Approximation Algorithms for Stochastic Knapsack and Stochastic Orienteering Problems

Xinchang Xie and Zhiyong Zhao

April 16, 2016

1 Introduction

In this report, we will focus on approximation algorithms for the problems about stochastic knapsack and stochastic orienteering, under several different setups. The main results are from [1] and [2].

Basically, in the stochastic knapsack problem, we have a knapsack of capacity B and a set of potential jobs, while each job has a random size and a reward that might be correlated. Our goal is to maximize the expected total reward by proceeding jobs while not exceeding the capacity constraint, i.e., the total size of all finished jobs is no greater than total capacity B. However, the realization of size and reward for a scheduled job could only be observed after it is completely finished, and we can only proceed one job at any point of time. Constant-factor algorithms have been well studied for the case when the size and reward of a job are independent. While in [1], they consider jobs with correlated size and reward, and consider the situation in which jobs can be canceled at any time of its proceeding process. The authors propose an LP-relaxation and a corresponding randomized algorithm to give a constant-factor approximation to the true optimal. In the same paper, the authors also consider constant-factor approximation algorithm for budgeted multi-armed bandits problem when the martingale condition does not hold.

On the other hand, [2] considers stochastic orienteering problems, in which we are given a graph equipped with a metric. Each node has a job with a deterministic reward and a random size. The goal is to maximize total expected reward while satisfying a budget constraint, i.e., the total cost of traveling plus the total size of selected and finished jobs is no greater than budget B. Similar to the stochastic knapsack problem setup we just described, the realization of a job's size will only be observed after it is completely processed. In this paper, the authors propose a constant-factor approximation algorithm for the best non-adaptive policy, i.e., a fixed-routine policy that yields maximal expected total reward (among all fixed-routine policies). They also show that the adaptivity gap between the best non-adaptive policy and the best adaptive policy is small. Hence the approximation algorithm they proposed is also a good one for the optimal adaptive policy.

2 Correlated Knapsack and Non-Martingale Bandit Problems

2.1 Stochastic Knapsack Problems

Let us start with the stochastic knapsack problem with correlated size and reward of jobs. Suppose we have a total capacity B and n different jobs. For each job $i \in [1, n]$, let $(\pi_{i,t}, R_{i,t})$ denote the probability $\pi_{i,t}$ of job i has size t and reward $R_{i,t}$. Let $ER_{i,t} = \sum_{s \leq B-t} \pi_{i,s} R_{i,s}$ be the expected reward of scheduling job i when we have used t capacity. The LP-relaxation proposed in [1] is basically a global time-indexed linear program. Let $x_{i,t} \in [0, 1]$ be the probability of scheduling job i at time t (i.e., when an amount of t capacity has been used so far). Then the LP is,

$$
\begin{aligned}
\max \quad & \sum_{i,t} ER_{i,t} \cdot x_{i,t} \\
& \sum_{t} x_{i,t} \leq 1 \quad \forall i \\
& \sum_{i,t'} x_{i,t'} \cdot \mathbb{E}[\min(S_i, t)] \leq 2t \quad \forall t \\
& x_{i,t} \in [0, 1] \quad \forall i, t
\end{aligned}
$$

After solving the above LP and get the solution $\{x^*_{i,t}\}$, a corresponding randomize algorithm with 1/8-competitive-ratio runs as follows,

Algorithm 1 Stochastic Knapsack-No Cancellation

for $i = 1$ to n **do**
  assign a random start-time $D_i = t$ with probability $x^*_{i,t}/4$ and $D_i = \infty$ with probability $1 - \sum_t x^*_{i,t}/4$
end for
for $j = 1$ to n **do**
  Choose job i which has the jth smallest finite deadline
  if the total size of selected jobs does not exceed D_i **then**
    select job i
  end if
end for

Now let us move to the case when job cancellations are allowed at any point of time. This new feature of the problem can actually lead to a large gap between policies with and without job cancellations. Because of this large gap, the previous LP-relaxation can perform poorly and a new relaxation is needed here to address this issue. Now consider to separate all jobs into two groups, the jobs whose size is no more than $B/2$ and the ones with size greater than or equal to $B/2$. We first deal with the small-size jobs and set the reward of all large-size jobs to be 0. The idea is to introduce two groups of variables $v_{i,t}$ and $s_{i,t}$. Let $v_{i,t}$ denote the probability that Opt will process the job i for at least t time steps, while $s_{i,t}$

denotes the probability that Opt stops processing job i exactly at time t. Based on these variables, an LP-relaxation for small-size jobs has the following formulation,

$$
\begin{aligned}
\max \quad & \sum_{1\le t\le B/2} \sum_{i} v_{i,t} \cdot R_{i,t} \frac{\pi_{i,t}}{\sum_{t'\ge t} \pi_{i,t'}} \\
& v_{i,t} = s_{i,t} + v_{i,t+1} \\
& s_{i,t} \ge \frac{\pi_{i,t}}{\sum_{t'\ge t} \pi_{i,t'}} \cdot v_{i,t} \\
& \sum_{i,t} t \cdot s_{i,t} \le B \\
& v_{i,0} = 1, \quad v_{i,t}, s_{i,t} \in [0,1]
\end{aligned}
$$

By solving the above LP-relaxation for small-size jobs for the solution $\{s^*_{i,t}, v^*_{i,t}\}$, the following randomize algorithm gets a competitive ratio 1/8 to the small-size instantiation problem.

Algorithm 2 Stochastic Knapsack-Small Size

for $i = 1$ to n **do**
  totally ignore job i with probability $3/4$
end for
for $0 \le t \le B/2$ **do**
  cancel job i at this point of time with probability $\frac{s^*_{i,t}}{v^*_{i,t}} - \frac{\pi_{i,t}}{\sum_{t'\ge t}\pi_{i,t'}}$
  if the total size of selected jobs does not exceed D_i **then**
    select job i
  end if
end for

On the other hand, if we set the reward of all small-size jobs to be 0, it can be shown that there is actually an optimal solution that does not cancel any jobs. Hence the LP-relaxation and the corresponding randomize algorithm that we discussed earlier can be used here to get an approximation with 1/8 competitive ratio. Now combining these two approximations, i.e., for small-size instantiations and large-size instantiations, we can run one of them at random and get a constant-factor approximation solution to the stochastic knapsack problems with job cancellations.

2.2 Multi-armed Bandits

Now let us move on to multi-armed bandit problems with budget constraint. The LP-relaxation is based on global time-index. Let variable $z_{u,t}$ indicate the DM plays state $u \in S_i$ at time t, and let $w_{u,t}$ be the indicator of arm i first enters state u at time t. For a tree, we can write down the LP as follows,

$$
\begin{aligned}
\max \quad & \sum_{u,t} r_u \cdot z_{u,t} \\
& w_{u,t} = z_{\text{parent}(u),t-1} \cdot p_{\text{parent}(u),u} \\
& \sum_{t' \le t} w_{u,t'} \ge \sum_{t' \le t} z_{u,t'} \\
& \sum_{u \in S} z_{u,t} \le 1 \\
& w_{\rho_i,1} = 1
\end{aligned}
$$

Lemma: The value of this LP is at least OPt, the expected reward of an optimal adaptive strategy.

The hard part is the rounding algorithm. There are three phases. Phase I:Convex Decomposition. Given an opt of the LP, decompose the fractional solution into a convex combination of integral strategy. Phase II: Eliminate the small gaps.In this phase, we need to consider the contiguous tree. Phase III: Schedule the arms through a randomized algorithm.

For a general transition graph, a naive way is to consider all the minimal spanning trees; however, this incurs an exponential blowup of the state space. In [1], they only consider layered DAGs, since in this case the state space will only increase by a factor of the horizon B.

The LP constraint is similar to the case of trees.

$$
\begin{aligned}
\max \quad & \sum_{u,t} r_u \cdot z_{u,t} \\
& w_{u,t} = \sum_{v} z_{v,t-1} \cdot p_{v,u} \quad \forall t \in [2, B],\ u \in S \setminus \cup_i \rho_i,\ v \in S \\
& \sum_{t' \le t} w_{u,t'} \ge \sum_{t' \le t} z_{u,t'} \\
& \sum_{u \in S} z_{u,t} \le 1 \\
& w_{\rho_i,1} = 1
\end{aligned}
$$

We also need to decompose the fractional solution into convex combinations of integral strategies. There are similar steps of eliminating small gaps and how to schedule arms.

3 Stochastic Orienteering Problems

In this section, we will discuss approximation algorithms for stochastic orienteering problem. The basic setups are as follows, we are give a graph with distance metric (V, d). For each node v of the graph, there is a job located there with a fixed reward r_v and a random processing time/size S_v, whose distribution is known as π_v. Given a starting root ρ, our goal is to maximize the expected reward that we can collect along our path, under a total budget constraint, i.e., the total traveling cost and total processing cost do not exceed a budget B.

Both the reward and the travel distances are assumed to be integers. Also, the processing time/size can only be resolved after it is completely processed.

The basic idea to approximate the optimal solution is two parts. First we focus on non-adaptive policies, i.e., those policies that choose path without taking history into consideration. The second part is to show that the best non-adaptive policy performs reasonably well when compared with the best adaptive policy. Specifically, one can show that the non-adaptive-Opt is $O(\log\log B)$-order of the adaptive-Opt. The approximation algorithm is based on the $O(1)$-approximation algorithm **AlgKO**[1] for the KnapOrient Problem that will be defined shortly.

The (deterministic) knapsack orienteering problem is defined as follows, we have two budgets, the traveling budget L and the knapsack budget W. For each node v, there is a job located there with reward $\widehat{r}_v$ and size $\widehat{s}_v$. The goal is to find the optimal path that leads to the maximum reward $\sum_{v\in P}\widehat{r}_v$ while not exceeding the knapsack budget constraint, i.e., $\sum_{v\in P}\widehat{s}_v \le W$. Based on the deterministic knapsack orienteering problem, let us define KnapOrient instance $I_{\text{ko}}(W)$ with parameter W as $I_{\text{ko}}(W) := \text{KnapOrient}(V, d, \{(\widehat{s}_u), \widehat{r}_u\} : \forall u \in V)$ where $L = B - W$, $\widehat{s}_u = \mu_u(W/2)^2$ adn $\widehat{r}_u = r_u$ if $\mathbb{P}_{S_u\sim\pi_u}[S_u > W/2] \le 1/2$ and $\widehat{r}_u = 0$ otherwise. Then the approximation algorithm runs as follows,

---

Algorithm 3 Stochastic Orienteering Approximation Algorithm

---

for all $v \in V$ **do**
  set $R_v := r_v \cdot \mathbb{P}_{S_v\sim\pi_v}[S_v \le (B - d(\rho, v)]$ be the expected reward of the single-vertex tour to v,
end for
With probability 1/2, visit the vertex v with the highest R_v and exit,
for $W = B, B/2, B/4, \ldots, B/2^{\lceil \log B\rceil}, 0$ **do**
  if $W \neq 0$ **then**
    set $i = \log(B/W)$
  else
    set $i = \lceil \log B\rceil + 1$
  end if
  let P_i be the path returned by **AlgKO** on the valid KnapOrient instance $I_{\text{ko}}(W)$, and let $\widehat{R}_i$ be the reward of this solution P_i,
  let P_{i^*} be the solution among $\{P_i\}$ with the maximum reward $\widehat{R}_i$.
  sample each vertex in P_{i^*} independently w.p. 1/4 and visit these sampled vertices in order given by P_{i^*}.
end for

It is discussed and proved in [2] that the above algorithm is an $O(1)$-approximation to the best non-adaptive policy for the stochastic orienteering problem. The paper also discusses that the gap between non-adaptive policy and adaptive policy is reasonably small. To be exact, the ratio between the best non-adaptive Opt and the best adaptive Opt is $O(\log\log B)$.

---

[1] The approximation algorithms for KnapOrient problem have been well studied in the existing literatures. We can just use any one of those approximating algorithms among those proposed in related literatures.

[2] Here $\mu_u(Z)$ is the truncated mean function defined as $\mu_u(Z) := \mathbb{E}_{S_u\sim\pi_u}[\min(S_u, Z)]$

Hence, the approximation solution we get will be an $O(\log \log B)$ approximation to the optimal solution of the original stochastic orienteering problem.

Beyond the approximation algorithm to the basic version of this problem, the authors also extend the work to the case where we have both the reward and the size to be random and correlated to each other. They further extend the result to the setup that allows job cancellations. However, the details about these extensions will not be discussed in this review, and we refer the interested reader to [2] and the references therein.

4 Conclusion and Future Directions

In [1], they presented the first constant-factor approximations for the stochastic knapsack problem with cancellations and correlated size/reward pairs, and for the budgeted learning problem without the martingale property. In [2], they present a constant-factor approximation algorithm for the best non-adaptive policy for the stochastic orienteering problem. They also showed a small adaptivity gap.

There are some possible further directions for the stochastic knapsack problem. We list some in the following.

- The size of the knapsack is not constant. It means the knapsack size can decrease as time evolves. An example is that as time evolves, the person that deals with the knapsack will lose patient and he cannot take usual size items as the beginning. The size will decrease drastically.
- In the original LP, there are B^2 variables, one for each item and each time step. Can we decrease the number of variables in the LP for the SKP and achieve similar results?
- In the original paper, the variables are defined according to the beginning time for scheduling a job. Can we consider variables of the ending time of a scheduled job rather than the beginning time?
- Suppose there are k parallel knapsacks which can take items simultaneously. And we can also assume that when a machine a dealing with some portion of an item, the left portion of the item can be transferred to another knapsack. Is there a constant-factor approximation to the OPT in this situation?
- Can we construct some better algorithms for some specific distribution, like exponential distribution? Or the algorithm is not sensitive to the distributions?
- Can we solve the stochastic knapsack problem via primal-dual methods?
- In [1], the results are same for SKP with or without cancellation. Can we improve the approximation ratio for the SKP with cancellation?

References

[1] Anupam Gupta, Ravishankar Krishnaswamy, Marco Molinaro, and R Ravi. Approximation algorithms for correlated knapsacks and non-martingale bandits. In *Foundations of Computer Science (FOCS), 2011 IEEE 52nd Annual Symposium on*, pages 827–836. IEEE, 2011.

[2] Anupam Gupta, Ravishankar Krishnaswamy, Viswanath Nagarajan, and R Ravi. Approximation algorithms for stochastic orienteering. In *Proceedings of the twenty-third annual ACM-SIAM symposium on Discrete Algorithms*, pages 1522–1538. SIAM, 2012.