Maximizing Expected Utility for Stochastic Combinatorial Optimization Problems

Jian Li
Institute for Interdisciplinary Information Sciences
Tsinghua University, Beijing, P.R.China
Email: lapordge@gmail.com

Amol Deshpande
Department of Computer Science
University of Maryland, College Park, USA
Email: amol@cs.umd.edu

Abstract— We study the stochastic versions of a broad class of combinatorial problems where the weights of the elements in the input dataset are uncertain. The class of problems that we study includes shortest paths, minimum weight spanning trees, and minimum weight matchings over probabilistic graphs, and other combinatorial problems like knapsack. We observe that the expected value is inadequate in capturing different types of *risk-averse* or *risk-prone* behaviors, and instead we consider a more general objective which is to maximize the *expected utility* of the solution for some given utility function, rather than the expected weight (expected weight becomes a special case). We show that we can obtain a polynomial time approximation algorithm with *additive error* ϵ for any $\epsilon > 0$, if there is a pseudopolynomial time algorithm for the *exact* version of the problem (This is true for the problems mentioned above) and the maximum value of the utility function is bounded by a constant. Our result generalizes several prior results on stochastic shortest path, stochastic spanning tree, and stochastic knapsack. Our algorithm for utility maximization makes use of the separability of exponential utility and a technique to decompose a general utility function into exponential utility functions, which may be useful in other stochastic optimization problems.

1. Introduction

The most common approach to deal with optimization problems in presence of uncertainty is to optimize the expected value of the solution. However, expected value is inadequate in expressing diverse people's preferences towards decision-making under uncertain scenarios. In particular, it fails at capturing different *risk-averse* or *risk-prone* behaviors that are commonly observed. Consider the following simple example where we have two lotteries L_1 and L_2. In L_1, the player could win 1000 dollars with probability 1.0, while in L_2 the player could win 2000 dollars with probability 0.5 and 0 dollars otherwise. It is easy to see that both have the same expected payoff of 1000 dollars. However, many, if not most, people would treat L_1 and L_2 as two completely different choices. Specifically, a risk-averse player is likely to choose L_1 and a risk-prone player may prefer L_2 (Consider a gambler who would like to spend 1000 dollars to play double-or-nothing). A more involved but also more surprising example is the *St. Petersburg paradox* (see e.g., [35], [1]) which has been widely used in the economics literature as a criticism of expected value. These observations and criticisms have led researchers, especially in Economics, to study the problem from a more fundamental perspective and to directly maximize user satisfaction, often called *utility*. The uncertainty present in the problem instance naturally leads us to optimize the *expected utility*.

Let $\mathcal{F}$ be the set of feasible solutions to an optimization problem. Each solution $S \in \mathcal{F}$ is associated with a random weight $w(S)$. For instance, $\mathcal{F}$ could be a set of lotteries and $w(S)$ is the (random) payoff of lottery S. We model the risk awareness of a user by a utility function $\mu : \mathbb{R} \to \mathbb{R}$: the user obtains $\mu(x)$ units of utility if the outcome is x, i.e., $w(S) = x$. Formally, the *expected utility maximization principle* is simply stated as follows: the most desirable solution S is the one that maximizes the expected utility, i.e.,

$$S = \arg\max_{S' \in \mathcal{F}} \mathbb{E}[\mu(w(S'))]$$

Indeed, the expected utility theory is a branch of the utility theory that studies "betting preferences" of people with regard to uncertain outcomes (gambles). The theory was formally initiated by von Neumann and Morgenstern in 1940s [51], [20] who gave an axiomatization of the theory (known as *von Neumann-Morgenstern expected utility theorem*). The theory is well known to be versatile in expressing diverse risk-averse or risk-prone behaviors.

In this paper, we focus on the following broad class of combinatorial optimization problems. The deterministic version of the problem has the following form: we are given a ground set of elements $U = \{e_i\}_{i=1...n}$; each element e is associated with a weight w_e; each feasible solution is a subset of the elements satisfying some property. Let $\mathcal{F}$ denote the set of feasible solutions. The objective for the deterministic problem is to find a feasible solution S with the minimum total weight $w(S) = \sum_{e \in S} w_e$. We can see that many combinatorial problems such as shortest path, minimum spanning tree, and minimum weight matching belong to this class. In the stochastic version of the problem, the weight w_e of each element e is a nonnegative random variable. We assume all w_es are independent of each other. We use $p_e(.)$ to denote the probability density function for w_e (or probability mass function in discrete case). We are also given a utility function $\mu : \mathbb{R}^+ \to \mathbb{R}^+$ which maps a weight value to a utility value. By the expected utility maximization principle, our goal here is to find a

feasible solution $S \in \mathcal{F}$ that maximizes the expected utility, i.e., $\mathbb{E}[\mu(w(S))]$. We call this problem the *expected utility maximization (*EUM*)* problem.

Let us use the following toy example to illustrate the rationale behind EUM. There is a graph with two nodes s and t and two parallel links e_1 and e_2. Edge e_1 has a fixed length 1 while the length of e_2 is 0.9 with probability 0.9 and 1.9 with probability 0.1 (the expected value is also 1). We want to choose one edge to connect s and t. It is not hard to imagine that a risk-averse user would choose e_1 since e_2 may turn out to be a much larger value with a nontrivial probability. We can capture such behavior using the utility function (1) (defined in Section 1.1). Similarly, we can capture the risk-prone behavior by using, for example, the utility function $\mu(x) = \frac{1}{x+1}$. It is easy to see that e_1 maximizes the expected utility in the former case, and e_2 in the latter.

1.1. Our Contributions

We discuss in detail our result for EUM. We assume μ is part of the specification of the problem but not part of the input. Moreover, we assume $\lim_{x\to\infty} \mu(x) = 0$. This captures the fact that if the weight of solution is too large, it becomes almost useless for us. W.l.o.g. we can also assume $0 \le \mu(x) \le 1$ for $x \ge 0$, by scaling. We say a function $\widetilde{\mu}(x)$ is an ϵ-*approximation* of $\mu(x)$ if $|\widetilde{\mu}(x) - \mu(x)| \le \epsilon \forall x \ge 0$. For ease of exposition, we let $\widetilde{\mu}(x)$ be a complex function. Recall that a polynomial time approximation scheme (PTAS) is an algorithm which takes an instance of a minimization problem and a parameter ϵ and produces a solution whose cost is within a factor $1+\epsilon$ of the optimum, and the running time, for any fixed ϵ, is polynomial in the size of the input. We use A to denote the deterministic combinatorial optimization problem under consideration. The *exact version* of a problem A asks the question whether there is a feasible solution of A with weight exactly equal to a given number K. We say an algorithm runs in *pseudopolynomial time* for the exact version of A if the running time is polynomial in n and K. Our first main theorem is the following.

Theorem 1: Assume that there is a pseudopolynomial algorithm for the exact version of A. Further assume that given any $\epsilon > 0$, we can find an ϵ-approximation of the utility function μ as $\widetilde{\mu}(x) = \sum_{k=1}^{L} c_k \phi_k^x$, where L is a constant and $|\phi_k| \le 1 \forall k$; ϕ_k may be complex numbers. Then, there is an algorithm that runs in time $(n/\epsilon)^{O(L)}$ that approximates EUM(A) with an *additive* error $O(\epsilon)$. If the optimal expected utility is $\Theta(1)$, we obtain a PTAS.

For many combinatorial problems, a pseudopolynomial algorithm for the exact version is known. Examples include shortest path, spanning tree, matching and knapsack. Hence, the only task left is to find a short exponential sum that ϵ-approximates μ. For this purpose, we adopt the Fourier series technique. However, the technique cannot be used directly since it works only for periodic functions with bounded periodicities. In order to get a good approximation for $x \in [0, \infty)$, we leverage the fact that $\lim_{x\to\infty} \mu(x) = 0$ and develop a general framework that uses the Fourier series decomposition as a subroutine. Generally speaking, such an approximation is only possible if the function is "well behaved", i.e., it satisfies some continuity or smoothness conditions. In particular, we prove Theorem 2. We say that the utility function μ satisfies the α-Hölder condition if $|\mu(x) - \mu(y)| \le C\,|x-y|^{\alpha}$, for some constant C and some constant α.

Theorem 2: If μ satisfies the α-Hölder condition for some constant $\alpha > 1/2$, then, for any $\epsilon > 0$, we can obtain an exponential sum with $O(poly(\frac{1}{\epsilon}))$ terms which is an ϵ-approximation of μ for $x \ge 0$.

Consider the utility function

$$\widetilde{\chi}(x) = \begin{cases} 1 & x \in [0,1] \\ -\frac{x}{\delta} + \frac{1}{\delta} + 1 & x \in [1, 1+\delta] \\ 0 & x > 1+\delta \end{cases} \tag{1}$$

where $\delta > 0$ is a small constant (See Figure 1(1)). We can verify that $\widetilde{\chi}$ satisfies 1-Hölder condition with $C = \frac{1}{\delta}$. Therefore, Theorem 2 is applicable. This example is interesting since it can be viewed as a continuous variant of the threshold function $\chi(x) = \begin{cases} 1 & x \in [0,1] \\ 0 & x > 1 \end{cases}$, for which maximizing the expected utility is equivalent to maximizing $\Pr(w(S) \le 1)$. This special case has been considered several times in literature for various problems including stochastic shortest path [41], stochastic spanning tree [28], [22], stochastic knapsack [23] and some other stochastic problems [3], [39].

It is interesting to compare our result with the result for the stochastic shortest path problem considered by Nikolova et al. [41], [39]. In [41], they show that there is an exact $O(n^{\log n})$ time algorithm for maximizing the probability that the length of the path is at most 1, i.e., $\Pr(w(S) \le 1)$, assuming all edges are normally distributed and there is a path with its mean at most 1. Later, Nikolova [39] extends the result to an FPTAS for any problem under the same assumptions, if the deterministic version of the problem has a polynomial time exact algorithm. We can see that under such assumptions, the optimal probability is at least $1/2$.[1] Therefore, provided the same assumption and further assuming that $\Pr(w_e < 0)$ is miniscule,[2] our algorithm is a PTAS for the continuous variant of the problem. Indeed, we can translate this result to a bi-criterion approximation result of the following form: for any fixed $\delta, \epsilon > 0$, we can

[1]The sum of multiple Gaussians is also a Gaussian. Hence, if we assume the mean of the length of a path (which is a Gaussian) is at most 1, the probability that the length of the path is at most 1 is at least $1/2$.

[2]Our technique can only handle distributions with positive supports. Thus, we have to assume that the probability that a negative value appears is miniscule and can be safely ignored.

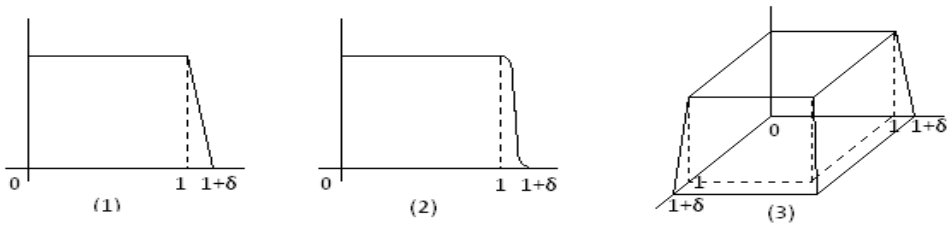

Figure 1. (1) The utility function $\widetilde{\chi}(x)$, a continuous variant of the threshold function $\chi(x)$; (2) A smoother variant of $\chi(x)$; (3) The utility function $\widetilde{\chi}_2(x)$, a continuous variant of the 2-d threshold function $\chi_2(x)$.

find in polynomial time a solution S such that

$$\Pr(w(S) \leq 1+\delta) \geq (1-\epsilon)\Pr(w(S^*) \leq 1).$$

where S^* is the optimal solution (Corollary 2). We note that such a bi-criterion approximation was only known for exponentially distributed edges before [41].

Let us consider another application of our results to the stochastic knapsack problems defined in [23]. Given a set U of independent random variables $\{x_1, \ldots, x_n\}$, with associated profits $\{v_1, \ldots, v_n\}$ and an overflow probability γ, we are asked to pick a subset S of U such that $\Pr(\sum_{i\in S} x_i \geq 1) \leq \gamma$ and the total profit $\sum_{i\in S} v_i$ is maximized. Goel and Indyk [23] showed that, for any $\epsilon > 0$, there is a polynomial time algorithm that can find a solution S with the profit as least the optimum and $\Pr(\sum_{i\in S} x_i \geq 1+\epsilon) \leq \gamma(1+\epsilon)$ for exponentially distributed variables. They also gave a quasi-polynomial time approximation scheme for Bernoulli distributed random variables. Quite recently, in parallel with our work, Bhalgat et al. [12] obtained the same result for arbitrary distributions under the assumption that $\gamma = \Theta(1)$. Their technique is based on discretizing the distributions and is quite involved. Our result, applied to stochastic knapsack, matches that of Bhalgat et al. We remark that our algorithm is much simpler and has a much better running time (Theorem 5). Despite a little loss in the approximation guarantees in some cases, our technique can be applied to almost all positive probability distributions, and a much richer class of utility functions.

Equally importantly, we can extend our basic approximation scheme to handle generalizations such as multiple utility functions and multidimensional weights. Interesting applications of these extensions include generalizations of stochastic knapsack, such as *stochastic multiple knapsack* (Theorem 8) and *stochastic multidimensional knapsack (stochastic packing)* (Theorem 9).

1.2. Related Work

In recent years stochastic optimization problems have drawn much attention from the computer science community and stochastic versions of many classical combinatorial optimization problems have been studied. In particular, a significant portion of the efforts has been devoted to the two-stage stochastic optimization problem. In such a problem, in a first stage, we are given probabilistic information about the input but the cost of selecting an item is low; in a second stage, the actual input is revealed but the costs for the elements are higher. We are asked to make decision after each stage and minimize the expected cost. Some general techniques have been developed [26], [46]. We refer interested reader to [50] for a comprehensive survey. Another widely studied type of problems considers designing *adaptive* probing policies for stochastic optimization problems where the existence or the exact weight of an element can be only known upon a probe. There is typically a budget for the number of probes (see e.g., [25], [17]), or we require an irrevocable decision whether to include the probed element in the solution right after the probe (see e.g., [19], [15], [5], [18], [12]). However, most of those works focus on optimizing the expected value of the solution. There is also sporadic work on optimizing the overflow probability or some other objectives subject to the overflow probability constraints. In particular, a few recent works have explicitly motivated such objectives as a way to capture the risk-averse type of behaviors [3], [39], [49]. Besides those works, there has been little work on optimizing more general utility functions for combinatorial stochastic optimization problems from an approximation algorithms perspective.

The most related work to ours is the stochastic shortest path problem (Stoch-SP), which was also the initial motivation for this work. The problem has been studied extensively for several special utility functions in operation research community. Sigal et al. [47] studied the problem of finding the path with greatest probability of being the shortest path. Loui [34] showed that Stoch-SP reduces to the shortest path (and sometimes longest path) problem if the utility function is linear or exponential. Nikolova et al. [40] identified more specific utility and distribution combinations that can be solved optimally in polynomial time. Much work considered dealing with more general utility functions, such as piecewise linear or concave functions, e.g., [37], [38], [7]. However, these algorithms are essentially heuristics and the worst case running times are still exponential. Nikolova et al. [41] studied the problem of maximizing the probability that the length of the chosen path is less than some given parameter. Besides the result we mentioned before, they also considered Poisson and exponential distributions. Despite much effort on this problem, no algorithm is known to run in polynomial time and have provable performance guarantees,

especially for more general utility functions or more general distributions. This is perhaps because the hardness comes from different sources, as also noted in [41]: the shortest path selection per se is combinatorial; the distribution of the length of a path is the convolution of the distributions of its edges; the objective is nonlinear; to list a few.

Kleinberg et al. [30] first considered the stochastic knapsack problem with Bernoulli-type distributions and provided a polynomial-time $O(\log 1/\gamma)$ approximation where γ is the given overflow probability. For item sizes with exponential distributions, Goel and Indyk [23] provided a bi-criterion PTAS, and for Bernoulli-distributed items they gave a quasi-polynomial approximation scheme. Chekuri and Khanna [14] pointed out that a PTAS can be obtained for the Bernoulli case using their techniques for the multiple knapsack problem. Goyal and Ravi [24] showed a PTAS for Gaussian distributed sizes. Quite recently, Bhalgat, Goel and Khanna [12] developed a general discretizaton technique that reduces the distributions to a small number of equivalent classes which we can efficiently enumerate for both adaptive and nonadaptive versions of stochastic knapsack. They used this technique to obtain improved results for several variants of stochastic knapsack, notably a bi-criterion PTAS for the *adaptive version* of the problem. Dean at al. [19] gave the first constant approximation for the adaptive version of stochastic knapsack. The adaptive version of stochastic multidimensional knapsack (or equivalently stochastic packing) has been considered in [18], [12] where constant approximations and a bi-criterion PTAS were developed.

This work is partially inspired by our prior work on top-k and other queries over probabilistic datasets [31], [33]. In fact, we can show that both the *consensus answers* proposed in [31] and the *parameterized ranking functions* proposed in [33] follow the expected utility maximization principle where the utility functions are materialized as distance metrics for the former and the weight functions for the latter. Our technique for approximating the utility functions is also similar to the approximation scheme used in [33] in spirit. However, no performance guarantees are provided in that work.

There is a large volume of work on approximating functions using short exponential sums over a bounded domain, e.g., [42], [8], [9], [10]. Some works also consider using linear combinations of Gaussians or other *kernels* to approximate functions with finite support over the entire real axis $(-\infty, +\infty)$ [16]. This is however impossible using exponentials since α^x is either periodic (if $|\alpha| = 1$) or approaches to infinity when $x \to +\infty$ or $x \to -\infty$ (if $|\alpha| \neq 1$).

2. ALGORITHM

We first note that EUM is #P-hard in general since the problem of computing the overflow probability of a set of items with Bernoulli distributions, a very special case of our problem, is #P-hard [30].

Our approach is very simple. We first observe that the problem is easy if the utility function is an exponential function. We approximate the utility function $\mu(x)$ by a short exponential sum, i.e., $\sum_{i=1}^{L} c_i \phi_i^x$ with L being a constant (c_i and ϕ_i may be complex numbers). Hence, $\mathbb{E}[\mu(w(S))]$ can be approximated by $\sum_{i=1}^{L} c_i \mathbb{E}[\phi_i^{w(S)}]$. Then, we consider the following multi-criterion version of the problem with L objectives $\{\mathbb{E}[\phi_i^{w(S)}]\}_{i=1,\ldots,L}$: given L complex numbers $v_1, \ldots, v_L$, we want to find a solution S such that $\mathbb{E}[\phi_i^{w(S)}] \approx v_i$ for $i = 1, \ldots, L$. We achieve this by utilizing the pseudopolynomial time algorithm for the exact version of the problem. We argue that we only need to consider a polynomial number of $v_1, \ldots, v_L$ combinations (which we call *configurations*) to find out the approximate optimum. In Section 2.1, we show how to solve the multi-criterion problem provided that a short exponential sum approximation of μ is given. In particular, we prove Theorem 1. Then, we show how to approximate μ by a short exponential sum by proving Theorem 2 in Section 2.2 and Section 2.3.

Let us first consider the exponential utility function $\mu(x) = \alpha^x$ for any $\alpha \in \mathbb{C}$. Fix an arbitrary solution S and $\alpha > 0$. Due to the independence of the elements, we can see that

$$\mathbb{E}[\alpha^{w(S)}] = \mathbb{E}[\alpha^{\sum_{e \in S} w_e}] = \mathbb{E}[\prod_{e \in S} \alpha^{w_e}] = \prod_{e \in S} \mathbb{E}[\alpha^{w_e}]$$

Taking log on both sides, we get $\log \mathbb{E}[\alpha^{w(S)}] = \sum_{e \in S} \log \mathbb{E}[\alpha^{w_e}]$. If α is a positive real number and $\mathbb{E}[\alpha^{w_e}] \leq 1$ (or equivalently, $-\log \mathbb{E}[\alpha^{w_e}] \geq 0$), this reduces to the deterministic optimization problem.

We still need to show how to compute $\mathbb{E}[\alpha^{w_e}]$. If w_e is a discrete random variable with a polynomial size support, we can easily compute $\mathbb{E}[\alpha^{w_e}]$ in polynomial time. If w_e has an infinite discrete or continuous support, we can not compute $\mathbb{E}[\alpha^{w_e}]$ directly and may need to approximate it. We leave the discussion of the issue to the full version of the paper [32].

2.1. Proof of Theorem 1

Now, we prove Theorem 1. We start with some notations. We use $|c|$ and $\arg(c)$ to denote the absolute value and the argument of the complex number c, respectively. In other words, $c = |c|(\cos(\arg(c)) + i \sin(\arg(c))) = |c| e^{i \arg(c)}$. We always require $\arg(c) \in [0, 2\pi)$ for any $c \in \mathbb{C}$. Recall that we say the exponential sum $\sum_{i=1}^{L} c_i \phi_i^x$ is an *ϵ-approximation* for $\mu(x)$ if the following holds:

$$|\mu(x) - \sum_{i=1}^{L} c_i \phi_i^x| \leq \epsilon \quad \forall x \geq 0$$

We first show that if the utility function can be decomposed exactly into a short exponential sum, we can approximate the optimal expected utility well.

Theorem 3: Assume $\widetilde{\mu}(x) = \sum_{k=1}^{L} c_k \phi_k^x$ is the utility function where $|\phi_k| \leq 1$ for $1 \leq k \leq L$. We also assume that there is a pseudopolynomial algorithm for the exact version of A. Then, for any $\epsilon > 0$, there is an algorithm that runs in time $(n/\epsilon)^{O(L)}$ and finds a solution S such that $|\mathbb{E}[\widetilde{\mu}(w(S))] - \mathbb{E}[\widetilde{\mu}(w(\widetilde{S}))]| < \epsilon$ where $\widetilde{S} = \arg\max_{S'} |\mathbb{E}[\widetilde{\mu}(w(S'))|$.

We use the scaling and rounding technique that has been used often in multi-criterion optimization problems (e.g., [45], [43]). Since our objective function is not additive and not monotone, the general results for multi-criterion optimization [43], [36], [45], [2] do not directly apply here. We briefly sketch our algorithm. Let $\gamma = \delta = \frac{\epsilon}{Ln}$. For each $e \in U$, we associate it with a $2L$ dimensional integer vector $\langle a_1(e), b_1(e), \ldots, a_L(e), b_L(e)\rangle$ where $a_i(e) = \lfloor \frac{-\ln|\mathbb{E}[\phi_i^{w_e}]|}{\gamma} \rfloor$ and $b_i(e) = \lfloor \frac{\arg(\mathbb{E}[\phi_i^{w_e}])}{\delta} \rfloor$. $a_i(e)$ and $b_i(e)$ are the scaled and rounded versions of $-\ln|\mathbb{E}[\phi_i^{w_e}]|$ and $\arg(\mathbb{E}[\phi_i^{w_e}])$, respectively. Since $|\phi_i| \leq 1$, we can see that $a_i(e) \geq 0$ for any $e \in U$. We maintain $(JK)^L$ configurations where $J = \lceil \frac{-\ln(\epsilon/L)}{\gamma} \rceil$ and $K = \lceil \frac{2\pi n}{\delta} \rceil$. The number of configurations is $(n/\epsilon)^{O(L)}$. Each configuration $\sigma(\mathbf{a})$ is indexed by a $2L$-dimensional vector $\mathbf{a} = \langle \alpha_1, \beta_1, \ldots, \alpha_L, \beta_L \rangle$ where $1 \leq \alpha_i \leq J$ and $1 \leq \beta_i \leq K$ for $i = 1, \ldots, L$. In other words, the configurations are $\sigma(\langle 1, 1, \ldots, 1, 1\rangle), \ldots, \sigma(\langle J, K, \ldots, J, K\rangle))$. For vector $\mathbf{a} = \langle \alpha_1, \beta_1, \ldots, \alpha_L, \beta_L \rangle$, configuration $\sigma(\mathbf{a}) = 1$ if and only if there is a feasible solution $S \in \mathcal{F}$ such that for all $j = 1, \ldots, L$, $\beta_j = \sum_{e \in S} b_j(e)$, and $\alpha_j = \min(J, \sum_{e \in S} a_j(e))$. Otherwise, $\sigma(\mathbf{a}) = 0$. Lemma 1 tells us the expected utility for the rounded instance is close to the true value of the expected utility. Lemma 2 shows we can compute those configurations in polynomial time. The missing proofs can be found in the full version of the paper [32].

Lemma 1: For vector $\mathbf{a} = \langle \alpha_1, \beta_1, \ldots, \alpha_L, \beta_L \rangle$, $\sigma_v(\mathbf{a}) = 1$ if and only if there is a solution S such that $|\mathbb{E}[\widetilde{\mu}(w(S))] - \sum_{k=1}^{L} c_k e^{-\alpha_k \gamma + i\beta_k \delta}| \leq O(\epsilon)$.

Lemma 2: Suppose there is a pseudopolynomial time algorithm for the exact version of A, which runs in time polynomial in n and t (t is the maximum integer in the instance of A). Then, we can compute the values for these configurations in time $(\frac{n}{\epsilon})^{O(L)}$.

From these two lemmas, Theorem 3 follows immediately. Theorem 1 can be readily obtained from Theorem 3 and the fact $\widetilde{\mu}$ is an ϵ-approximation of μ.

Proof of Theorem 1: Suppose S is our solution and S^* is the optimal solution for utility function μ. From Theorem 3, we know that $|\mathbb{E}[\widetilde{\mu}(w(S))] \geq \mathbb{E}[\widetilde{\mu}(w(S^*))]| - \epsilon$. Since $\widetilde{\mu}$ is an ϵ-approximation of μ, we can see that

$$\begin{aligned}\left|\mathbb{E}[\mu(w(S))] - \mathbb{E}[\widetilde{\mu}(w(S))]\right| &= \left|\int (\mu(x) - \widetilde{\mu}(x)) p_S(x) \mathrm{d}x\right| \\ &\leq \left|\int \epsilon p_S(x) \mathrm{d}x\right| \leq \epsilon\end{aligned}$$

for any solution S, where p_S is the probability density function of S. Therefore, we have

$$\begin{aligned}|\mathbb{E}[\mu(w(S))]| &\geq |\mathbb{E}[\widetilde{\mu}(w(S))]| - \epsilon \geq |\mathbb{E}[\widetilde{\mu}(w(S^*))]| - 2\epsilon \\ &\geq |\mathbb{E}[\mu(w(S^*))]| - 3\epsilon\end{aligned}$$

The proof is complete. □

Algorithm: ESUM

1. Initially, we slightly change function $\mu(x)$ to a new function $\widehat{\mu}(x)$ as follows: We require $\widehat{\mu}(x)$ is a "smooth " function in $[-2T_\epsilon, 2T_\epsilon]$ such that $\widehat{\mu}(x) = \mu(x)$ for all $x \in [0, T_\epsilon]$; $\widehat{\mu}(x) = 0$ for $|x| \geq 2T_\epsilon$. We choose $\widehat{\mu}(x)$ in $[-2T_\epsilon, 0]$ and $[T_\epsilon, 2T_\epsilon]$ such that $\widehat{\mu}(x)$ is smooth. We do not specify the exact smoothness requirements now since they may depend on the choice of AP. Note that there may be many ways to interpolate μ such that the above conditions are satisfied. The only properties we need are: (1) $\widehat{\mu}$ is amenable to algorithm AP; (2) $|\widehat{\mu}(x) - \mu(x)| \leq \epsilon \quad \forall x \geq 0$.
2. We apply AP to $f(x) = \eta^x \widehat{\mu}(x)$ over domain $[-hT_\epsilon, hT_\epsilon]$ ($\eta \geq 1$ and $h \geq 2$ are constants to be determined later). Suppose the resulting exponential sum $\widehat{f}(x) = \sum_{i=1}^{L} c_i \phi_i^x$ which is an ϵ-approximation of f on $[-hT_\epsilon, hT_\epsilon]$.
3. Let $\widetilde{\mu}(x) = \sum_{i=1}^{L} c_i (\frac{\phi_i}{\eta})^x$, which is our final approximation of $\mu(x)$ on $[0, \infty)$.

2.2. Approximating the Utility Function

In this subsection, we discuss the issue of approximating μ. In particular, we develop a generic algorithm that takes as a subroutine an algorithm AP for approximating functions in a bounded interval domain, and approximates $\mu(x)$ in the infinite domain $[0, +\infty)$. In the next subsection, we use the Fourier series expansion as the choice of AP and show that important classes of utility functions can be approximated well.

There are many works on approximating functions using short exponential sums, e.g., the Fourier decomposition approach [48], Prony's method [42], and many others [8], [9]. However, their approximations are done over a finite interval domain, say $[-\pi, \pi]$ or over a finite number of discrete points. No error bound can be guaranteed outside the domain. Our algorithm is a generic procedure that turns an algorithm that can approximate functions over $[-\pi, \pi]$ into one that can approximate our utility function μ over $[0, +\infty)$, by utilizing the fact that $\lim_{x \to \infty} \mu(x) = 0$.

Since $\lim_{x \to \infty} \mu(x) = 0$, for any ϵ, there exist a point T_ϵ such that $\mu(x) \leq \epsilon \quad \forall x > T_\epsilon$. Since we assume the utility function μ is specified as a part of the problem but not a part of the input instance, T_ϵ is a constant for any constant ϵ. We also assume there is an algorithm AP that,

for any function f (under some conditions specified later), can produce an exponential sum $\widehat{f}(x) = \sum_{i=1}^{L} c_i \phi_i^x$ which is an ϵ-approximation of $f(x)$ in $[-\pi, \pi]$ such that $|\phi_i| \leq 1$ and L depends only on ϵ and f. In fact, we can assume w.l.o.g. that AP can approximate $f(x)$ over $[-B, B]$ for any $B = O(1)$. This is because we can apply AP to the scaled version $g(x) = f(x \cdot \frac{B}{\pi})$ (which is defined on $[-\pi, \pi]$) and then scale the obtained approximation $\widehat{g}(x)$ back to $[-B, B]$, i.e., the final approximation is $\widehat{f}(x) = \widehat{g}(\frac{\pi}{B} \cdot x)$. Scaling a function by a constant factor $\frac{B}{\pi}$ typically does not affect the smoothness of f in any essential way and we can still apply AP. Recall that our goal is to produce an exponential sum that is an ϵ-approximation for $\mu(x)$ in $[0, +\infty)$. We denote this procedure by ESUM.

By setting $\eta = 2$ and

$$h \geq \frac{\log(\sum_{i=1}^{L} |c_i|/\epsilon)}{T_\epsilon}, \tag{2}$$

we can show the following theorem.

Lemma 3: $\widetilde{\mu}(x)$ is a 2ϵ-approximation of $\mu(x)$.

Proof: We know that $|\widehat{f}(x) - f(x)| \leq \epsilon$ for $x \in [0, hT_\epsilon]$. Therefore, we have that

$$|\widetilde{\mu}(x) - \widehat{\mu}(x)| = |\frac{\widehat{f}(x)}{\eta^x} - \frac{f(x)}{\eta^x}| \leq \frac{\epsilon}{\eta^x} \leq \epsilon.$$

Combining with $|\widehat{\mu}(x) - \mu(x)| \leq \epsilon$, we obtain $|\widetilde{\mu}(x) - \mu(x)| \leq 2\epsilon$ for $x \in [0, hT_\epsilon]$. For $x > hT_\epsilon$, we can see

$$|\widetilde{\mu}(x)| = |\sum_{i=1}^{L} c_i (\frac{\phi_i}{\eta})^x| \leq \sum_{i=1}^{L} |c_i (\frac{\phi_i}{\eta})^x|$$
$$\leq \frac{1}{2^x} \sum_{i=1}^{L} |c_i| \leq \frac{1}{2^{hT_\epsilon}} \sum_{i=1}^{L} |c_i| \leq \epsilon$$

Since $\mu(x) < \epsilon$ for $x > hT_\epsilon$, the proof is complete. □ ■

Remark: Since we do not know c_i before applying AP, we need to set h to be a constant (only depending on μ and ϵ) such that (2) is always satisfied. In particular, we need to provide an upper bound for $\sum_{i=1}^{L} |c_i|$. In the next subsection, we use the Fourier series decomposition as the choice for AP, which allows us to provide such a bound for a large class of functions.

2.3. A Choice of AP*: The Fourier Series Approach*

Now, we discuss the choice of algorithm AP and the conditions that $f(x)$ needs to satisfy so that it is possible to approximate $f(x)$ by a short exponential sum in a bounded interval. In fact, if we know in advance that there is a short exponential sum that can approximate f, we can use the algorithms developed in [9], [10] (for continuous case) and [8] (for discrete case). However, those works do not provide an easy characterization of the class of functions. From now on, we restrict ourselves to the classic Fourier series technique, which has been studied extensively and allows such characterizations.

Consider the partial sum of the Fourier series of the function $f(x)$: $(S_N f)(x) = \sum_{k=-N}^{N} c_k e^{ikx}$ where the Fourier coefficient $c_k = \frac{1}{2\pi} \int_{-\pi}^{\pi} f(x) e^{-ikx} \mathrm{d}x$. It has $L = 2N + 1$ terms. Since $f(x)$ is a real function, we have $c_k = c_{-k}$ and the partial sum is also real. We are interested in the question under which conditions does the function $S_N f$ converge to f (as N increases) and what is convergence rate? Roughly speaking, the more "smooth" f is, the faster $S_N f$ converges to f. In general, this question is extremely intricate and deep and is one of the central topics in the area of harmonic analysis. In the following, we give one classic result about the convergence of Fourier series and show how to use it in our problem. Then we provide a few concrete examples.

We say f satisfies the *α-Hölder condition* if $|f(x) - f(y)| \leq C\,|x - y|^\alpha$, for some constant C and $\alpha > 0$ and any x and y. The constant C is called the *Hölder coefficient* of f, also denoted as $|f|_{C^{0,\alpha}}$. We say f is *C-Lipschitz* if f satisfies 1-Hölder condition with coefficient C.

We need the following classic result of Jackson.

Theorem 4: (See e.g., [44]) If f satisfies the α-Hölder condition, it holds that

$$|f(x) - (S_N f)(x)| \leq O\Big(\frac{|f|_{C^{0,\alpha}} \ln N}{N^\alpha}\Big).$$

For later development, we need a few simple lemmas. The proofs are straightforward and thus omitted here.

Lemma 4: Suppose $f : [a, c] \to \mathbb{R}$ is a continuous function which consists of two pieces $f_1 : [a, b] \to \mathbb{R}$ and $f_2 : [b, c] \to \mathbb{R}$. If both f_1 and f_2 satisfy the α-Hölder condition with Hölder coefficient C, then $|f|_{C^{0,\alpha}} \leq 2C$.

Lemma 5: Suppose $f : [a, c] \to \mathbb{R}$ is a continuous function satisfying the α-Hölder condition with Hölder coefficient C. Then, for $g(x) = f(hx)$ for some constant h, we have $|g|_{C^{0,\alpha}} \leq Ch^\alpha$.

Using Theorem 4 and Lemma 5, we obtain the following corollary.

Corollary 1: Suppose $f \in C^0[-hT_\epsilon, hT_\epsilon]$ satisfies the α-Hölder condition with $|f|_{C^{0,\alpha}} = O(1)$ and $N = O\big(hT_\epsilon (\frac{1}{\epsilon} \log \frac{1}{\epsilon})^{1/\alpha}\big)$. Then, it holds that $|f(x) - (S_N f)(x)| \leq \epsilon$ for $x \in [-hT_\epsilon, hT_\epsilon]$.

Everything is in place to prove Theorem 2. Consider the algorithm AP. If μ is α-Hölder with coefficient $O(1)$, we can construct $\widehat{\mu}$ which is also α-Hölder with coefficient $O(1)$, by Lemma 4. Then, we can easily see that $f(x) = \eta^x \widehat{\mu}(x)$ is also α-Hölder with coefficient $O(1)$ in $[-hT_\epsilon, hT_\epsilon]$ for any $\eta = 2$. Hence, we can apply Corollary 1. By Lemma 3, we complete the proof of Theorem 2.

How to Choose h: Recall from Section 2.2 that we need to choose h (the value should be independent of c_is and L) to satisfy (2). In fact, if μ satisfies the α-Hölder condition for some $\alpha > 1/2$, we can choose $h = O(\frac{1}{T_\epsilon} \log \frac{1}{\epsilon})$. Please refer to the full version of the paper [32] for the details.

3. APPLICATIONS

We first consider two utility functions $\chi(x)$ and $\widetilde{\chi}(x)$ presented in the introduction. Note that maximizing $\mathbb{E}[\chi(w(S))]$ is equivalent to maximizing $\Pr(w(S) \leq 1)$. The following lemma is straightforward.

Lemma 6: For any solution S,

$$\Pr(w(S) \leq 1) \leq \mathbb{E}[\widetilde{\chi}(w(S))] \leq \Pr(w(S) \leq 1 + \delta).$$

By Theorem 1, Theorem 2 and Lemma 6, we can easily obtain the following corollary.

Corollary 2: Suppose there is a pseudopolynomial time algorithm for the exact version of A. Then, for any fixed constants $\epsilon > 0$ and $\delta > 0$, there is an algorithm that runs in time $(\frac{n}{\epsilon})^{O(\frac{1}{\epsilon^2}\log\frac{1}{\epsilon})}$, and produces a solution S such that

$$\Pr(w(S) \leq 1 + \delta) + \epsilon \geq \max_{S' \in \mathcal{F}} \Pr(w(S') \leq 1)$$

Now, let us see some applications of our general results to specific problems.

Stochastic Shortest Path: Finding a path with the exact target length (we allow non-simple paths)[3] can be easily done in pseudopolynomial time by dynamic programming. Therefore, as discussed in Section 1.1, Corollary 2 generalizes several results for stochastic shortest path in prior work [41], [39].

Stochastic Spanning Tree: Our objective is to find a spanning tree T in the given probabilistic graph such that $\Pr(w(T) \leq 1)$ is maximized. Polynomial time algorithms have been developed for Gaussian distributed edges [28], [22]. To the best of our knowledge, no approximation algorithm with provable guarantee is known for other distributions. Noticing there exists a pseudopolynomial time algorithm for the exact spanning tree problem [6], we can directly apply Corollary 2.

Stochastic k-Median on Trees: The problem asks for a set S of k nodes in the given probabilistic tree G such that $\Pr(\sum_{v \in V(G)} \mathsf{dis}(v, S) \leq 1)$ is maximized, where $\mathsf{dis}(v, S)$ is the minimum distance from v to any node in S in the tree metric. The k-median problem can be solved optimally in polynomial time on trees by dynamic programming [29]. In fact, we can easily modify the dynamic program to get a pseudopolynomial time algorithm for the exact version.

Stochastic Knapsack with Random Sizes: We are given a set U of n items. Each item i has a random size w_i and a deterministic profit v_i. We are also given a positive constant $0 \leq \gamma \leq 1$. The goal is to find a subset $S \subseteq U$ such that $\Pr(w(S) \leq 1) \geq \gamma$ and the total profit $v(S) = \sum_{i \in S} v_i$ is maximized.

If the profits of the items are polynomially bounded integers, we can see the optimal profit is also a polynomially bounded integer. We can first guess the optimal profit. For each guess g, we solve the following problem: find a subset S of items such that the total profit of S is exactly g and $\mathbb{E}[\widetilde{\chi}(w(S))]$ is maximized. The exact version of the deterministic problem is to find a solution S with a given total size and a given total profit, which can be easily solved in pseudopolynomial time by dynamic programming. Therefore, by Corollary 2, we can easily show that we can find in polynomial time a set S of items such that the total profit $v(S)$ is at least the optimum and $\Pr(w(S) \leq 1+\epsilon) \geq (1-\epsilon)\gamma$ for any constant ϵ and γ.

If the profits are general integers, we can use the standard scaling technique to get a $(1-\epsilon)$-approximation for the total profit. See the full version for the details. In sum, we have obtained the following result.

Theorem 5: For any constants $\epsilon > 0$ and $\gamma > 0$, there is a polynomial time algorithm to compute a set S of items such that the total profit $v(S)$ is within a $1-\epsilon$ factor of the optimum and $\Pr(w(S) \leq 1+\epsilon) \geq (1-\epsilon)\gamma$.

Recently, Bhalgat et al. [12, Theorem 8.1] obtained the same result, with a running time $n^{2^{poly(1/\epsilon)}}$, while our running time is $(\frac{n}{\epsilon})^{O(\frac{1}{\epsilon^2}\log\frac{1}{\epsilon})} = n^{poly(1/\epsilon)}$.

Moreover, we can easily extend our algorithm to generalizations of the knapsack problem if the corresponding exact version has a pseudopolynomial time algorithm. For example, we can get the same result for the partial-ordered knapsack problem with tree constraints [21], [45]. In this problem, items must be chosen in accordance with specified precedence constraints and these precedence constraints form a partial order and the underlining undirected graph is a tree (or forest). A pseudopolynomial algorithm for this problem is presented in [45].

Stochastic Knapsack with Random Profits: We are given a set U of n items. Each item i has a deterministic size w_i and a random profit v_i. The goal is to find a subset of items that can be packed into a knapsack with capacity 1 and the probability that the profit is at least a given threshold T is maximized. Henig [27] and Carraway et al. [13] studied this problem for normally distributed profits and presented dynamic programming and branch and bound heuristics to solve this problem optimally.

We can solve the equivalent problem of minimizing the probability that the profit is at most the given threshold. It is straightforward to modify our algorithm to work for the minimization problem and we can also get an ϵ additive error for any $\epsilon > 0$. In fact, we can show that violation of the capacity constraint is necessary unless $P = NP$. The details can be found in the full version [32] of the paper .

Theorem 6: If the optimal probability is $\Omega(1)$, we can find in polynomial time a subset S of items such that $\Pr(v(S) > (1-\epsilon)T) \geq (1-\epsilon)OPT$ and $w(S) \leq 1+\epsilon$, for any constant $\epsilon > 0$.

[3]The exact version of *simple* path is NP-hard, since it includes the Hamiltonian path problem as a special case.

4. EXTENSIONS

In this section, we discuss some extensions to our basic approximation scheme. We first consider optimizing a constant number of utility functions in Section 4.1. Then, we study the problem where the weight of each element is a random vector in Section 4.2.

4.1. Multiple Utility Functions

The problem we study in this section contains a set U of n elements. Each element e has a random weight w_e. We are also given d utility functions $\mu_1, \ldots, \mu_d$ and d positive numbers $\lambda_1, \ldots, \lambda_d$. We assume d is a constant. A feasible solution consists of d subsets of elements that satisfy some property. Our objective is to find a feasible solution $S_1, \ldots, S_d$ such that $\mathbb{E}[\mu_i(w(S_i))] \geq \lambda_i$ for all $1 \leq i \leq d$.

We can easily extend our basic approximation scheme to the multiple utility functions case as follows. We decompose these utility functions into short exponential sums using ESUM as before. Then, for each utility function, we maintain $(n/\epsilon)^{O(L)}$ configurations. Therefore, we have $(n/\epsilon)^{O(dL)}$ configurations in total and we would like to compute the values for these configurations. We denote the deterministic version of the problem under consideration by A. The exact version of A asks for a feasible solution $S_1, \ldots, S_d$ such that the total weight of S_i is exactly the given number t_i for all i. Following an argument similar to Lemma 2, we can easily get the following generalization of Theorem 1.

Theorem 7: Assume that there is a pseudopolynomial algorithm for the exact version of A. Further assume that given any $\epsilon > 0$, we can ϵ-approximate each utility function by an exponential sum with at most L terms. Then, there is an algorithm that runs in time $(n/\epsilon)^{O(dL)}$ and finds a feasible solution $S_1, \ldots, S_d$ such that $\mathbb{E}[\mu_i(w(S_i)] \geq \lambda_i - \epsilon$ for $1 \leq i \leq d$, if there is a feasible solution for the original problem.

Now let us consider two simple applications of the above theorem.

Stochastic Multiple Knapsack: In this problem we are given a set U of n items, d knapsacks with capacity 1, and d constants $0 \leq \gamma_i \leq 1$. We assume d is a constant. Each item i has a random size w_i and a deterministic profit v_i. Our objective is to find d disjoint subsets $S_1, \ldots, S_d$ such that $\Pr(w(S_i) \leq 1) \geq \gamma_i$ for all $1 \leq i \leq d$ and $\sum_{i=1}^{d} v(S_i)$ is maximized. The exact version of the problem is to find a packing such that the load of each knapsack i is *exactly* the given value t_i. It is not hard to show this problem can be solved in pseudopolynomial time by standard dynamic programming. If the profits are general integers, we also need the scaling technique as in stochastic knapsack with random sizes. In sum, we can get the following generalization of Theorem 5.

Theorem 8: For any constants $d \in \mathbb{N}$, $\epsilon > 0$ and $0 \leq \gamma_i \leq 1$ for $1 \leq i \leq d$, there is a polynomial time algorithm to compute d disjoint subsets $S_1, \ldots, S_d$ such that the total profit $\sum_{i=1}^{d} v(S_i)$ is within a $1 - \epsilon$ factor of the optimum and $\Pr(w(S_i) \leq 1 + \epsilon) \geq (1 - \epsilon)\gamma_i$ for $1 \leq i \leq d$.

Stochastic Multidimensional Knapsack: In this problem we are given a set U of n items and a constant $0 \leq \gamma \leq 1$. Each item i has a deterministic profit v_i and a random size which is a random d-dimensional vector $\mathbf{w}_i = \{w_{i1}, \ldots, w_{id}\}$. We assume d is a constant. Our objective is to find a subset S of items such that $\Pr(\bigwedge_{j=1}^{d}(\sum_{i \in S} w_{ij} \leq 1)) \geq \gamma$ and total profit is maximized. This problem can be also thought as the fixed set version of the stochastic packing problem considered in [18], [12]. We first assume the components of each size vector are independent. The correlated case will be addressed in the next subsection.

For ease of presentation, we assume $d = 2$ from now on. Extension to general constant d is straightforward. We can solve the problem by casting it into a multiple utility problem as follows. For each item i, we create two copies i_1 and i_2. The copy i_j has a random weight w_{ij}. A feasible solution consists of two sets S_1 and S_2 such that S_1 (S_2) only contains the first (second) copies of the elements and S_1 and S_2 correspond to exactly the same subset of original elements. We enumerate all such pairs (γ_1, γ_2) such that $\gamma_1\gamma_2 \geq \gamma$ and $\gamma_i \in [\gamma, 1]$ is a power of $1 - \epsilon$ for $i = 1, 2$. Clearly, there are a polynomial number of such pairs. For each pair (γ_1, γ_2), we solve the following problem: find a feasible solution S_1, S_2 such that $\Pr(\sum_{i \in S_j} w_{ij} \leq 1) \geq \gamma_j$ for all $j = 1, 2$ and total profit is maximized. Using the scaling technique and Theorem 7 for optimizing multiple utility functions, we can get a $(1 - \epsilon)$-approximation for the optimal profit and $\Pr(\bigwedge_{j=1}^{2}(\sum_{i \in S_j} w_{ij} \leq 1)) = \prod_{j=1}^{2} \Pr(\sum_{i \in S_j} w_{ij} \leq 1) \geq (1 - O(\epsilon))\gamma_1\gamma_2 \geq (1 - O(\epsilon))\gamma$.

We note that the same result for independent components can be also obtained by using the discretization technique developed for the adaptive version of the problem in [12] [4]. If the components of each size vector are correlated, we can not decompose the problem into two 1-dimensional utilities as in the independent case. Now, we introduce a new technique to handle the correlated case.

4.2. Multidimensional Weight

The general problem we study contains a set U of n elements. Each element e has a random weight vector $w_i = (w_{i1}, \ldots, w_{id})$. We assume d is a constant. We are also given a utility functions $\mu : \mathbb{R}^d \to \mathbb{R}^+$. A feasible solution is a subset of elements satisfying some property. We use $w(S)$ as a shorthand notation for vector $(\sum_{i \in S} w_{i1}, \ldots, \sum_{i \in S} w_{id})$. Our objective is to find a feasible solution S such that $\mathbb{E}[\mu_i(w(S)]$ is maximized.

From now on, x and k denote d-dimensional vectors and kx (or $k \cdot x$) denotes the inner product of k and

[4]With some changes of the discretization technique, the correlated case can be also handled [11].

x. As before, we assume $\mu(x) \in [0,1]$ for all $x \geq 0$ and $\lim_{|x|\to+\infty} \mu(x) = 0$, where $|x| = \max(x_1, \ldots, x_d)$, Our algorithm is almost the same as in the one dimension case and we briefly sketch it here. We first notice that expected utilities decompose for exponential utility functions, i.e., $\mathbb{E}[\phi^{k\cdot w(S)}] = \prod_{i\in S} \mathbb{E}[\phi^{k\cdot w_i}]$. Then, we attempt to ϵ-approximate the utility function $\mu(x)$ by a short exponential sum $\sum_{|k|\leq N} c_k \phi_k^{kx}$ (there are $O(N^d)$ terms). If this can be done, $\mathbb{E}[\phi^{k\cdot w(S)}]$ can be approximated by $\sum_{|k|\leq N} c_k \mathbb{E}[\phi^{k\cdot w(S)}]$. Using the same argument as in Theorem 1, we can show that there is a polynomial time algorithm that can find a feasible solution S with $\mathbb{E}[\mu(w(S))] \geq OPT - \epsilon$ for any $\epsilon > 0$, provided that a pseudopolynomial algorithm exists for the exact version of the deterministic problem.

To approximate the utility function $\mu(x)$, we need the multidimensional Fourier series expansion of a function $f : \mathbb{C}^d \to \mathbb{C}$ (assuming f is 2π-periodic in each axis): $f(x) \sim \sum_{k\in\mathbb{Z}^d} c_k e^{ikx}$ where $c_k = \frac{1}{(2\pi)^d} \int_{x\in[-\pi,\pi]^d} f(x) e^{-ikx}\, dx$. The *rectangular partial sum* is defined to be

$$S_N f(x) = \sum_{|k_1|\leq N} \cdots \sum_{|k_d|\leq N} c_k e^{ikx}.$$

It is known that the rectangular partial sum $S_N f(x)$ converges uniformly to $f(x)$ in $[-\pi,\pi]^d$ for many function classes as n tends to infinity. In fact, a generalization of Theorem 4 to $[-\pi,\pi]^d$ also holds [4]: If f satisfies the α-Hölder condition, then

$$|f(x) - (S_N f)(x)| \leq O\Big(\frac{|f|_{C^{0,\alpha}} \ln^d N}{N^\alpha}\Big) \quad \text{for } x \in [-\pi,\pi]^d.$$

Now, we have an algorithm AP that can approximate a function in a bounded domain. It is also straightforward to extend ESUM to the multidimensional case. Hence, we can ϵ-approximate μ by a short exponential sum in $[0,+\infty)^d$, thereby proving the multidimensional generalization of Theorem 2. Let us consider an application of our result.

Stochastic Multidimensional Knapsack (Revisited): We consider the case where the components of each weight vector can be correlated. Note that the utility function χ_2 corresponding to this problem is the two dimensional threshold function: $\chi_2(x,y) = 1$ if $x \leq 1$ and $y \leq 1$; $\chi_2(x,y) = 0$ otherwise. As in the one dimensional case, we need to consider a continuous version $\widetilde{\chi}_2$ of χ_2 (see Figure 1(3)). By the result in this section and a generalization of Lemma 6 to higher dimension, we can get the following.

Theorem 9: For any constants $d \in \mathbb{N}$, $\epsilon > 0$ and $0 \leq \gamma \leq 1$, there is a polynomial time algorithm for finding a set S of items such that the total profit $v(S)$ is $1-\epsilon$ factor of the optimum and $\Pr(\bigwedge_{j=1}^d (\sum_{i\in S} w_{ij} \leq 1+\epsilon)) \geq (1-\epsilon)\gamma$.

5. Conclusion

We considered the problem of maximizing expected utility for many stochastic combinatorial problems, such as shortest paths, spanning trees, matchings, and knapsack. We developed a polynomial time approximation scheme with additive error ϵ for any $\epsilon > 0$. A key ingredient of our algorithm is decomposition of the utility function into a short exponential sum. In this paper, we use the Fourier series technique to fulfill this task. Exploring other decomposition approaches is an interesting direction for future work. Our general approximation framework may be useful for other stochastic optimization problems. One major open problem is to obtain approximations with reasonable multiplicative factors, or nontrivial inapproximability results, for the utility maximization problem.

6. Acknowledgments

We would like to thank Evdokia Nikolova for providing an extended version of [41] and many helpful discussions. We also would like to thank Chandra Chekuri for pointing to us the recent work [12] and Anand Bhalgat for some clarifications of the same work. This work was supported in part by NSF Grant IIS-0916736, a DARPA grant, National Basic Research Program of China Grants 2007CB807900, 2007CB807901, and the National Natural Science Foundation of China Grants 61033001, 61061130540, 61073174.

References

[1] "St. Petersburg paradox," http://en.wikipedia.org/wiki/St._Petersburg_paradox.

[2] H. Ackermann, A. Newman, H. Röglin, and B. Vöcking, "Decision making based on approximate and smoothed pareto curves," *Algorithms and Computation*, pp. 675–684, 2005.

[3] S. Agrawal, A. Saberi, and Y. Ye, "Stochastic Combinatorial Optimization under Probabilistic Constraints," *Arxiv preprint arXiv:0809.0460*, 2008.

[4] S. Alimov, R. Ashurov, and A. Pulatov, "Multiple fourier series and fourier integrals, in commutative harmonic analysis. IV: Harmonic analysis in $\mathbb{R}_n$," *Encyclopedia of Mathematical Science*, vol. 42, 1992.

[5] N. Bansal, A. Gupta, J. Li, J. Mestre, V. Nagarajan, and A. Rudra, "When LP is the Cure for Your Matching Woes: Improved Bounds for Stochastic Matchings," *European Symposium on Algorithms*, pp. 218–229, 2010.

[6] F. Barahona and W. Pulleyblank, "Exact arborescences, matchings and cycles," *Discrete Applied Mathematics*, vol. 16, no. 2, pp. 91–99, 1987.

[7] J. Bard and J. Bennett, "Arc reduction and path preference in stochastic acyclic networks," *Management Science*, vol. 37, no. 2, pp. 198–215, 1991.

[8] G. Beylkin and L. Monzón, "On Generalized Gaussian Quadratures for Exponentials and Their Applications* 1," *Applied and Computational Harmonic Analysis*, vol. 12, no. 3, pp. 332–373, 2002.

[9] ——, "On approximation of functions by exponential sums," *Applied and Computational Harmonic Analysis*, vol. 19, no. 1, pp. 17–48, 2005.

[10] ——, "Approximation by exponential sums revisited," *Applied and Computational Harmonic Analysis*, vol. 28, no. 2, pp. 131–149, 2010.

[11] A. Bhalgat, 2011, personal Communication.

[12] A. Bhalgat, A. Goel, and S. Khanna, "Improved approximation results for stochastic knapsack problems," in *ACM-SIAM Symposium on Discrete algorithms*, 2011.

[13] R. Carraway, R. Schmidt, and L. Weatherford, "An algorithm for maximizing target achievement in the stochastic knapsack problem with normal returns," *Naval research logistics*, vol. 40, no. 2, pp. 161–173, 1993.

[14] C. Chekuri and S. Khanna, "A PTAS for the multiple knapsack problem," in *ACM-SIAM Symposium on Discrete algorithms*, 2000, pp. 213–222.

[15] N. Chen, N. Immorlica, A. Karlin, M. Mahdian, and A. Rudra, "Approximating matches made in heaven," *International Colloquium on Automata, Languages and Programming*, pp. 266–278, 2009.

[16] W. Cheney and W. Light, *A Course in Approximation Theory*. Brook/Cole Publishing Company, 2000.

[17] R. Cheng, J. Chen, and X. Xie, "Cleaning uncertain data with quality guarantees," *Proceedings of the VLDB Endowment*, vol. 1, no. 1, pp. 722–735, 2008.

[18] B. Dean, M. Goemans, and J. Vondrák, "Adaptivity and approximation for stochastic packing problems," in *ACM-SIAM Symposium on Discrete algorithms*, 2005, pp. 395–404.

[19] B. Dean, M. Goemans, and J. Vondrak, "Approximating the Stochastic Knapsack Problem: The Benefit of Adaptivity," *Mathematics of Operations Research*, vol. 33, no. 4, 2008.

[20] P. Fishburn, *Utility Theory and Decision Making*. John Wiley & Sons, Inc, 1970.

[21] M. Garey and D. Johnson, *"Computers and Intractability: A Guide to the Theory of NP-Completeness"*. W.H. Freeman, 1979.

[22] S. Geetha and K. Nair, "On stochastic spanning tree problem," *Networks*, vol. 23, no. 8, pp. 675–679, 1993.

[23] A. Goel and P. Indyk, "Stochastic load balancing and related problems," in *Annual Symp. on Foundations of Computer Science*, 1999, p. 579.

[24] V. Goyal and R. Ravi, "Chance constrained knapsack problem with random item sizes," *To appear in Operation Research Letter*, 2009.

[25] S. Guha and K. Munagala, "Adaptive Uncertainty Resolution in Bayesian Combinatorial Optimization Problems," *To appear in ACM Transactions on Algorithms*, 2008.

[26] A. Gupta, M. Pál, R. Ravi, and A. Sinha, "Boosted sampling: approximation algorithms for stochastic optimization," in *ACM Symp. on Theory of Computing*, 2004, pp. 417–426.

[27] M. Henig, "Risk criteria in a stochastic knapsack problem," *Operations Research*, vol. 38, no. 5, pp. 820–825, 1990.

[28] H. Ishii, S. Shiode, and T. Nishida Yoshikazu, "Stochastic spanning tree problem," *Discrete Applied Mathematics*, vol. 3, no. 4, pp. 263–273, 1981.

[29] O. Kariv and S. Hakimi, "An algorithmic approach to network location problems. II: The p-medians," *SIAM Journal on Applied Mathematics*, vol. 37, no. 3, pp. 539–560, 1979.

[30] J. Kleinberg, Y. Rabani, and É. Tardos, "Allocating bandwidth for bursty connections," in *ACM Symp. on Theory of Computing*, 1997, p. 673.

[31] J. Li and A. Deshpande, "Consensus answers for queries over probabilistic databases," in *ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*, 2009.

[32] ——, "Maximizing expected utility for stochastic combinatorial optimization problems," 2011, http://arxiv.org/abs/1012.3189.

[33] J. Li, B. Saha, and A. Deshpande, "A unified approach to ranking in probabilistic databases," in *Proceedings of the VLDB Endowment*, 2009.

[34] R. Loui, "Optimal paths in graphs with stochastic or multidimensional weights," *Communications of the ACM*, vol. 26, no. 9, pp. 670–676, 1983.

[35] R. Martin, "The St. Petersburg Paradox," *The Stanford Encyclopedia of Philosophy*, 2004, http://plato.stanford.edu/archives/fall2004/entries/paradox-stpetersburg.

[36] S. Mittal and A. Schulz, "A general framework for designing approximation schemes for combinatorial optimization problems with many objectives combined into one," *Approximation, Randomization and Combinatorial Optimization. Algorithms and Techniques*, pp. 179–192, 2008.

[37] I. Murthy and S. Sarkar, "Exact algorithms for the stochastic shortest path problem with a decreasing deadline utility function," *European Journal of Operational Research*, vol. 103, no. 1, pp. 209–229, 1997.

[38] ——, "Stochastic shortest path problems with piecewise-linear concave utility functions," *Management Science*, vol. 44, no. 11, pp. 125–136, 1998.

[39] E. Nikolova, "Approximation Algorithms for Reliable Stochastic Combinatorial Optimization," *International Workshop on Approximation Algorithms for Combinatorial Optimization Problems*, pp. 338–351, 2010.

[40] E. Nikolova, M. Brand, and D. Karger, "Optimal route planning under uncertainty," in *Proceedings of International Conference on Automated Planning and Scheduling*, 2006.

[41] E. Nikolova, J. Kelner, M. Brand, and M. Mitzenmacher, "Stochastic shortest paths via quasi-convex maximization," in *European Symposium on Algorithms*, 2006, pp. 552–563.

[42] M. R. Osborne and G. K. Smyth, "A modified prony algorithm for fitting sums of exponential functions," *SIAM Journal of Scientific Computing*, 1995.

[43] C. Papadimitriou and M. Yannakakis, "On the approximability of trade-offs and optimal access of web sources," in *Annual Symp. on Foundations of Computer Science*, 2000.

[44] M. J. D. Powell, *Approximation theory and methods*. Cambridge University Press, 1981.

[45] H. Safer, J. B. Orlin, and M. Dror, "Fully polynomial approximation in multi-criteria combinatorial optimization," 2004, mIT Working Paper.

[46] D. Shmoys and C. Swamy, "An approximation scheme for stochastic linear programming and its application to stochastic integer programs," *J. ACM*, vol. 53, no. 6, p. 1012, 2006.

[47] C. Sigal, A. Pritsker, and J. Solberg, "The stochastic shortest route problem," *Operations Research*, vol. 28, no. 5, pp. 1122–1129, 1980.

[48] E. Stein and R. Shakarchi, *Fourier analysis: an introduction*. Princeton University Press, 2003.

[49] C. Swamy, "Risk-Averse Stochastic Optimization: Probabilistically-Constrained Models and Algorithms for Black-Box Distributions." *ACM-SIAM Symposium on Discrete algorithms*, 2010.

[50] C. Swamy and D. Shmoys, "Approximation algorithms for 2-stage stochastic optimization problems," *ACM SIGACT News*, vol. 37, no. 1, p. 46, 2006.

[51] J. von Neumann and O. Morgenstern, *Theory of Games and Economic Behavior*, 2nd ed. Princeton Univ. Press, 1947.