Solving robust bin packing problems with a branch-and-price approach

Xavier Schepler[a], André Rossi[b], Evgeny Gurevsky[c], Alexandre Dolgui[d]

[a] Data Lab, Generix Group,
[b]Université Paris Dauphine - PSL, LAMSADE UMR CNRS 7243
[c]Université de Nantes, LS2N UMR CNRS 6004
[d]IMT Atlantique, LS2N UMR CNRS 6004

February 2nd, 2024

Context and objectives

Context: standard 1D Bin Packing (*BP*) under uncertainty

BP

A set V of items of different sizes ($w_i > 0$) have to be assigned to a minimum number of identical bins of capacity $W \geq \max_{i \in V} w_i$

Model of uncertainty

A given subset $\widetilde{V} \subseteq V$ of item sizes have random deviations from their nominal values.

Application examples

- Hospital administration (Cardoen et al., 2010; Lamiri et al., 2008).
- Logistics capacity planning (Crainic et al., 2016).
- Assembly balancing (Boysen et al., 2007; Battaïa and Dolgui, 2013).

Considered robust BP variants

Stability radius calculated for the ℓ_∞ norm: $RBP_\infty(r)$

Each uncertain size can be increased by a maximum of r.

Relative resiliency: $RBP_{rr}(\alpha)$

Each uncertain size w_i can be increased by a maximum of αw_i

Stability radius calculated for the ℓ_1 norm: $RBP_1(r)$

The sum of the size increases over $\widetilde{V}$ is at most r.

Objectives of the study

- Solve to optimality these robust variants of BP.
- Investigate what protection against uncertainty is offered.
- Evaluate the cost of robustness.

Presentation plan

Context and objectives

Literature review

A Branch-and-Price algorithm

Numerical results

Conclusion and perspectives

Literature review

Column generation for *BP* under uncertainty

Robust machine availability problem (Song et al., 2018)

- Budgeted uncertainty (Bertsimas and Sim, 2004).
- Set-cover reformulation, Branch-and-Price algorithm (B&P)
  - Master problem: selection of filled bin patterns (set cover problem).
  - Pricing sub-problem : generation of filled bin patterns (knapsack-like problem).
- Can handle instances with up to 180 items with limited uncertainty within 1200 seconds.

Cutting stock with random demand (Alem et al., 2010)

- Two stage stochastic non-linear program.
- Column generation based heuristics.

Selected previous B&P algorithms

Focus on Ryan and Foster branching

Branch on a fractionally packed pair of items, so that they are:

1. in the same bin → standard Knapsack (*KS*) sub-problem,
2. in two different bins → *KS* with conflicts sub-problem.

Previous studies with this branching scheme

Study	Uncertainty	Node selection	*KS* with conflicts
Vance et al. (1994)	No	No conflicts first	MILP solver
Gamrath et al. (2016)	No	Integer feasibility, objective function	MILP solver
Song et al. (2018)	Budgeted	?	Binary decision diagrams

A Branch-and-Price algorithm

Compact formulations

Stability radius calculated for the ℓ_∞ norm: $RBP_\infty(r)$

$$
\begin{aligned}
\min \quad & \sum_{k\in K} y_k && \text{(1a)}\\
\text{s.t.} \quad & \sum_{k\in K} x_{ik} \geq 1, \quad \forall i \in V, && \text{(1b)}\\
& \sum_{i\in\widetilde{V}} (w_i + r)x_{ik} + \sum_{i\in V\setminus\widetilde{V}} w_i x_{ik} \leq W y_k, \quad \forall k \in K, && \text{(1c)}\\
& y_k \in \{0,1\}, \quad \forall k \in K, && \text{(1d)}\\
& x_{ik} \in \{0,1\}, \quad \forall i \in V, \quad \forall k \in K. && \text{(1e)}
\end{aligned}
$$

Relative resiliency: $RBP_{rr}(r)$

Constraints (1c) are replaced by:

$$\sum_{i\in\widetilde{V}} w_i(1+\alpha)x_{ik} + \sum_{i\in V\setminus\widetilde{V}} w_i x_{ik} \leq W y_k, \quad \forall k \in K.$$

Observation

Solving $RBP_\infty(r)$ or $RBP_{rr}(r)$ reduces to solving BP.

Compact formulations

Stability radius calculated for the ℓ_1 norm: $RBP_1(r)$

$$
\begin{aligned}
\min \quad & \sum_{k \in K} y_k && \text{(2a)}\\
\text{s.t.} \quad & \sum_{k \in K} x_{ik} \geq 1, \quad \forall i \in V, && \text{(2b)}\\
& x_{ik} \leq a_k, \quad \forall i \in \widetilde{V}, \quad \forall k \in K, && \text{(2c)}\\
& \sum_{i \in V} w_i x_{ik} + r a_k \leq W y_k, \quad \forall k \in K, && \text{(2d)}\\
& y_k \in \{0,1\}, \quad \forall k \in K, && \text{(2e)}\\
& a_k \in \{0,1\}, \quad \forall k \in K, && \text{(2f)}\\
& x_{ik} \in \{0,1\}, \quad \forall i \in V, \quad \forall k \in K. && \text{(2g)}
\end{aligned}
$$

Observation

- $RBP_1(r)$ is a generalization of BP.
- An algorithm for $RBP_1(r)$ can also solve BP, $RBP_\infty(r)$ and $RBP_{rr}(\alpha)$

Set-cover reformulation

Master problem: selection of filled bin patterns

$$\min \quad \sum_{B \in \mathcal{B}} \lambda_B \tag{3a}$$

$$\text{s.t.} \quad \sum_{B \in \mathcal{B}} x_i^B \lambda_B \geq 1, \quad \forall i \in V, \tag{3b}$$

$$\sum_{B \in \mathcal{B}} \lambda_B \leq K \tag{3c}$$

$$\lambda_B \in \{0,1\}, \quad \forall B \in \mathcal{B}. \tag{3d}$$

Pricing sub-problem: generation of filled bin patterns

$$\max \quad \sum_{i \in V} \pi_i x_i \tag{4a}$$

$$\text{s.t.} \quad x_i \leq a, \quad \forall i \in \widetilde{V}, \tag{4b}$$

$$\sum_{i \in V} w_i x_i + ra \leq W, \tag{4c}$$

$$x_i \in \{0,1\}, \quad \forall i \in V, \tag{4d}$$

$$a \in \{0,1\}. \tag{4e}$$

Sub-problems without conflict

Property

The pricing sub-problem without conflict can be solved in $\mathcal{O}(n \cdot W)$, by solving two instances of the 0-1 knapsack problem.

→ Two calls to the Combo algorithm (Martello et al., 1999):

- one for the case without uncertain-sized item (*i.e.* $a = 0$)
- one for the case with uncertain-sized items (*i.e.* $a = 1$)

Sub-problems with conflicts

Property

The pricing sub-problem with conflicts can be solved by addressing two instances of the 0-1 knapsack problem with conflicts.

$\rightarrow$ Two calls to a dedicated branch-and-bound algorithm (Sadykov and Vanderbeck, 2013).

Property (Dominance rule)

Let $C_i \subset V$ be the set of conflicting items with $i \in V$, i.e., for each $j \in C_i$ we have $x_i + x_j \leq 1$. If the following four conditions hold for some $j \in C_i$, then item j is "dominated" by item i, and can be removed from the instance of the robust 0-1 knapsack problem with conflicts:

1. $i \notin \widetilde{V}$ or $j \in \widetilde{V}$
2. $\pi_i \geq \pi_j$,
3. $w_i \leq w_j$,
4. $C_i \setminus \{j\} \subseteq C_j \setminus \{i\}$.

Primal heuristics

Adapted First-Fit Decreasing heuristic (FFD)

1. First pack uncertain-sized items with FFD:
   Bins capacity is $W - r$
2. Then pack remaining items with FFD:
   All extra bins have capacity W

Integer programming based heuristics

- Simple rounding,
- One-opt,
- ZI Round.

Numerical results

Settings

- The branch-and-price algorithm is implemented in C.
  - SCIP Optimization Suite 7.0.2 framework.
  - IBM CPLEX 12.8 simplex.
- Experiments were run on an Intel Core i7-6700HQ processor at 2.6 GHz with 4 GB of RAM
- Depth first search was in use, restarted from the best-bound node every 100 nodes.

Performance with BP

Set	#INSTANCE	#OPT	#OPT Vance	#OPT SCIP	#OPT Belov	#OPT Song
Falkenauer U	74	60	53	18	74*	-
Falkenauer T	80	79*	76	35	57	52
Scholl 1	323	323*	323*	244	323*	-
Scholl 2	244	197	204	67	244*	-
Scholl 3	10	10*	10*	0	10*	0
Wascher	17	15	6	0	17*	6
Schwerin 1	100	100*	100*	0	100*	100*
Schwerin 2	100	99	100*	0	100*	100*
Hard 28	28	26	11	7	28*	11
Random 50	165	165*	165*	165*	165*	-
Random 100	271	271*	271*	271*	271*	-
Random 200	359	359*	358	293	359*	-
Random 300	393	393*	387	155	393*	-
Random 400	425	425*	416	114	425*	-
Random 500	414	412	394	69	414*	-
Random 750	433	407	99	22	433*	-
Random 1000	441	282	62	0	441*	-
Total	3877	3623	3035	1460	3854*	-

Table 1: Comparative results with *BP*: number of instances solved in at most one minute per instance

Performance with $RBP_1(r)$

$\widetilde{V}$, (%)	r	Min. CPU, (s.)	Avg. CPU, (s.)	Max. CPU, (s.)	Std. CPU, (s.)	Avg. #BIN	#OPT, (%)	#SOL, (%) with GAP = 1
10%	All	0.09	52.28	616.78	161.45	39.69	92.3%	7.7%
	0.2W (290)	0.11	38.46	605.82	137.26	38.52	94.83%	5.17%
	0.3W (290)	0.1	40.26	616.78	141.86	39.44	94.14%	5.86%
	0.4W (290)	0.09	78.1	603.47	195.99	41.1	87.93%	12.07%
30%	All	0.03	55.19	601.78	168.88	44.2	91.26%	8.74%
	0.2W (290)	0.06	63.11	601.78	179.58	40.64	90%	10%
	0.3W (290)	0.08	38.08	600.21	140.7	43.71	94.14%	5.86%
	0.4W (290)	0.03	64.37	600.21	182.4	48.26	89.66%	10.34%
50%	All	0.02	64.59	600.88	182.2	48.63	89.66%	10.34%
	0.2W (290)	0.05	65.43	600.02	182.21	42.68	89.66%	10.34%
	0.3W (290)	0.02	62.53	600.88	179.7	47.76	90%	10%
	0.4W (290)	0.02	65.8	600.01	185.27	55.44	89.31%	10.69%
All	All	0.02	57.35	616.78	171.07	44.17	91.07%	8.93%

Table 2: Results of the proposed branch-and-price algorithm with $RBP_1(r)$, 870 runs

Cost of robustness

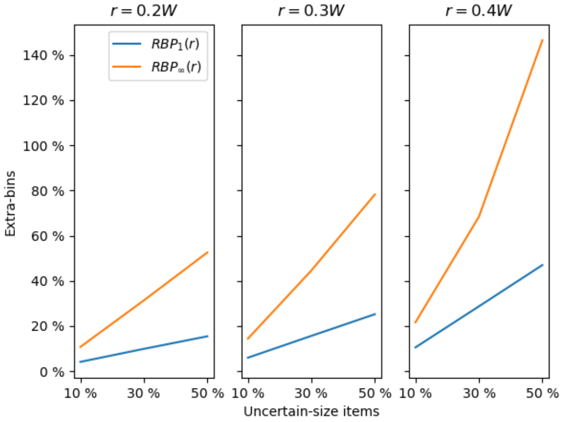

Figure 1: Increase of the extra cost for robustness, varying the percentage of uncertain-size items and r, for $RBP_1(r)$ and $RBP_\infty(r)$

Cost of robustness

α	$\widetilde{V}, (\%)$	Extra bins, (%)
0.2	10%	1.97%
	30%	5.80%
	50%	9.69%
0.3	10%	2.48%
	30%	8.56%
	50%	14.34%
0.4	10%	3.68%
	30%	11.54%
	50%	19.47%

Table 3: Extra cost for robustness increasing relative resiliency, for $RPB_{rr}(\alpha)$

Conclusion and perspectives

Conclusion

- Three robust variants of BP with items of uncertain size.
- One set-cover reformulation valid for the three robust variants.
- The proposed branch-and-price algorithm is able:
  - to obtain an optimal solution to every instance of $RBP_\infty(r)$ and $RBP_{rr}(\alpha)$, on average in less than three seconds per instance,
  - to obtain a proven optimal solution to 91% of $RBP_1(r)$ instances and to obtain a solution either optimal or requiring one extra bin to the remaining ones, on average in less than one minute.
- The extra cost for robustness ranged from 2% to 146% extra bins, depending on the robust approach, the percentage of uncertain items, and the value of r or α.
- We observed that standard BP provides solutions with a feasibility probability always close to 0, and that the proposed approaches provide the expected protections against uncertainty.

Perspectives

- Different robustness models:
  - For example uncertain bins, in which all item sizes can deviate.
  - Different kinds of uncertainties, affecting different subsets of items.
- Improvement of the branch-and-price algorithm:
  - Acceleration of the lower bound evaluation
  - Improvement of this lower bound.
  - Design of primal heuristics.