Approximation Algorithms for Offline Risk-averse Combinatorial Optimization

Evdokia Nikolova [*]

November 5, 2010

Abstract

We consider generic optimization problems that can be formulated as minimizing the cost of a feasible solution $\mathbf{w}^T\mathbf{x}$ over a combinatorial feasible set $\mathcal{F} \subset \{0,1\}^n$. For these problems we describe a framework of risk-averse stochastic problems where the cost vector $\mathbf{W}$ has independent random components, unknown at the time of solution. A natural and important objective that incorporates risk in this stochastic setting is to look for a feasible solution whose stochastic cost has a small tail or a small convex combination of mean and standard deviation. Our models can be equivalently reformulated as nonconvex programs for which no efficient algorithms are known. In this paper, we make progress on these hard problems.

Our results are several efficient general-purpose approximation schemes. They use as a black-box (exact or approximate) the solution to the underlying deterministic problem and thus immediately apply to arbitrary combinatorial problems. For example, from an available δ-approximation algorithm to the linear problem, we construct a $\delta(1+\epsilon)$-approximation algorithm for the stochastic problem, which invokes the linear algorithm only a logarithmic number of times in the problem input (and polynomial in $\frac{1}{\epsilon}$), for any desired accuracy level $\epsilon > 0$. The algorithms are based on a geometric analysis of the curvature and approximability of the nonlinear level sets of the objective functions.

1 Introduction

Suppose we have to catch a flight and need to find a route to the airport. If there is no traffic, this is an application of the classical shortest path problem and can be solved with a variety of existing algorithms such as Dijkstra's shortest path algorithm, etc. More often, however, not only is there traffic but also traffic conditions are *uncertain*. What then do we mean by the shortest path to the airport? Such a question is ill-posed. We may instead attempt definitions such as the path with the shortest *expected* travel time, although, when we have a flight to catch, this does not seem like an appropriate objective. What we need instead is a definition that captures our risk aversion.

The definition of the risk-averse model need not be unique. Indeed, the natural objectives may change depending on *when* we are submitting the route query: ahead of time, when we are debating how much time to budget for our trip, *or* at the start of our trip, when we want to maximize our chance of on-time arrival over the fixed time period we now have to get to the destination. In the former setting, we would typically want to allocate enough time to ensure some confidence of on-time arrival, say 95%. In the latter, given a deadline to reach our destination, we need to find the route with which we will most likely reach by the deadline. For example, this optimal route may give us only 60% chance of arriving on time if we have not allocated enough time for the trip. A third objective, used for example by the Federal Highway Administration [15] as a travel time reliability criterion, is given by the mean plus standard deviation of a route. This third criterion has been considered in the context of stochastic minimum spanning trees as well [3], and is sometimes referred to as mean-risk optimization (*e.g.,* [3]).

In this paper, inspired by the route planning application above, we consider generic combinatorial problems that can be formulated as minimizing the cost of a feasible solution $\mathbf{w}^T\mathbf{x}$ over a combinatorial feasible set $\mathcal{F} \subset \{0,1\}^n$ and ask what happens when the associated costs are stochastic. The most common approach in stochastic optimization

[*]Massachusetts Institute of Technology, Cambridge, MA. Email: nikolova @ mit.edu.

is to find the solution of minimum expected cost. However, in many applications such as the one above reliability considerations are very important: risk-averse users need reassurance regarding the level of risk, and not just the expected cost of the provided solution. For example, the transportation community has recognized the importance of reliable route plans (*e.g.*, [9, 36, 33, 46, 14]). However, the algorithms for finding these reliable routes are typically inefficient or heuristic with unknown approximation guarantee. Risk-aversion is clearly very important as well in finance and other *continuous* optimization settings [42]. While risk models have a long history in the finance setting, their study is much more recent in *combinatorial* optimization settings and there are hardly any studies on general risk-averse models and unified approaches for solving them from an approximation algorithms perspective in the complexity theoretic sense. (We describe related work below.) One challenge with such research is that incorporating risk-aversion transforms the problems into *nonconvex* ones [42, 37] for which there are no known efficient algorithms and rigorous approximative analysis is scarce. In addition, having to perform nonconvex optimization over *combinatorial* feasible sets adds an extra layer of difficulty and necessitates merging the traditionally distinct *continuous* and *discrete* optimization approaches.

In this paper, we provide a rigorous unified treatment of offline risk-averse combinatorial optimization problems, offering fully-polynomial approximation schemes (FPTAS) for the following risk-averse models:

1. *Mean-risk model:* minimize $(mean + c \cdot standard\ deviation)$ where $c \geq 0$ is the risk-aversion coefficient.

2. *Probability tail model:* maximize $\Pr(solution\ cost \leq budget)$ for a given *budget*.

3. *Value-at-risk model:* minimize $budget$ such that $\Pr(solution\ cost \leq budget) \geq p$ for a given confidence probability p.

In contrast with the diversity in risk-averse model specifications above, we will show that the same approximation algorithm design can simultaneously solve all. In our analysis, we assume that the cost distributions are independent although in Section 5 we show how our algorithms also extend to the case of correlations of neighboring edges in a graph. For example, for shortest path problems, the graph with correlated edges is transformed into a slightly larger graph with independent edges and thus all our results immediately carry through. A more in-depth analysis of correlations in stochastic optimization is offered by Agrawal *et al.* [2].

To be precise, all our algorithms run in *oracle*-polynomial time, in that they call an algorithm (oracle) for the underlying deterministic problem polynomially many times in the problem input (and in $\frac{1}{\epsilon}$ for a given $\epsilon > 0$ in the case of FPTAS). For simplicity, instead of *oracle*-FPTAS, we shall simply refer to them as FPTAS, defined more formally as follows:

Definition 1.1 *A fully-polynomial approximation scheme (FPTAS) is an algorithm for an optimization problem that, given an input I and desired accuracy $\epsilon > 0$, finds in time polynomial in $\frac{1}{\epsilon}$ and the input size, a solution of value $OPT'(I)$ that satisfies*

$$|OPT(I) - OPT'(I)| \leq \epsilon OPT(I),$$

for all inputs I, where $OPT(I)$ is the optimal solution value on input I.

In Section 4 we give approximation algorithms for the stochastic versions of NP-hard combinatorial problems, for whose deterministic versions there are available δ-approximations. This notion of approximation is defined more formally below:

Definition 1.2 *A δ-approximation algorithm for a minimization problem is a polynomial-time algorithm that, given an input instance I, finds a solution with value $OPT'(I)$, satisfying*

$$OPT(I) \leq OPT'(I) \leq \delta OPT(I),$$

for all instances I, where $OPT(I)$ is the optimal solution value on input I. The definition of approximation for a maximization problem is analogous.

Contributions. We start our discussion with the relatively simpler mean-risk model, which is equivalent to minimizing $\left(mean + c \cdot \sqrt{variance}\right)$. We provide fully-polynomial approximation algorithms that apply to *arbitrary* cost distributions with given means and variances, and achieve essentially the same approximation factor as what is possible for the underlying deterministic problem. Our algorithms use as a black-box an algorithm for the deterministic problem. We summarize our results for this setting below:

Theorem 1.3 (See Theorems 3.1, 4.1) *There is a fully-polynomial approximation scheme for the mean-risk stochastic model, when there is an exact or fully-polynomial approximation algorithm for the underlying deterministic problem.*

In addition, there is a $(1+\epsilon)\delta$-approximation for the stochastic model running in time polynomial in $\frac{1}{\epsilon}$, when there is an available δ-approximation for the corresponding deterministic problem.

A rigorous approximation-algorithmic analysis of the probability tail and value-at-risk models in the framework, which involve optimization of the probability tails, necessitates an assumption on the distribution: in the absence of any knowledge on the distributions, the best one can do is bound the tails, for example using Chernoff or Chebyshev bounds, and optimize those tail bounds instead—this will yield a conservative overestimate of the probability of exceeding the budget.

We provide strict approximation results under the commonly assumed Gaussian distributions; we then show how the same algorithmic techniques can apply to arbitrary distributions using tail bounds. In the Gaussian setting, minimizing the probability tail in the probability tail model is equivalent to maximizing $\frac{budget-mean}{\sqrt{variance}}$ and we get the following approximations:

Theorem 1.4 (See Theorems 3.1, 4.2) *There is a fully-polynomial approximation scheme for the probability tail model, when there is an exact or fully-polynomial approximation algorithm for the underlying deterministic problem.*

In addition, when there is an available δ-approximation for the deterministic problem, there is a $\sqrt{1-\left[\frac{\delta-(1-\epsilon^2/4)}{(2+\epsilon)\epsilon/4}\right]}$-approximation for the corresponding stochastic model running in time polynomial in $\frac{1}{\epsilon}$.

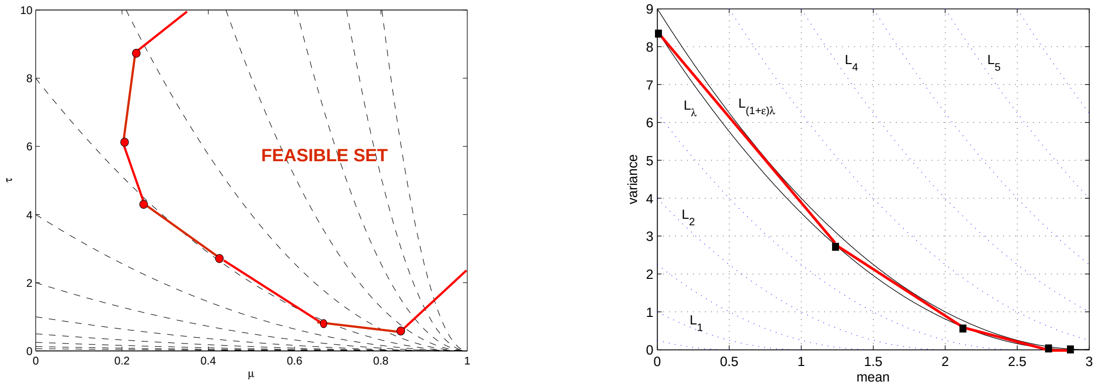
The value-at-risk model under Gaussian distributions is equivalent to the mean-risk model, with risk-aversion coefficient $c = \Phi^{-1}(p)$, where $\Phi^{-1}(\cdot)$ is the inverse cumulative distribution function of the standard normal $N(0,1)$.

For *arbitrary distributions*, the value-at-risk model reduces to the mean-risk model, but with a more conservative risk-aversion coefficient $c = \sqrt{\frac{p}{1-p}}$, which causes our algorithms to provide an overestimate of the true error probability of exceeding the budget.

Background and Challenges. Our algorithms build on the fact that the model formulations in our framework are all instances of concave or quasi-concave minimization, for which it is known that the optimal solution is attained at an extreme point of the feasible set (see, *e.g.,* [5]). In addition, our objective functions depend only on the means and variances of feasible solutions. Thus, we can project the feasible set on the plane spanned by the mean and variance vectors and only consider extreme points on the projection (see Figure 1(a)). This greatly restricts the number of relevant extreme points. For example, for minimum spanning trees and matroids, we can efficiently enumerate the polynomially many extreme points. Therefore, the corresponding risk-averse stochastic spanning trees and matroids can be found in polynomial time. We provide more of these background details and a description of the algorithm in Section 2. However, an arbitrary combinatorial problem typically has too many extreme points, even on a two-dimensional projection (for example, shortest paths have $n^{\log n}$ such points [38]), *hence our focus on approximation in this paper.*

We can geometrically visualize the objective function in terms of its level sets on the mean-variance plane. These form parabolas, corresponding to higher objective function values at greater mean and variance values. The optimal solution is obtained at the lowest parabola touching the projected feasible set. Figure 1(a) depicts these parabolas and the challenge that arises with concave minimization problems: along the convex hull boundary of the feasible set, the objective function may fluctuate. In particular, many extreme points might be local optima and thus local search algorithms can fail to find a good approximation.

Another technique, which might seem promising for obtaining a fully polynomial approximation algorithm for our risk-averse framework, is parametric search: for a given bound on the variance, find the solution with smallest mean,



(a) Objective function level sets and feasible polytope

(b) Nonlinear separation oracle

Figure 1: *(a)* Level sets of the probability tail objective function and the convex hull of the projected feasible set on the mean-variance plane. *(b)* Level sets and approximate *nonlinear separation oracle* on the mean-variance plane.

and then search for the variance bound yielding the best answer. There are two problems with this approach. First, finding the solution with smallest mean subject to a constraint on the variance is NP-hard and it is not always known or even not always possible to approximate it [40]. Second, even when we know how to solve it, an approximation for it would not necessarily yield a corresponding approximation to our probability tail objective due to the presence of the budget parameter in the objective.

Overview of Algorithms and Techniques. [For the case of *easy* deterministic problems.] A brief conceptual description of our approach is as follows: The algorithm constructs a *nonlinear separation oracle* that determines, for a given function level set,[1] if there is a feasible solution below the level set (with value less than the given function value) or the entire feasible set is above the given level (see Lemma 3.2 for a more formal definition of the oracle). Afterwards, a binary search on the optimum objective function value combined with the separation oracle finds the desired approximate solution.

The separation oracle approximates a given level set curve by inscribing a (partial) polygon in it, as shown in Figure 1(b). Each side of the polygon induces a linear objective over the feasible set, which we minimize via a black-box call to the algorithm for the deterministic problem. If the resulting solution is below the current level set (more precisely, its associated original objective function value is smaller than $(1+\epsilon)$ times the given level), the separation oracle returns that solution. Else, if after minimizing with respect to all linear segments we do not find any solutions below the level set, the separation oracle returns a negative answer, namely that the entire feasible set is above the level set.

The subtlety arises in how to construct the polygonal segments to ensure a good and efficient approximation. To get an efficient algorithm, we need to approximate the level set curves with as few linear segments as possible. On the other hand, to get a good approximation factor, we need a finer polygon (with more and smaller sides), which is sandwiched between the desired level set with function value λ and the level set with function value $\lambda(1+\epsilon)$ (See Figure 1(b)). In the worst case, when the level sets touch, as is the case for the probability tail objective, a polygon sandwiched between the two level sets will have infinitely many sides. We resolve this problem by carefully bounding the optimal solution so that we do not need all infinitely many linear segments from the polygon and, in particular, we prove that it suffices to consider only polynomially many such segments.

To the best of our knowledge, our concept and use of the approximate *nonlinear separation oracle* for the design of approximation algorithms for nonconvex optimization problems are novel. We believe that our approach would be useful for approximating other low-rank concave minimization and possibly more general nonlinear or nonconvex

[1] The level set of a function f for value λ is the subset of the domain on which the function equals λ, $L_\lambda = \{\mathbf{x} \mid f(\mathbf{x}) = \lambda\}$.

optimization problems beyond the ones considered here. We remark that the method applies to both discrete and continuous (non-polyhedral) constraint sets.

[*Hard* deterministic problems.] We could use the same algorithm design as above, by appropriately modifying its analysis and approximation factors, when we have a δ-approximation rather than an exact algorithm for solving the underlying deterministic problem. It turns out that for this case, a cruder and simpler algorithm gives the same approximation factor. In particular, all we need to do here is apply the algorithm for the deterministic problem on a small sequence of linear cost functions of the form $mean + k \cdot variance$, for a geometric progression of coefficients k.

However, even if we know what single choice of k would find the optimal solution, the difficulty is to translate the approximation given by the deterministic black-box algorithm for its associated *linear objective* into an approximation for the *original concave objective*. The two functions have nothing in common (except that the former is a gradient of the latter at some point), and, a priori, it is not clear that an approximation of the linear objective would at all yield a meaningful approximation factor for the original nonconvex objective. This is the key technical challenge which makes the analysis of this setting more mathematically involved. Fortunately, all objective functions in our framework admit such an approximation (the probability tail objective is again more challenging due to the given budget and requires us to know that there is a feasible solution at least a small distance away from the budget).

Related Work. A rich body of work in stochastic combinatorial optimization focuses on two-stage and multistage optimization (*e.g.*, [45, 22, 29, 21, 26]). The models there typically look for solutions of minimum *expected cost* and thus do not incorporate risk. In 2006 Swamy and Shmoys remarked that "it would be interesting to explore stochastic models that incorporate risk" [49]. There are models that have incorporated additional budget constraints [47] or threshold constraints for specific problems such as knapsack, load balancing and others [10, 18, 31].

At the other end of the risk-aversion spectrum is the paradigm of robust optimization (see survey [6]), which provides completely reliable (robust) solutions, though this is only possible when the uncertainty is bounded, namely the random variables have bounded support. Our framework for risk-averse optimization falls between the traditional stochastic optimization approach, which minimizes expected cost, and robust optimization, which minimizes the maximum cost. Interestingly, part of our framework (the mean-risk model) arises in robust discrete optimization under ellipsoidal uncertainty sets [7]. Bertsimas and Sim provide pseudopolynomial algorithms and an algorithm converging to a locally optimal solution, assuming that the underlying deterministic problem can be solved exactly. This contrasts with our fully polynomial approximation schemes that work with both exact and approximate algorithms for the deterministic problem.

Atamtürk and Narayanan [3] also consider mean-risk minimization in discrete optimization, giving a characterization in terms of submodular minimization. Our feasible set is an arbitrary subset of the hypercube vertices, on which it is not known how to do submodular minimization. As a curiosity, we mention here that the mean-risk objective is also supermodular via the Lovász extension [32]. However, supermodular minimization is even harder and this perspective does not help our problem at hand.

The probability tail objective was previously considered in the special context of stochastic shortest paths and an *exact* algorithm was given based on enumerating relevant extreme points from the path polytope [38]. The same type of algorithm readily extends to arbitrary combinatorial problems. However, in general the exact algorithm is inefficient (superpolynomial or exponential in the problem size), therefore we provide approximation algorithms in this paper.

The value-at-risk objective in our framework can be classified under research on probabilistic programming, or optimization with probabilistic (chance) constraints. Most of the existing literature concerns continuous optimization settings (*e.g.*, [37, 13]; see also Chapter 4 in Shapiro *et al.* [44]) and concentrates on convergence to optimal solutions, rather than the design of approximation algorithms in the complexity theoretic sense. One example of work in the discrete setting is on giving bounds for integer programming problems with probabilistic constraints [12]. This work considers a different problem formulation from ours, in which the uncertainty is in the demand, rather than the cost of a feasible solution (that is, it is in the right-hand-side, rather than the left-hand-side of the inequality in the probabilistic constraint) and the solution is via convexification of a cone generation method. In a separate line of research, Swamy considers two-stage risk-averse optimization for covering and packing problems [48]. Other than the high-level idea of incorporating risk, his models (assuming sampling access to distributions) and techniques (LP-relaxation and convex minimization) are entirely different from ours.

A comprehensive survey of models that incorporate risk in *continuous* settings is provided by Rockafellar [42] as well as in the recent book by Shapiro, Dentcheva and Ruszczyński [44]. A different framework that allows for uncertainty in the assumed distributions and distribution parameters is that of distributionally robust optimization (see, *e.g.,* [11]). The solution concepts and continuous nature of the problems make this line of research very different from ours.

Additional related work on the *combinatorial* optimization side includes research on multi-criteria optimization (*e.g.,* [40, 1, 43, 50]) and combinatorial optimization with a ratio of linear objectives [35, 41]. Our models can also be seen as instances of concave discrete minimization. However, the existing work in this area requires assumptions that do not hold in our framework, such as restrictive properties on the feasible set, strictly positive range of the objective function, or boundedness/positivity of the objective function gradient [39, 4, 30, 19].

2 Model definitions and preliminaries

In this section, we formally define the models in our risk-averse optimization framework and give the necessary background for our algorithms in the next sections.

Suppose we have an arbitrary combinatorial set of feasible solutions $\mathcal{F} \subset \{0,1\}^n$, together with an oracle for optimizing linear objectives over the set. In addition, we are given nonnegative vectors of means $\boldsymbol{\mu} \in \mathbb{R}^n$ and variances $\boldsymbol{\tau} \in \mathbb{R}^n$ for the stochastic cost vector $\mathbf{W}$, coming from independent distributions so that the mean and variance of a solution $\mathbf{x} \in \mathcal{F}$ is $\boldsymbol{\mu}^T\mathbf{x}$ and $\boldsymbol{\tau}^T\mathbf{x} \geq 0$ respectively. We are interested in finding a feasible solution with optimal cost, where the notion of optimality incorporates risk.

1. *Mean-risk model:* A family of objectives that has been analyzed in continuous optimization settings (mostly in the context of finance [16, 34]) and in some discrete optimization settings (minimum spanning trees [3]), as well as under an equivalent robust optimization framework [7], is the family of convex combinations of mean and standard deviation. Formally, this problem is to:

$$\begin{array}{ll} \text{minimize} & \boldsymbol{\mu}^T\mathbf{x} + c\sqrt{\boldsymbol{\tau}^T\mathbf{x}} \\ \text{subject to} & \mathbf{x} \in \mathcal{F}, \end{array} \tag{1}$$

   where the constant $c \geq 0$ parametrizes the degree of the user's risk aversion.

2. *Probability tail model:* An alternative natural model maximizes the probability that the stochastic solution cost is within a desired budget or threshold t: maximize $\Pr\left(\mathbf{W}^T\mathbf{x} \leq t\right)$ subject to $\mathbf{x} \in \mathcal{F}$. When the stochastic costs $\mathbf{W}$ are Gaussian, we can directly compute the above probability as $\Phi(\frac{t-\boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}})$, where $\Phi(\cdot)$ denotes the cumulative distribution function of the standard normal random variable $N(0,1)$. Since the function $\Phi(\cdot)$ is monotone increasing, the problem has the following equivalent formulation (which is also approximation-preserving by Lemma C.1 in the Appendix):

$$\begin{array}{ll} \text{maximize} & \dfrac{t-\boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}} \\ \text{subject to} & \mathbf{x} \in \mathcal{F}. \end{array} \tag{2}$$

   When the stochastic costs $\mathbf{W}$ come from arbitrary distributions, the maximum probability is lower-bounded by $\frac{(t-\boldsymbol{\mu}^T\mathbf{x})^2}{(t-\boldsymbol{\mu}^T\mathbf{x})^2+(\boldsymbol{\tau}^T\mathbf{x})}$ (by the one-sided Chebyshev bound, also known as Cantelli's inequality [20], $\Pr(X \leq E[X] + k\sqrt{Var(X)}) \geq 1 - \frac{1}{1+k^2}$, with $k = \frac{t-\boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}}$). While maximizing a lower-bound will not yield a strict approximation of the probability tail objective, it is the best one can achieve in the absence of distributional information other than the mean and variance—and our techniques can strictly approximate this bound as well:

$$\begin{array}{ll} \text{maximize} & \dfrac{(t-\boldsymbol{\mu}^T\mathbf{x})^2}{(t-\boldsymbol{\mu}^T\mathbf{x})^2+\boldsymbol{\tau}^T\mathbf{x}} \\ \text{subject to} & \mathbf{x} \in \mathcal{F}. \end{array} \tag{3}$$

For both formulations of the probability tail model we assume that there is a solution with mean that is within the given threshold t. This condition expresses that we are in a *risk-averse* situation and corresponds to the assumption that the risk-aversion coefficient $c \geq 0$ in the mean-risk model above. (From a mathematical standpoint, if we suppose that $\boldsymbol{\mu}^T\mathbf{x} > t$ for all $\mathbf{x} \in \mathcal{F}$, the maximum of problem (2) will be negative, therefore solutions with *higher variance would be preferred*, corresponding to a *risk-loving* situation.)

3. *Value-at-risk model:* Finally, we may wish to minimize the budget t such that the probability of not exceeding it is at least a given confidence level p:

$$\begin{array}{ll} \text{minimize} & t \\ \text{subject to} & \Pr(\mathbf{W}^T\mathbf{x} \leq t) \geq p \\ & \mathbf{x} \in \mathcal{F}. \end{array} \tag{4}$$

Depending on whether we have Gaussian or arbitrary distributions, this problem is exactly equivalent to, or its solution can be upper-bounded using Chebyshev's bound by the mean-risk model (1) with $c = \Phi^{-1}(p)$ or $c = \sqrt{\frac{p}{1-p}}$ (See Ghaoui *et al.* [17]; more details are provided in the next Section 2.1).

We should mention here that even when the random variables have *arbitrary* independent distributions, normal approximation for problems with probabilistic constraints has been suggested as a reasonable approach in the Lectures on Stochastic Programming by Shapiro *et al.* [44] (See p. 141-144 in Chapter 4.4). In particular, from the Central Limit Theorem, we have that when each variable has finite mean and finite variance and satisfies a mild additional condition (informally that the sum of third moments is small relative to sum of second moments), then the sum of random variables $\mathbf{W}^T\mathbf{x}$ converges to a normal distribution with mean $\boldsymbol{\mu}^T\mathbf{x}$ and variance $\boldsymbol{\tau}^T\mathbf{x}$ as the number of variables grows to infinity. In particular, for a fixed problem size, the approximation would be reasonable when the dimension n is sufficiently large and the incidence vector $\mathbf{x}$ has sufficiently many nonzero components.

Our algorithms make oracle calls to an *exact* or *approximate* algorithm for solving the underlying deterministic (linear) problem:

$$\begin{array}{ll} \text{minimize} & \mathbf{w}^T\mathbf{x} \\ \text{subject to} & \mathbf{x} \in \mathcal{F}. \end{array} \tag{5}$$

We sometimes refer to the algorithm for solving the deterministic problem as a *linear oracle* after its linear objective, in contrast with the risk-averse stochastic problems that have *nonlinear* objectives. This is not to be confused with linear programming (LP) or LP relaxation: the deterministic problem (5) is an integer problem which might be polynomially solvable or NP-hard.

We first establish that all models are instances of quasi-concave minimization (equivalently, quasi-convex maximization) over $\mathbf{x} \in \mathcal{F}$, consequently they attain their optima at extreme points of the feasible set [5].

2.1 Quasi-concave properties of the objectives

Concave (convex) functions are special cases of quasi-concave (quasi-convex) functions.

Definition 2.1 *A function g from a convex set C to $\mathbb{R}$ is* quasi-convex *if all its lower level sets $\underline{L}_\lambda = \{\mathbf{x} \mid g(\mathbf{x}) \leq \lambda\}$ are convex.*

Theorem 2.2 *[25, 5] Let $\mathcal{C} \subset \mathbb{R}^n$ be a compact convex set. A quasi-convex function $f : \mathcal{C} \to \mathbb{R}$ that attains a maximum over $\mathcal{C}$, attains the maximum at some extreme point of $\mathcal{C}$.*

We next show that the models in our risk-averse framework above are instances of quasi-concave minimization. The mean-risk objective in Eq. (1) is clearly concave. The maximization objectives in Eq. (2) and (3) are quasiconvex in the risk-averse settings[2] and the proofs are routine; we provide one such proof for completeness.

[2]Quasi-convexity is lost on the negative range of the objective $f(\mathbf{x}) = \frac{t-\boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}}$: as explained before, this situation corresponds to a *risk-loving* setting, which is mathematically different and is not the focus of this work.

2.1.1 Probability tail model

Lemma 2.3 *The function $f(\mathbf{x}) = \frac{t-\boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}}$ is quasi-convex on its positive range.*

Proof: From the definition of quasi-convexity, we have to show that for all $\mathbf{x}, \mathbf{y} \in L_\lambda$ and $\alpha \in [0,1]$, $\alpha\mathbf{x} + (1-\alpha)\mathbf{y} \in L_\lambda$, when $\lambda > 0$. To show this, we need to verify that

$$
\begin{aligned}
& \frac{t - \boldsymbol{\mu}^T[\alpha\mathbf{x} + (1-\alpha)\mathbf{y}]}{\sqrt{\boldsymbol{\tau}^T[\alpha\mathbf{x} + (1-\alpha)\mathbf{y}]}} \leq \lambda \\
\Leftrightarrow \quad & (t - \alpha\boldsymbol{\mu}^T\mathbf{x} - (1-\alpha)\boldsymbol{\mu}^T\mathbf{y})^2 \leq \alpha\lambda^2\boldsymbol{\tau}^T\mathbf{x} + (1-\alpha)\lambda^2\boldsymbol{\tau}^T\mathbf{y}. \\
\Leftrightarrow \quad & t^2 + (\alpha\mu^T\mathbf{x})^2 + ((1-\alpha)\mu^T y)^2 - 2t\alpha\mu^T\mathbf{x} - 2t(1-\alpha)\mu^T y + 2\alpha(1-\alpha)(\mu^T\mathbf{x})(\mu^T y) \leq \lambda^2\alpha\tau^T\mathbf{x} + \lambda^2\alpha\tau^T y
\end{aligned}
$$

Since $\alpha \in [0,1]$, we have $\alpha(1-\alpha) \geq 0$, hence

$$
\begin{aligned}
& -\alpha(1-\alpha)u^2 + 2\alpha(1-\alpha)uv - \alpha(1-\alpha)v^2 \leq 0 \qquad && \forall u, v \in \mathbb{R} \\
\Rightarrow \quad & \alpha^2u^2 + 2\alpha(1-\alpha)uv + (1-\alpha)^2v^2 \leq \alpha u^2 + (1-\alpha)v^2 && \forall u, v.
\end{aligned}
$$

Applying the above inequality with $u = \boldsymbol{\mu}^T\mathbf{x}$, $v = \boldsymbol{\mu}^T\mathbf{y}$, we get

$$
\begin{aligned}
& (t - \alpha\boldsymbol{\mu}^T\mathbf{x} - (1-\alpha)\boldsymbol{\mu}^T\mathbf{y})^2 \\
= \quad & t^2 + \alpha^2(\boldsymbol{\mu}^T\mathbf{x})^2 + 2\alpha(1-\alpha)(\boldsymbol{\mu}^T\mathbf{x})(\boldsymbol{\mu}^T\mathbf{y}) + (1-\alpha)^2(\boldsymbol{\mu}^T\mathbf{y})^2 - 2t\alpha\boldsymbol{\mu}^T\mathbf{x} - 2t(1-\alpha)\boldsymbol{\mu}^T\mathbf{y} \\
\leq \quad & t^2 + \alpha(\boldsymbol{\mu}^T\mathbf{x})^2 + (1-\alpha)(\boldsymbol{\mu}^T y)^2 - 2t\alpha\boldsymbol{\mu}^T\mathbf{x} - 2t(1-\alpha)\boldsymbol{\mu}^T\mathbf{y} \\
= \quad & \alpha(t - \boldsymbol{\mu}^T\mathbf{x})^2 + (1-\alpha)(t - \boldsymbol{\mu}^T\mathbf{y})^2 \\
\leq \quad & \alpha\lambda^2\boldsymbol{\tau}^T\mathbf{x} + (1-\alpha)\lambda^2\boldsymbol{\tau}^T\mathbf{y},
\end{aligned}
$$

where the last inequality follows from the fact that $\mathbf{x}, \mathbf{y} \in L_\lambda$. □

Lemma 2.4 *The function $f(\mathbf{x}) = \frac{(t-\boldsymbol{\mu}^T\mathbf{x})^2}{(t-\boldsymbol{\mu}^T\mathbf{x})^2 + \boldsymbol{\tau}^T\mathbf{x}}$ is quasi-convex on its entire range.*

The formal proof of this lemma is analogous to above. It can also be seen geometrically: the lower-level sets of this function are the epigraphs (the areas above the graphs) of upward-facing parabolas, and hence are convex.

2.1.2 Value-at-risk model

In this section we show how the value-at-risk objective reduces to the problem of minimizing a linear combination of mean and standard deviation. We first establish the equivalence under normal distributions, and then show a reduction for arbitrary distributions using Chebyshev's bound.

Lemma 2.5 *The value-at-risk model*

$$
\begin{aligned}
\textit{minimize} \quad & t \\
\textit{subject to} \quad & \Pr(\mathbf{W}^T\mathbf{x} \leq t) \geq p \\
& \mathbf{x} \in \mathcal{F}
\end{aligned}
$$

for a given probability p is equivalent to the mean-risk model

$$
\begin{aligned}
\textit{minimize} \quad & \boldsymbol{\mu}^T\mathbf{x} + c\sqrt{\boldsymbol{\tau}^T\mathbf{x}} \\
\textit{subject to} \quad & \mathbf{x} \in \mathcal{F}
\end{aligned}
$$

with $c = \Phi^{-1}(p)$, when the element costs come from independent normal distributions.

Proof: As before, $\Phi(\cdot)$ denotes the cumulative distribution function of the standard normal random variable $N(0,1)$, and $\Phi^{-1}(\cdot)$ denotes its inverse. For normally distributed costs $\mathbf{W}$ we have

$$\begin{aligned}
&\Pr(\mathbf{W}^T\mathbf{x} \leq t) \geq p \\
\Leftrightarrow\quad &\Pr\left(\frac{\mathbf{W}^T\mathbf{x} - \boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}} \leq \frac{t - \boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}}\right) \geq p \\
\Leftrightarrow\quad &\Phi\left(\frac{t - \boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}}\right) \geq p \\
\Leftrightarrow\quad &\frac{t - \boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}} \geq \Phi^{-1}(p) \\
\Leftrightarrow\quad &t \geq \boldsymbol{\mu}^T\mathbf{x} + \Phi^{-1}(p)\sqrt{\boldsymbol{\tau}^T\mathbf{x}}.
\end{aligned}$$

Because the stochastic value-at-risk problem is minimizing over both t and $\mathbf{x}$, the smallest threshold t is equal to the minimum of $\boldsymbol{\mu}^T\mathbf{x} + c\sqrt{\boldsymbol{\tau}^T\mathbf{x}}$ over the feasible set $\mathbf{x} \in \mathcal{F}$, where the constant $c = \Phi^{-1}(p)$. □

For arbitrary distributions, we can apply the one-sided Chebyshev bound $\Pr(\mathbf{W}^T\mathbf{x} \geq \boldsymbol{\mu}^T\mathbf{x} + c\sqrt{\boldsymbol{\tau}^T\mathbf{x}}) \leq \frac{1}{1+c^2}$, or equivalently $\Pr(\mathbf{W}^T\mathbf{x} < \boldsymbol{\mu}^T\mathbf{x} + c\sqrt{\boldsymbol{\tau}^T\mathbf{x}}) > 1 - \frac{1}{1+c^2}$. Taking $c = \sqrt{\frac{p}{1-p}}$ gives the inequality $\Pr(\mathbf{W}^T\mathbf{x} < \boldsymbol{\mu}^T\mathbf{x} + c\sqrt{\boldsymbol{\tau}^T\mathbf{x}}) > p$. This yields the following lemma:

Lemma 2.6 *The value-at-risk model with arbitrary distributions reduces to:*

$$\begin{aligned}
\textit{minimize}\quad & \boldsymbol{\mu}^T\mathbf{x} + \sqrt{\frac{p}{1-p}}\sqrt{\boldsymbol{\tau}^T\mathbf{x}} \\
\textit{subject to}\quad & \mathbf{x} \in \mathcal{F}
\end{aligned}$$

In particular, the optimal value of the above concave minimization problem will provide an upper bound for the minimum threshold t in the value-at-risk problem with given probability p.

We remark that in the absence of more information on the distributions, other than their means and standard deviations, the best one can do is to upper-bound the probability tail in the value-at-risk problem.

For an illustration of the difference between the above lemmas, consider the following shortest path application:

Example 2.7 *Suppose we need to reach the airport by a certain time. We want to find the minimum time (and route) that we need to allocate for our trip so as to arrive on time with probability at least $p = .95$. (That is, how close can we cut it to the deadline and not be late?) If we know that the travel times on the edges are normally distributed, the minimum time equals $\min_{\mathbf{x}\in\mathcal{F}} \boldsymbol{\mu}^T\mathbf{x} + 1.645\sqrt{\boldsymbol{\tau}^T\mathbf{x}}$, since $\Phi^{-1}(.95) = 1.645$. On the other hand, if we had no information about the distributions, we should instead allocate the upper bound $\min_{\mathbf{x}\in\mathcal{F}} \boldsymbol{\mu}^T\mathbf{x} + 4.5\sqrt{\boldsymbol{\tau}^T\mathbf{x}}$, since $\frac{1}{\sqrt{1-0.95}} \approx 4.5$ (which still guarantees that we would arrive with probability at least 95%).*

2.2 Exact algorithms

In the previous section we established that all models in our risk-averse framework reduce to instances of quasi-concave minimization (or equivalently, quasi-convex maximization). In this section, we give exact algorithms based on this property.

An exact algorithm of this nature was previously proposed for the special case of the stochastic shortest path problem [38]. This algorithm and its analysis readily extend to general problems and all objectives in our risk-averse framework. We include the generalized statement and analysis here for completeness, and as a prelude to the approximation algorithms in the next sections.

Theorem 2.8 *The optimal solution to all models in our risk-averse framework is an extreme point of the dominant[3] of the projected feasible set onto the mean-variance plane $span(\boldsymbol{\mu}, \boldsymbol{\tau})$.*

[3] The *dominant* of a set S is defined as the set of points that are coordinate-wise bigger than points in S, namely $\{\mathbf{y} \mid \mathbf{y} \geq \mathbf{x} \text{ for some } \mathbf{x} \in S\}$.

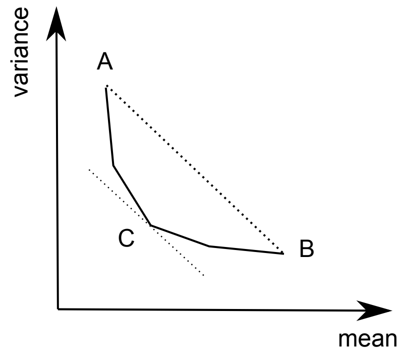


Figure 2: Enumerating extreme points.

Proof: In all models the objective functions depend only on the mean $\boldsymbol{\mu}^T\mathbf{x}$ and variance $\boldsymbol{\tau}^T\mathbf{x}$ of the feasible solution $\mathbf{x}$. Therefore, we can project the objectives and feasible set onto the mean-variance plane given by $span(\boldsymbol{\mu}, \boldsymbol{\tau})$ and work in this 2-dimensional subspace. The quasi-concavity/convexity is retained in this projected space (this follows immediately by Definition 2.1 and properties of projections [5]), and moreover the optimizer in the projected space is the projection of the optimizer in the original problem. Therefore, by Theorem 2.2, the optimal solution is an extreme point of the projected feasible set. Furthermore, this implies that the optimal solution of the relaxed continuous programs over the convex hull of the feasible set $\mathcal{F}$ is also optimal for the original discrete versions.

On the other hand, the risk-aversion in our models implies that our objective functions are monotone in the mean and variance so that the optimum is obtained at the Pareto boundary of smallest mean-variance combinations of the feasible solutions. Therefore, the optimum to each of our models is an extreme point on the dominant of the projected feasible set. $\square$

Theorem 2.8 establishes correctness of the exact algorithm for finding the optimal risk-averse solution, presented in Figure 3. The extreme point enumeration can be done in multiple ways via oracle calls to the underlying deterministic problem, for a carefully selected sequence of weight vectors as follows: All extreme points on the dominant of the projected feasible set minimize some linear objective $(\boldsymbol{\mu} + \gamma\boldsymbol{\tau})^T\mathbf{x}$ over the feasible set, for some $\gamma \geq 0$. We first find the two optimal solutions minimizing the mean $\boldsymbol{\mu}^T\mathbf{x}$ and variance $\boldsymbol{\tau}^T\mathbf{x}$. We then compute the slope of the line connecting their corresponding projections (A and B in Figure 2) on the mean-variance plane. This slope induces a new linear objective $(\boldsymbol{\mu} + \gamma_1\boldsymbol{\tau})^T\mathbf{x}$ for some γ_1 (the punctuated line parallel to AB in Figure 2) and we find the new optimal solution (represented by point C in the figure) with respect to this objective. We continue recursively to find the extreme points between A and C and between C and B. If the new returned extreme point is identical to one of the endpoints, we know that there are no further extreme points in the corresponding interval. The whole process will terminate after $2k$ deterministic oracle calls where k is the number of extreme points.

We remark that finding the extreme points in our risk-averse framework is equivalent to finding the breakpoints in a parametric optimization framework [23, 8], where for two given weight vectors $\boldsymbol{\mu}$ and $\boldsymbol{\tau}$, the goal is to find the feasible solutions minimizing the parametric cost $\boldsymbol{\mu} + \gamma\boldsymbol{\tau}$, for all values of the parameter $\gamma \in [0, \infty)$. (A breakpoint is a parameter value where the optimal solution changes.) The *parametric complexity* of this problem is defined as the number of breakpoints, and it determines the complexity of our exact algorithm. We summarize this in the following theorem.

Theorem 2.9 *There is an exact algorithm for our risk-averse optimization framework whose running time is determined by the parametric complexity of the underlying deterministic problem. In particular, the algorithm runs $2k$ oracle calls to the underlying deterministic problem, where k is the number of parametric breakpoints with respect to the parametric objective $\boldsymbol{\mu} + \gamma\boldsymbol{\tau}$, $\gamma \in [0, \infty)$.*

Corollary 2.10 *The exact algorithm for risk-averse*

1. *minimum spanning trees and matroids is polynomial.*
2. *shortest paths is $n^{O(\log n)}$.*

Problem: Maximize or minimize $f(\mathbf{x})$ over $\mathbf{x} \in \mathcal{F}$.
Output: Optimal solution $\mathbf{x} \in \mathcal{F}$
Algorithm:

1. Enumerate all extreme points on the dominant[3] of the projected feasible set $\mathcal{F}$ onto the mean-variance plane $span(\boldsymbol{\mu}, \boldsymbol{\tau})$.
2. Evaluate the objective function f at each extreme point.
3. Output the extreme point with optimal objective function value.

Figure 3: Exact algorithm for risk-averse optimization.

The result about minimum spanning trees and matroids under the *mean-risk model*, with a different line of reasoning through submodular minimization, appears in Atamtürk and Narayanan [3]. The result about shortest paths under the *probability tail model* appears in Nikolova *et al.* [38]. For many other problems of interest, the parametric complexity is exponential in the worst-case [8].

3 An FPTAS for the risk-averse framework for easy combinatorial problems

In this section, we present a general-purpose FPTAS design that applies to all models in the risk-averse framework defined in Section 2. The FPTAS uses as a black-box an exact algorithm for the underlying deterministic problem and is based on a geometric analysis of the curvature and approximability of the level sets of the objective functions. The black-box calls to the exact algorithm are made for a carefully chosen *small* set of linear objectives $\mathbf{w} \geq 0$. We remark that, in general, such a set may not even exist. For example, the necessary number of linear objectives may be large or even infinite if the objective function has unbounded gradient (as is the case in the second model above). From a complexity perspective, minimizing a concave function over some feasible set may be hard to approximate even if minimizing a linear function over the same set can be done in polynomial time [30].

As in Section 2.2, all objectives (1)-(4) can be projected onto the mean-variance plane $span(\boldsymbol{\mu}, \boldsymbol{\tau})$ and can be thought of as functions on two dimensions. The projected level sets of the objective functions on the mean-variance plane $span(\boldsymbol{\mu}, \boldsymbol{\tau})$ are parabolas. We construct an approximate separation oracle, which tells us whether for a given function value λ there is a feasible solution below the $(1-\epsilon)\lambda$-level set or else if the entire feasible set is above the λ-level set. We do this by inscribing a (partial) polygon between these two level sets. Geometrically, the optimal polygon choice (with fewest sides) is such that its vertices are on one level set and its sides are tangent to the other, as shown in Figure 1(b). The FPTAS template for a maximization problem is described more formally in Figure 4 (it is analogous for a minimization problem).

Theorem 3.1 *There is an oracle fully-polynomial time approximation scheme for all problems in our risk-averse stochastic framework, which uses as a black-box an exact algorithm for solving the underlying deterministic problem (5).*

In the rest of this section we prove this theorem. The crux of the proof is in establishing that the approximate separation oracle can be constructed from polynomially many linear segments as described in the following main technical lemma. (Lemma 3.2 is stated for a stochastic maximization problem as in Eq. (2); the analogous statement holds for a stochastic minimization problem as in Eq. (1).) The argument for how Theorem 3.1 follows from the Lemma is provided at the end of this section.

Lemma 3.2 (Approximate Nonlinear Separation Oracle) *Suppose we have an exact algorithm for solving the deterministic problem (5). Then, we can construct an oracle which solves the following approximate separation problem: given a level λ and $\epsilon \in (0,1)$, the oracle returns*

Problem: Maximize $f(\mathbf{x})$ over $\mathbf{x} \in \mathcal{F}$.
Output: Solution $\mathbf{x}'$ such that $f(\mathbf{x}') \geq (1-\epsilon) f_{max}(\mathbf{x})$
Algorithm:

1. For appropriate lower and upper bounds of $f(\cdot)$, denoted f_l and f_u respectively, apply *approximate nonlinear separation oracle* below with $\epsilon' = 1 - \sqrt{1-\epsilon}$ successively on the function values $f_u, (1-\epsilon')f_u, (1-\epsilon')^2 f_u, \ldots$ until we find a value, for which the separation oracle returns a feasible solution $\mathbf{x}'$.

2. Run the available black-box algorithm for the deterministic problem on subset of elements with zero mean, to find the smallest-variance solution among the solutions with mean zero. Compare with the solution above and return the solution with better objective function value.

Approximate Nonlinear Separation Oracle.
Input: Function value λ, approximation factor $\epsilon' > 0$; black-box access to algorithm for minimizing linear functions over $\mathbf{x} \in \mathcal{F}$.
Output:

(a) A solution $\mathbf{x}' \in \mathcal{F}$ with $f(\mathbf{x}') \geq (1-\epsilon')\lambda$, *or*

(b) An answer that $f(\mathbf{x}) < \lambda$ for all $\mathbf{x} \in \mathcal{F}$.

Algorithm:

1. Inscribe a polygon between the level sets corresponding to function values λ and $(1-\epsilon')\lambda$.

2. For each side of the polygon, minimize the induced linear objective.

3. If a resulting solution $\mathbf{x}'$ satisfies $f(\mathbf{x}') \geq (1-\epsilon)\lambda$, return $\mathbf{x}'$. Else return that $f(\mathbf{x}) < \lambda$ for all $\mathbf{x} \in \mathcal{F}$.

Figure 4: FPTAS template for solving risk-averse stochastic problems.

1. *A solution $\mathbf{x} \in \mathcal{F}$ with $f(\mathbf{x}) \geq (1-\epsilon)\lambda$,* or

2. *An answer that $f(\mathbf{x}) < \lambda$ for all $\mathbf{x} \in \mathcal{F}$,*

and the number of linear oracle calls it makes is polynomial in $\frac{1}{\epsilon}$ and the size of the input.

The proof-construction of the approximate nonlinear separation oracle in Lemma 3.2 follows from a series of lemmas about bounding the size and number of the linear segments that approximate a level set and comprise the separation oracle. Since the level sets and their position with respect to each other is different for the different objectives, the actual computations of the size and number of linear segments differs. We provide the proof for the probability tail formulation (2), which is more subtle due to the budget threshold and the fact the level sets are tangent to each other. The proofs for the remaining objectives are analogous; for completeness we provide them in the appendix for the mean-risk objective whose level sets, though still parabolas, are differently situated with respect to each other.

Consider the lower level sets $\underline{L}_\lambda = \{\mathbf{z} \mid f(\mathbf{z}) \leq \lambda\}$ of the projected probability tail objective function $f(m,s) = \frac{t-m}{\sqrt{s}}$, where $m, s \in \mathbb{R}$. Denote $L_\lambda = \{\mathbf{z} \mid f(\mathbf{z}) = \lambda\}$. We will prove that any level set boundary can be approximated by a small number of linear segments. The main work here involves deriving a condition for a linear segment with endpoints on L_λ, to have objective function values within $(1-\epsilon)$ of λ (See Fig. 5).

Lemma 3.3 *Consider the points $(m_1, s_1), (m_2, s_2) \in L_\lambda$ with $s_1 > s_2 > 0$. The segment connecting these two points is contained in the level set region $\underline{L}_\lambda \backslash \underline{L}_{\lambda(1-\epsilon)}$ whenever $s_2 \geq (1-\epsilon)^4 s_1$, for every $\epsilon \in (0,1)$.*

Proof: Any point on the segment $[(m_1, s_1), (m_2, s_2)]$ can be written as a convex combination of its endpoints, $(\alpha m_1 + (1-\alpha)m_2, \alpha s_1 + (1-\alpha)s_2)$, where $\alpha \in [0,1]$. Consider the function $h(\alpha) = f(\alpha m_1 + (1-\alpha)m_2, \alpha s_1 +$

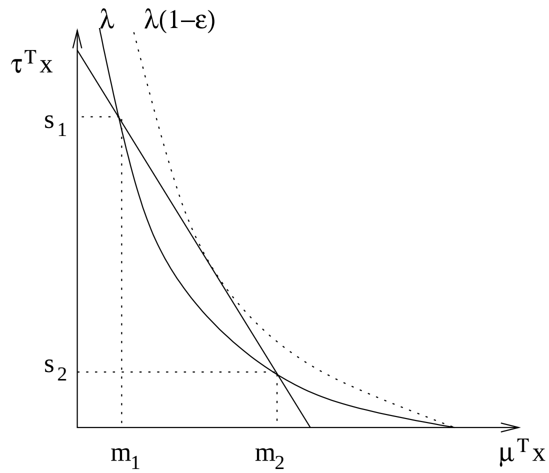


Figure 5: The objective value along a segment is not too far from the objective value at the endpoints of the segment, provided s_1 and s_2 are not too far. λ and $\lambda(1-\epsilon)$ are the objective function values along the drawn level sets.

$(1-\alpha)s_2)$. We have,

$$h(\alpha) \quad = \quad \frac{t - \alpha m_1 - (1-\alpha)m_2}{\sqrt{\alpha s_1 + (1-\alpha)s_2}} = \frac{t - \alpha(m_1 - m_2) - m_2}{\sqrt{\alpha(s_1 - s_2) + s_2}}$$

We want to find the point on the segment with smallest objective value, so we minimize with respect to α.

$$\begin{aligned} h'(\alpha) &= \frac{(m_2 - m_1)\sqrt{\alpha(s_1 - s_2) + s_2} - [t - \alpha(m_1 - m_2) - m_2] * \frac{1}{2}(s_1 - s_2)/\sqrt{\alpha(s_1 - s_2) + s_2}}{\alpha(s_1 - s_2) + s_2} \\ &= \frac{2(m_2 - m_1)[\alpha(s_1 - s_2) + s_2] - [t - \alpha(m_1 - m_2) - m_2](s_1 - s_2)}{2[\alpha(s_1 - s_2) + s_2]^{3/2}} \\ &= \frac{\alpha(m_2 - m_1)(s_1 - s_2) + 2(m_2 - m_1)s_2 - (t - m_2)(s_1 - s_2)}{2[\alpha(s_1 - s_2) + s_2]^{3/2}}. \end{aligned}$$

Setting the derivative to 0 is equivalent to setting the numerator above to 0, thus we get:

$$\alpha_{\min} = \frac{(t - m_2)(s_1 - s_2) - 2(m_2 - m_1)s_2}{(m_2 - m_1)(s_1 - s_2)} = \frac{t - m_2}{m_2 - m_1} - \frac{2s_2}{s_1 - s_2}.$$

Note that the denominator of $h'(\alpha)$ is positive and its numerator is linear in α, with a positive slope, therefore the derivative is negative for $\alpha < \alpha_{\min}$ and positive otherwise, so $\alpha_{\min}$ is indeed a global minimum as desired. In fact, $h(\alpha)$ is strictly decreasing for $\alpha < \alpha_{\min}$ and strictly increasing for $\alpha > \alpha_{\min}$, and since $h(0) = h(1) = f(m_i, s_i) = \lambda$ for $i = 1, 2$, it must be that $\alpha_{\min} \in (0, 1)$. (One can also check directly that $h'(0) < 0$ and $h'(1) > 0$.)

It remains to verify that $h(\alpha_{\min}) \geq (1-\epsilon)\lambda$. Note that $t - m_i = \lambda\sqrt{s_i}$ for $i = 1, 2$ since $(m_i, s_i) \in L_\lambda$ and consequently, $m_2 - m_1 = \lambda(\sqrt{s_1} - \sqrt{s_2})$. We use this in the following expansion of $h(\alpha_{\min})$.

$$\begin{aligned} h(\alpha_{\min}) &= \frac{t + \alpha_{\min}(m_2 - m_1) - m_2}{\sqrt{\alpha_{\min}(s_1 - s_2) + s_2}} = \frac{t + (\frac{t - m_2}{m_2 - m_1} - \frac{2s_2}{s_1 - s_2})(m_2 - m_1) - m_2}{\sqrt{(\frac{t - m_2}{m_2 - m_1} - \frac{2s_2}{s_1 - s_2})(s_1 - s_2) + s_2}} \\ &= \frac{t + t - m_2 - 2s_2 \frac{m_2 - m_1}{s_1 - s_2} - m_2}{\sqrt{(t - m_2)\frac{s_1 - s_2}{m_2 - m_1} - 2s_2 + s_2}} = \frac{2(t - m_2) - 2s_2 \frac{\lambda(\sqrt{s_1} - \sqrt{s_2})}{s_1 - s_2}}{\sqrt{\lambda\sqrt{s_2}\frac{s_1 - s_2}{\lambda(\sqrt{s_1} - \sqrt{s_2})} - s_2}} \\ &= \frac{2\lambda\sqrt{s_2} - 2s_2\frac{\lambda}{\sqrt{s_1} + \sqrt{s_2}}}{\sqrt{\sqrt{s_2}(\sqrt{s_1} + \sqrt{s_2}) - s_2}} = 2\lambda\frac{\sqrt{s_2} - \frac{s_2}{\sqrt{s_1} + \sqrt{s_2}}}{\sqrt{\sqrt{s_1 s_2}}} \\ &= 2\lambda\frac{\sqrt{s_1 s_2} + s_2 - s_2}{(s_1 s_2)^{1/4}(\sqrt{s_1} + \sqrt{s_2})} = 2\lambda\frac{(s_1 s_2)^{1/4}}{\sqrt{s_1} + \sqrt{s_2}}. \end{aligned}$$

We need to show that when the ratio s_1/s_2 is sufficiently close to 1, $h(\alpha_{\min}) \geq (1 - \epsilon)\lambda$, or equivalently

$$\frac{2(s_1 s_2)^{1/4}}{\sqrt{s_1} + \sqrt{s_2}} \geq 1 - \epsilon \qquad \Leftrightarrow \qquad 2(s_1 s_2)^{1/4} \geq (1 - \epsilon)(s_1^{1/2} + s_2^{1/2})$$

$$\Leftrightarrow \quad (1 - \epsilon)\Big(\frac{s_1}{s_2}\Big)^{1/2} - 2\Big(\frac{s_1}{s_2}\Big)^{1/4} + (1 - \epsilon) \leq 0 \tag{6}$$

The minimum of the last quadratic function above is attained at $\Big(\frac{s_1}{s_2}\Big)^{1/4} = \frac{1}{1-\epsilon}$ and we can check that at this minimum the quadratic function is indeed negative:

$$(1 - \epsilon)\Big(\frac{1}{1 - \epsilon}\Big)^2 - 2\Big(\frac{1}{1 - \epsilon}\Big) + (1 - \epsilon) = (1 - \epsilon) - \frac{1}{1 - \epsilon} < 0,$$

for all $0 < \epsilon < 1$. The inequality (6) is satisfied at $\frac{s_1}{s_2} = 1$, therefore it holds for all $\Big(\frac{s_1}{s_2}\Big) \in [1, \frac{1}{(1-\epsilon)^4}]$. Hence, a sufficient condition for $h(\alpha_{\min}) \leq (1 - \epsilon)\lambda$ is $s_2 \geq (1 - \epsilon)^4 s_1$, and we are done. $\square$

Using Lemma 3.3, we next show that any level set L_λ can be approximated within a multiplicative factor of $(1 - \epsilon)$ via a small number of segments. Let s_{min} and s_{max} be a lower and upper bound respectively for the variance of the optimal solution. For example, take s_{min} to be the smallest positive coordinate of the variance vector, and s_{max} the variance of the feasible solution with smallest mean.

Lemma 3.4 *The level set $L_\lambda = \{(m, s) \in \mathbb{R}^2 \mid \frac{t-m}{\sqrt{s}} = \lambda\}$ can be approximated within a factor of $(1 - \epsilon)$ by $\left\lceil \frac{1}{4}\log\Big(\frac{s_{\max}}{s_{\min}}\Big) / \log\frac{1}{1-\epsilon} \right\rceil$ linear segments.*

Proof: By definition of s_{min} and s_{max}, the variance of the optimal solution ranges from $s_{\min}$ to $s_{\max}$. By Lemma 3.3, the segments connecting the points on L_λ with variances $s_{\max}, s_{\max}(1 - \epsilon)^4, s_{\max}(1 - \epsilon)^8, ..., s_{\min}$ all lie in the level set region $\underline{L}_\lambda \backslash \underline{L}_{\lambda(1-\epsilon)}$, that is they underestimate and approximate the level set L_λ within a factor of $(1 - \epsilon)$. The number of these segments is $\lceil \frac{1}{4}\log\Big(\frac{s_{\max}}{s_{\min}}\Big) / \log\frac{1}{1-\epsilon} \rceil$. $\square$

The above lemma yields the approximate separation oracle for the level set L_λ and the feasible set $\mathcal{F}$, by applying the black-box algorithm for the deterministic problem to cost vectors $a\boldsymbol{\mu} + \boldsymbol{\tau}$, for all possible slopes $(-a)$ of the segments approximating the level set. This concludes the proof-construction for the separation oracle in Lemma 3.2.

We now show how to obtain a fully polynomial approximation algorithm for the nonconvex problems in our risk-averse framework by using the nonlinear separation oracle from Lemma 3.2.

Proof of Theorem 3.1: We prove the theorem for a maximization problem; the proof is analogous for a minimization problem. We first need to bound the optimum value f_{opt} of the objective function f. A lower bound f_l is provided by the solution $\mathbf{x}_{mean}$ with smallest mean or the solution $\mathbf{x}_{var}$ with smallest positive variance, whichever has a higher objective value: $f_l = \max\{f(\mathbf{x}_{mean}), f(\mathbf{x}_{var})\}$. On the other hand, $\boldsymbol{\mu}^T\mathbf{x} \geq \boldsymbol{\mu}^T\mathbf{x}_{mean}$ and $\boldsymbol{\tau}^T\mathbf{x} \geq \boldsymbol{\tau}^T\mathbf{x}_{var}$ for all $\mathbf{x} \in \mathcal{F}$, so an upper bound f_u for the objective f is given by f evaluated at $\boldsymbol{\mu}^T\mathbf{x}_{mean}$ for the mean and $\boldsymbol{\tau}^T\mathbf{x}_{var}$ for the variance.

Now, apply the approximate separation oracle from Lemma 3.2 with $\epsilon' = 1 - \sqrt{1-\epsilon}$ successively on the levels $f_u, (1-\epsilon')f_u, (1-\epsilon')^2 f_u, ...$ until we reach a level $\lambda = (1-\epsilon')^i f_u \geq f_l$ for which the oracle returns a feasible solution $\mathbf{x}'$ with

$$f(\mathbf{x}') \geq (1-\epsilon')\lambda = (\sqrt{1-\epsilon})^{i+1} f_u.$$

From running the oracle on the previous level $f_u(1-\epsilon')^{i-1}$, we know that $f(\mathbf{x}) \leq f(\mathbf{x}_{opt}) < (\sqrt{1-\epsilon})^{i-1} f_u$ for all $\mathbf{x} \in \mathcal{F}$, where $\mathbf{x}_{opt}$ denotes the optimal solution. Therefore,

$$(\sqrt{1-\epsilon})^{i+1} f_u \leq f(\mathbf{x}') \leq f(\mathbf{x}_{opt}) < (\sqrt{1-\epsilon})^{i-1} f_u, \qquad \text{and hence}$$

$$(1-\epsilon) f(\mathbf{x}_{opt}) < f(\mathbf{x}') \leq f(\mathbf{x}_{opt}).$$

So the solution $\mathbf{x}'$ gives a $(1-\epsilon)$-approximation to the optimum $\mathbf{x}_{opt}$. In the process, we run the approximate nonlinear separation oracle at most $\log\left(\frac{f_u}{f_l}\right) / \log\frac{1}{1-\epsilon'}$ times, which is polynomial in $\frac{1}{\epsilon}$ and the input size, and each separation oracle call itself makes polynomially many black-box queries to the algorithm for the deterministic problem, hence the algorithm makes polynomially many black-box queries, QED. $\square$

4 Approximating the risk-averse versions of hard combinatorial problems

In this section, we show that a δ-approximate oracle to the deterministic problem (5), which we sometimes call a linear oracle, can be used to construct efficient approximation algorithms for the risk-averse stochastic models. As in the approximative analysis for easy combinatorial problems, we first check whether the optimal solution has zero variance and if not, proceed with the algorithm and analysis below.

We can use the same approximation algorithm template that constructs a nonlinear separation oracle as in the previous section, but it turns out that a cruder algorithm which simply tests a geometric progression of mean-variance tradeoffs provides the same approximation guarantees. The main technical challenge in the algorithm analysis is that even if we know the optimal mean-variance tradeoff to query from the black-box algorithm for the deterministic problem, it is not obvious and not intuitive what approximation factor one can get for the risk-averse objectives from a δ-approximation factor for the deterministic one.

We obtain a sharp approximation result for the mean-risk objective—we can get essentially the same approximation factor as the available one for the deterministic problem:

Theorem 4.1 *Suppose we have a δ-approximation oracle for solving the deterministic combinatorial problem (5). The mean-risk model (1) can be approximated to a multiplicative factor of $\delta(1+\epsilon)$ by calling the oracle for the deterministic problem polynomially many times in the input size and $\frac{1}{\epsilon}$.*

We can also get the following approximation for the probability tail formulation (2):

Theorem 4.2 *Suppose we have a δ-approximation oracle for solving the deterministic combinatorial problem (5). The probability tail model (2) has a $\sqrt{1 - \left[\frac{\delta - (1-\epsilon^2/4)}{(2+\epsilon)\epsilon/4}\right]}$ -approximation algorithm that calls the algorithm for the deterministic problem polynomially many times in $\frac{1}{\epsilon}$ and the input size, assuming the optimal solution to (2) satisfies $\boldsymbol{\mu}^T\mathbf{x}^* \leq (1-\epsilon)t$.*

The high-level analysis for these approximation algorithms is the same; it differs in the computation of the approximation factors. Below, we present the proofs for Theorem 4.2, which are technically more subtle. The proof of Theorem 4.1 is provided in the appendix.

We first prove several geometric lemmas that enable us to derive the approximation factor. The first lemma is key for the transition from approximating a linear objective (by the algorithm for the deterministic problem) to approximating the nonconvex probability tail objective. See Figure 6 for visualizing the notation.

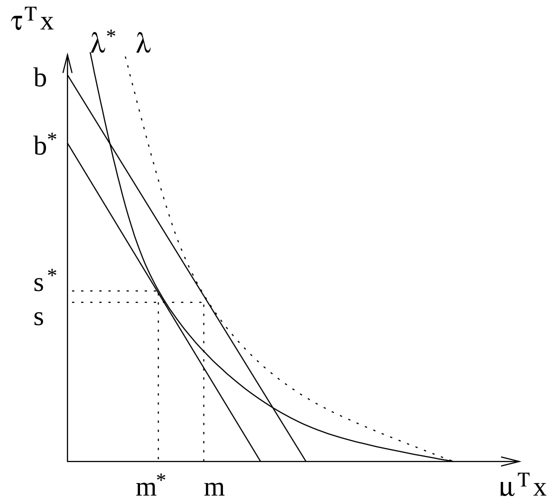


Figure 6: Applying the approximate linear oracle on the optimal linear objective (slope) gives an approximate value b of the optimal linear objective value b^*. The resulting solution has nonlinear objective function value of at least λ, which is an equally good approximation for the optimal value λ^*.

Lemma 4.3 (Geometric lemma) *Consider two objective function values $\lambda^* > \lambda$ and points $(m^*, s^*) \in L_{\lambda^*}$, $(m, s) \in L_\lambda$ with positive coordinates, such that the tangents to the points at the corresponding level sets are parallel. Then, the y-intercepts b^*, b of the two tangent lines satisfy*

$$b - b^* = s^*\Big[1 - \Big(\frac{\lambda}{\lambda^*}\Big)^2\Big].$$

Proof: Suppose the slope of the tangents is $(-a)$, where $a > 0$. Then the y-intercepts of the two tangent lines satisfy

$$b = s + am, \qquad b^* = s^* + am^*.$$

In addition, since the points (m, s) and (m^*, s^*) lie on the level sets L_λ, L_{λ^*}, they satisfy

$$t - m = \lambda\sqrt{s}, \qquad t - m^* = \lambda^*\sqrt{s^*}.$$

Since the first line is tangent at (m, s) to the parabola $y = (\frac{t-x}{\lambda})^2$, the slope equals the first derivative at this point, $-\frac{2(t-x)}{\lambda^2}|_{x=m} = -\frac{2(t-m)}{\lambda^2} = -\frac{2\lambda\sqrt{s}}{\lambda^2} = -\frac{2\sqrt{s}}{\lambda}$, so the absolute value of the slope is $a = \frac{2\sqrt{s}}{\lambda}$. Similarly the absolute value of the slope also satisfies $a = \frac{2\sqrt{s^*}}{\lambda^*}$, therefore

$$\sqrt{s^*} = \frac{\lambda^*}{\lambda}\sqrt{s}.$$

Note that for $\lambda^* > \lambda$, this means that $s^* > s$. From here, we can represent the difference $m - m^*$ as

$$m - m^* \quad = \quad (t - m^*) - (t - m) = \lambda^*\sqrt{s^*} - \lambda\sqrt{s} = \frac{(\lambda^*)^2}{\lambda}\sqrt{s} - \lambda\sqrt{s} = \Big[\Big(\frac{\lambda^*}{\lambda}\Big)^2 - 1\Big]\lambda\sqrt{s}.$$

Substituting the slope $a = \frac{2\sqrt{s}}{\lambda}$ in the tangent line equations, we get

$$\begin{aligned}
b - b^* &= s + \frac{2\sqrt{s}}{\lambda}m - s^* - \frac{2\sqrt{s}}{\lambda}m^* \\
&= s - \Big(\frac{\lambda^*}{\lambda}\Big)^2 s + \frac{2\sqrt{s}}{\lambda}(m - m^*) \\
&= s - \Big(\frac{\lambda^*}{\lambda}\Big)^2 s + \frac{2\sqrt{s}}{\lambda}\lambda\sqrt{s}\Big[\Big(\frac{\lambda^*}{\lambda}\Big)^2 - 1\Big] \\
&= s - \Big(\frac{\lambda^*}{\lambda}\Big)^2 s + 2s\Big[\Big(\frac{\lambda^*}{\lambda}\Big)^2 - 1\Big] \\
&= s\Big[\Big(\frac{\lambda^*}{\lambda}\Big)^2 - 1\Big] = s^*\Big[1 - \Big(\frac{\lambda}{\lambda^*}\Big)^2\Big],
\end{aligned}$$

as desired. □

The next lemma shows that if we know the optimal linear objective to use with the available δ-approximate algorithm for the deterministic problem (5), then we can approximate the optimal solution well.

Lemma 4.4 (Optimal Linear Objective Lemma) *Suppose we have a δ-approximate linear oracle for optimizing over the feasible set $\mathcal{F}$ and suppose that the optimal solution satisfies $\boldsymbol{\mu}^T\mathbf{x}^* \leq (1-\epsilon)t$. If we can guess the slope of the tangent to the corresponding level set at the optimal point $\mathbf{x}^*$, then we can find a $\sqrt{1 - \delta\frac{2-\epsilon}{\epsilon}}$-approximate solution to the nonconvex problem (2).*

In particular, setting $\epsilon = \sqrt{\delta}$ gives a $(1 - \sqrt{\delta})$-approximate solution.

Proof: Denote the projection of the optimal point $\mathbf{x}^*$ on the plane by $(m^*, s^*) = (\boldsymbol{\mu}^T\mathbf{x}^*, \boldsymbol{\tau}^T\mathbf{x}^*)$. We apply the linear oracle with respect to the slope $(-a)$ of the tangent to the level set L_{λ^*} at (m^*, s^*). The value of the linear objective at the optimum is $b^* = s^* + am^*$, which is the y-intercept of the tangent line. The linear oracle returns a δ-approximate solution, that is a solution on a parallel line with y-intercept $b \leq \delta b^*$. Suppose the original (nonlinear) objective value at the returned solution is lower-bounded by λ, that is it lies on a line tangent to L_λ (See Figure 6). From Lemma 4.3, we have $b - b^* = s^*\left[1 - \left(\frac{\lambda}{\lambda^*}\right)^2\right]$, therefore

$$\Big(\frac{\lambda}{\lambda^*}\Big)^2 = 1 - \frac{b - b^*}{s^*} = 1 - \left(\frac{b - b^*}{b^*}\right)\frac{b^*}{s^*} \geq 1 - \delta\frac{b^*}{s^*}. \tag{7}$$

Recall that $b^* = s^* + m^*\frac{2\sqrt{s^*}}{\lambda^*}$ and $m^* \leq (1-\epsilon)t$, then

$$\frac{b^*}{s^*} = 1 + \frac{2m^*}{\lambda^*\sqrt{s^*}} = 1 + \frac{2m^*}{t - m^*} \leq 1 + \frac{2m^*}{\frac{\epsilon}{1-\epsilon}m^*} = 1 + \frac{2(1-\epsilon)}{\epsilon} = \frac{2-\epsilon}{\epsilon}.$$

Together with Eq. (7), this gives a $\sqrt{1 - \delta\frac{2-\epsilon}{\epsilon}}$-approximation factor to the optimal.

On the other hand, setting $\epsilon = \sqrt{\delta}$ gives the approximation factor $\sqrt{1 - \delta\frac{2-\sqrt{\delta}}{\sqrt{\delta}}} = 1 - \sqrt{\delta}$. □

Next, we prove a geometric lemma that will be needed to analyze the approximation factor we get when applying the linear oracle on an approximately optimal slope. (See Fig. 7 for some of the notation.)

Lemma 4.5 *Consider the level set L_λ and points (m^*, s^*) and (m, s) on it, at which the tangents to L_λ have slopes $-a$ and $-a(1+\xi)$ respectively. Let the y-intercepts of the tangent line at (m, s) and the line parallel to it through (m^*, s^*) be b_1 and b respectively. Then $\frac{b}{b_1} \leq \frac{1}{1-\xi^2}$.*

Proof: The equation of the level set L_λ is $y = (\frac{t-x}{\lambda})^2$ so the slope at a point $(m, s) \in L_\lambda$ is given by the derivative at $x = m$, that is $-\frac{2(t-m)}{\lambda^2} = -\frac{2\sqrt{s}}{\lambda}$. So, the slope of the tangent to the level set L_λ at point (m^*, s^*) is $-a = -\frac{2\sqrt{s^*}}{\lambda}$. Similarly the slope of the tangent at (m, s) is $-a(1+\xi) = -\frac{2\sqrt{s}}{\lambda}$. Therefore, $\sqrt{s} = (1+\xi)\sqrt{s^*}$, or equivalently $(t - m) = (1+\xi)(t - m^*)$.

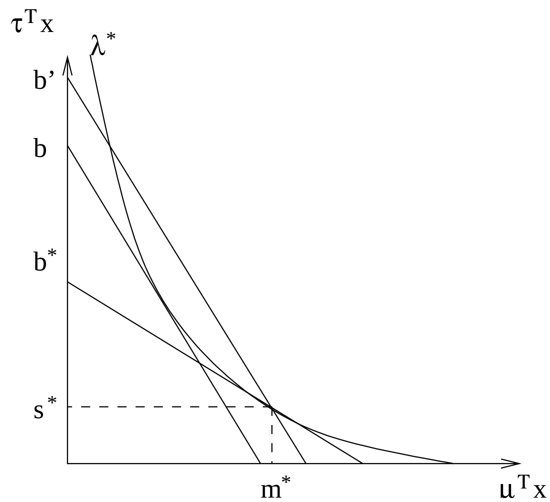


Figure 7: Applying the linear oracle with an approximate linear function (slope) still gives a solution with good approximate objective function value.

Since b, b_1 are intercepts with the y-axis, of the lines with slopes $-a(1+\xi) = -\frac{2\sqrt{s}}{\lambda}$ containing the points (m^*, s^*), (m, s) respectively, we have

$$\begin{aligned} b_1 &= s + \frac{2\sqrt{s}}{\lambda} m = \frac{t^2 - m^2}{\lambda^2} \\ b &= s^* + (1+\xi)\frac{2\sqrt{s^*}}{\lambda} m^* = \frac{t - m^*}{\lambda^2}(t + m^* + 2\xi m^*). \end{aligned}$$

Therefore

$$\begin{aligned} \frac{b}{b_1} &= \frac{(t-m^*)(t+m^*+2\xi m^*)}{(t-m)(t+m)} = \frac{1}{1+\xi}\frac{t+m^*+2\xi m^*}{t+m} = \frac{1}{1+\xi}\frac{t+(1+2\xi)m^*}{(1-\xi)t+(1+\xi)m^*} \\ &\leq \frac{1}{1+\xi}\left(\frac{1}{1-\xi}\right) = \frac{1}{1-\xi^2}, \end{aligned}$$

where we use $m = t - (1+\xi)(t - m^*)$ from above. □

We now show that we get a good approximation even when we use an approximately optimal linear objective with our linear oracle.

Lemma 4.6 *Suppose that we use an approximately optimal linear objective with a δ-approximate linear oracle for solving the probability tail model (2). In particular, suppose the linear objective (slope) that we use is within $(1+\xi)$ of the slope of the tangent at the optimal solution. Then this will give a solution to the probability tail model (2) with value at least $\sqrt{1 - \left[\frac{\delta}{1-\xi^2} - 1\right]\frac{2-\epsilon}{\epsilon}}$ times the optimal, provided the optimal solution satisfies $\boldsymbol{\mu}^T\mathbf{x}^* \leq (1-\epsilon)t$.*

Proof: Suppose the optimal solution is (m^*, s^*) and it lies on the optimal level set λ^* (see Figure 8). Let the slope of the tangent to the level set boundary at the optimal solution be $(-a)$. We apply our δ-approximation linear oracle with respect to a slope that is $(1+\xi)$ times the optimum slope $(-a)$. Suppose the resulting black box solution lies on the line with y-intercept b_2, and the true optimum lies on the line with y-intercept b'. We know $b' \in [b_1, b]$, where b_1 and b are the y-intercepts of the lines with slope $-(1+\xi)a$ that are tangent to L_{λ^*} and pass through (m^*, s^*) respectively. Then we have $\frac{b_2}{b} \leq \frac{b_2}{b'} \leq \delta$.

Furthermore, by Lemma 4.5 we have $\frac{b}{b_1} \leq \frac{1}{1-\xi^2}$.

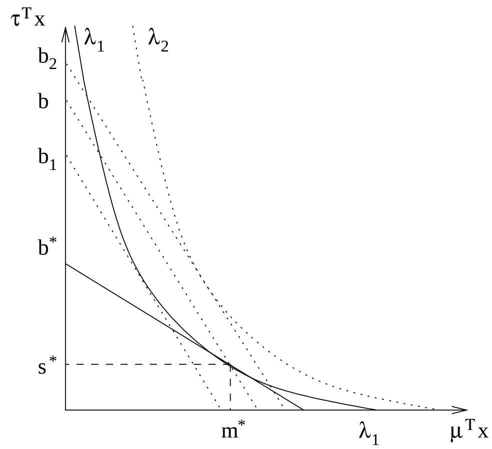


Figure 8: Approximating the objective value λ_1 of the optimal solution (m^*, s^*).

On the other hand, from Lemma 4.3, $b_2 - b_1 = s[1 - (\frac{\lambda_2}{\lambda^*})]$, where λ_2 is the smallest possible objective function value along the line with slope $-a(1+\xi)$ and y-intercept b_2, in other words the smallest possible objective function value that the solution returned by the approximate linear oracle may have; (m, s) is the tangent point of the line with slope $-(1+\xi)a$, tangent to L_{λ^*}.

Therefore, applying the above inequalities, we get

$$\left(\frac{\lambda_2}{\lambda^*}\right)^2 = 1 - \frac{b_2 - b_1}{s} = 1 - \frac{b_2 - b_1}{b_1}\frac{b_1}{s} = 1 - \left(\frac{b_2}{b}\frac{b}{b_1} - 1\right)\frac{b_1}{s} \geq 1 - \left(\frac{\delta}{1-\xi^2} - 1\right)\frac{2-\epsilon}{\epsilon},$$

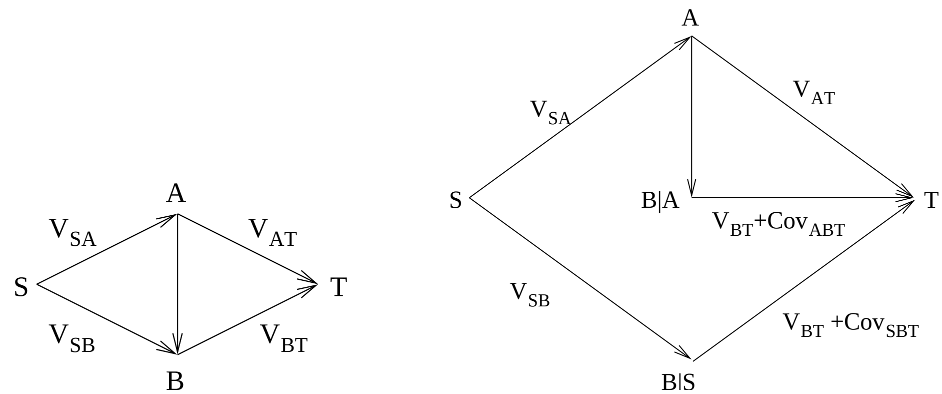
where $\frac{b_1}{s} \leq \frac{2-\epsilon}{\epsilon}$ follows as in the proof of Lemma 4.4. The result follows. $\square$

Finally, we are ready to give the approximation algorithm and its analysis in the proof of our main theorem:

*Proof of Theorem 4.2:*The algorithm applies the linear approximation oracle with respect to a small number of linear functions, and chooses the best resulting solution. In particular, suppose the optimal slope (tangent to the corresponding level set at the optimal solution point) lies in the interval $[L, U]$ (for lower and upper bound). We find approximate solutions with respect to the slopes $L, L(1+\xi), L(1+\xi)^2, ..., L(1+\xi)^k \geq U$, namely we apply the approximate linear oracle $\frac{\log(U/L)}{\log(1+\xi)}$ times, where $\xi = \frac{\epsilon^3}{2(1+\epsilon^3)}$. With this, we are certain that the optimal slope will lie in some interval $[L(1+\xi)^i, L(1+\xi)^{i+1}]$ and by Lemma 4.6 the solution returned by the linear oracle with respect to slope $L(1+\xi)^{i+1}$ will give a $\sqrt{1 - \left[\frac{\delta}{1-\xi^2} - 1\right]\frac{2-\epsilon}{\epsilon}}$- approximation to our nonlinear objective function value. Since we are free to choose ξ, setting it to $\xi = \epsilon/2$ gives the desired number of queries.

We conclude the proof by noting that we can take L to be the slope tangent to the corresponding level set at (m_L, s_L) where s_L is the minimum positive coordinate of the variance vector and $m_L = t(1-\epsilon)$. Similarly let U be the slope tangent at (m_U, s_U) where $m_U = 0$ and s_U is the sum of coordinate of the variance vector. $\square$

When $\delta = 1$, that is when we can solve the underlying linear problem exactly in polynomial time, the above algorithm gives an approximation factor of $\sqrt{\frac{1}{1+\epsilon/2}}$, or equivalently $1 - \epsilon'$, where $\epsilon = 2[\frac{1}{(1-\epsilon')^2} - 1]$. While this algorithm is still an oracle-fully polynomial time approximation scheme, it gives a bi-criteria approximation: it requires that there is a small gap between the mean of the optimal solution and the budget t so it is weaker than our previous algorithm from Section 3, which had no such requirement. This is expected since, of course, this algorithm is cruder, simply taking a geometric progression of linear functions rather than tailoring the black-box algorithm calls for the deterministic problem to the objective function value that it is searching for, as does the approximate separation oracle that the FPTAS from the previous section is based on.



(a) Original graph with correlated adjacent edges

(b) Transformed graph with independent edges

Figure 9: Solution with correlated adjacent edges.

5 Extensions: correlations

Our study of the risk-averse optimization framework presented here was motivated by route planning problems. Clearly, in a route planning application, one cannot assume that the edge delays are independently distributed: for example, an accident in one edge would increase congestion in the edges that follow it. On the other hand, it is reasonable to assume that the delay on an edge affects and is affected by other nearby edges. In such situations, our results can be readily extended with an appropriate graph transformation used in belief-propagation.[4] For clarity, we describe the transformation when only adjacent edges are pairwise correlated; one can deduce the analogous transformation when up to a constant number of consecutive edges can be pairwise correlated.

Suppose there are pairwise correlations between adjacent edges (the first one incoming and the second outgoing from their common node). Consider the following graph transformation. For every node B with incoming edges (A, B) in the original graph G, create nodes $B|A$ in the new graph G'. An edge (A, B) in G yields edges $(A|X, B|A)$ in G', for all nodes X that precede node A in G. Denote the covariance between edges (A, B) and (B, T) in G by Cov_{ABT}, and their variances by V_{AB} and V_{BT} respectively. Then in the transformed graph G', define the variance of edge $(B|A, T|B)$ by $V_{BT} + Cov_{ABT}$ as in Figure 9. Notice that these definitions of variance decouple the correlations so now the edge distributions are independent. We can thus run our existing algorithms on G' and thus solve the problems for correlated edges in the original graph G. We can apply this method of decoupling correlated edges for not just correlations among two neighboring edges, but up to a constant number of consecutive edges (in order to maintain polynomial size for the transformed graph G').

6 Conclusion

We have presented a framework for risk-averse stochastic combinatorial optimization that includes mean-risk minimization and models involving the probability tail of the stochastic cost of a solution. Our algorithms are independent of the feasible set structure and use solutions for the underlying linear (deterministic) problems as oracles for solving the corresponding stochastic models. As such, they apply to very general combinatorial settings for which *exact* or *approximate* linear oracles are available.

Our primary motivation for this work was to design an approximation algorithm for finding the most reliable route in a network with uncertain edge delays (in the sense that the route maximizes the probability of arriving on time under a given deadline), which consequently extended to the rich class of problems and risk-averse models considered here. An implementation of our approximation algorithm in the context of finding risk-averse routes reveals that they

[4] We thank Alexander Hartemink [24] for pointing this out and telling us the transformation.

are also very practical: for example, they achieve 99.9%-accuracy with only up to 6 iterations of an algorithm for the deterministic problem.

In future work, it would be interesting to extend our offline stochastic models to online models, as has previously been done with offline linear to online linear problems [28, 27]. It would be also useful to consider adaptive stochastic models, building on the framework of multistage stochastic optimization.

Other open directions include considering convex risk-measures such as the ones described in Rockafellar [42] that have been analyzed in continuous settings. We note that although the models in this paper are nonconvex, this nonconvexity (concavity) is beneficial because it preserves integrality of the desired solution. This is not true for convex objectives: convex *discrete* optimization is yet another challenging and exciting area of research.

Acknowledgements The author thanks Hari Balakrishnan, Dimitris Bertsimas, Shuchi Chawla, Costis Daskalakis, Alexander Hartemink, Nick Harvey, Michel Goemans, David Karger, David Kempe, Asu Ozdaglar, Christos Papadimitriou, Pablo Parrilo, Andreas Schulz, David Shmoys, Nicolás Stier Moses, Madhu Sudan and John Tsitsiklis for valuable suggestions and feedback at various stages of this work, as well as Matthew Fontana for the careful proofreading of this manuscript.

References

[1] H. Ackermann, A. Newman, H. Röglin, and B. Vöcking. Decision making based on approximate and smoothed pareto curves. In *Proc. of 16th ISAAC*, pages 675–684, 2005.

[2] S. Agrawal, Y. Ding, A. Saberi, and Y. Ye. Price of correlations in stochastic optimization. *Manuscript*, 2010.

[3] A. Atamtürk and V. Narayanan. Polymatroids and risk minimization in discrete optimization. *Operations Research Letters*, 36:618–622, 2008.

[4] Y. Berstein, J. Lee, S. Onn, and R. Weismantel. Nonlinear optimization for matroid intersection and extensions. *Manuscript at arXiv:0807.3907*, 2008.

[5] D. Bertsekas, A. Nedić, and A. Ozdaglar. *Convex Analysis and Optimization*. Athena Scientific, Belmont, MA, 2003.

[6] D. Bertsimas, D. Brown, and C. Caramanis. Theory and applications of robust optimization. *Manuscript*, 2007.

[7] D. Bertsimas and M. Sim. Robust discrete optimization and network flows. *Manuscript*, 2004.

[8] P. Carstensen. *The complexity of some problems in parametric linear and combinatorial programming*. Ph.D. Dissertation, Mathematics Dept., Univ. of Michigan, Ann Arbor, Mich., 1983.

[9] A. Chen and Z. Ji. Path finding under uncertainty. *Journal of advanced transportation*, 39(1):19–37, 2005.

[10] B. Dean, M. X. Goemans, and J. Vondrák. Approximating the stochastic knapsack: the benefit of adaptivity. In *Proceedings of the 45th Annual Symposium on Foundations of Computer Science*, pages 208–217, 2004.

[11] E. Delage and Y. Ye. Distributionally robust optimization under moment uncertainty with application to data-driven problems. *Oper. Res.*, 58(3):595–612, 2010.

[12] D. Dentcheva, A. Prékopa, and A. Ruszczyński. Bounds for probabilistic integer programming problems. *Discrete Appl. Math.*, 124:55–65, December 2002.

[13] E. Erdoan and G. Iyengar. On two-stage convex chance constrained problems. *Mathematical Methods of Operations Research*, 65:115–140, 2007. 10.1007/s00186-006-0104-2.

[14] Y. Fan, R. Kalaba, and I. J. E. Moore. Arriving on time. *Journal of Optimization Theory and Applications*, 127(3):497–513, 2005.

[15] Federal Highway Administration. *Traffic Congestion and Reliability: Trends and advanced strategies for congestion mitigation*. Cambridge Systematics Inc. and Texas Transportation Institute, 2005.

[16] H. Föllmer and A. Schied. *Stochastic Finance: An Introduction in Discrete Time*. Walter de Gruyter, 2004.

[17] L. E. Ghaoui, M. Oks, and F. Oustry. Worst-case value-at-risk and robust portfolio optimization: A conic programming approach. *Oper. Res.*, 51(4):543–556, 2003.

[18] A. Goel and P. Indyk. Stochastic load balancing and related problems. In *Proceedings of the 40th Symposium on Foundations of Computer Science*, 1999.

[19] V. Goyal. An FPTAS for minimizing a class of quasi-concave functions over a convex set. Technical Report Tepper WP 2008-E24, Carnegie Mellon University Tepper School of Business, 2008.

[20] G. Grimmett and D. Stirzaker. *Probability and Random Processes, 3rd ed.* Oxford Univ. Press, 2001.

[21] A. Gupta, M. Pál, R. Ravi, and A. Sinha. Boosted sampling: approximation algorithms for stochastic optimization. In *Proceedings of the 36th Annual ACM Symposium on Theory of Computing*, pages 365–372, Chicago, IL, USA, 2004.

[22] A. Gupta, M. Pál, R. Ravi, and A. Sinha. What about Wednesday? Approximation algorithms for multistage stochastic optimization. In *Proceedings of 8th APPROX*, pages 86–98, 2005.

[23] D. Gusfield. Parametric combinatorial computing and a problem of program module distribution. *Journal of the ACM*, 30(3):551–563, 1983.

[24] A. Hartemink. Personal communication. 2008.

[25] R. Horst, P. M. Pardalos, and N. V. Thoai. *Introduction to Global Optimization*. Kluwer Academic Publishers, Dordrecht, The Netherlands, 2000.

[26] N. Immorlica, D. Karger, M. Minkoff, and V. S. Mirrokni. On the costs and benefits of procrastination: approximation algorithms for stochastic combinatorial optimization problems. In *Proceedings of the fifteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 691–700, 2004.

[27] S. M. Kakade, A. T. Kalai, and K. Ligett. Playing games with approximation algorithms. In *STOC '07: Proceedings of the thirty-ninth annual ACM symposium on Theory of computing*, pages 546–555, New York, NY, USA, 2007. ACM.

[28] A. Kalai and S. Vempala. Efficient algorithms for on-line optimization. *Journal of Computer and System Sciences*, 71:291–307, 2005.

[29] I. Katriel, C. Kenyon-Mathieu, and E. Upfal. Commitment under uncertainty: Two-stage stochastic matching problems. *Theor. Comput. Sci.*, 408(2-3):213–223, 2008.

[30] J. A. Kelner and E. Nikolova. On the hardness and smoothed complexity of quasi-concave minimization. In *FOCS '07: Proceedings of the 48th Annual IEEE Symposium on Foundations of Computer Science*, pages 472–482, Washington, DC, USA, 2007. IEEE Computer Society.

[31] J. Kleinberg, Y. Rabani, and É. Tardos. Allocating bandwidth for bursty connections. *SIAM Journal on Computing*, 30(1):191–217, 2000.

[32] L. Lovász. Submodular functions and convexity. In *Mathematical Programming—The State of the Art, A. Bachem, M, Grőtschel and B. Korte, eds.*, pages 235–257. Springer-Verlag, 1983.

[33] Y. (Marco), Nie and X. Wu. Shortest path problem considering on-time arrival probability. *Transportation Research Part B: Methodological*, 43(6):597–613, 2009.

[34] H. M. Markowitz. *Mean-Variance Analysis in Portfolio Choice and Capital Markets*. Basil Blackwell, Cambridge, MA, 1987.

[35] N. Megiddo. Combinatorial optimization with rational objective functions. *Mathematics of Operations Research*, 4:414–424, 1979.

[36] E. Miller-Hooks and H. Mahmassani. Path comparisons for a priori and time-adaptive decisions in stochastic, time-varying networks. *European Journal of Operational Research*, 146(1):67–82, April 2003.

[37] A. Nemirovski and A. Shapiro. Convex approximations of chance constrained programs. *SIAM Journal on Optimization*, 17(4):969–996, 2006.

[38] E. Nikolova, J. A. Kelner, M. Brand, and M. Mitzenmacher. Stochastic shortest paths via quasi-convex maximization. In *Lecture Notes in Computer Science 4168 (ESA 2006)*, pages 552–563, Springer-Verlag, 2006.

[39] S. Onn. Convex discrete optimization. *Encyclopedia of Optimization*, pages 513–550, 2009.

[40] C. H. Papadimitriou and M. Yannakakis. On the approximability of trade-offs and optimal access of web sources. In *Proceedings of the 41st Annual Symposium on Foundations of Computer Science*, pages 86–92, Washington, DC, USA, 2000.

[41] T. Radzik. Newton's method for fractional combinatorial optimization. In *Proceedings of the 33rd Annual Symposium on Foundations of Computer Science*, pages 659–669, 1992.

[42] R. T. Rockafellar. Coherent approaches to risk in optimization under uncertainty. In *Tutorials in Operations Research INFORMS*, pages 38–61, 2007.

[43] H. Safer, J. B. Orlin, and M. Dror. Fully polynomial approximation in multi-criteria combinatorial optimization. *MIT Working Paper*, February 2004.

[44] A. Shapiro, D. Dentcheva, and A. Ruszczyński. *Lectures on Stochastic Programming: Modeling and Theory*. SIAM, Philadelphia, PA, USA, 2009.

[45] D. B. Shmoys and C. Swamy. An approximation scheme for stochastic linear programming and its application to stochastic integer programs. *Journal of the ACM*, 53(6):978–1012, 2006.

[46] C. E. Sigal, A. A. B. Pritsker, and J. J. Solberg. The stochastic shortest route problem. *Operations Research*, 28(5):1122–1129, 1980.

[47] A. Srinivasan. Approximation algorithms for stochastic and risk-averse optimization. In *SODA '07: Proceedings of the eighteenth annual ACM-SIAM symposium on Discrete algorithms*, pages 1305–1313, Philadelphia, PA, USA, 2007.

[48] C. Swamy. Risk-averse stochastic optimization: Probabilistically-constrained models and algorithms for black-box distributions. In *SODA '11: Proceedings of the annual ACM-SIAM symposium on Discrete algorithms*, Philadelphia, PA, USA, 2011. Society for Industrial and Applied Mathematics. forthcoming.

[49] C. Swamy and D. B. Shmoys. Approximation algorithms for 2-stage stochastic optimization problems. *ACM SIGACT News*, 37(1):33–46, 2006.

[50] A. Warburton. Approximation of pareto optima in multiple-objective, shortest-path problems. *Oper. Res.*, 35(1):70–79, 1987.

A Proof of Theorem 3.1 for mean-risk objective (FPTAS for easy combinatorial problems)

Similarly to the probability tail objective, we construct a nonlinear separation oracle by approximating a level set with a polygon whose sides induce linear objectives. Geometrically, the optimal choice of linear objectives is determined by drawing segments starting from one endpoint of the level set L_λ and repeatedly drawing tangents to the level set $L_{(1+\epsilon)\lambda}$.

In order to establish that the resulting linear segments are few, we first show that the tangent-segments to $L_{(1+\epsilon)\lambda}$ starting at the endpoints of L_λ are sufficiently long.

Lemma A.1 *Consider points (m_1, s_1) and (m_2, s_2) on L_λ with $0 \leq m_1 < m_2 \leq \lambda$ such that the segment with these endpoints is tangent to $L_{(1+\epsilon)\lambda}$ at the point $\alpha(m_1, s_1) + (1-\alpha)(m_2, s_2)$. Then, $\alpha = \frac{c^2}{4}\frac{s_1 - s_2}{(m_2 - m_1)^2} - \frac{s_2}{s_1 - s_2}$ and the objective value at the tangent point is $\left[\frac{c^2}{4}\frac{s_1 - s_2}{m_2 - m_1} + s_2\frac{m_2 - m_1}{s_1 - s_2} + m_2\right]$.*

Proof: Let $\bar{f} : \mathbb{R}^2 \to \mathbb{R}$, $\bar{f}(m, s) = m + c\sqrt{s}$ be the projection of the objective $f(\mathbf{x}) = \boldsymbol{\mu}^T\mathbf{x} + c\sqrt{\boldsymbol{\tau}^T\mathbf{x}}$. The objective values along the segment with endpoints (m_1, s_1), (m_2, s_2) are given by

$$h(\alpha) = \bar{f}\big(\alpha(m_1, s_1) + (1-\alpha)(m_2, s_2)\big) = \alpha(m_1 - m_2) + m_2 + c\sqrt{\alpha(s_1 - s_2) + s_2},$$

for $\alpha \in [0, 1]$. The point along the segment with maximum objective value (that is, the tangent point to the minimum level set bounding the segment) is found by setting the derivative $h'(\alpha) = m_1 - m_2 + c\frac{s_1 - s_2}{2\sqrt{\alpha(s_1 - s_2) + s_2}}$, to zero:

$$\begin{aligned}
& m_2 - m_1 = c\frac{s_1 - s_2}{2\sqrt{\alpha(s_1 - s_2) + s_2}} \\
\Leftrightarrow \quad & \sqrt{\alpha(s_1 - s_2) + s_2} = c\frac{s_1 - s_2}{2(m_2 - m_1)} \\
\Leftrightarrow \quad & \alpha(s_1 - s_2) + s_2 = c^2\frac{(s_1 - s_2)^2}{4(m_2 - m_1)^2} \\
\Leftrightarrow \quad & \alpha(s_1 - s_2) = c^2\frac{(s_1 - s_2)^2}{4(m_2 - m_1)^2} - s_2 \\
\Leftrightarrow \quad & \alpha = c^2\frac{s_1 - s_2}{4(m_2 - m_1)^2} - \frac{s_2}{s_1 - s_2}.
\end{aligned}$$

This is a maximum, since the derivative $h'(\alpha)$ is decreasing in α. The objective value at the maximum is

$$\begin{aligned}
h(\alpha_{\max}) &= \alpha_{\max}(m_1 - m_2) + m_2 + c\sqrt{\alpha_{\max}(s_1 - s_2) + s_2} \\
&= \left[c^2\frac{s_1 - s_2}{4(m_2 - m_1)^2} - \frac{s_2}{s_1 - s_2}\right](m_1 - m_2) + m_2 + c^2\frac{s_1 - s_2}{2(m_2 - m_1)} \\
&= -\frac{c^2}{4}\frac{s_1 - s_2}{m_2 - m_1} - s_2\frac{m_1 - m_2}{s_1 - s_2} + m_2 + \frac{c^2}{2}\frac{s_1 - s_2}{m_2 - m_1} \\
&= \frac{c^2}{4}\frac{s_1 - s_2}{m_2 - m_1} + s_2\frac{m_2 - m_1}{s_1 - s_2} + m_2.
\end{aligned}$$

Further, since $s_1 = (\frac{\lambda - m_1}{c})^2$ and $s_2 = (\frac{\lambda - m_2}{c})^2$, their difference satisfies $s_1 - s_2 = \frac{1}{c^2}(m_2 - m_1)(2\lambda - m_1 - m_2)$, so $\frac{s_1 - s_2}{m_2 - m_1} = \frac{2\lambda - m_1 - m_2}{c^2}$ and the above expression for the maximum function value on the segment becomes

$$h(\alpha_{\max}) = \frac{c^2}{4}\frac{2\lambda - m_1 - m_2}{c^2} + \frac{c^2 s_2}{2\lambda - m_1 - m_2} + m_2 = \frac{2\lambda - m_1 - m_2}{4} + \frac{(\lambda - m_2)^2}{2\lambda - m_1 - m_2} + m_2.$$

$\square$

Now we can show that the tangent segments at the ends of the level set L_λ are long.

Lemma A.2 *Consider the endpoint $(m_2, s_2) = (\lambda, 0)$ of L_λ. Then either the single segment connecting the two endpoints of L_λ is entirely below the level set $L_{(1+\epsilon)\lambda}$, or the other endpoint of the segment tangent to $L_{(1+\epsilon)\lambda}$ is $(m_1, s_1) = (\lambda(1-4\epsilon), (\frac{4\epsilon\lambda}{c})^2)$.*

Proof: Since $0 \le m_1 < \lambda$, we can write $m_1 = \beta\lambda$ for some $\beta \in [0, 1)$. Consequently, $s_1 = (\frac{\lambda - m_1}{c})^2 = \frac{\lambda^2(1-\beta)^2}{c^2}$ and $\frac{s_1 - s_2}{m_2 - m_1} = \frac{\lambda^2(1-\beta)^2}{c^2\lambda(1-\beta)} = \frac{\lambda(1-\beta)}{c^2}$. By Lemma A.1, the objective value at the tangent point is

$$\frac{c^2}{4}\frac{\lambda(1-\beta)}{c^2} + \lambda = \lambda\left(\frac{1-\beta}{4} + 1\right) = (1+\epsilon)\lambda.$$

The last equality follows by our assumption that the tangent point lies on the $L_{(1+\epsilon)\lambda}$ level set. Hence, $\beta = 1 - 4\epsilon$, so $m_1 = (1-4\epsilon)\lambda$ and $s_1 = (\frac{\lambda - m_1}{c})^2 = (\frac{4\epsilon\lambda}{c})^2$. $\square$

Next, we characterize the segments with endpoints on L_λ that are tangent to the level set $L_{\lambda(1+\epsilon)}$.

Lemma A.3 *Consider two points (m_1, s_1), (m_2, s_2) on L_λ with $0 \le m_1 < m_2 \le \lambda$ such that the segment connecting the two points is tangent to $L_{(1+\epsilon)\lambda}$. Then, $\frac{s_1}{s_2} \ge (1+2\epsilon)^2$.*

Proof: Let point (m, s) on the segment with endpoints (m_1, s_1), (m_2, m_2) be the tangent point to the level set $L_{(1+\epsilon)\lambda}$. Then the slope $\frac{s_1 - s_2}{m_1 - m_2}$ of the segment is equal to the derivative of the function $y = (\frac{(1+\epsilon)\lambda - x}{c})^2$ at $x = m$, which is $-2\frac{(1+\epsilon)\lambda - m}{c^2} = -\frac{2\sqrt{s}}{c}$. Since $\frac{s_1 - s_2}{m_1 - m_2} = \frac{s_1 - s_2}{(\lambda - m_2) - (\lambda - m_1)} = \frac{s_1 - s_2}{c(\sqrt{s_2} - \sqrt{s_1})} = -\frac{\sqrt{s_2} + \sqrt{s_1}}{c}$, equating the two expressions for the slope we get $2\sqrt{s} = \sqrt{s_2} + \sqrt{s_1}$.

On the other hand, since $(m, s) \in L_{(1+\epsilon)\lambda}$, we have

$$m = (1+\epsilon)\lambda - c\sqrt{s} = (1+\epsilon)\lambda - \frac{c\sqrt{s_2} + c\sqrt{s_1}}{2} = (1+\epsilon)\lambda - \frac{\lambda - m_2 + \lambda - m_1}{2} = \epsilon\lambda + \frac{m_1 + m_2}{2}$$

and $m = \alpha(m_1 - m_2) + m_2$ for some $\alpha \in (0, 1)$. Therefore $\alpha = \frac{1}{2} - \frac{\epsilon\lambda}{m_2 - m_1} = \frac{1}{2} - \frac{\epsilon\lambda}{c(\sqrt{s_1} - \sqrt{s_2})}$.

Next,

$$\begin{aligned} s &= \alpha(s_1 - s_2) + s_2 = \left[\frac{1}{2} - \frac{\epsilon\lambda}{c(\sqrt{s_1} - \sqrt{s_2})}\right](s_1 - s_2) + s_2 = \frac{s_1 - s_2}{2} - \frac{\epsilon\lambda}{c}(\sqrt{s_1} + \sqrt{s_2}) + s_2 \\ &= \frac{s_1 + s_2}{2} - \frac{\epsilon\lambda}{c}(\sqrt{s_1} + \sqrt{s_2}) \end{aligned}$$

therefore using $2\sqrt{s} = \sqrt{s_2} + \sqrt{s_1}$ from above, we get two equivalent expressions for $4s$:

$$\begin{aligned} & 2(s_1 + s_2) - \frac{4\epsilon\lambda}{c}(\sqrt{s_1} + \sqrt{s_2}) = s_1 + s_2 + 2\sqrt{s_1 s_2} \\ \Leftrightarrow\quad & s_1 + s_2 - \frac{4\epsilon\lambda}{c}(\sqrt{s_1} + \sqrt{s_2}) - 2\sqrt{s_1 s_2} = 0 \\ \Leftrightarrow\quad & \frac{s_1}{s_2} + 1 - \frac{4\epsilon\lambda}{c\sqrt{s_2}}(\sqrt{\frac{s_1}{s_2}} + 1) - 2\sqrt{\frac{s_1}{s_2}} = 0 \end{aligned}$$

Denote for simplicity $z = \sqrt{\frac{s_1}{s_2}}$ and $w = \frac{2\epsilon\lambda}{c\sqrt{s_2}}$, then we have to solve the following quadratic equation for z in terms of w:

$$\begin{aligned} & z^2 + 1 - 2w(z+1) - 2z = 0 \\ \Leftrightarrow\quad & z^2 - 2z(w+1) + 1 - 2w = 0. \end{aligned}$$

The discriminant of this quadratic expression is $D = (w+1)^2 - 1 + 2w = w^2 + 4w$ and its roots are $z_{1,2} = 1 + w \pm \sqrt{w^2+4w}$, Since $\frac{s_1}{s_2} > 1$, we choose the bigger root $z_2 = 1 + w + \sqrt{w^2+4w}$. Therefore since $w = \frac{2\epsilon\lambda}{c\sqrt{s_2}} \geq 0$ we have

$$\sqrt{\frac{s_1}{s_2}} = 1 + w + \sqrt{w^2+4w} \geq 1 + w = 1 + \frac{2\epsilon\lambda}{c\sqrt{s_2}} \geq 1 + \frac{2\epsilon\lambda}{c\frac{\lambda}{c}} = 1 + 2\epsilon,$$

where the last inequality follows from the fact that $\sqrt{s_2} < \sqrt{s_1} \leq \frac{\lambda}{c}$. This concludes the proof. $\square$

The previous lemma shows that each segment is sufficiently long so that overall the number of tangent segments approximating the level set L_λ is small. In particular, the number of segments is polynomial in $\frac{1}{\epsilon}$ (and does not depend on the problem size n). This gives us the desired approximate nonlinear separation oracle for the level sets of the objective function.

Lemma A.4 *A nonlinear $(1+\epsilon)$-approximate separation oracle to any level set of the nonconvex objective $f(\mathbf{x})$ in problem (1) can be found with $(1 + \frac{\log(\frac{1}{16\epsilon^2})}{2\log(1+2\epsilon)})$ queries to the available linear oracle for solving problem (5).*

The nonlinear oracle takes as inputs λ, ϵ and returns either a feasible solution $\mathbf{x} \in \mathcal{F}$ with $f(\mathbf{x}) \leq (1+\epsilon)\lambda$ or an answer that $f(\mathbf{x}) > \lambda$ for all $\mathbf{x}$ in $\mathcal{F}$.

Proof: Apply the available linear oracle to the slopes of the segments with endpoints on the specified level set, say L_λ, and which are tangent to the level set $L_{(1+\epsilon)\lambda}$. By Lemma A.3 and Lemma A.2, the y-coordinates of endpoints of these segments are given by

$$\begin{aligned} s_1 &= (\frac{\lambda}{c})^2, \\ s_2 &\leq \frac{s_1}{(1+2\epsilon)^2}, \\ s_3 &\leq \frac{s_1}{(1+2\epsilon)^4}, \\ &\cdots \\ s_k &\leq \frac{s_1}{(1+2\epsilon)^{2(k-1)}}, \\ s_{k+1} &= 0, \end{aligned}$$

where $s_k = (\frac{4\epsilon\lambda}{c})^2$, so $k = 1 + \log(\frac{1}{16\epsilon^2})/2\log(1+2\epsilon)$, which is precisely the number of segments we use and the result follows. $\square$

Finally, applying the approximate nonlinear separation oracle from Lemma A.4 on a suitable geometric progression of function values gives an approximation algorithm for the mean-standard deviation model (1). We can use the following bounds f_l, f_u for the function values in the algorithm. For a lower bound, set $f_l = s_{min}$, the smallest positive variance coordinate, and for an upper bound take $f_u = nm_{max} + c\sqrt{ns_{max}}$, where m_{max} and s_{max} are the largest coordinates of the mean and variance vectors respectively. Additionally, run the linear oracle once with weight vector equal to the vector of means, over the subset of coordinates with zero variances and return that solution if it is better. In particular, we can solve the problem even if the optimal objective value is zero.

Theorem A.5 *There is an oracle-polynomial time approximation scheme for the stochastic problem (1), which uses an exact oracle for solving the underlying deterministic problem (5). This algorithm returns a $(1+\epsilon)$-approximate solution and makes $(1 + \frac{2}{\epsilon}\log(\frac{f_u}{f_l}))(1 + \frac{\log(\frac{1}{16\epsilon^2})}{2\log(1+2\epsilon)})$ oracle queries, namely logarithmic in the input and polynomial in $\frac{1}{\epsilon}$.*

Proof: Apply the $(1+\xi)$-approximate nonlinear oracle successively on the sequence of function values $f_l,\ (1+\xi)f_l,\ (1+\xi)^2 f_l,\ \ldots$ for $\xi = \sqrt{1+\epsilon} - 1$, until we reach a level $\lambda = (1+\xi)^i f_l \leq f_u$ for which the oracle returns a feasible solution $\mathbf{x}'$ with

$$f(\mathbf{x}') \leq (1+\xi)\lambda = (1+\xi)^{i+1} f_l.$$

From running the oracle on the previous level $(1+\xi)^{i-1}f_l$, we know that $f(\mathbf{x}) \geq f(\mathbf{x}_{opt}) > (1+\xi)^{i-1}f_l$ for all $\mathbf{x}$ in the feasible set, where $\mathbf{x}_{opt}$ denotes the optimal solution. Therefore,

$$(1+\xi)^{i-1}f_l < f(\mathbf{x}_{opt}) \leq f(\mathbf{x}') \leq (1+\xi)^{i+1}f_l, \qquad \text{and hence}$$

$$f(\mathbf{x}_{opt}) \leq f(\mathbf{x}') < (1+\xi)^2 f(\mathbf{x}_{opt}) = (1+\epsilon)f(\mathbf{x}_{opt}).$$

So the solution $\mathbf{x}'$ gives an $(1+\epsilon)$-approximation to the optimum $\mathbf{x}_{opt}$. In the process, we run the approximate nonlinear separation oracle at most $2\log(\frac{f_u}{f_l})/\log(1+\epsilon)$ times. In addition, we run the linear oracle once more with linear objective given by the vector of means, over the subset of components with zero variances and return that solution if it is better than the above. $\square$

B Proof of Theorem 4.1 (Approximation of mean-risk model for hard combinatorial problems)

Suppose we have a δ-approximate linear oracle for solving the deterministic problem (5). We will provide an algorithm for the stochastic problem (1) with approximation factor $\delta(1+\epsilon)$, which invokes the linear oracle a small number of times that is logarithmic in the problem input and polynomial in $\frac{1}{\epsilon}$.

First, we show that if we can guess the optimal linear objective, given by the slope of the tangent to the corresponding level set at the optimal solution, then applying the approximate linear oracle returns an approximate solution with the same multiplicative approximation factor δ. The above statement reduces to showing the following geometric fact.

Lemma B.1 *Consider levels $0 \leq \lambda_1 < \lambda_2$ and two parallel lines tangent to the corresponding level sets L_{λ_1} and L_{λ_2} at points (m_1, s_1) and (m_2, s_2) respectively. Further, suppose the corresponding y-intercepts of these lines are b_1 and b_2. Then $\frac{b_2}{b_1} = \frac{\lambda_2+m_2}{\lambda_1+m_1} \geq \frac{\lambda_2}{\lambda_1}$.*

Proof: The function defining a level set L_λ has the form $y = \frac{(\lambda-x)^2}{c^2}$, and thus the slope of the tangent to the level set at a point $(m,s) \in L_\lambda$ is given by the first derivative at the point, $-\frac{2(\lambda-x)}{c^2}|_{x=m} = -\frac{2(\lambda-m)}{c^2} = -\frac{2\sqrt{s}}{c}$. Therefore the equation of the tangent line is $y = -\frac{2\sqrt{s}}{c}x + b$, where

$$b = s + \frac{2\sqrt{s}}{c}m = \sqrt{s}(\sqrt{s} + \frac{2m}{c}) = \sqrt{s}(\frac{\lambda-m}{c} + \frac{2m}{c}) = \sqrt{s}(\frac{\lambda+m}{c})$$

Since the two tangents from the lemma are parallel, their slopes are equal: $-\frac{2\sqrt{s_1}}{c} = -\frac{2\sqrt{s_2}}{c}$, therefore $s_1 = s_2$ and equivalently $(\lambda_1 - m_1) = (\lambda_2 - m_2)$.

Therefore the y-intercepts of the two tangents satisfy

$$\frac{b_2}{b_1} = \frac{\sqrt{s_2}(\frac{\lambda_2+m_2}{c})}{\sqrt{s_1}(\frac{\lambda_1+m_1}{c})} = \frac{\lambda_2+m_2}{\lambda_1+m_1} \geq \frac{\lambda_1}{\lambda_2}.$$

The last inequality follows from the fact that $\lambda_2 > \lambda_1$ and $\lambda_1 - m_1 = \lambda_2 - m_2$ (and equality is achieved when $m_1 = \lambda_1$ and $m_2 = \lambda_2$). $\square$

Corollary B.2 *Suppose the optimal solution to the nonconvex problem (1) is (m_1, s_1) with objective value λ_1. If we can guess the slope $-a$ of the tangent to the level set L_{λ_1} at the optimal solution, then applying the approximate linear oracle for solving problem (5) with respect to that slope will give a δ-approximate solution to problem (1).*

Proof: The approximate linear oracle will return a solution (m', s') with value $b_2 = s' + am' \leq \delta b_1$, where $b_1 = s_1 + am_1$. The objective function value of (m', s') is at most λ_2, which is the value at the level set tangent to the line $y = -ax + b_2$. By Lemma B.1, $\frac{\lambda_2}{\lambda_1} \leq \frac{b_2}{b_1} \leq \delta$, therefore the approximation solution has objective function value at most δ times the optimal value, QED. □

If we cannot guess the slope at the optimal solution, we have to approximate it. Lemma B.3 proves that if we apply the approximate linear oracle to a slope that is within $(1+\sqrt{\frac{\epsilon}{1+\epsilon}})$ of the optimal slope, we would still get a good approximate solution with approximation factor $\delta(1+\epsilon)$.

Lemma B.3 *Consider the level set L_λ and points (m^*, s^*) and (m, s) on it, at which the tangents to L_λ have slopes $-a$ and $-a(1+\sqrt{\frac{\epsilon}{1+\epsilon}})$ respectively. Let the y-intercepts of the tangent line at (m, s) and the line parallel to it through (m^*, s^*) be b_1 and b respectively. Then $\frac{b}{b_1} \leq 1+\epsilon$.*

Proof: Let $\xi = \sqrt{\frac{\epsilon}{1+\epsilon}}$. As established in the proof of Lemma B.1, the slope of the tangent to the level set L_λ at point (m^*, s^*) is $-a = -\frac{2\sqrt{s^*}}{c}$. Similarly the slope of the tangent at (m, s) is $-a(1+\xi) = -\frac{2\sqrt{s}}{c}$. Therefore, $\sqrt{s} = (1+\xi)\sqrt{s^*}$, or equivalently $(\lambda - m) = (1+\xi)(\lambda - m^*)$.

Since b, b_1 are intercepts with the y-axis, of the lines with slopes $-a(1+\xi) = -\frac{2\sqrt{s}}{c}$ containing the points (m^*, s^*), (m, s) respectively, we have

$$\begin{aligned} b_1 &= s + \frac{2\sqrt{s}}{c} m = \frac{\lambda^2 - m^2}{c^2} \\ b &= s^* + (1+\xi)\frac{2\sqrt{s^*}}{c} m^* = \frac{\lambda - m^*}{c^2}(\lambda + m^* + 2\xi m^*). \end{aligned}$$

Therefore

$$\frac{b}{b_1} = \frac{(\lambda - m^*)(\lambda + m^* + 2\xi m^*)}{(\lambda - m)(\lambda + m)} = \frac{1}{1+\xi}\frac{\lambda + m^* + 2\xi m^*}{\lambda + m} \leq \frac{1}{1+\xi}\left(\frac{1}{1-\xi}\right) = \frac{1}{1-\xi^2} = 1+\epsilon,$$

where the last inequality follows by Lemma B.4. □

Lemma B.4 *Following the notation of Lemma B.3, $\frac{\lambda + m^* + 2\xi m^*}{\lambda + m} \leq \frac{1}{1-\xi}$.*

Proof: Recall from the proof of Lemma B.3 that $(\lambda - m) = (1+\xi)(\lambda - m^*)$, therefore $m = \lambda - (1+\xi)(\lambda - m^*) = -\xi\lambda + (1+\xi)m^*$. Hence,

$$\frac{\lambda + m^* + 2\xi m^*}{\lambda + m} = \frac{\lambda + (1+2\xi)m^*}{(1-\xi)\lambda + (1+\xi)m^*} = \frac{\frac{\lambda}{m^*} + (1+2\xi)}{(1-\xi)\frac{\lambda}{m^*} + (1+\xi)} \leq \frac{1}{1-\xi},$$

since $\frac{1+2\xi}{1+\xi} \leq \frac{1}{1-\xi}$ for $\xi \in [0, 1)$. □

A corollary from Lemma B.1 and Lemma B.3 is that applying the linear oracle with respect to a slope that is within $(1+\sqrt{\frac{\epsilon}{1+\epsilon}})$ times of the optimal slope yields an approximate solution with objective value within $(1+\epsilon)\delta$ times of the optimal.

Lemma B.5 *Suppose the optimal solution to the nonconvex problem (1) is (m^*, s^*) with objective value λ and the slope of the tangent to the level set L_λ at it is $-a$. Then running the δ-approximate oracle for solving problem (5) with respect to a slope that is in $[-a, -a(1+\sqrt{\frac{\epsilon}{1+\epsilon}})]$ returns a solution to (1) with objective function value no greater than $(1+\epsilon)\delta\lambda$.*

Proof: Suppose the optimal solution with respect to the linear objective specified by slope $-a(1+\sqrt{\frac{\epsilon}{1+\epsilon}})$ has value $b' \in [b_1, b]$, where b_1, b are the y-intercepts of the lines with that slope, tangent to L_λ and passing through (m^*, s^*) respectively (See Figure 8). Then applying the δ-approximate linear oracle to the same linear objective returns solution with value $b_2 \geq \delta b'$. Hence $\frac{b_2}{b} \leq \frac{b_2}{b'} \leq \delta$.

On the other hand, the approximate solution returned by the linear oracle has value of our original objective function equal to at most λ_2, where L_{λ_2} is the level set tangent to the line on which the approximate solution lies. By Lemma B.1, $\frac{\lambda_2}{\lambda} \leq \frac{b_2}{b_1} = \frac{b_2}{b}\frac{b}{b_1} \leq \delta(1+\epsilon)$, where the last inequality follows by Lemma B.3 and the above bound on $\frac{b_2}{b}$. □

Finally, we are ready to state our theorem for solving the stochastic model (1). The theorem says that there is an algorithm for this problem with essentially the same approximation factor as for the underlying deterministic combinatorial problem (5), which makes only logarithmically many calls to the latter.

Theorem B.6 (Theorem 4.1 in main body of paper.) *Suppose we have a δ-approximation oracle for solving the deterministic combinatorial problem (5). The mean-risk model (1) can be approximated to a multiplicative factor of $\delta(1+\epsilon)$ by calling the oracle logarithmically many times in the input parameters and polynomially many times in $\frac{1}{\epsilon}$.*

Proof: We use the same type of algorithm as in Theorem 4.2: apply the available approximate linear oracle on a geometric progression of cost vectors (slopes), determined by the lemmas above. In particular, apply it to slopes $U, (1+\xi)U, ..., (1+\xi)^i U = L$, where $\xi = \sqrt{\frac{\epsilon}{1+\epsilon}}$, L is a lower bound for the optimal slope and U is an upper bound for it. For each approximate feasible solution obtained, compute its objective function value and return the solution with minimum objective function value. By Lemma B.5, the value of the returned solution would be within $\delta(1+\epsilon)$ of the optimal.

Note that it is possible for the optimal slope to be 0: this would happen when the optimal solution satisfies $m^* = \lambda$ and $s^* = 0$. We have to handle this case differently: run the linear oracle just over the subset of coordinates with zero variance-values, to find the approximate solution with smallest m. Return this solution if its value is better than the best solution among the above.

It remains to bound the values L and U. We established earlier that the optimal slope is given by $\frac{2\sqrt{s^*}}{c}$, where s^* is the variance of the optimal solution. Among the solutions with nonzero variance, the variance of a feasible solution is at least s_{min}, the smallest possible nonzero variance of a single element, and at most $(\lambda_{max})^2 \leq (nm_{max} + c\sqrt{ns_{max}})^2$, where m_{max} is the largest possible mean of a single element and s_{max} is the largest possible variance of a single element (assuming that a feasible solution uses each element in the ground set at most once). Thus, set $U = -\frac{2\sqrt{s_{min}}}{c}$ and $L = -\frac{2(nm_{max}+c\sqrt{ns_{max}})}{c}$ □

C Gap-preserving approximation lemma for probability tail objective

Lemma C.1 *A δ-approximation for the nonconvex threshold objective (2) yields a δ-approximation for the stochastic threshold objective $\Phi\left(\frac{t-\boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}}\right)$, where Φ denotes the cumulative distribution function of the standard normal random variable $N(0,1)$.*

Proof: Denote the approximate and the optimal solutions by $\mathbf{x}, \mathbf{x}_{opt}$ respectively. A δ-approximation for maximizing the nonconvex threshold objective means that

$$\frac{t-\boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}} \geq \delta \frac{t-\boldsymbol{\mu}^T\mathbf{x}_{opt}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}_{opt}}}.$$

Denote $f = \frac{t-\boldsymbol{\mu}^T\mathbf{x}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}}}$ and $f_{opt} = \frac{t-\boldsymbol{\mu}^T\mathbf{x}_{opt}}{\sqrt{\boldsymbol{\tau}^T\mathbf{x}_{opt}}}$. Since by assumption, $f_{opt} \geq 0$, a line going through 0 and between the points $(f, \Phi(f)), (f_{opt}, \Phi(f_{opt}))$ on the graph of the function Φ will cross the vertical lines through this points below the graph and above the graph respectively (at the points A' and B' in Figure 10). Using the notation from Figure 10, we thus have that $\Phi(f) \geq y$ (the y-coordinate of A') and $\Phi(f_{opt}) \leq y_{opt}$. On the other hand, since the lines AA' and BB' are parallel, we have the equality below:

$$\delta \leq \frac{f}{f_{opt}} = \frac{y}{y_{opt}} \leq \frac{\Phi(f)}{\Phi(f_{opt})},$$

therefore $\Phi(f) \geq \delta\Phi(f_{opt})$, QED. □

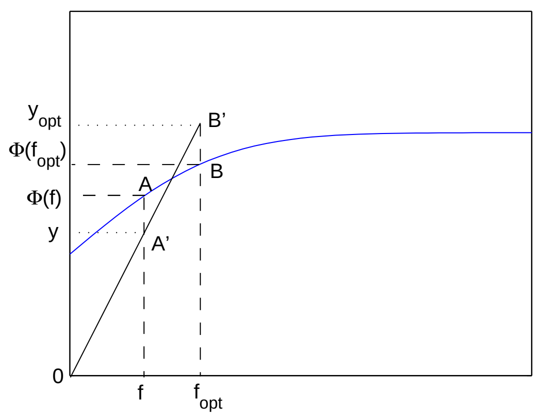


Figure 10: A plot of the stochastic threshold objective (the cumulative distribution function Φ of the standard normal random variable).