Lagrangian-based Online Stochastic Bin Packing

Varun Gupta
University of Chicago
varun.gupta@chicagobooth.edu

Ana Radovanovic
Google
anaradovanovic@google.com

ABSTRACT

Motivated by the problem of packing Virtual Machines on physical servers in the cloud, we study the problem of online stochastic bin packing under two settings – packing with permanent items, and packing under item departures.

In the setting with permanent items, we present the first truly distribution-oblivious bin packing heuristic that achieves $O(\sqrt{n})$ regret compared to OPT for all distributions. Our algorithm is essentially gradient descent on suitably defined Lagrangian relaxation of the bin packing Linear Program. We also prove guarantees of our heuristic against non *i.i.d.* input using a randomly delayed Lyapunov function to smoothen the input.

For the setting where items eventually depart, we are interested in minimizing the steady-state number of bins. Our algorithm extends as is to the case of item departures. Further, leveraging the Lagrangian approach, we generalize our algorithm to a setting where the processing time of an item is inflated by a certain known factor depending on the configuration it is packed in.

Categories and Subject Descriptors

F.2.2 [**Nonnumerical Algorithms and Problems**]: Sequencing and Scheduling; G.3 [**Probability and Statistics**]: Stochastic Processes

Keywords

Stochastic bin packing; Primal-Dual algorithm; Best Fit

1. MODEL NOTATION AND DEFINITIONS

There is a sequence of items that are packed online using algorithm A. Items can be of different types $j \in [J] = \{1, \ldots, J\}$. Each arrival is of type j i.i.d. with probability p_j. We abbreviate this distribution by F. The set $\mathcal{C}$ denotes the feasible set of configurations for the bins, where each $c \in \mathcal{C}$ is a multiset of items with x_{cj} representing the number of type j items in configuration c. Denote the empty configuration by $\emptyset$, a configuration with a single type j item by e_j, and $c \cup j$ and $c \backslash j$ denotes configurations with one more and one less item j respective (assuming they are in $\mathcal{C}$). We assume that we have an infinite number of bins available, and we use $N_c^A(n)$ to denote the number of bins in configuration c after nth item has been packed.

The performance metric that we want to minimize is the number of bins opened by a packing algorithm on distribution F, i.e.,

$$N_F^A(n) \triangleq \sum_{c \in \mathcal{C} \backslash \emptyset} N_c^A(n)$$

Departure Model: In Section 2.2 we extend our model to the case where items depart after spending some random time in the system. We assume that items arrive according to a Poisson process in time with rate $0 < \lambda < \infty$ and leave after some random time with finite mean $1/\mu_j$ and $M \triangleq \{1/\mu_1, \ldots, 1/\mu_J\}$, and finite higher moments.Additionally, an item packed in configuration c experiences a slowdown, so that the departure rate of a type j item which is currently in configuration c becomes $\mu_j s_{cj}$. We are interested in both the steady-state behavior (the number of bins used is parameterized by λ in this case), as well as the transient behavior (convergence rate to steady-state).

1.1 Review of Bin Packing Literature

Online bin packing with infinite collection of bins and permanent items: The Sum of Squares (SS) rule [2, 1] is the current state-of-the-art bin packing policy when item sizes and bin size B are integral. However, for certain class of distributions ("linear waste") SS achieves a constant factor more waste than OPT, and in [1] the authors fix this problem by essentially learning some information about the distribution. Our proposed policy achieves $O(\sqrt{n})$ regret for all distributions and is truly distribution-oblivious.

Bin packing with infinitely many bins and item departures: Stolyar [5] proposes a greedy packing heuristic that achieves $OPT \times (1+\epsilon)$ number of bins in steady state for arbitrarily small $\epsilon > 0$, which was improved to $OPT + o(\lambda)$ by Stolyar and Zhong [6]. Ghaderi et al. [3] propose a randomized Best Fit heuristic which also achieves $OPT \times (1+\epsilon)$ number of bin for arbitrary $\epsilon > 0$. However, due to specialized nature of the proposed algorithms, none of the above extend to the case of congestion-dependent slowdown.

2. ALGORITHMS

2.1 Bin Packing with Permanent Items

Given the item size distribution F, the optimal *bin rate* for

SIGMETRICS'15, June 15-19, 2015, Portland, OR, USA
Copyright © 2015 ACM 978-1-4503-3486-0/15/06 ...$15.00.
http://dx.doi.org/10.1145/2745844.2745897.

F (that is, the average number of bins used per item) can be computed by solving the following Linear Program (called the configuration LP):

$$b(F) = \min_{n_c} \sum_{c \in \mathcal{C}} n_c \qquad (\mathbf{P_{nodep}})$$

subject to

$$\forall c \in \mathcal{C} : n_c \geq 0$$

$$\forall j \in [J] : \sum_c n_c x_{cj} = P_j$$

The variable n_c denote the expected number of configuration c bins opened per item from F. (In a prior unpublished technical report [4], we had looked at the 1-d level packing problem via *flow LP* which is a more compact representation of the configuration LP in that setting, and gives slightly better constants in suboptimality gap).

The algorithm PD-exp is a straightforward Lagrangian minimization of P_{nodep} with exponential penalty function:

```
Algorithm PD-exp : At time $t$
    • Define configuration potentials:
          $V_c(t) = 1 - \kappa e^{-\epsilon(t) N_c(t)}$
      with $V_\emptyset \doteq 0$.
    • Place arriving item, say of type $j$, in configuration
      $c^*$ to create $c^* \cup j$, where:
          $c^* = \arg\ \min_c V_{c \cup j}(t) - V_c(t)$
```

Theorem 1. *For the PD-exp algorithm with* $\epsilon(t) = \sqrt{\frac{|\mathcal{C}|}{2(|\mathcal{C}|+t)}}$,

$$\mathbf{E}\Big[N_F^{PD}(n)\Big] \leq \mathbf{E}\Big[N_F^{OPT}(n)\Big] + \sqrt{8|\mathcal{C}|(n + |\mathcal{C}|)}$$

2.1.1 Guarantees against non-i.i.d. input

Adversarial Model: At time t, the adversary samples an item size S_t from distribution F_t that is a function of the history of samples generated by him. That is:

$$F_t = f_t(S_1, \ldots, S_{t-1}) \qquad (1)$$

Let $\{\mathcal{F}_t\}$ denote the filtration generated by $\{S_1, \ldots, S_t\}$ where as is usual $\mathcal{F}_0 = \{\emptyset, \Omega\}$.

Theorem 2. *For a given window size L, define the* smoothed arrival distribution *at time k conditioned on $\mathcal{F}_t$ as:*

$$\hat{F}_{k|t} \doteq \mathbf{E}\left[\frac{1}{L+1} \sum_{m=k}^{k+L} F_m \middle| \mathcal{F}_t\right]$$

Denote the optimal bin-rate of the bin packing LP for $\hat{F}_k$ by $\hat{b}_{k|t}$. For succinctness, $\hat{b}_k \doteq \hat{b}_{k|0}$.
If $L, \epsilon, \kappa, |\mathcal{C}|$ satisfy (i) $\epsilon L < 1$ and (ii) $L \leq \frac{1}{\epsilon} \log \frac{\kappa|\mathcal{C}|}{|\mathcal{C}|-1}$ then

$$\mathbf{E}\Big[N^{PD}(n)|\mathcal{F}_t\Big] \leq \sum_{k=1}^{n} \hat{b}_{k|t} + n\kappa\epsilon\left(\frac{2L+3}{4}\right) + \frac{|\mathcal{C}|\kappa}{\epsilon} + L.$$

Corollary 1. *If $L = \Theta(n^c)$ for arbitrary $0 \leq c < 1$, choosing $\epsilon = \Theta(n^{\frac{1+c}{2}})$:*

$$\mathbf{E}\Big[N^{PD}(n)\Big] \leq \sum_{k=1}^{n} \hat{b}_k + O(n^{\frac{1+c}{2}}).$$

2.2 Bin Packing with item departures

Our proposed heuristics extend almost as-is to the case of item departures. As before we define configuration potentials as:

$$V_c(t) = 1 - \frac{1}{\epsilon(t)} e^{-\epsilon(t) N_c(t)}$$

where we set $\epsilon(t) = \sqrt{\frac{|\mathcal{C}|}{2(|\mathcal{C}|+n(t))}}$, $n(t)$ denoting the total number of items in the system.

2.2.1 Packing with Heterogeneous slowdowns: The Proxy Dual method

In this section we consider the more general problem of heterogeneous slowdowns: the departure rate of a type j item in configuration c is given by $\mu_j s_{cj}$. Our convention will be $s_{e_j,j} = 1$, that is the slowdown of an item type when it is the only item in a bin is the benchmark of slowdown.

```
Algorithm PD$_{\mathbf{het}}$ :
    • Define configuration duals:
          $\alpha_c(t) = 1 - \kappa e^{-\epsilon(t) N_c(t)}$
      where $\kappa \geq m \times \max_{c, j \in c} \frac{1}{s_{cj}}$ ($m$ denotes the maxi-
      mum number of items in any configuration).
    • Define configuration potentials:
          $V_c(t) = (1 - \alpha_c(t)) - \sum_j x_{cj} s_{cj} (1 - \alpha_{e_j})$
      with $V_\emptyset \doteq 0$.
    • Place arriving item, say of type $j$, in configuration
      $c^*$ to create $c^* \cup j$, where:
          $c^* = \arg\ \min_c V_{c \cup j}(t) - V_c(t)$
```

Figure 1: Primal-Dual algorithm for heterogeneous slowdowns

3. REFERENCES

[1] János Csirik, David S. Johnson, Claire Kenyon, James B. Orlin, Peter W. Shor, and Richard R. Weber. On the sum-of-squares algorithm for bin packing. *J. ACM*, 53(1):1–65, 2006.

[2] János Csirik, David S. Johnson, Claire Kenyon, Peter W. Shor, and Richard R. Weber. A self organizing bin packing heuristic. In *ALENEX*, pages 246–265, London, UK, 1999. Springer-Verlag.

[3] Javad Ghaderi, Yuan Zhong, and R. Srikant. Asymptotic optimality of bestfit for stochastic bin packing. *SIGMETRICS Perform. Eval. Rev.*, 42(2):64–66, September 2014.

[4] Varun Gupta and Ana Radovanovic. Online stochastic bin packing. *CoRR*, abs/1211.2687, 2012.

[5] Alexander L. Stolyar. An infinite server system with general packing constraints. *Operations Research*, 61(5):1200–1217, 2013.

[6] Alexander L. Stolyar and Yuan Zhong. A large-scale service system with packing constraints: Minimizing the number of occupied servers. In *ACM SIGMETRICS*, pages 41–52, New York, NY, USA, 2013. ACM.