Approximating the Orthogonal Knapsack Problem for Hypercubes

Rolf Harren

University of Dortmund

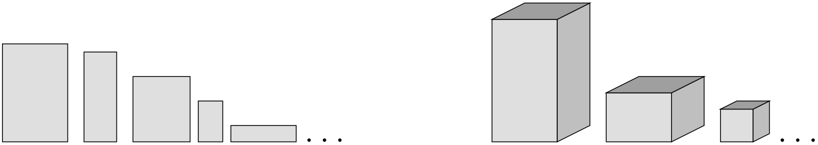
2007.09.07

Outline

Outline

Multidimensional Orthogonal Packing Problems

Input:
d-dimensional cuboid items $a_1, \ldots, a_n$

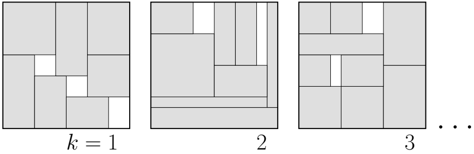


Objective:
an orthogonal, non-rotational and non-overlapping packing into a given space such that...

Multidimensional Orthogonal Packing Problems

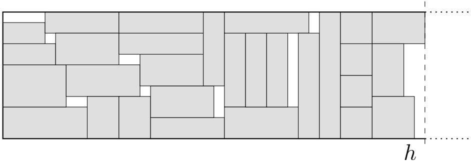
Bin Packing

...the number of bins is minimized



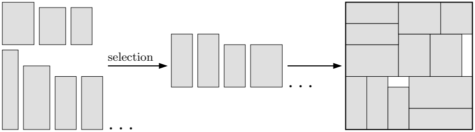
Strip Packing

...the total height is minimized



Knapsack Packing

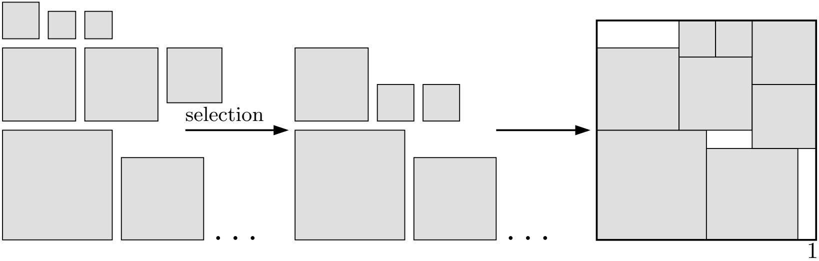
...the profit of the packed selection of items is maximized



Orthogonal Knapsack Packing for Hypercubes

Considered problem

All items are squares, cubes or hypercubes $0 < a_1, \ldots, a_n \leq 1$
Items have profits p_i
Bin has unit size



Results

	Bin Packing	Strip Packing
2- dim	1.525.. Bansal, Correa, APX-complete Sviridenko	AFPTAS Kenyon, Rémila
3- dim	3.382..	1.691.. Bansal, Han, Iwama Sviridenko, Zhang
d- dim	open	open

	Hypercube Bin Packing	Hypercube Strip Packing
2- dim	APTAS Bansal, Correa, Kenyon, Sviridenko	AFPTAS
d- dim	APTAS Bansal, Correa, Kenyon, Sviridenko	APTAS

Results

	Bin Packing	Strip Packing
2- dim	1.525.. APX-complete — Bansal, Correa, Sviridenko	AFPTAS — Kenyon, Rémila
3- dim	3.382..	1.691.. — Bansal, Han, Iwama Sviridenko, Zhang
d- dim	open	open

	Hypercube Bin Packing	Hypercube Strip Packing
2- dim	APTAS — Bansal, Correa, Kenyon, Sviridenko	AFPTAS
d- dim	APTAS — Bansal, Correa, Kenyon, Sviridenko	APTAS

Results

	Knapsack Packing		
	General		Hypercube
2- dim	$2+\epsilon$	Jansen, Zhang	$\frac{5}{4}+\epsilon$
3- dim	$7+\epsilon$ APX-complete	Diedrich, H., Jansen Thöle, Thomas	$\frac{9}{8}+\epsilon$
d- dim	open		$\frac{2^d+1}{2^d}+\epsilon$

Further results exist on

- Packing with large resources,
- maximizing the volume,
- maximizing the number, ...

Results

	Knapsack Packing		
	General		Hypercube
2- dim	$2+\epsilon$	Jansen, Zhang	$\frac{5}{4}+\epsilon$
3- dim	$7+\epsilon$ APX-complete	Diedrich, H., Jansen Thöle, Thomas	$\frac{9}{8}+\epsilon$
d- dim	open		$\frac{2^d+1}{2^d}+\epsilon$

Further results exist on

PACKING WITH LARGE RESOURCES,
MAXIMIZING THE VOLUME,
MAXIMIZING THE NUMBER, ...

Open Question

NP-Completeness

It is unknown for all previous packing problems whether the restriction to Hypercube packing is NP-hard for $d \geq 3$.

Applications

Cutting Problems

All packing problems can be seen as cutting problems, e.g., cutting textile or wood

Transportation Industry

- Arranging container on a ship
- Arranging items inside a container

Applications

Cutting Problems

All packing problems can be seen as cutting problems, e.g., cutting textile or wood

Transportation Industry

- Arranging container on a ship
- Arranging items inside a container

Applications

Advertisement Placement

- Arranging ads in a newspaper
- Arranging ads on a flash page

Scheduling

- Bounded running time on a computer with a grid layout for the processors
- Tasks need a fixed running time on a rectangular grid of processors

Applications

Advertisement Placement

- Arranging ads in a newspaper
- Arranging ads on a flash page

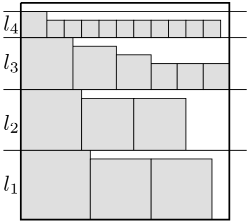
Scheduling

- Bounded running time on a computer with a grid layout for the processors
- Tasks need a fixed running time on a rectangular grid of processors

Outline

NFDH

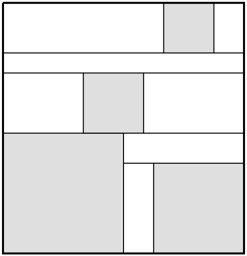
NEXT-FIT-DECREASING-HEIGHT (NFDH) is a very efficient layer based packing algorithm for small items



Lemma

For small items $a_i \leq \delta$, the unfilled volume is bounded by δd

Gaps in a Packing

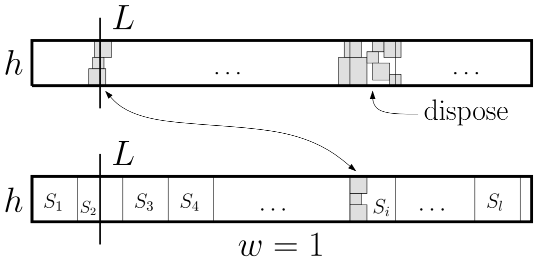


Lemma

Given a packing P of m squares we can partition the free space into at most $3m$ rectangles

At least one item in P has to be aligned to the bottom of the bin

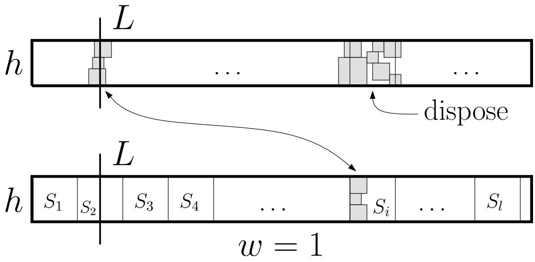
Shifting Technique



Lemma

For small items $a_i \leq \delta$ it is possible to free a given line L by shifting the items into a gap losing not more than $O(\delta)p(I)$ of the profit.

Shifting Technique



Lemma

For small items $a_i \leq \delta$ it is possible to free a given line L by shifting the items into a gap losing not more than $O(\delta)p(I)$ of the profit.

Rectangle Packing with Large Resources

If the bin is much bigger than the items we can derive a good approximation ratio

Lemma

There is an approximation algorithm for RECTANGLE PACKING *into a bin* $B = (a, b)$ *where* $a = 1$ *and* $b \geq \frac{1}{\epsilon^4}$ *with approximation ratio* $(1 + \epsilon)$

Idea:

- Bin has strip-like shape
- Pack a selection of items with the AFPTAS for STRIP PACKING and apply a shifting technique to the overhang

Rectangle Packing with Large Resources

If the bin is much bigger than the items we can derive a good approximation ratio

Lemma

There is an approximation algorithm for RECTANGLE PACKING *into a bin $B = (a, b)$ where $a = 1$ and $b \geq \frac{1}{\epsilon^4}$ with approximation ratio $(1 + \epsilon)$*

Idea:

- Bin has strip-like shape
- Pack a selection of items with the AFPTAS for STRIP PACKING and apply a shifting technique to the overhang

Outline

Outline

Separation

A little bit technical...

Separation

- Let $r = \lceil \frac{1}{\epsilon} \rceil$ and $\alpha_0 = \epsilon, \alpha_{i+1} = \alpha_i^4 \epsilon$
- Divide (unknown) optimal solution I_{opt}
- $M_i = \{s \in I_{opt} : s \in [\alpha_{i+1}, \alpha_i[\}$ for $1 \leq i \leq r$
- $\Rightarrow \exists i^* \in \{1, \ldots, r\}$ with $p(M_{i^*}) \leq \epsilon \cdot p(I_{opt})$

Large, medium and small items

- $L_{opt} := \{s \in I_{opt} : s \geq \alpha_{i^*}\}$
- $M := M_{i^*}$
- $S := \{s \in I : s < \alpha_{i^*+1}\}$

Separation

A little bit technical...

Separation

- Let $r = \lceil \frac{1}{\epsilon} \rceil$ and $\alpha_0 = \epsilon, \alpha_{i+1} = \alpha_i^4 \epsilon$
- Divide (unknown) optimal solution I_{opt}
- $M_i = \{s \in I_{opt} : s \in [\alpha_{i+1}, \alpha_i[\} \quad \text{for } 1 \leq i \leq r$
- $\Rightarrow \exists i^* \in \{1, \ldots, r\}$ with $p(M_{i^*}) \leq \epsilon \cdot p(I_{opt})$

Large, medium and small items

- $L_{opt} := \{s \in I_{opt} : s \geq \alpha_{i^*}\}$
- $M := M_{i^*}$
- $S := \{s \in I : s < \alpha_{i^*+1}\}$

Separation

A little bit technical...

Separation

- Let $r = \lceil \frac{1}{\epsilon} \rceil$ and $\alpha_0 = \epsilon, \alpha_{i+1} = \alpha_i^4 \epsilon$
- Divide (unknown) optimal solution I_{opt}
- $M_i = \{s \in I_{opt} : s \in [\alpha_{i+1}, \alpha_i[\} \quad \text{for } 1 \leq i \leq r$
- $\Rightarrow \exists i^* \in \{1, \ldots, r\}$ with $p(M_{i^*}) \leq \epsilon \cdot p(I_{opt})$

Large, medium and small items

- $L_{opt} := \{s \in I_{opt} : s \geq \alpha_{i^*}\}$
- $M := M_{i^*}$
- $S := \{s \in I : s < \alpha_{i^*+1}\}$

Separation

A little bit technical...

Separation

- Let $r = \lceil \frac{1}{\epsilon} \rceil$ and $\alpha_0 = \epsilon, \alpha_{i+1} = \alpha_i^4 \epsilon$
- Divide (unknown) optimal solution I_{opt}
- $M_i = \{s \in I_{opt} : s \in [\alpha_{i+1}, \alpha_i[\} \quad \text{for } 1 \leq i \leq r$
- $\Rightarrow \exists i^* \in \{1, \ldots, r\}$ with $p(M_{i^*}) \leq \epsilon \cdot p(I_{opt})$

Large, medium and small items

- $L_{opt} := \{s \in I_{opt} : s \geq \alpha_{i^*}\}$
- $M := M_{i^*}$
- $S := \{s \in I : s < \alpha_{i^*+1}\}$

Separation

A little bit technical...

Separation

- Let $r = \lceil \frac{1}{\epsilon} \rceil$ and $\alpha_0 = \epsilon, \alpha_{i+1} = \alpha_i^4 \epsilon$
- Divide (unknown) optimal solution I_{opt}
- $M_i = \{s \in I_{opt} : s \in [\alpha_{i+1}, \alpha_i[\} \quad \text{for } 1 \leq i \leq r$
- $\Rightarrow \exists i^* \in \{1, \ldots, r\}$ with $p(M_{i^*}) \leq \epsilon \cdot p(I_{opt})$

Large, medium and small items

- $L_{opt} := \{s \in I_{opt} : s \geq \alpha_{i^*}\}$
- $M := M_{i^*}$
- $S := \{s \in I : s < \alpha_{i^*+1}\}$

Separation

A little bit technical...

Separation

- Let $r = \lceil \frac{1}{\epsilon} \rceil$ and $\alpha_0 = \epsilon, \alpha_{i+1} = \alpha_i^4 \epsilon$
- Divide (unknown) optimal solution I_{opt}
- $M_i = \{s \in I_{opt} : s \in [\alpha_{i+1}, \alpha_i[\} \quad \text{for } 1 \leq i \leq r$
- $\Rightarrow \exists i^* \in \{1, \ldots, r\}$ with $p(M_{i^*}) \leq \epsilon \cdot p(I_{opt})$

Large, medium and small items

- $L_{opt} := \{s \in I_{opt} : s \geq \alpha_{i^*}\}$
- $M := M_{i^*}$
- $S := \{s \in I : s < \alpha_{i^*+1}\}$

Separation

In a nutshell

Large items are large (i.e., $\geq \alpha_{i^*}$)

Small items are small (i.e., $< \alpha_{i^*+1}$)

Medium items are unimportant (i.e., $p(M) \leq \epsilon \cdot \mathrm{OPT}(I)$).

Separation

In a nutshell

Large items are large (i.e., $\geq \alpha_{i^*}$)

Small items are small (i.e., $< \alpha_{i^*+1}$)

Medium items are unimportant (i.e., $p(M) \leq \epsilon \cdot \mathrm{OPT}(I)$).

Separation

In a nutshell

Large items are large (i.e., $\geq \alpha_{i^*}$)

Small items are small (i.e., $< \alpha_{i^*+1}$)

Medium items are unimportant (i.e., $p(M) \leq \epsilon \cdot \mathrm{OPT}(I)$).

Outline

Enumeration

Enumeration

- All relevant values are constant:

$$r, \alpha_i \text{ and } i^* \in \{1, \ldots, r\}$$

- at most $\frac{1}{\alpha_{i^*}^2}$ large items fit into the bin (volume argument)
- try all values for i^* and all possible selection of $\leq \frac{1}{\alpha_{i^*}^2}$ large items

we assume i^* and an optimal packing of the large items are known

Enumeration

Enumeration

- All relevant values are constant:

$$r, \alpha_i \text{ and } i^* \in \{1, \dots, r\}$$

- at most $\frac{1}{\alpha_{i^*}^2}$ large items fit into the bin (volume argument)
- try all values for i^* and all possible selection of $\leq \frac{1}{\alpha_{i^*}^2}$ large items

we assume i^* and an optimal packing of the large items are known

Enumeration

Enumeration

- All relevant values are constant:

$$r, \alpha_i \text{ and } i^* \in \{1, \ldots, r\}$$

- at most $\frac{1}{\alpha_{i^*}^2}$ large items fit into the bin (volume argument)
- try all values for i^* and all possible selection of $\leq \frac{1}{\alpha_{i^*}^2}$ large items

we assume i^* and an optimal packing of the large items are known

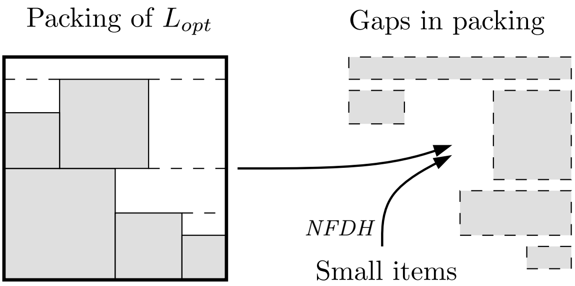
Enumeration

Enumeration

- All relevant values are constant:
$$r, \alpha_i \text{ and } i^* \in \{1, \ldots, r\}$$
- at most $\frac{1}{\alpha_{i*}^2}$ large items fit into the bin (volume argument)
- try all values for i^* and all possible selection of $\leq \frac{1}{\alpha_{i*}^2}$ large items

we assume i^* and an optimal packing of the large items are known

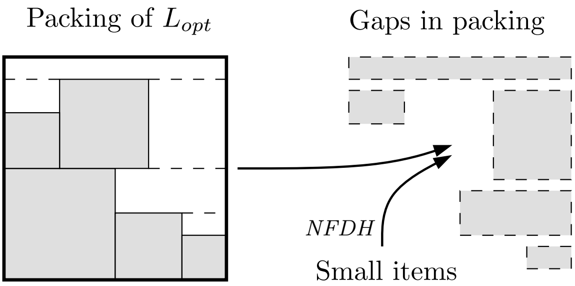
Unfilled Volume



At most $3m$ gaps

Unfilled volume per gap $\leq 2\delta$

Unfilled Volume



At most $3m$ gaps
Unfilled volume per gap $\leq 2\delta$

Unfilled Volume

Adding small items with NEXT-FIT-DECREASING-HEIGHT

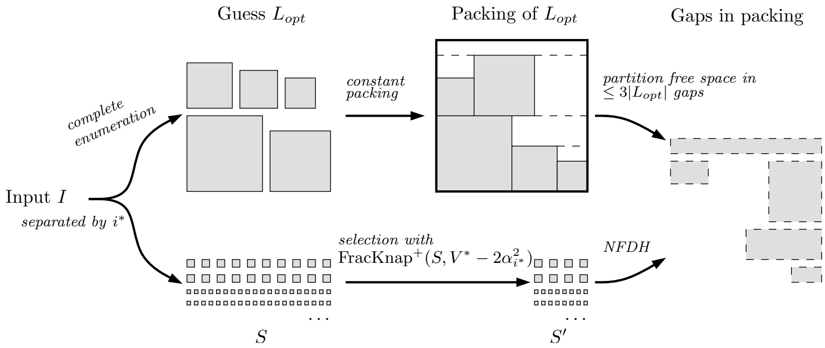
$$
\begin{aligned}
\text{Unfilled volume} &\le \overbrace{3m}^{\text{number of gaps}} \cdot \overbrace{2\delta}^{\text{unfilled volume per gap}} \\
&\le 3 \cdot \frac{1}{\alpha_{i*}^2} \cdot 2\alpha_{i^*+1} \\
&= 3 \cdot \frac{1}{\alpha_{i*}^2} \cdot 2\alpha_{i*}^4 \epsilon \\
&\le \alpha_{i*}^2
\end{aligned}
$$

Outline

3 Methods

- Enough remaining space
- Several large items
- Only one very large item

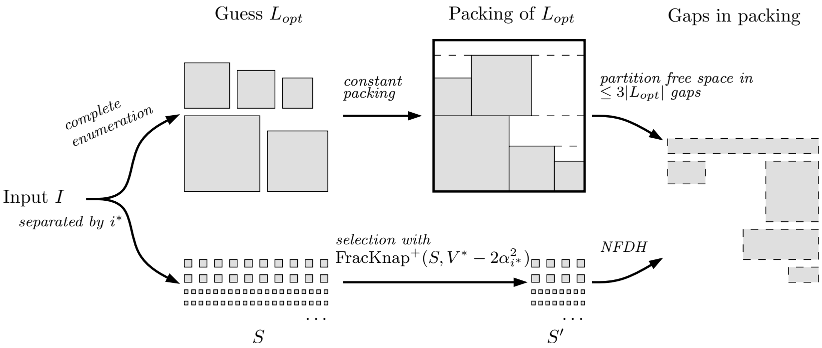
Enough Remaining Space $\mathrm{Vol}(L_{opt}) \leq 1 - \alpha_{i^*}$



$$p(S') \geq \frac{V^* - 2\alpha_{i^*}^2}{V^*}\mathrm{OPT}(S) \geq (1 - \frac{2\alpha_{i^*}^2}{\alpha_{i^*}})\mathrm{OPT}(S)$$

$$p(L_{opt} \cup S') \geq (1 - O(\epsilon))\mathrm{OPT}(I)$$

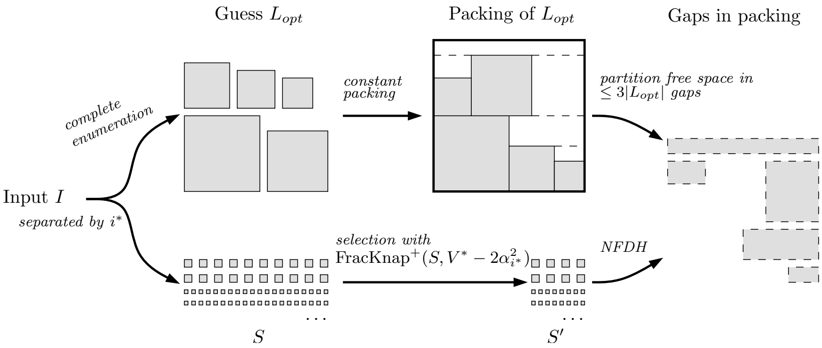
Enough Remaining Space $\mathrm{Vol}(L_{opt}) \leq 1 - \alpha_{i^*}$



$$p(S') \geq \frac{V^* - 2\alpha_{i^*}^2}{V^*}\mathrm{OPT}(S) \geq (1 - \frac{2\alpha_{i^*}^2}{\alpha_{i^*}})\mathrm{OPT}(S)$$

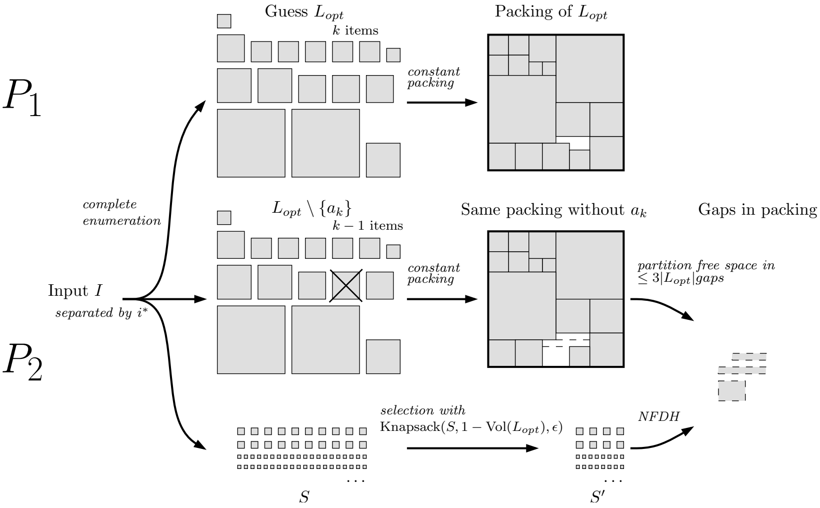
$$p(L_{opt} \cup S') \geq (1 - O(\epsilon))\mathrm{OPT}(I)$$

Enough Remaining Space $\mathrm{Vol}(L_{opt}) \leq 1 - \alpha_{i^*}$



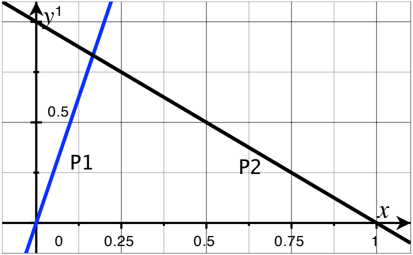
$$p(S') \geq \frac{V^* - 2\alpha_{i^*}^2}{V^*}\mathrm{OPT}(S) \geq \left(1 - \frac{2\alpha_{i^*}^2}{\alpha_{i^*}}\right)\mathrm{OPT}(S)$$
$$p(L_{opt} \cup S') \geq (1 - O(\epsilon))\mathrm{OPT}(I)$$

Several Large Items



Several Large Items

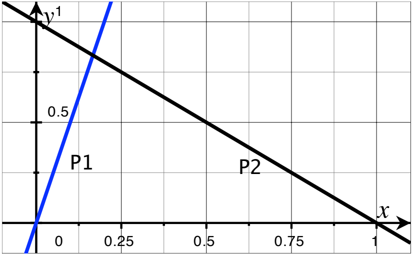
Profit $P_1 \geq k \cdot p_k$
Profit $P_2 \geq p(L_{opt} \cup S') - p_k$



$$\max(P_1, P_2) \geq (\frac{k}{k+1} - O(\epsilon))\mathrm{OPT}(I)$$

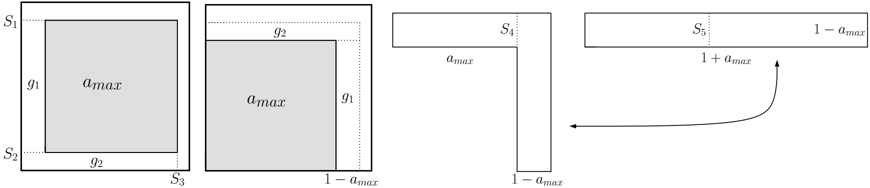
Several Large Items

Profit $P_1 \geq k \cdot p_k$
Profit $P_2 \geq p(L_{opt} \cup S') - p_k$



$$\max(P_1, P_2) \geq (\frac{k}{k+1} - O(\epsilon))\mathrm{OPT}(I)$$

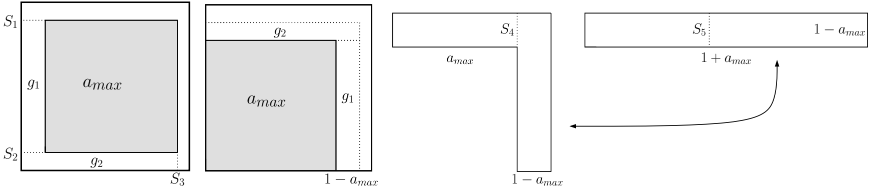
Only One Very Large Item $a_{max} \geq 1 - \epsilon^4$



Use RECTANGLE PACKING WITH LARGE RESOURCES for the free space

$$p(L_{opt} \cup S') \geq (1 - O(\epsilon))\mathrm{OPT}(I)$$

Only One Very Large Item $a_{max} \geq 1 - \epsilon^4$



Use Rectangle Packing with Large Resources for the free space

$$p(L_{opt} \cup S') \geq (1 - O(\epsilon))\mathrm{OPT}(I)$$

Outline

1. Introduction
   - Problem
   - Preparation
2. Square Packing
   - Separation into Large, Medium and Small Items
   - Packing the Large Items Optimally
   - Adding the Small Items
   - Putting Everything Together
3. Hypercube Packing
   - Generalization
4. Summary

Putting Everything Together

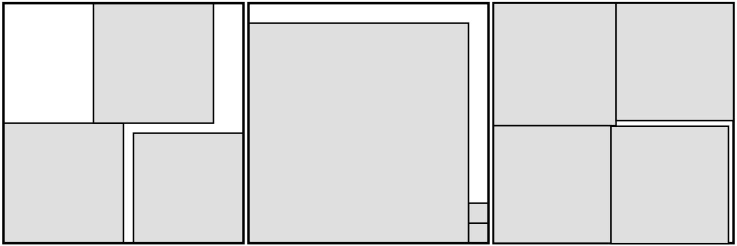
We derived methods for

Case 1	Enough remaining space	$(1 - O(\epsilon))\mathrm{OPT}(I)$
Case 2	Several large items	$(\frac{k}{k+1} - O(\epsilon))\mathrm{OPT}(I)$
Case 3	Only one very large item	$(1 - O(\epsilon))\mathrm{OPT}(I)$

We show that $k < 4$ can be reduced to Case 1 or Case 3

Main Idea

Three similarly large squares cannot fill a square bin almost completely



Outline

Methods for Hypercube Packing

Direct adoption of 2-dim methods

With suitable separation parameters

- Case 1 Enough remaining space
- Case 2 Several large items

work as well

More work needed

For Case 3 we need an approximation algorithm for

ORTHOGONAL KNAPSACK PACKING WITH LARGE RESOURCES FOR HYPERCUBES

with ratio $(1 + \epsilon)$ if the bin is big enough

Methods for Hypercube Packing

Direct adoption of 2-dim methods

With suitable separation parameters

Case 1 Enough remaining space

Case 2 Several large items

work as well

More work needed

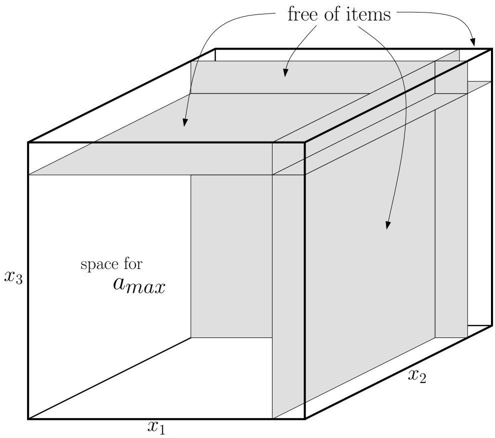
For Case 3 we need an approximation algorithm for

ORTHOGONAL KNAPSACK PACKING WITH LARGE RESOURCES FOR HYPERCUBES

with ratio $(1 + \epsilon)$ if the bin is big enough

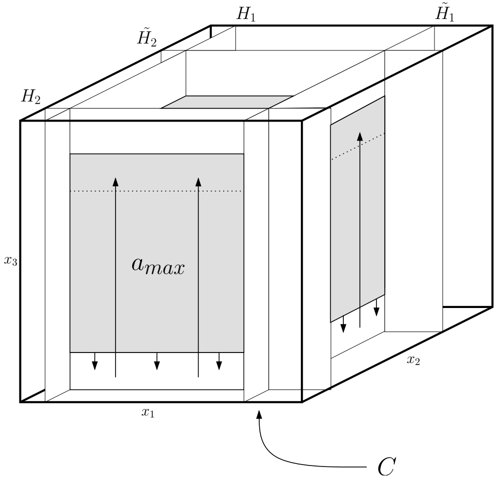
Wellstructured Packing

a_{max} big enough



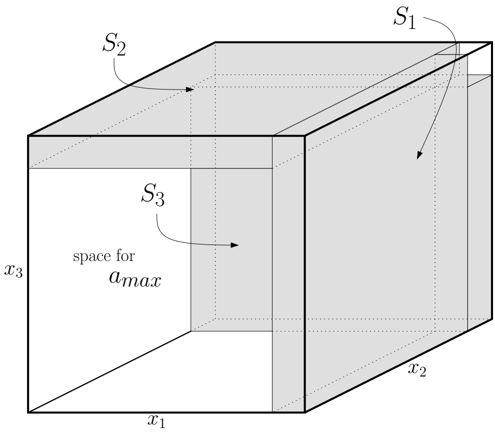
Construction of a Wellstructured Packing

a_{max} big enough



Packing the Small Items

Applying the algorithm for ORTHOGONAL KNAPSACK PACKING WITH LARGE RESOURCES FOR HYPERCUBES



Improving Approximation Ratio

Seven similarly large cubes cannot fill a cube bin almost completely

In general

For d-dim Hypercube Packing, we can reduce Case 2 with $k < 2^d$ to Case 1 or Case 3

Improving Approximation Ratio

Seven similarly large cubes cannot fill a cube bin almost completely

In general

For d-dim Hypercube Packing, we can reduce Case 2 with $k < 2^d$ to Case 1 or Case 3

Outline

Summary

Result

We developed an approximation algorithm with approximation ratio $1 + \frac{1}{2^d} + \epsilon$ for d-dimensional ORTHOGONAL KNAPSACK PACKING FOR HYPERCUBES

Main Steps

- Separation of large, medium and small items
- Packing the large items
- Adding the small items
- "Three similarly large squares cannot fill a square bin almost completely"

Summary

Result

We developed an approximation algorithm with approximation ratio $1 + \frac{1}{2^d} + \epsilon$ for d-dimensional ORTHOGONAL KNAPSACK PACKING FOR HYPERCUBES

Main Steps

- Separation of large, medium and small items
- Packing the large items
- Adding the small items
- "Three similarly large squares cannot fill a square bin almost completely"

Additional notes

Practical application of this algorithm

The running time is dominated by huge enumerations, making the algorithm practically unusable.

Asymptotic behavior

The structure of the problem does not allow asymptotic algorithms. Neither in the size of the input, nor in the value of an optimal solution.

Additional notes

Practical application of this algorithm

The running time is dominated by huge enumerations, making the algorithm practically unusable.

Asymptotic behavior

The structure of the problem does not allow asymptotic algorithms. Neither in the size of the input, nor in the value of an optimal solution.

The End

Thanks for your attention