Approximation Algorithms for Multidimensional Bin Packing

Arindam Khan
Indian Institute of Science (IISc), Bangalore
I GET IDEAS ABOUT WHAT’S ESSENTIAL WHEN PACKING MY SUITCASE.

— Diane von Furstenberg
Bin Packing Problem

• **Given**: $n$ items with sizes $s_1, s_2, \ldots, s_n$, s.t. $s_i \in (0,1]$,
• **Goal**: Pack all items into min # of unit bins.

  • Example: items $\{0.8, 0.6, 0.3, 0.2, 0.1\}$ can be packed in 2 unit bins: $\{0.8, 0.2\}$ and $\{0.6, 0.3, 0.1\}$.
  • $3/2$ hardness of approximation (from *Partition*).
    - This does not rule out OPT+1 guarantee.
  • delaVega-Lueker, Combinatorica ’81: APTAS,
  • Karp-Karmarkar, FOCS ’82: OPT $+ O(\log^2(OPT))$,
  • Hoberg-Rothvoss, SODA ’17: OPT $+ O(\log(OPT))$. 
Talk Overview

• Five generalizations of Bin Packing:
  1. Geometric Bin Packing (GBP),
  2. Strip Packing (2SP),
  3. Geometric Knapsack (2GK),
  4. Vector Bin Packing (VBP),
  5. Weighted Bipartite Edge Coloring (WBEC).
Pitas
**PTAS**

- **Polynomial Time Approximation Schemes (PTAS):**
  If for every $\varepsilon > 0$, there exists a poly-time ($O(n^{f(\varepsilon)})$-time) algorithm $A_\varepsilon$ such that $A_\varepsilon(I) \leq (1 + \varepsilon) \OPT(I)$.

- **Efficient PTAS (EPTAS):** if running time is $O(f(\varepsilon).n^c)$.

- **Fully PTAS (FPTAS):** if running time is $O((n/\varepsilon)^c)$.

- **Asymptotic PTAS (APTAS):** $A_\varepsilon(I) \leq (1 + \varepsilon) \OPT(I) + O(1)$.

- **QuasiPTAS (QPTAS):** $(1 + \varepsilon)$-approximation in $n^{(\log n)^{O(1)}}$-time.

- **PseudoPTAS (PPTAS):** $(1 + \varepsilon)$-approximation in $n^{O(1)}$-time, where $n$ is the number of items and the numeric data is polynomially bounded in $n$. 
1. Geometric Bin Packing
2-D Geometric Bin Packing

- **Given:** Collection of rectangles (by width, height),
- **Goal:** Pack them into minimum number of unit square bins.

- Orthogonal Packing: rectangles packed parallel to bin edges.
- With 90 degree *rotations* and *without rotations*.
- Reduces to 1-D bin packing, if all items have height = 1.
- For d-D GBP, we have d-D cuboids and bins instead of rectangles.
Applications:

- Cloth cutting, steel cutting, wood cutting
- Placing ads in newspapers
- Memory allocation in paging systems
- Truck Loading
- Palletization by robots
2D BP: Tale of approximability

<table>
<thead>
<tr>
<th>Algorithm (Asymptotic)</th>
<th>Hardness</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.125 [Chung Garey Johnson, JACM ‘82]</td>
<td>No APTAS (from 3D Matching) [Bansal-Sviridenko, SODA’04],</td>
</tr>
<tr>
<td>2+(\varepsilon) [Kenyon-Remilla, FOCS’96]</td>
<td>3793/3792 (with rotation), 2197/2196 (w/o rotation) [Chlebik-Chlebikova, CIAC’06]</td>
</tr>
<tr>
<td>1.69 [Caprara, FOCS’02]</td>
<td></td>
</tr>
<tr>
<td>1.52 [Bansal-Caprara-Sviridenko, FOCS’06]</td>
<td></td>
</tr>
<tr>
<td>1.5 [Jansen-Praedel, SODA’13]</td>
<td></td>
</tr>
<tr>
<td>1.405 [Bansal-K., SODA’14] (with and w/o rotations)</td>
<td></td>
</tr>
</tbody>
</table>

- d-dimensional (d>2) GBP (without rotations): 1.69\(d-1\) [Caprara, FOCS’02]. (with rotations): 1.69\(d-1\) [Sharma, ‘21].
- APTAS for d-dimensional squares: [Bansal-Sviridenko, SODA’04].
Configuration LP

- \( \mathbb{C} \): set of configurations (possible way of feasibly packing a bin).

- *Set* of (maximal) configurations without rotations.
Configuration LP

• $\mathbb{C}$: set of configurations (possible way of feasibly packing a bin).

• *Set* of (maximal) configurations with 90 degree rotations.
Configuration LP

- \( \mathbb{C} \): set of configurations (possible way of feasibly packing a bin).

**Primal:**
\[
\min \{ \sum_{C} x_{C} : \sum_{C \ni i} x_{C} \geq 1 \ (i \in I), x_{C} \geq 0 \ (C \in \mathbb{C}) \} 
\]

**Dual:**
\[
\max \{ \sum_{i \in I} v_{i} : \sum_{i \in C} v_{i} \leq 1 \ (C \in \mathbb{C}), v_{i} \geq 0 \ (i \in I) \} 
\]

- **Problem:** Exponential number of configurations!
- **Solution:** Can be solved within \((1 + \epsilon)\) accuracy using separation problem for the dual.

Separation problem of dual:
Given one bin, pack as much area as possible.
- PTAS [BCJPS, ISAAC 2009]
Round and Approx (R&A) Framework [Bansal-K. ‘14]

• Given a packing problem $\Pi$

1. If the configuration LP is solved within $(1 + \epsilon)$ factor
   \[
   \min \left\{ \sum_{C} x_C : \sum_{C \ni i} x_C \geq 1 \ (i \in I), x_C \geq 0 \ (C \in \mathbb{C}) \right\}
   \]

2. There is a $\rho$ approximation rounding-based algorithm.
   • Then there is $(1 + \ln \rho)$ approximation for $\Pi$. 

Rounding based Algorithms:

- Rounding based algorithms are ubiquitous in bin packing.
- In general, packing of small items is easy.
- Big items are problematic.
- **Big Items** are replaced by larger items from **$O(1)$ types**.
- **Loss:** Due to larger items.
- **Gain:** Fewer configurations. $O(1)$ types of large items imply rounded instance can be solved optimally.
- Example: Linear grouping [delaVega-Luker, Kenyon-Remilla], Geometric Grouping [Karp-Karmarkar], Harmonic Rounding [Lee-Lee, Caprara, Bansal et al.], JP rounding [JansenPradel].
Rounding based Algorithms in 2D

- **Classification** of items into big, wide, long, medium and small by defining two parameters $\delta, \mu (\ll \delta)$ such that total area of medium rectangles is very small.
Rounding based Algorithms in 2D

• Small & medium rectangles are packed separately, not incurring much loss.
• The main difficulty is in packing large, long/vertical & wide/horizontal items.
Rounding in 2D: container-based packing.
Rounding in 2D : container-based packing.

- **Container** is an axis-aligned rectangular region such that
Container is an axis-aligned rectangular region such that
- either it contains one large item.
Container is an axis-aligned rectangular region such that

- either it contains one large item.
- or items are packed inside the containers either as a horizontal stack or vertical stack.
Rounding in 2D: container-based packing.

- Container is an axis-aligned rectangular region such that
- either it contains one large item.
- or items are packed inside the containers either as a horizontal stack or vertical stack
Rounding in 2D: container-based packing.

- Container is an axis-aligned rectangular region such that
- either it contains one large item.
- or items are packed inside the containers either as a horizontal stack or vertical stack
- or all items inside it are very small in both dimensions.
Rounding in 2D: container-based packing.

- If there are $O(1)$ types of containers, then one can view that all large dimensions are rounded to $O(1)$ number of values.
- In polynomial time we can guess the sizes of containers.
- The gap between $\delta$ and $\mu$ ensures fractional and integral packing of wide/long items are very close.
Rounding in 2D: $\alpha$-approximation using container-based packing.
Rounding in 2D: $\alpha$-approximation using container-based packing.

• **Existence:** For any arbitrary feasible packing in $m$ bins, items can be packed in $\alpha m + O(1)$ bins of container-based packing with $O(1)$ type of containers.

• **Guess the packing:** Guess the sizes and positions) of $C$ containers in $n^{O(C)}$ time.

• **Pack the items:** Containers can be packed using a Dynamic Program based PTAS for multiple-knapsack problem.
Round and Approx Framework (R & A)

• 1. Solve configuration LP using APTAS. Let $z^* = \sum_{C \in \mathbb{C}} x^*_C$. 

Primal:

\[
\min \left\{ \sum_C x_C : \sum_{C \ni i} x_C \geq 1 \ (i \in I), \ x_C \geq 0 \ (C \in \mathbb{C}) \right\}
\]
Round and Approx Framework (R & A)

• 1. Solve configuration LP using APTAS. Let $z^* = \sum_{C \in \mathbb{C}} x^*_C$.

• 2. Randomized Rounding: For $q$ iterations:
  select a configuration $C'$ at random with probability $\frac{x^*_{C'}}{z^*}$.

Primal:
\[
\min \left\{ \sum_C x_C : \sum_{C \ni i} x_C \geq 1 \ (i \in I), x_C \geq 0 \ (C \in \mathbb{C}) \right\}
\]
Round and Approx Framework (R & A)

• 1. Solve configuration LP using APTAS. Let \( z^* = \sum_{C \in \mathbb{C}} x_C^* \).

• 2. Randomized Rounding: For \( q \) iterations:

   select a configuration \( C' \) at random with probability \( \frac{x_{C'}^*}{z^*} \).

• 3. Approx: Apply a \( \rho \) approximation rounding based algorithm \( A \) on the residual instance \( S \).

• 4. Combine: the solutions from step 2 and 3.
R & A Rounding Based Algorithms

• Probability item $i$ left uncovered after rand. rounding
  \[
  \left(1 - \sum_{c \ni i} \frac{x_c^*}{z^*}\right)^q \leq \frac{1}{\rho}
  \]
  by choosing $q = (\ln \rho) \cdot LP(I)$.

• Number of items of each type shrinks by a factor $\rho$
  e.g., $E[|B_j \cap S|] = \frac{|B_j|}{\rho}$ for some item type $B_j$.

• Concentration using Independent Bounded Difference Inequality.
Proof Sketch

• Rounding based Algo: $O(1)$ types of items
  $= O(1)$ number of constraints in configuration LP.
  
  • $ALGO(S) \approx OPT(\tilde{S}) \approx LP(\tilde{S})$.

  • As # items for each item type shrinks by $\rho$, $LP(\tilde{S}) \approx \frac{1+\epsilon}{\rho} LP(\tilde{I})$.

  • $\rho$ — approximation: $ALGO(I) \approx LP(\tilde{I}) \leq \rho OPT(I) + O(1)$.
  
  • $ALGO(S) \approx OPT(I)$. 

3/6/21
Proof Sketch

• **Thm:** R&A gives a \((1 + \ln \rho + \epsilon)\) approximation.

• **Proof:**

  • Randomized Rounding : \(q = \ln \rho \cdot LP(I)\)
  • Residual Instance \(S = (1 + \epsilon)OPT(I) + O(1)\).

• Round + Approx => \((\ln \rho + 1 + \epsilon)OPT(I) + O(1)\).
Guillotine Packing
Guillotine Packing

**Guillotine Cut:** Edge to Edge cut across a bin

---

**Guillotine packing:** All items can be separated by a sequence of guillotine cuts.  
**Objective:** Minimize number of bins such that packing in each bin is a guillotine packing.
Connection between guillotine & general packing

- APTAS for guillotine 2-D bin packing [Bansal Lodi Sviridenko, FOCS’05].
- **Conjecture:** Given any packing of $m$ bins, there is a guillotine packing in $4m/3 + O(1)$ bins. This will imply $(4/3 + \varepsilon)$-approximation for 2-D BP.
- **Conjecture:** Given any packing of $m$ bins, there is a 2-stage packing in $3m/2 + O(1)$ bins.
2. Strip Packing
Strip Packing Problem: (2-D)
Strip Packing Problem: (2-D)

• Input:
Strip Packing Problem: (2-D)

- **Input:**
  - Rectangles $R_1, R_2, \ldots, R_n$: Each $R_j$ has integral width and height $(w_i, h_j)$. 
Strip Packing Problem: (2-D)

- Input:
  - Rectangles $R_1, R_2, \ldots, R_n$: Each $R_i$ has integral width and height $(w_i, h_i)$. 

![Diagram of rectangles](image.png)
Strip Packing Problem: (2-D)

- **Input:**
  - Rectangles $R_1, R_2, \ldots, R_n$: Each $R_i$ has integral width and height $(w_i, h_i)$.
  - A strip of integral width $W$ and infinite height.
Strip Packing Problem: (2-D)

- **Input**: Rectangles $R_1, R_2, \ldots, R_n$: Each $R_i$ has integral width and height $(w_i, h_i)$.
- A strip of integral width $W$ and infinite height.

Width $W = 6$

Infinite height
Strip Packing Problem: (2-D)

- **Input:**
  - Rectangles $R_1, R_2, \ldots, R_n$: Each $R_i$ has integral width and height $(w_i, h_i)$.
  - A strip of integral width $W$ and infinite height.

- **Goal:**
  - Pack all rectangles **minimizing** the height of the strip.

![Diagram of rectangles and strip]

- Width $W = 6$
- Infinite height

$R_1 (1,6)$  $R_2 (3,2)$  $R_3 (2,2)$  $R_4 (1,3)$  $R_5 (3,1)$
Strip Packing Problem: (2-D)

• **Input**:  
  - Rectangles $R_1, R_2, \ldots, R_n$: Each $R_i$ has integral width and height $(w_i, h_i)$.  
  - A strip of integral width $W$ and infinite height.

• **Goal**:  
  - Pack all rectangles minimizing the height of the strip.  
  - *Axis-parallel* non-overlapping packing.

---

**Diagram**

- Rectangles $R_1(1,6)$, $R_2(3,2)$, $R_3(2,2)$, $R_4(1,3)$, $R_5(3,1)$.
- Strip width $W = 6$.
- Infinite height.
Strip Packing Problem: (2-D)

• Input:
  - Rectangles $R_1, R_2, \ldots, R_n$: Each $R_i$ has integral width and height $(w_i, h_i)$.
  - A strip of integral width $W$ and infinite height.

• Goal:
  - Pack all rectangles minimizing the height of the strip.
  - Axis-parallel non-overlapping packing.

Variant 1: No rotations are allowed!
Strip Packing Problem: (2-D)

• **Input:**
  - Rectangles \( R_1, R_2, ..., R_m \): Each \( R_i \) has integral width and height \((w_i, h_i)\).
  - A strip of integral width \( W \) and infinite height.

• **Goal:**
  - Pack all rectangles minimizing the height of the strip.
  - Axis-parallel non-overlapping packing.

### Variant 2:
90° rotations are allowed!
Strip Packing:

- Strip Packing generalizes bin packing (when all rectangles have same height),
Strip Packing:

- Strip Packing generalizes
  - bin packing (when all rectangles have same height),
  - makespan minimization (when all rectangles have same width).
Tale of approximability.

- **Asymptotic PTAS** [Kenyon-Remila, FOCS’96 ] (Without rotations),
- **Asymptotic PTAS** [Jansen-vanStee, STOC’05] (With rotations).
- **Absolute Approximation**: (Polynomial time)
  - **2.7**-appx. [First-Fit-Decreasing-Height, Coffman-Garey-Johnson-Tarjan ‘80].
  - **2**-appx [Steinberg’97]
  - **5/3+ε** [Harren-Jansen-Pradel-vanStee, Comp.Geom.‘14].
- **Hardness of appx in poly-time**: 3/2 (from Bin Packing).
- **(3/2+ε)**-appx for non-large rectangles [GGJJKR; APPROX’20]
Tale of approximability.

- **Pseudopolynomial time** \(O(nW)^c\):
  - **Algorithms:**
    - 1.5+\(\varepsilon\) [Jansen-Thole, SICOMP’10]
    - 1.4 +\(\varepsilon\) [Nadiradze-Wiese, SODA’16]
    - 4/3 +\(\varepsilon\) [Galvez, Grandoni, Ingala, K., FSTTCS’16; Jansen-Rau, WALCOM’17]
    - 5/4 +\(\varepsilon\) [Jansen, Rau, ESA’19]
  - **Hardness:**
    - 12/11 Adamaszek, Kociumaka, Pilipczuk, and Pilipczuk, TOCT’17
    - 5/4 Henning, Jansen, Rau, and Schmarje, CSR’18
3. Geometric Knapsack
Geometric Knapsack: (2-D)

Input:
- Rectangles $I := \{R_1, R_2, \ldots, R_n\}$; Each $R_i$ has integral width and height $(w_i, h_i)$ and profit $p_i$.
- A Square $K \times K$ knapsack.
Geometric Knapsack: (2-D)

- **Input:**
  - Rectangles $I := \{R_1, R_2, \ldots, R_n\}$; Each $R_i$ has integral width and height $(w_i, h_i)$ and profit $p_i$.
  - A Square $K \times K$ knapsack.
Geometric Knapsack: (2-D)

• Input:
  - Rectangles $I: = \{R_1, R_2, ..., R_n\}$; Each $R_i$ has integral width and height $(w_i, h_i)$ and profit $p_i$.
  - A Square $K \times K$ knapsack.

• Goal: Find an axis-parallel non-overlapping packing of a subset of input rectangles into the knapsack that maximizes the total profit.
Geometric Knapsack: (2-D)

- **Input**: 
  - Rectangles $I := \{R_1, R_2, ..., R_n\}$; Each $R_i$ has integral width and height $(w_i, h_i)$ and profit $p_i$.
  - A Square $K \times K$ knapsack.

- **Goal**: Find an axis-parallel non-overlapping packing of a subset of input rectangles into the knapsack that maximizes the total profit.

**Variant 1: 2DK**

- No rotations are allowed!
- $OPT = 155$

---

<table>
<thead>
<tr>
<th>rectangles</th>
<th>profit</th>
<th>position</th>
</tr>
</thead>
<tbody>
<tr>
<td>(9,6)</td>
<td>100 $</td>
<td></td>
</tr>
<tr>
<td>(7,6)</td>
<td>95 $</td>
<td></td>
</tr>
<tr>
<td>(5,8)</td>
<td>90 $</td>
<td></td>
</tr>
<tr>
<td>(4,6)</td>
<td>60 $</td>
<td></td>
</tr>
<tr>
<td>(4,4)</td>
<td>5 $</td>
<td></td>
</tr>
<tr>
<td>(2,3)</td>
<td>1 $</td>
<td></td>
</tr>
</tbody>
</table>

---

<table>
<thead>
<tr>
<th>rectangles</th>
<th>profit</th>
<th>position</th>
</tr>
</thead>
<tbody>
<tr>
<td>(9,6)</td>
<td>90 $</td>
<td></td>
</tr>
<tr>
<td>(4,6)</td>
<td>5 $</td>
<td></td>
</tr>
<tr>
<td>(2,3)</td>
<td>5 $</td>
<td></td>
</tr>
<tr>
<td>(4,6)</td>
<td>60 $</td>
<td></td>
</tr>
</tbody>
</table>
Geometric Knapsack: (2-D)

• **Input:**
  - Rectangles $I := \{R_1, R_2, \ldots, R_n\}$; Each $R_i$ has integral width and height $(w_i, h_i)$ and profit $p_i$.
  - A Square $K \times K$ knapsack.

• **Goal:** Find an axis-parallel non-overlapping packing of a subset of input rectangles into the knapsack that maximizes the total profit.

---

**Variant 2: (2DKR)**

90 degree rotations are allowed!

$\text{OPT}=165$
Geometric Knapsack: Complexity

• Geometric Knapsack is Strongly NP-hard (even when all items are squares with profit 1), [Leung et al., 1990]
  - No exact algorithm even in pseudo-polynomial time (unless P=NP).

• W[1]-hard [Grandoni, Kratsch, Wiese, ESA’19], So no EPTAS.

• Not known whether the problem is APX-hard.

• The existence of a PTAS/QPTAS/PPTAS is still open!

• (1+ε)-approximation known if
  - profit of an item is equal to its area. [Bansal et al., ISAAC ‘09].
  - items are relatively small [Fishkin et al., MFCS ‘05].
  - items are squares [Wiese-Heydrich, SODA ’17].
Geometric Knapsack:

• *(2+ε)*-approximation [Jansen-Zhang, SODA’04]
  - for both with and without rotations.
  - even in the cardinality case (when all profits are 1).

• Broke the barrier of 2 [Galvez-Grandoni-Ingala-K.-Wiese, FOCS’17]
  - Without rotations: *(17/9+ε)*<1.89-appx.
  - With rotations: *(1.5+ε)*-appx.
  - Cardinality case: 1.72, *(4/3+ε)*-appx., resp.

• Pseudopolynomial time *(4/3+ε)*-appx.
  [Galvez-Grandoni-K.-Romero-Wiese, SoCG’21]

• PPTAS for guillotine 2-D knapsack [K.-Maiti-Sharma-Wiese, SoCG ’21]
Corridor decomposition

• Any feasible packing can be partitioned into $O(1)$ corridors (rectilinear polygons) defined by $O(1)$ number of line segments and intersecting only rectangles of profit $\leq \varepsilon p(OPT)$.

• Process the corridors so that we can retain a large fraction of profits in a packing that only have $O(1)$ types of containers or corridors with two bends.
4. Vector packing
Vector Packing: Multidimensional Bin Packing

Jobs (items)  

Resources (dimensions): 
CPU, Memory, Network, Disk, I/O

Servers (bins)  

Goal: Assign all jobs to the servers s.t. min number of servers are needed.
Vector packing

• Input:
  Set of $d$-dimensional vectors: $[0,1]^d$

• Goal:
  pack all vectors into minimum # of unit vector bins such that for each bin for each dimension the coordinate wise sum of packed vector in it is $\leq 1$. 

$d = 2$
A tale of approximability

• Dimension $d$ is constant, otherwise approximation hardness $\Omega(d^{1-\epsilon})$.

• Asymptotic Approximation:
  • $d + \epsilon$ [Linear grouping: de la Vega-Lueker ‘81]
  • $2 + \ln(d) + \epsilon$ [Assignment LP: Chekuri-Khanna ‘99]
  • $1 + \ln(d) + \epsilon$ [Configuration LP: Bansal-Caprara-Sviridenko FOCS ‘06]

• Absolute/Nonasymptotic: 2 for $d = 2$ [Kellerer-Kotov 2003]

• Hardness:

• No APTAS (from 3D Matching)[Woeginger 1997].
Recent progress:

• Bansal, Elias, K., SODA’ 16: [Multiobjective matching+R&A framework]
• Almost tight \((1.5 + \epsilon)\) (Absolute) Approximation for 2-D.
  \(1.405\) Asymptotic Approximation for 2-D.
• \(0.807 + \ln(d + 1)\) Asymptotic Approximation for \(d\) dim.
• Hardness of \(d\) for constant rounding based algorithms.
• If we allow extra resource of \(\epsilon\) in \((d - 1)\) dimensions, we can find a packing in polynomial time in \((1 + \epsilon)Opt + O(1)\) number of bins.
• Sai Sandeep, 2021: \(\Omega(\log d)\)-hardness, when \(d\) is large.
5. Weighted Bipartite Edge Coloring

Edge Coloring          Meets          Bin Packing
Weighted Bipartite Edge Coloring

• **Given:** An edge-weighted bipartite multi-graph $G = (L \cup R, E)$ with edge-weights $w: E \to [0,1]$.

• **Goal:** Find a proper weighted coloring with minimum number of colors.

• **Proper weighted coloring:** Sum of the edges incident to any vertex of any color is $\leq 1$. 

![Graph with edge weights]
Weighted Bipartite Edge Coloring

• **Given:** An edge-weighted bipartite multi-graph $G := (L \cup R, E)$ with edge-weights $w: E \rightarrow [0,1]$.

• **Goal:** Find a proper weighted coloring with minimum number of colors.

• **Proper weighted coloring:** Sum of the edges incident to any vertex of any color is $\leq 1$.

• Reduces to bin packing if $|L| = |R| = 1$. 
Weighted Bipartite Edge Coloring: Previous Works

- **Conjecture 1.** [Chung & Ross, 1991]
  There is a proper weighted coloring with \(2m - 1\) colors where
  \[m = \max \{\min \text{# bins to pack } w_e's \mid e \in \delta(v)\}\].

Lower bound:
- Ngo -Vu SODA’03: \(1.25m\)

Upper bound:
- Correa-Goemans STOC ‘04: \(2.548m\)
- Feige-Singh ESA ‘08: \(9m/4 = 2.25m\)
- K.-Singh, FSTTCS’15: \(2.22m\). [FF-based algorithm, Config LP-based analysis]
Other related problems.

- Storage Allocation Problem [Momke, Wiese, ICALP’20].
- Unsplittable Flow on a Path [Grandoni, Momke, Wiese, Zhou, STOC’18].
- Dynamic Storage Allocation [Buchsbaum et al., STOC’03].
- Maximum Independent Set of Rectangles [Mitchell’21].
- Pach-Tardos Conjecture [K., Reddy, APPROX’20].
## Summary of present status

<table>
<thead>
<tr>
<th>Problem</th>
<th>Approximation Guarantee</th>
<th>Hardness</th>
</tr>
</thead>
<tbody>
<tr>
<td>1-D Bin Packing</td>
<td>OPT+O(log OPT)</td>
<td>Strongly NP-hard</td>
</tr>
<tr>
<td>2-D Geometric BP (NR/R)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d-D Geometric BP (NR/R) (d&gt;2)</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.405 [BK, SODA’14]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.69^{d-1} [C, MathOR’08]</td>
<td></td>
</tr>
<tr>
<td>d-D Vector BP</td>
<td>0.81+O(log d) [BEK, SODA 2016]</td>
<td>Ω(\log d)[Sai Sandeep, ‘21]</td>
</tr>
<tr>
<td>2-D Strip Packing (NR/R)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>3-D Strip Packing (NR)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d-D Strip Packing (NR)</td>
<td>PT (abs.): 5/3 + \epsilon. [HJPS, CompGeo’14]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PPT(abs.): 5/4 + \epsilon [JR, ESA’19]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>3/2 + \epsilon [JP, SOFSEM’14]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>1.69^{d-1} [C, MathOR’08]</td>
<td></td>
</tr>
<tr>
<td>2-D Geometric Knapsack (NR/R)</td>
<td></td>
<td></td>
</tr>
<tr>
<td>d-D Geometric Knapsack (NR)</td>
<td>PT: 1.89, 1.5 + \epsilon [GGHKW FOCS’17]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PPT: 4/3+\epsilon [GKKRW SoCG’21]</td>
<td></td>
</tr>
<tr>
<td></td>
<td>PT: 3^d (1 + \epsilon) [Sharma ’21]</td>
<td></td>
</tr>
<tr>
<td>Weighted Bipartite Edge Coloring</td>
<td>2.22 [KS FSTTCS’15]</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Strongly NP-hard.</td>
</tr>
</tbody>
</table>

*W[1]-hard [GKW ESA’19] APX-hard (d>2)*

1. Strongly NP-hard.

2. No APTAS [BCKS, MathOR’06].
Top 10 open problems

1. Algorithm with \( \text{OPT}+O(1) \)-guarantee for bin packing.
2. Resolve integrality gap of configuration LP.
3. A poly(d)-approximation or hardness for d-dim geometric bin packing?
4. Improve 1.405- asymp approximation for 2D GBP.
5. Resolve guillotine conjecture (4/3) for 2-D BP.
6. Resolve 2-stage conjecture (3/2) for 2-D BP.
7. \((3/2+\varepsilon)\)-(absolute) approximation for 2D strip packing?
8. PTAS (or PPTAS or QPTAS) for 2-D GK (even with rotations)?
10. Study parameterized approximation/exact/practical algorithms.
THANK YOU