}$

Objective:

Find a *feasible* schedule of the jobs with minimal makespan

$m = 4$

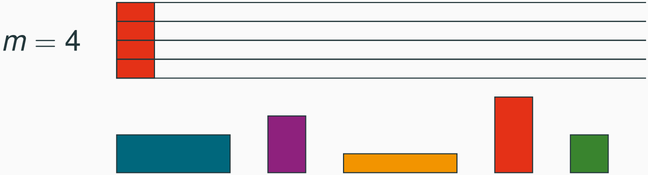


Given:

- $m \in \mathbb{N}$ parallel machines
- Set of n jobs J with width processing time $p_j \in \mathbb{N}_{\leq W}$ machine requirement $q_j \in \mathbb{N}$

Objective:

Find a *feasible* schedule of the jobs with minimal makespan



Scheduling Parallel Tasks

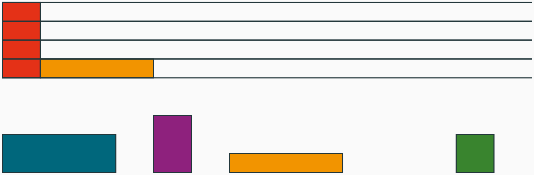
Given:

- $m \in \mathbb{N}$ parallel machines
- Set of n jobs J with width processing time $p_j \in \mathbb{N}_{\leq w}$ machine requirement $q_j \in \mathbb{N}$

Objective:

Find a *feasible* schedule of the jobs with minimal makespan

$m = 4$



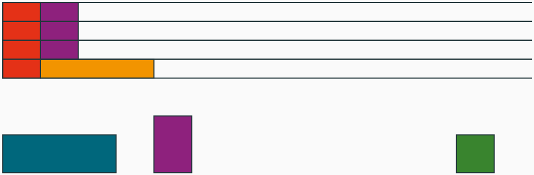
Given:

- $m \in \mathbb{N}$ parallel machines
- Set of n jobs J with width processing time $p_j \in \mathbb{N}_{\leq W}$ machine requirement $q_j \in \mathbb{N}$

Objective:

Find a *feasible* schedule of the jobs with minimal makespan

$m = 4$



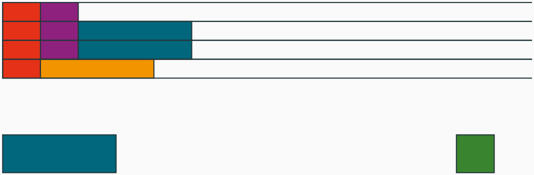
Given:

- $m \in \mathbb{N}$ parallel machines
- Set of n jobs J with width processing time $p_j \in \mathbb{N}_{\leq w}$ machine requirement $q_j \in \mathbb{N}$

Objective:

Find a *feasible* schedule of the jobs with minimal makespan

$m = 4$

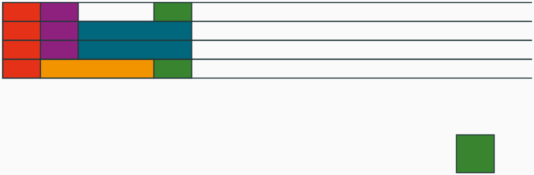


Given:

- $m \in \mathbb{N}$ parallel machines
- Set of n jobs J with width processing time $p_j \in \mathbb{N}_{\leq w}$
  machine requirement $q_j \in \mathbb{N}$

Objective:
Find a *feasible* schedule of the jobs with minimal makespan

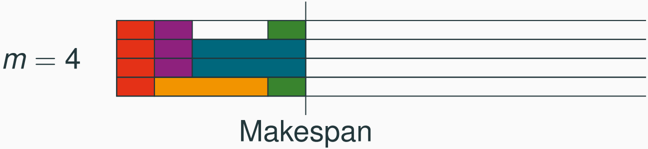
$m = 4$



Given:

- $m \in \mathbb{N}$ parallel machines
- Set of n jobs J with width processing time $p_j \in \mathbb{N}_{\leq W}$ machine requirement $q_j \in \mathbb{N}$

Objective:
Find a *feasible* schedule of the jobs with minimal makespan



Known Results

	Lower Bound	Algorithm
m arbitrary	$3/2$ [12]	$3/2+\varepsilon$ [13]
$m \in poly(n)$	strongly NP-hard [14]	PTAS [15]
constant $m \geq 5$	strongly NP-hard [14]	PTAS [16,17]
$m \in \{2,3\}$	NP-hard [12]	pseudo polynomial algorithm [14]

[12] folklore, Partition-Problem

[13] Jansen, 2012

[14] Du, Leung 1989

[15] Jansen, Thöle, 2010

[16] Amoura, Bampis, Kenyon, Manoussakis, 2002

[17] Jansen, Porkolab, 2002

	Lower Bound	Algorithm
m arbitrary	$3/2$ [12]	$3/2+\varepsilon$ [13]
$m \in poly(n)$	strongly NP-hard [14]	PTAS [15]
constant $m \geq 5$	strongly NP-hard [14]	PTAS [16,17]
$m \in \{2,3\}$	NP-hard [12]	pseudo polynomial algorithm [14]

Open Question

Is the problem strongly NP-hard when $m = 4$?

Known Results

	Lower Bound	Algorithm
m arbitrary	$3/2$ [12]	$3/2 + \varepsilon$ [13]
$m \in poly(n)$	strongly NP-hard [14]	PTAS [15]
constant $m \geq 5$	strongly NP-hard [14]	PTAS [16,17]
$m \in \{2, 3\}$	NP-hard [12]	pseudo polynomial algorithm [14]

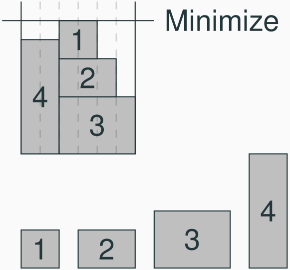
Open Question

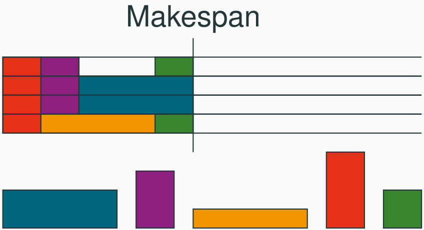
Is the problem strongly NP-hard when $m = 4$?

Our Answer

Yes, it is.

Connection of the Problems:





Scheduling Parallel Tasks

Is there a schedule on 4 Machines with makespan W (in which all jobs are scheduled on contiguous machines)?

Strip Packing

Given a Strip with width W, can we find a feasible packing with height at most 4?

3-Partition

Given

$3z$ integral numbers I with $\sum_{i \in I} i = zD$ and $\frac{1}{4}D < i < \frac{1}{2}D$ for each $i \in I$

Question

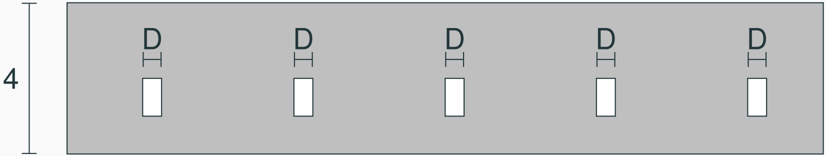
Can we partition the set I into sets $I_1, \ldots, I_z$ such that $\sum_{i \in I_j} i = D$ for each $j = 1, \ldots z$?

Properties

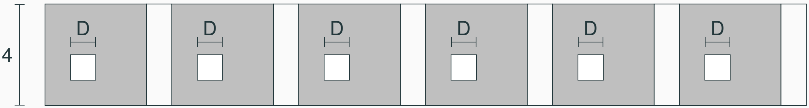
strongly NP-complete $\Rightarrow$ no pseudo polynomial algorithm, unless $P = NP$

Main Idea

Find a set of items forcing the 3-Partition instance to be a yes-instance, if there is a packing with height 4



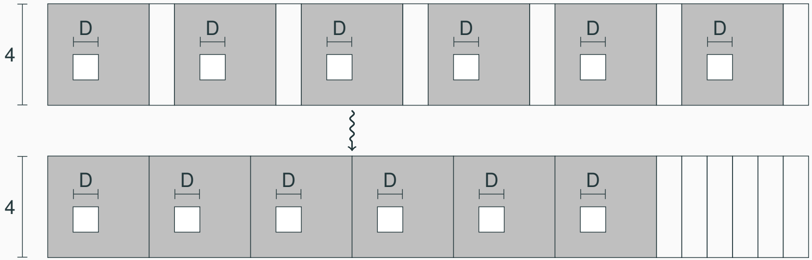
Finding a Structure



Properties

- Items with height 4 are not helpful

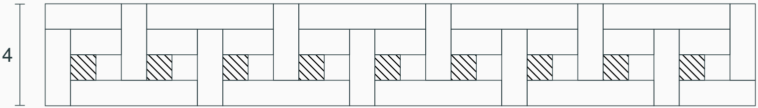
Finding a Structure



Properties

- Items with height 4 are not helpful

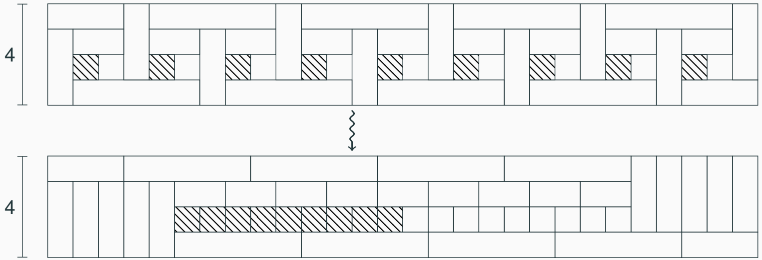
Finding a Structure



Properties

- Items with height 4 are not helpful
- No items with height 2: reordering is always possible

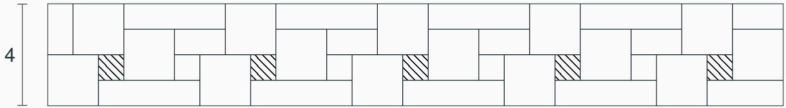
Finding a Structure



Properties

- Items with height 4 are not helpful
- No items with height 2: reordering is always possible

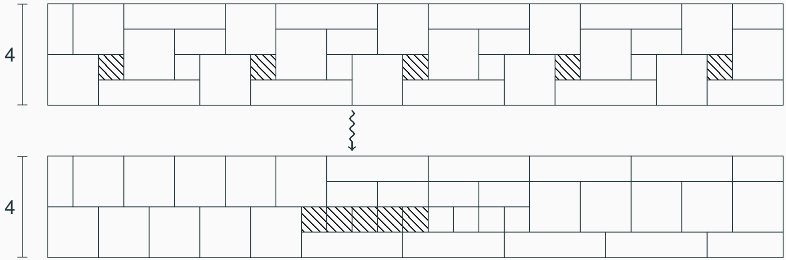
Finding a Structure



Properties

- Items with height 4 are not helpful
- No items with height 2: reordering is always possible
- No items with height 3: reordering is always possible

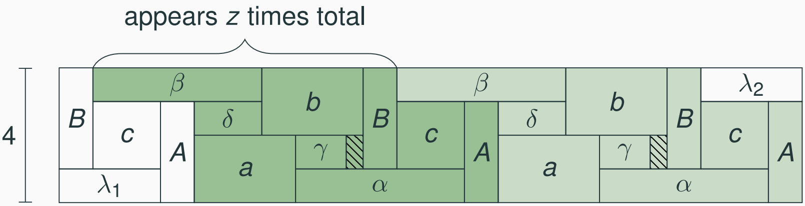
Finding a Structure



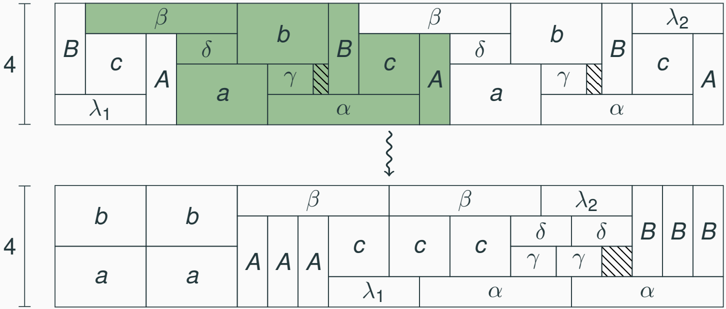
Properties

- Items with height 4 are not helpful
- No items with height 2: reordering is always possible
- No items with height 3: reordering is always possible

Packing Structure



A wrong choice of width can open the possibility to a reordering of the items:

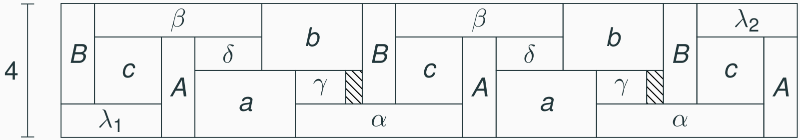


Objective

Find fitting widths for the items which preclude a reordering, that fuses areas for 3-partition items

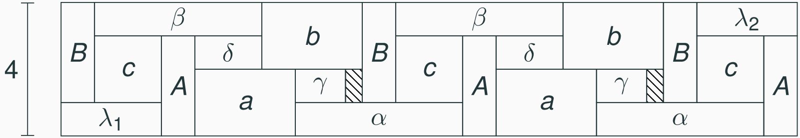
Observation

Each x-coordinate of any item is defined by a combination of width of other items



Observation

Each x-coordinate of any item is defined by a combination of width of other items

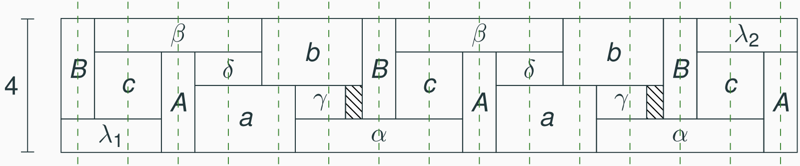


Idea:

Give each set of items which can be cut by a vertical line an individual token, which is added to the width of these items

Observation

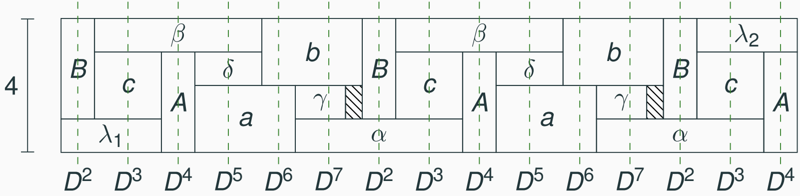
Each x-coordinate of any item is defined by a combination of width of other items



Idea:

Give each set of items which can be cut by a vertical line an individual token, which is added to the width of these items

Each x-coordinate of any item is defined by a combination of width of other items

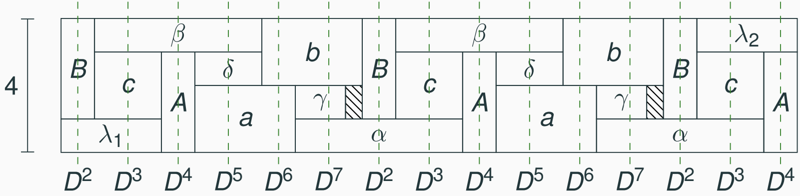


Idea:

Give each set of items which can be cut by a vertical line an individual token, which is added to the width of these items

Observation

Each x-coordinate of any item is defined by a combination of width of other items



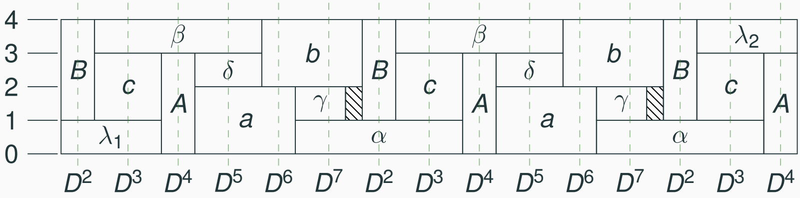
Idea:

Give each set of items which can be cut by a vertical line an individual token, which is added to the width of these items

Requirement

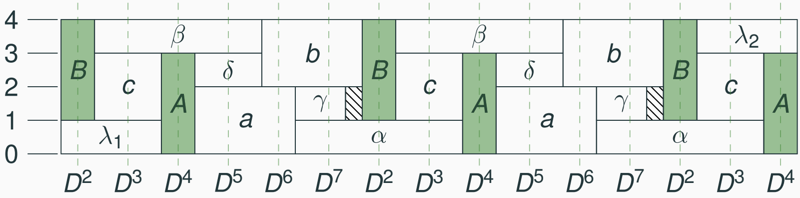
D has to be large, e.g., $D > 4(z+1)$.

Consequences



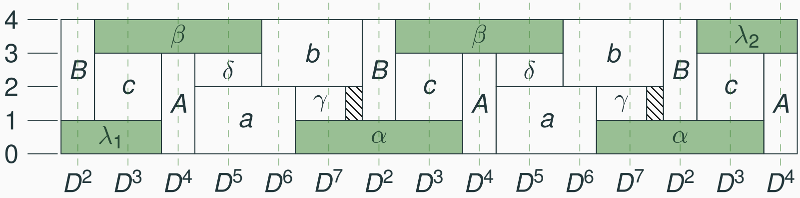
- Total width: $(z+1)(D^2 + D^3 + D^4) + z(D^5 + D^6 + D^7)$

Consequences



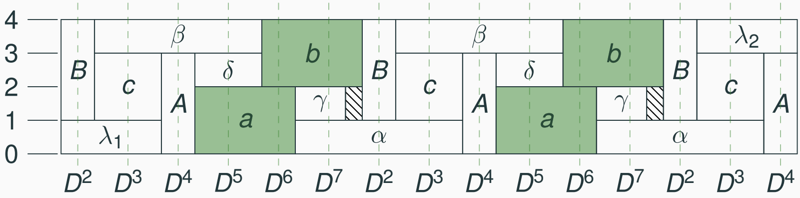
- Total width: $(z+1)(D^2 + D^3 + D^4) + z(D^5 + D^6 + D^7)$
- Items A and B will always overlap height 1 to 3

Consequences



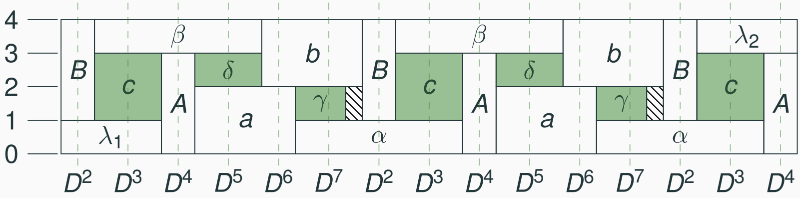
- Total width: $(z+1)(D^2 + D^3 + D^4) + z(D^5 + D^6 + D^7)$
- Items A and B will always overlap height 1 to 3
- Items α, β, λ can not overlap heights between 1 and 3.

Consequences



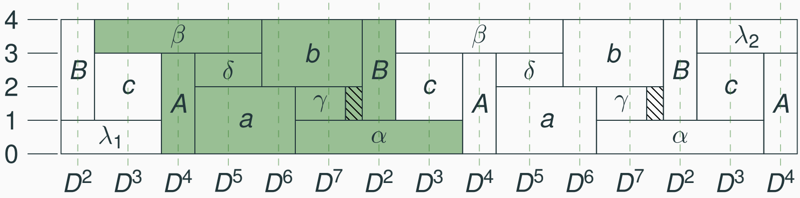
- Total width: $(z+1)(D^2+D^3+D^4)+z(D^5+D^6+D^7)$
- Items A and B will always overlap height 1 to 3
- Items α, β, λ can not overlap heights between 1 and 3.
- There have to be z items of type a or b which overlap the height 3 to 4 and z other, which overlap 0 to 1.

Consequences



- Total width: $(z+1)(D^2+D^3+D^4)+z(D^5+D^6+D^7)$
- Items A and B will always overlap height 1 to 3
- Items α, β, λ can not overlap heights between 1 and 3.
- There have to be z items of type a or b which overlap the height 3 to 4 and z other, which overlap 0 to 1.
- c, γ and δ have to overlap heights 1 to 3.

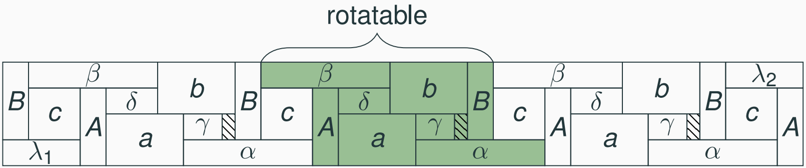
Consequences



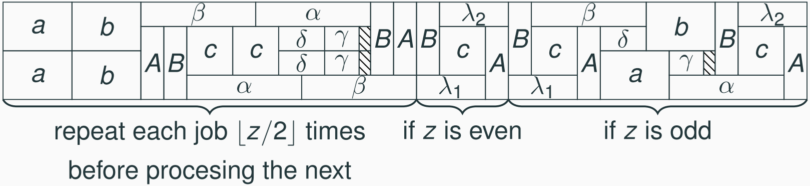
- Total width: $(z+1)(D^2+D^3+D^4)+z(D^5+D^6+D^7)$
- Items A and B will always overlap height 1 to 3
- Items α, β, λ can not overlap heights between 1 and 3.
- There have to be z items of type a or b which overlap the height 3 to 4 and z other, which overlap 0 to 1.
- c, γ and δ have to overlap heights 1 to 3.

These conditions are not enough

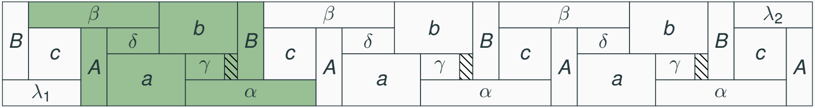
Reordering is still possible:



Rotate each second rotatable bock:



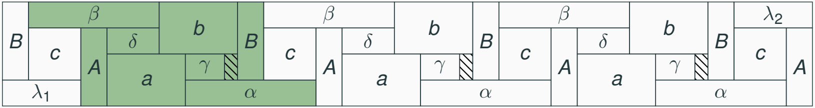
Use a new token D^8 to prevent swapping



Idea: add D^8 to α, β and λ, such that:

- $x_1 D^8(\alpha) + x_2 D^8(\beta) + x_3 D^8(\lambda_1) + x_4 D^8(\lambda_2) = (zD^8(\alpha) + zD^8(\beta) + D^8(\lambda_1) + D^8(\lambda_2))/2$
- $x_1 + x_2 + x_3 + x_4 = z + 1$
- $x_1 \in \{0, z\}$, $x_2 \in \{0, z\}$, $x_3 \in \{0, 1\}$, and $x_4 \in \{0, 1\}$ has exactly two solutions
  $x_1 = z$, $x_2 = 0$, $x_3 = 1$ and $x_4 = 0$ and
  $x_1 = 0$, $x_2 = z$, $x_3 = 0$ and $x_4 = 1$.

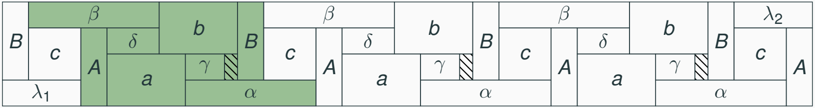
Use a new token D^8 to prevent rotation



Idea: use varying values for c, δ and γ to create a different amount of D^8 for α and β

- $D^8(c_i) = z + i, \forall i = 0, \ldots z$
- $D^8(\delta_i) = 3z - i, \forall i = 1, \ldots z$
- $D^8(\gamma_i) = 3z - i, \forall i = 1, \ldots z$

Use a new token D^8 to prevent rotation



Idea: use varying values for c, δ and γ to create a different amount of D^8 for α and β

- $D^8(c_i) = z + i, \forall i = 0, \ldots z$
- $D^8(\delta_i) = 3z - i, \forall i = 1, \ldots z$
- $D^8(\gamma_i) = 3z - i, \forall i = 1, \ldots z$

Consequence:

- $D^8(\alpha) = 4z$; $D^8(\beta) = (4z - 1)$; $D^8(\lambda_1) = z$; $D^8(\lambda_2) = 2z$;
- Total Width:

  $(z + 1)(D^2 + D^3 + D^4) + z(D^5 + D^6 + D^7) + (7z^2 + z)D^8$

Summary of the Techniques

- add a token for each vertical line to force certain items to be placed on these lines
- add an additional token D^8 to force certain items to be placed on a horizontal line
- Requirement: $D > 4(7z^2 + z)$
  $\rightarrow$ scale the partition instance

Theorem

For each $\varepsilon > 0$, there is no pseudo polynomial algorithm, which approximates strip packing with ratio $(\frac{5}{4} - \varepsilon)\mathrm{OPT}$, unless $P = NP$

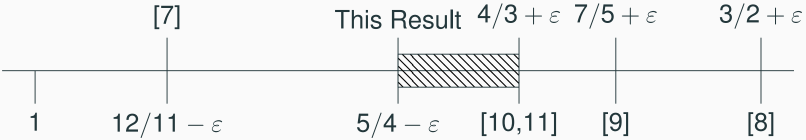
Theorem

Parallel Task Scheduling is strongly NP-complete for the case of $m = 4$

Open Questions

polynomial time

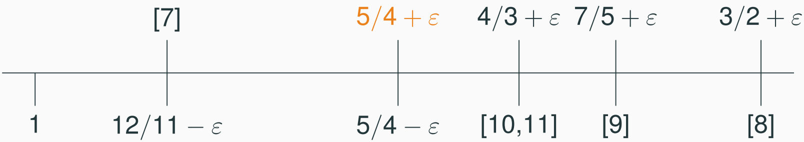
pseudo polynomial time



Open Questions

polynomial time

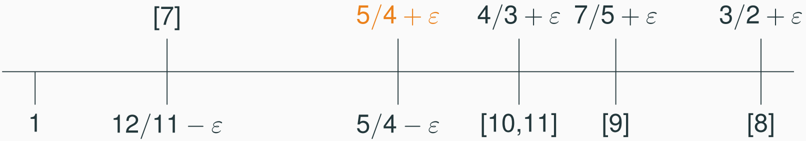
pseudo polynomial time



Open Questions

polynomial time

pseudo polynomial time



Thank you very much!