Polynomial Time Approximation Schemes for Class-Constrained Packing Problems

Hadas Shachnai and **Tami Tamir**

The Technion, Haifa

The Class-Constrained Multiple Knapsack Problem (CCMK)

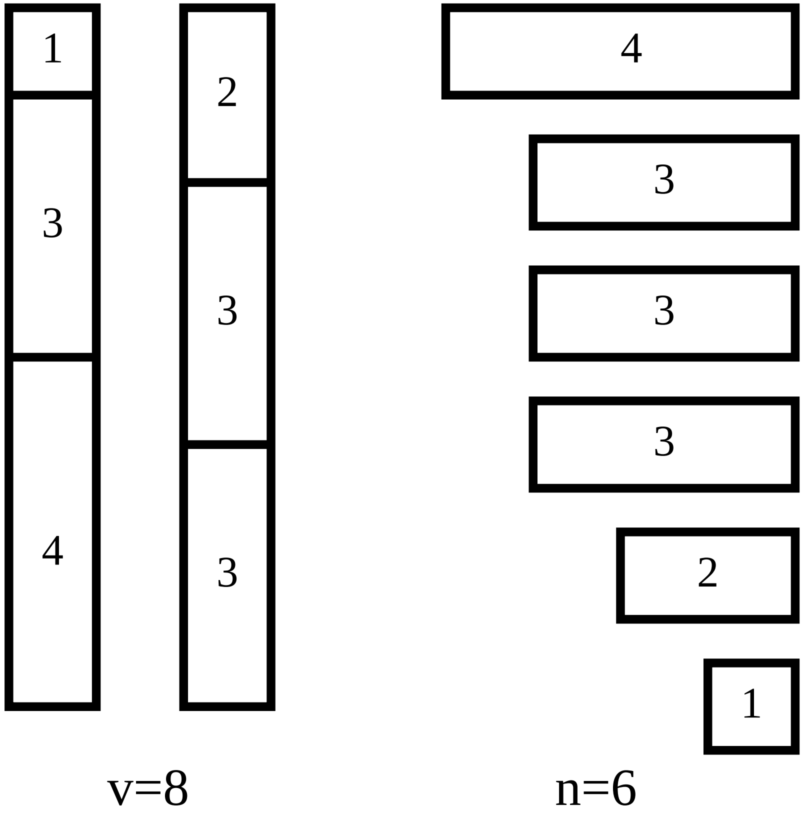
- A set of N knapsacks, $K_1, ..., K_N$.
  Each knapsack, K_j, has limited volume, v_j,
  and a limited number, c_j, of compartments.

- A set, I, of n items, $u_1, ..., u_n$.
  Each item, $u \in I$, has a size, $s(u)$, a value, $p(u)$,
  and a type (color), $c(u) \in \{1, ..., M\}$.

- The objective: A legal placement of some $U \subseteq I$ with maximal value.
  Legal = For all j, the total size of items placed in K_j is at most v_j, and the number of different colors included in K_j is at most c_j.

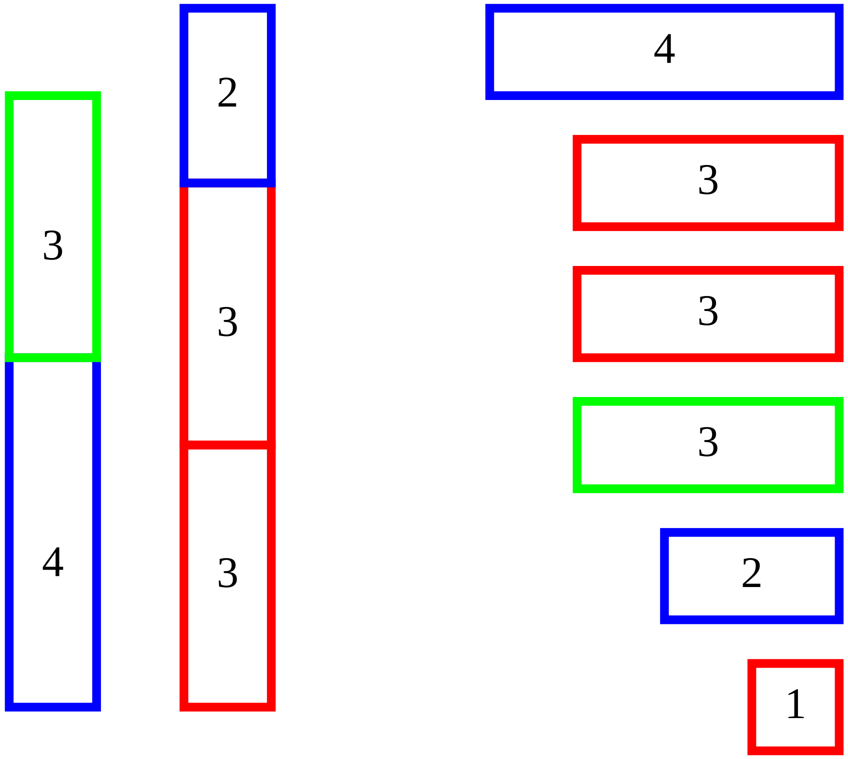
An Example (MK)

For all $u \in I$, $p(u) = s(u)$.



optimal value = 16

An Optimal Packing with Class-Constraints



v=8
c=2

n=6
M=3

optimal value = 15

The Class-Constrained Bin Packing Problem (CCBP)

- A set of identical bins, each having size 1 and c compartments.

- A set, I, of n items, $u_1, ..., u_n$. Each item, $u \in I$, has a size, $s(u) < 1$, and a color, $c(u) \in \{1, ..., M\}$.

- The objective: A legal placement of I in a minimal number of bins.
  Legal = The total size of items placed in each bin is at most 1, and the number of different colors included in each bin is at most c.

Generalized Class-Constrained Packing (GCCP)

- **Each item, $u \in I$, has a size, $s(u)$, a value, $p(u)$, and a set, $c(u) \subseteq \{1, ..., M\}$, of colors.**
- **It is legal to place u in any knapsack that allocate a compartment to some color in $c(u)$.**

Motivation

1. The data placement problem
   - The knapsacks are disks, each having
     - limited load capacity (v_j)
     - limited storage capacity (c_j)
   - The items are requests for broadcasts, each for one out of M video program files.
   - Allocate to each file disk space and fraction of the load capacity, such that the load generated due to access requests to that file is satisfied.
2. Production planning.
   - The knapsacks are machines, each having
     - limited physical resource (v_j)
     - limited number of configurations (c_j)
   - The items are orders for products of M distinct types.
3. Scheduling parallelizable tasks with machine allotment constraints.

Related Work

- IP formulation of CCMK (Dawande and Kalagnanam, 1998)

- A PTAS for MK (Chekuri and Khanna, 2000)

- A dual PTAS for BP (Hochbaum and Shmoys, 1987; Epstein and Sgall, 1999)

- Cardinality-constrained packing problems - when the *number* of items is limited (no types) (Kellerer and Pferschy, 1999; Caprara, Kellerer, Pferschy, and Pisinger, 2000)

Related Work (cont.)

- **Data placement** (Shachnai and Tamir, 1999; Golubchik, Khanna, Khuller, Thurimella, and Zhu, 2000).
- **Multi-dimensional packing problems:**
  May seem related, but there is a crucial difference:
  In class-constrained packing each bin has the 2-dim size $v_j \times c_j$,
  Each item has the 2-dim size

$$s(u) = \begin{cases} s(u) \times 1 & \text{if } u \text{ is the first item of its color in } K_j \\ s(u) \times 0 & \text{otherwise} \end{cases}$$

Our Results

- **A dual PTAS for CCBP,**
  (packs any instance, I, into $m \le OPT(I)$ bins of size $1+\varepsilon$).
- **A PTAS for CCMK,**
  (packs a set $U \subseteq I$ such that $P(U) \ge (1-\varepsilon)OPT(I)$).
- **An FPTAS for the 0-1 class-constrained knapsack (CCKP).**
  - optimal when all the items have the same value.
- **APX-hardness proofs for GCCP (already for the case where all items have the same size and the same value).**

- For the PTASs we assume that M is a fixed constant.

- The FPTAS is suitable for $M = \theta(n)$.

- In the APX-hardness proof $M = \theta(n)$.

A Dual PTAS for CCBP

First Attempt: Follow with some class-constrained adjustments a known PTAS for bin packing.

The problem: These PTASs consider the large (of size $\geq \varepsilon$) items first, and then add the small ones.
This approach yields an *asymptotic* PTAS, as in [DKS98].
(consider instances with small items from many colors).

Second Attempt: Use a PTAS that first eliminates somehow the small items.
For example, in [AAWY97] and [ES99], the small items are replaced by items of size ε.

The problem: We will need to group the small items of each color separately. This results in a PTAS whose complexity depends on $\varepsilon' = \frac{\varepsilon}{M}$.

A Dual PTAS for CCBP (cont.)

Third (winning) Attempt: Group and handle the small items 'carefully':

1. Let $\delta = \frac{\varepsilon}{4}$.
2. Transform I into an instance I' with at most one item of size $< \delta$ from each color. The small items of each color are grouped greedily: start to form a new group, when the total size of the previous one exceeds δ.

Lemma: Given a packing of I in m bins of size 1, we can pack I' in m bins of size $1 + 2\delta$.

Proof sketch: Show by induction on m that the small items of I can be replaced by the grouped items of I', such that for any bin B and a set of colors C_1, the small items with colors in C_1 packed in B, are replaced by grouped items in I' whose colors are in C_1, and whose total size is larger by at most 2δ.

A Dual PTAS for CCBP (cont.)

Base: $m = 1$, no overflow.
Step: $m = k$.

1. replace two arbitrary bins B_1, B_2 by a single bin B_{new}.
2. pack I' in the resulting instance (of $k-1$ bins).
3. split the grouped items in B_{new} between B_1 and B_2. ■

Conclusion: Let m be the minimal number of bins of size $1+2\delta$ needed to pack I', then $m \leq OPT(I)$.

3. Use A_{ES} - an algorithm based on [ES99] - to pack I' into at most m bins of size $(1+2\delta)(1+\delta) \leq (1+\varepsilon)$.

4. The placement of I' induces a placement of I in m bins of size $\leq (1+\varepsilon)$.

Theorem: The algorithm uses at most $OPT(I)$ bins. The sum of the sizes of the items in any bin does not exceed $1+\varepsilon$. The running time is $O(n^f)$ where $f = \frac{16M}{\varepsilon^2}$.

Summary

- A dual PTAS for CCBP
- A PTAS for CCMK:
  1. Use the guessing technique of Chekuri and Khanna to find:
     - The value, P, of an optimal packing
     - A set, U, of items with total profit P that has a feasible packing.
     - The partition of the bins into color-sets (items of which colors can be placed in each bin).
     - The partition of U to the bins;
  2. Show that the number of different bin types in any instance can be reduced to $O(\log n/\varepsilon)$.
- An FPTAS for the class-constrained 0-1 knapsack problem - using a *two-level* dynamic programming.
- APX-hardness proofs for GCCP - reductions from the 3-bounded 3-dim matching problem.