Computers and Chemical Engineering 171 (2023) 108170

Contents lists available at ScienceDirect

Computers and Chemical Engineering

journal homepage: www.elsevier.com/locate/cace

Optimal blending under general uncertainties: A chance-constrained programming approach

Yu Yang

Department of Chemical Engineering, California State University Long Beach, Long Beach, CA 90840, USA

ARTICLE INFO

Keywords:
Stochastic programming
Blend planning
Chance-constrained optimization
Gaussian mixture model

ABSTRACT

An optimization algorithm is proposed for blend planning with linear mixing law and general parameter uncertainties. The objective is to make the final product satisfy all quality specifications with high probability, and maximize the production profit by carefully determining the feedstock ratio. Conventional approaches that rely on deterministic optimization fail to account for parameter uncertainty, and thus may not generate a probabilistic feasible solution. The proposed work formulates the blend planning problem as a joint chance-constrained program (CCP). Using Boole's inequality to decompose joint constraints and the Gaussian mixture model to characterize uncertainty distributions, a conservative deterministic approximation of CCP can be formulated. Through second-order cone relaxation, branch-and-bound, optimality-based bound tightening, and reformulate-linearization techniques, the global optimum of deterministic approximation can be found. A risk level adjustment procedure is presented to reduce the conservativeness and further improve the objective value of the solution if posterior evaluation is allowed. Two numerical cases, including steel and gasoline productions, are studied to show the solving time, probabilistic feasibility, and solution quality of the proposed optimization method.

1. Introduction

The model-based blend planning and optimization is an essential step to improve the profitability and quality of gasoline, metal, and pharmaceutical manufacturing. However, the blendstock qualities are usually unknown at the time of blend planning. If such parameter uncertainties are not adequately characterized and integrated into the optimization formula, the resulting deterministic optimum may not be feasible in practice (Ben-Tal and Nemirovski, 2002; Ning and You, 2018). Such infeasible blending recipes may yield unqualified products failing to meet environmental, safety, and healthy regulations. Hence, the difference between nominal and real values of model parameters should be considered in the optimization to guarantee the economic efficiency, safety, and sustainability of the designed process.

The mathematical programming community has recognized the significance of explicitly accounting for model uncertainty in the optimization. Some comprehensive reviews about optimization under uncertainty and its successful applications in chemical engineering can be found (Sahinidis, 2004; Grossmann et al., 2016). Popular approaches include stochastic programming (SP), robust optimization (RO), and chance-constrained programming (CCP). The SP is widely used for long-term multi-stage scheduling because recourses actions can be employed to compensate for undesired outcomes after the first stage operations. RO tends to guarantee the feasibility under a pre-specified uncertainty set. Even though RO is less computationally expensive than SP, its solution can be more conservative (Grossmann et al., 2016). CCP overcomes RO's drawback because it takes advantage of uncertainty distributions and allows constraints to be violated at a small level. Related applications include chemical process design (Peng et al., 2022), power system operation (Fathabad et al., 2023), and waste solid management (Sun et al., 2013). All operations are determined in a single stage for blending recipe design, and the parameter distribution can be characterized through historical data. Therefore, CCP will be the research focus of this paper.

CCP was firstly introduced in 1959 (Charnes and Cooper, 1959) and has become increasingly attractive to the optimization community because of its flexibility to balance robustness and optimality (Li et al., 2008). Due to parameter uncertainty, requiring constraints to be satisfied under any scenarios could be conservative or even infeasible. CCP allows the solution to violate constraints for a small chance, denoted as risk level ϵ, and thus is less conservative than RO. However, solving CCP is non-trivial. A single linear chance constraint with normally distributed parameters or right-hand-side uncertainty with log-concave distribution can be converted to a convex form (Prékoba, 1995). However, other uncertainty distributions do not enable such an equivalent reformulation. If a program contains multiple chance constraints, the resulting joint CCP is even more challenging to solve.

E-mail address: yu.yang@csulb.edu.

https://doi.org/10.1016/j.compchemeng.2023.108170
Received 1 November 2022; Received in revised form 5 January 2023; Accepted 6 February 2023
Available online 8 February 2023

Thus, most works of CCP focus on searching for a locally optimum within a reasonable solution time. Scenario approximation (SA) and sample average approximation (SAA) draw a large number of samples to convert the CCP into a deterministic approximation. The distribution approximation method aims to characterize uncertainty distribution in a solvable form based on the historical data. Their details are discussed below:

Scenario Approximation: This type of methods, consisting of two categories, generates a scenario-based deterministic approximation of CCP based on independent Monte Carlo samples. The first group of SA generates a large number of scenarios (constraints) and requires $1-\epsilon$ of them to be satisfied (Luedtke and Ahmed, 2008; Luedtke, 2014). As the number of scenarios increases, the solution of this method will converge to the original CCP (Peña-Ordieres et al., 2020). However, because binary variables are introduced to represent the selection of relaxed constraints, solving the resulting formula is NP-hard and finally intractable as the number of samples becomes sufficiently large. Another shortcoming is that the resulting solution profile is non-smooth due to the stochastic nature of samples (Peña-Ordieres et al., 2020). The second group of SA focuses on the convex problem and requires all sampled constraints to be satisfied to yield a feasible solution of CCP with high confidence (Calafiore and Campi, 2006; Campi and Garatti, 2008). This method is attractive because no binary variable is introduced, and thus the formula is more scalable. Note that the fundamental idea of this method is to determine the sample complexity, namely, the number of samples. A series of works have been proposed to reduce the sample complexity and achieve a less conservative solution (Campi and Garatti, 2011; Alamo et al., 2015). Another effort extends this methodology to nonconvex cases relying on the posterior evaluation (Esfahani et al., 2015). However, the drawback of this type of approach is also significant because there is no link between the optimal solutions of CCP and scenario approximation (Peña-Ordieres et al., 2020).

Sample Average Approximation: Because the expectation on indicator function can be used to represent the probability of constraint satisfaction, there are plenty of works to conservatively approximate indicator function through sample average schemes. The conditional value-at-risk (CVaR) approximation and the Bernstein approximation (Nemirovski and Shapiro, 2006) are two typical schemes. The sigmoidal approximation is another option as a smooth method to replace the indicator function in the chance constraints (Tovar-Facio et al., 2018). Alternatively, the difference-of-convex functions can be employed to approximate the indicator function tightly (Hong et al., 2011; Shan et al., 2014). However, the resulting optimization is difficult to solve because its gradient information is inaccurate in the interested region (Peña-Ordieres et al., 2020). A recent work adopts the projected stochastic subgradient algorithm to solve a convergent sequence of smooth approximation of CCP and reports promising results (Kannan and Luedtke, 2021). In summary, SAA is a general but conservative method for CCP without explicitly exploring the shape of distributions (Tovar-Facio et al., 2018).

Distribution Approximation: This methodology uses historical data to estimate the probability density function (PDF), cumulative distribution function (CDF), or quantile function, and then converts the probabilistic constraint to an algebraic form. The kernel smoothing method can be applied for the PDF, CDF, or quantile function estimation to solve chance constraints with right-hand side uncertainty (Calfa et al., 2015). The kernel method is non-parametric but may lead to a computationally intensive formula. In Ref. Jiang and Guan (2016), ψ-divergence is used to describe the confidence set of an estimated PDF, and then a perturbed risk level replaces the original one in CCP to guarantee the robustness under ambiguous distributions. Those references show that data-driven PDF estimation is a promising and robust method to solve general CCP.

This paper employs a Gaussian mixture model (GMM) to characterize the distribution of uncertain parameters from historical data. It then develops a global optimization scheme for a conservative approximation of the original joint CCP. In previous work (Yang et al., 2017), we have established a global optimization algorithm for CCP with linear mixing law and Gaussian-distributed uncertainties. Further removing the restriction of Gaussian distribution enables the optimal blend planning under more general uncertainties. Note that a recent paper (Hu et al., 2022) also studied GMM-CCP but can only solve the global optimum for single chance-constrained program (SCCP), because that method only evaluates the risk level on the bound of each subregion, which is not applicable to joint chance-constrained cases. In our paper, the contributions of solving joint GMM-CCP are listed below:

- Boole's inequality is used to decompose the joint chance constraints.
- An outer approximation is built to successively approach the inverse CDF of each Gaussian component by following Cheng et al. (2012).
- Branch-and-bound, optimality-based bound tightening (OBBT), reformulate-linearization techniques (RLT), and risk level adjustment are presented to form the global optimization scheme.

The rest of this paper is organized as follows. The chance-constrained program for optimal blend planning, and its reformulation under GMM are stated in Section 2. A global optimization framework for the deterministic approximation is proposed in Section 3. Two case studies are presented in Section 4 to show comparative results and highlight the effectiveness of the proposed method. Finally, conclusions are drawn in Section 5.

Notation. Throughout this paper, vectors are denoted by boldface letters. The space of symmetric positive semi-definite matrices of dimension n is denoted by $\mathbb{S}_+^n$. Let $\mathbf{e}$ represent a vector of all ones. Its dimension will be clear from the context. All random parameters are denoted with tilde mark. Let $f_{\tilde{\boldsymbol{\xi}}}$ and $F_{\tilde{\boldsymbol{\xi}}}$ denote the PDF and CDF for $\tilde{\boldsymbol{\xi}}$, respectively. In particular, for $\boldsymbol{\mu} \in \mathbb{R}^n$ and $\boldsymbol{\Sigma} \in \mathbb{S}_+^n$, we let $\mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) : \mathbb{R}^n \to \mathbb{R}$ and $\Phi(\cdot; \boldsymbol{\mu}, \boldsymbol{\Sigma}) : \mathbb{R}^n \to \mathbb{R}$ denote the PDF and CDF of the n-variate normal distribution with mean vector $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$, respectively. We write $\tilde{\boldsymbol{\xi}} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ to express that $\tilde{\boldsymbol{\xi}}$ is normally distributed with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. Similarly, for $\boldsymbol{w} \in \mathbb{R}_+^S$ such that $\mathbf{e}^\top \boldsymbol{w} = 1$, $\boldsymbol{\mu}_s \in \mathbb{R}^n$ and $\boldsymbol{\Sigma}_s \in \mathbb{S}_+^n$, $\forall s \in \{1, \ldots, S\}$, we write $\tilde{\boldsymbol{\xi}} \sim \sum_{s=1}^S w_s \mathcal{N}(\boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s)$ to express that $\tilde{\boldsymbol{\xi}}$ follows the Gaussian mixture distribution.

2. Methodology

The conventional linear CCP assumes that qualities are normally distributed, and then chance-constraints could be reformulated equivalently as deterministic conic constraints, resulting in a tractable conservative approximation to the joint chance-constrained blending problem. Unfortunately, in many practical settings, the assumption of normally distributed qualities is unrealistic. For example, blendstock's chemical or physical property usually has its lower and upper limits, or its distribution is non-symmetric. It is thus desirable to develop a solution approach for CCP that is applicable to a general class of distributions. In this section, we relax the assumption of normally distributed qualities, and use the Gaussian mixture model to approximate the true distribution based on sampled data points. Subsequently, an efficient approach is developed to reformulate the chance-constrained optimization problem as a second-order cone program (SOCP).

2.1. Preliminary

The blending under uncertainties can be cast as a chance-constrained linear program (Yang et al., 2017), shown below:

$$\begin{aligned} \min_{\boldsymbol{x} \in \mathcal{X}} \ & \boldsymbol{r}^\top \boldsymbol{x} \\ \text{s.t.} \ & \mathbb{P}\left(\tilde{\boldsymbol{\xi}}_q^\top \boldsymbol{x} \leqslant \boldsymbol{g}_q^\top \boldsymbol{x}, \ \forall q \in \mathcal{Q}\right) \geqslant 1-\epsilon, \end{aligned} \tag{CBP}$$

where ϵ is the total risk level. Vector $\boldsymbol{g}_q$ is determined by the quality specification. $\mathcal{Q}$ is the set of quality. $\mathcal{X} \subset \mathbb{R}^B$ is the feasible region of blending flow $\boldsymbol{x}$ with dimension B. Note that $(\mathcal{CBP})$ may incorporate multiple chance constraints. A common approach is to decompose the joint chance constraints into $|\mathcal{Q}|$ individuals with risk level ϵ_q. This yields problem:

$$\begin{aligned} &\min_{\boldsymbol{x}\in\mathcal{X}} \ \boldsymbol{r}^{\mathrm{T}}\boldsymbol{x} \\ &\text{s.t.} \ \ \mathbb{P}\left(\tilde{\boldsymbol{\xi}}_q^{\mathrm{T}}\boldsymbol{x} \leqslant \boldsymbol{g}_q\boldsymbol{x}\right) \geqslant 1-\epsilon_q, \ \forall q \in \mathcal{Q}, \\ &\quad \textstyle\sum_{q\in\mathcal{Q}} \epsilon_q = \epsilon. \end{aligned} \tag{$\mathcal{ICB}$}$$

Based on Boole's inequality, $(\mathcal{ICB})$ is a conservative (inner) approximation to problem $(\mathcal{CBP})$.

2.2. Reformulation under mixture model

Univariate Gaussian mixtures. We begin our analysis for the case when the uncertain qualities in the various blendstocks are uncorrelated and each follows a Gaussian mixture distribution with known weights. This model is adequate when the blendstocks, within set $\mathcal{B} := \{1, 2, \dots, B\}$, come from different sources with disjoint processing procedures, implying that the modes of production of the blendstocks are unrelated to one another. Formally, this assumption can be stated as follows for each quality:

Assumption 1. For each blendstock $b \in \mathcal{B}$, it holds that $\tilde{\xi}_b \sim \sum_{s=1}^{S} w_{b,s}\mathcal{N}\left(\mu_{b,s}, \sigma_{b,s}^2\right)$ with known weight vector $\boldsymbol{w}_b \in \mathbb{R}_+^S$ satisfying $\mathbf{e}^{\mathrm{T}}\boldsymbol{w}_b = 1$, means $\mu_{b,s} \in \mathbb{R}$, and standard deviations $\sigma_{b,s} \in \mathbb{R}_+$, $\forall s \in \{1, \dots, S\}$.

Two remarks about Assumption 1 are presented. First, the distribution of each random parameter consists of the same number of Gaussian components to simplify the notation. Such a restriction can be relaxed without any modification on the algorithm. Second, the number of Gaussian components S and the mixture weights $\boldsymbol{w}_b$, $\forall b \in \mathcal{B}$, can be identified from measurement data using the Expectation-Maximization (EM) algorithm. If the approximation error of GMM is large, then a posterior check is necessary to ensure the solution meeting chance constraints.

The following proposition enables us to derive a conservative approximation of $(\mathcal{ICB})$ in the form of a second-order cone program under Assumption 1.

Proposition 1. *Given Assumption 1, for any fixed $\boldsymbol{x} \in \mathbb{R}^B$, $\boldsymbol{g} \in \mathbb{R}^B$ and $\epsilon \in (0, 1)$, we introduce the following statements:*

(a) $\mathbb{P}\left(\tilde{\boldsymbol{\xi}}^{\mathrm{T}}\boldsymbol{x} \leq \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x}\right) \geqslant 1-\epsilon$, *and*

(b) $\hat{\boldsymbol{\mu}}_k^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{\Phi}^{-1}(\frac{\gamma_k}{\delta_k})\sqrt{\boldsymbol{x}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_k\boldsymbol{x}} \leqslant \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x}, \ \forall k \in \mathcal{S}^B$,

where $\mathcal{S}^B := \{1, 2, \dots, S^B\}$. Let us define $\mathcal{S}^B$ distinct B-tuples $C_k := (c_k^1, c_k^2, \dots, c_k^B), \forall k \in \mathcal{S}^B$, where $c_k^b \in \{1, 2, \dots, S\}$, $\hat{\boldsymbol{\mu}}_k := [\mu_{1,c_k^1}, \dots, \mu_{B,c_k^B}]^{\mathrm{T}}$, $\hat{\boldsymbol{\Sigma}}_s := \left(\mathrm{diag}(\sigma_{1,c_k^1}, \dots, \sigma_{B,c_k^B})\right)^2$, $\delta_k := \prod_{b\in\mathcal{B}} w_{b,c_k^b}$, and the γ_k satisfy

$$\sum_{k\in\mathcal{S}^B} \gamma_k \geqslant 1-\epsilon.$$

Then, statement (b) above implies statement (a).

Proof. For each $b \in \mathcal{B}$, let $\tilde{z}_b := \tilde{\xi}_b x_b$ and the probability density function of $\tilde{z}_b$ is given by

$$f_{\tilde{z}_b} = \sum_{s=1}^{S} w_{b,s}\mathcal{N}\left(x_b\mu_{b,s}, (x_b\sigma_{b,s})^2\right)$$

Let us denote $\mathcal{N}\left(x_b\mu_{b,s}, (x_b\sigma_{b,s})^2\right)$ as $\mathcal{N}_{b,s}$. For $\tilde{z} = \tilde{\boldsymbol{\xi}}^{\mathrm{T}}\boldsymbol{x} = \sum_{b\in\mathcal{B}} \tilde{z}_b$, its probability density function is

$$\begin{aligned} f_{\tilde{z}} &= \left(\textstyle\sum_{s=1}^{S} w_{1,s}\mathcal{N}_{1,s}\right) * \left(\sum_{s=1}^{S} w_{2,s}\mathcal{N}_{2,s}\right) * \cdots * \left(\sum_{s=1}^{S} w_{B,s}\mathcal{N}_{B,s}\right) \\ &= \textstyle\sum_{s_1=1}^{S}\sum_{s_2=1}^{S}\cdots\sum_{s_B=1}^{S} \\ &\quad \times \left(w_{1,s_1}\mathcal{N}_{1,s_1} * w_{2,s_2}\mathcal{N}_{2,s_2} * \cdots * w_{B,s_B}\mathcal{N}_{B,s_B}\right) \end{aligned} \tag{1}$$

Let $c_k^1 = s_1, c_k^2 = s_2, \dots, c_k^B = s_B$, then Eq. (1) can be written as

$$\begin{aligned} f_{\tilde{z}} &= \textstyle\sum_{k\in\mathcal{S}^B}\prod_{b\in\mathcal{B}} w_{b,c_k^b}\mathcal{N}_{1,c_k^1} * \mathcal{N}_{2,c_k^2} * \cdots * \mathcal{N}_{B,c_k^B} \\ &= \textstyle\sum_{k\in\mathcal{S}^B} \delta_k\mathcal{N}\left(\hat{\boldsymbol{\mu}}_k^{\mathrm{T}}\boldsymbol{x}, \boldsymbol{x}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_k\boldsymbol{x}\right), \end{aligned} \tag{2}$$

where $*$ is the convolution operator. The last equality of Eq. (2) holds since the convolution of the probability distributions of independent random variables equals the distribution of their sum. Thus, $\tilde{\boldsymbol{\xi}}^{\mathrm{T}}\boldsymbol{x}$ also follows a Gaussian mixture distribution. Its cumulative distribution function is expressible as $F_{\tilde{z}} = \sum_{k\in\mathcal{S}^B} \delta_k\boldsymbol{\Phi}\left(\cdot; \hat{\boldsymbol{\mu}}_k^{\mathrm{T}}\boldsymbol{x}, \boldsymbol{x}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_k\boldsymbol{x}\right)$. Then, there are

$$\begin{aligned} &\hat{\boldsymbol{\mu}}_k^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{\Phi}^{-1}(\tfrac{\gamma_k}{\delta_k})\sqrt{\boldsymbol{x}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_k\boldsymbol{x}} \leqslant \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x}, \ \forall k \in \mathcal{S}^B \\ \Leftrightarrow\ &\delta_k \ \boldsymbol{\Phi}\left(\boldsymbol{g}^{\mathrm{T}}\boldsymbol{x}; \hat{\boldsymbol{\mu}}_k^{\mathrm{T}}\boldsymbol{x}, \boldsymbol{x}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_k\boldsymbol{x}\right) \geqslant \gamma_k, \ \forall k \in \mathcal{S}^B \\ \Rightarrow\ &\textstyle\sum_{k\in\mathcal{S}^B} \delta_k \ \boldsymbol{\Phi}\left(\boldsymbol{g}^{\mathrm{T}}\boldsymbol{x}; \hat{\boldsymbol{\mu}}_k^{\mathrm{T}}\boldsymbol{x}, \boldsymbol{x}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_k\boldsymbol{x}\right) \geqslant \sum_{k\in\mathcal{S}^B}\gamma_k \\ \Rightarrow\ &\textstyle\sum_{k\in\mathcal{S}^B} \delta_k \ \boldsymbol{\Phi}\left(\boldsymbol{g}^{\mathrm{T}}\boldsymbol{x}; \hat{\boldsymbol{\mu}}_k^{\mathrm{T}}\boldsymbol{x}, \boldsymbol{x}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_k\boldsymbol{x}\right) \geqslant 1-\epsilon \\ \Leftrightarrow\ &\mathbb{P}(\tilde{\boldsymbol{\xi}}^{\mathrm{T}}\boldsymbol{x} \leqslant \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x}) \geqslant 1-\epsilon \end{aligned}$$

This concludes the proof. □

Proposition 1 can be applied to each quality chance constraint by introducing index q for $\boldsymbol{w}$, $\hat{\boldsymbol{\mu}}$, $\boldsymbol{\sigma}$ and $\hat{\boldsymbol{\Sigma}}$. Then, the following conservative approximation to $(\mathcal{ICB})$ under Assumption 1 is developed as a deterministic optimization problem

$$\begin{aligned} \underset{\boldsymbol{x}\in\mathcal{X},\gamma_{q,k},\alpha_q}{\text{maximize}} \ & \boldsymbol{r}^{\mathrm{T}}\boldsymbol{x} \\ \text{s.t.} \ \ & \hat{\boldsymbol{\mu}}_{q,k}^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{\Phi}^{-1}(\tfrac{\gamma_{q,k}}{\delta_{q,k}})\sqrt{\boldsymbol{x}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_{q,k}\boldsymbol{x}} \leqslant \boldsymbol{g}_q^{\mathrm{T}}\boldsymbol{x} \quad \forall q \in \mathcal{Q}, \ \forall k \in \mathcal{S}^B, \\ & \overline{\gamma}_{q,k} \geqslant \gamma_{q,k} \geqslant \underline{\gamma}_{q,k}, \forall q \in \mathcal{Q}, \ \forall k \in \mathcal{S}^B, \\ & \textstyle\sum_{k\in\mathcal{S}^B}\gamma_{q,k} \geqslant 1-\alpha_q, \forall q \in \mathcal{Q}, \\ & \textstyle\sum_{q\in\mathcal{Q}}\alpha_q = \epsilon, \end{aligned} \tag{$\mathcal{LCB}_{\mathrm{U}}$}$$

where $\hat{\boldsymbol{\mu}}_{q,k}$ and $\hat{\boldsymbol{\Sigma}}_{q,k}$ are defined by extending $\hat{\boldsymbol{\mu}}_k$ and $\hat{\boldsymbol{\Sigma}}_k$ in Proposition 1 to each quality constraint; $\delta_{q,k} := \prod_{b\in\mathcal{B}} w_{q,b,c_k^b}$. The individual risk level for each quality constraint is defined as $\alpha_q, \forall q \in \mathcal{Q}$. Here we still use Boole's inequality to decompose joint constraints and require the sum of individual risk levels to be ϵ. The lower bound on $\gamma_{q,k}$ is set as $0.5\delta_{q,k}$ to ensure that $(\mathcal{LCB}_{\mathrm{U}})$ is a second-order cone program when $\gamma_{q,k}$ is pre-determined. Theoretically, $\gamma_{q,k}$ can arbitrarily approach $\delta_{q,k}$, but will render $\boldsymbol{\Phi}^{-1}(\gamma_{q,k}/\delta_{q,k})$ to infinity. Thus, $0.9999\delta_{q,k}$ can be set as an upper limit of $\gamma_{q,k}$ for practical computations.

Several comments about $(\mathcal{LCB}_{\mathrm{U}})$ are presented. First, the number of conic constraints in $(\mathcal{LCB}_{\mathrm{U}})$ is given by $|\mathcal{Q}|S^B$. For a moderate number of blendstocks commonly employed in practice and small S, the problem size will be manageable. If an uncertainty distribution needs many Gaussian components to accurately approximate, the proposed approach may not be applicable and we recommend sampling-based or traditional robust optimization approaches. Second, the risk level will be determined through a branch-and-bound scheme in order to further improve the solution quality. However, the resulting computational complexity will increase significantly because many bilinear terms should be addressed. Third, using Boole's inequality to approximate joint chance constraints brings conservativeness. However, the risk level ϵ can be gradually increased to improve the solution quality and evaluated via posterior samples for its probabilistic feasibility.

Multivariate Gaussian mixtures. We now investigate the case when the qualities across blendstocks are correlated. This scenario may happen when some blendstocks are produced by the same material flows or process units. The set of blendstocks can be divided into I subsets,

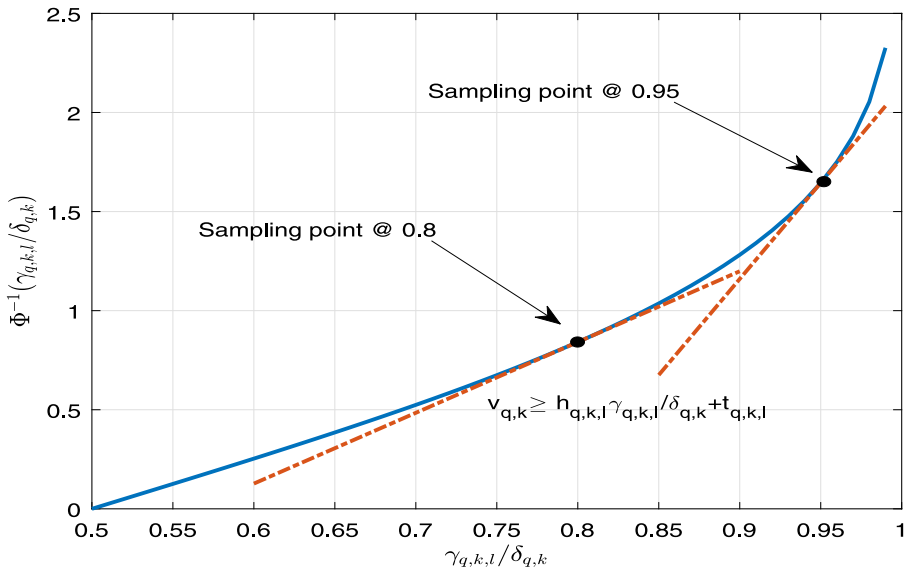

Fig. 1. The outer approximation of $\Phi^{-1}(\frac{\gamma_{q,k,l}}{\delta_{q,k}})$.

denoted by $\mathcal{B}_i$ with $i \in \mathcal{I} := \{1, \ldots, I\}$, such that random variables within $\mathcal{B}_i$ are correlated. Moreover, $\mathcal{B}_i$ should be mutually exclusive and collectively exhaustive. Vectors $\tilde{\boldsymbol{\xi}}$ and $\boldsymbol{x}$ are reordered such that all elements within the same $\mathcal{B}_i$ are adjacent, and then let $\tilde{\boldsymbol{\xi}}_{\mathcal{B}_i}$ to represent the random variables in the subset $\mathcal{B}_i$. In particular, it is assumed that the quality of blendstocks belonging to the same subset $\mathcal{B}_i$ follows a multivariate Gaussian mixture model:

Assumption 2. For each $i \in \mathcal{I}$, it holds that $\tilde{\boldsymbol{\xi}}_{\mathcal{B}_i} \sim \sum_{s=1}^{S} w_{i,s}\mathcal{N}(\boldsymbol{\mu}_{\mathcal{B}_i,s}, \boldsymbol{\Sigma}_{\mathcal{B}_i,s})$ with known weight vectors $\boldsymbol{w}_i \in \mathbb{R}_+^S$ satisfying $\mathbf{e}^{\mathrm{T}}\boldsymbol{w}_i = 1$, mean vectors $\boldsymbol{\mu}_{\mathcal{B}_i,s} \in \mathbb{R}^{|\mathcal{B}_i|}$ and covariance matrices $\boldsymbol{\Sigma}_{\mathcal{B}_i,s} \in \mathbb{R}^{|\mathcal{B}_i|\times|\mathcal{B}_i|}$.

The following proposition enables us to derive a conservative approximation $(\mathcal{ICB})$ in the form of a second-order cone program under Assumption 2. It naturally extends Proposition 1 to the case of multivariate Gaussian mixture distributions.

Proposition 2. *Let $\tilde{\boldsymbol{\xi}} := (\tilde{\boldsymbol{\xi}}_{\mathcal{B}_1}, \ldots, \tilde{\boldsymbol{\xi}}_{\mathcal{B}_I}) \in \mathbb{R}^B$, with $\tilde{\boldsymbol{\xi}}_{\mathcal{B}_i} \in \mathbb{R}^{|\mathcal{B}_i|}$, $\forall i \in \mathcal{I}$. Given Assumption 2, for any fixed $\boldsymbol{x} \in \mathbb{R}_+^B$, $\boldsymbol{g} \in \mathbb{R}^B$ and $\epsilon \in (0,1)$, define the following statements:*

(a) $\mathbb{P}\left(\tilde{\boldsymbol{\xi}}^{\mathrm{T}}\boldsymbol{x} \leqslant \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x}\right) \geqslant 1-\epsilon$, *and*

(b) $\hat{\boldsymbol{\mu}}_k^{\mathrm{T}}\boldsymbol{x} + \Phi^{-1}(\frac{\gamma_k}{\delta_k})\sqrt{\boldsymbol{x}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_k\boldsymbol{x}} \leqslant \boldsymbol{g}^{\mathrm{T}}\boldsymbol{x} \quad \forall k \in \mathcal{S}^I$,

Let us define all distinct I-tuples $C_k := (c_k^1, c_k^2, \ldots, c_k^I)$, $\forall k \in \{1, 2, \ldots, S^I\}$ on the domain $\{1, 2, \ldots, S\}$, such that $\hat{\boldsymbol{\mu}}_k := [\boldsymbol{\mu}_{\mathcal{B}_1,c_k^1}, \ldots, \boldsymbol{\mu}_{\mathcal{B}_I,c_k^I}]^{\mathrm{T}}$, $\hat{\boldsymbol{\Sigma}}_k := \mathrm{blkdiag}\left(\boldsymbol{\Sigma}_{\mathcal{B}_1,c_k^1}, \ldots, \boldsymbol{\Sigma}_{\mathcal{B}_I,c_k^I}\right)$, $\delta_k := \prod_{i\in\mathcal{I}} w_{i,c_k^i}$, and the γ_k satisfy

$$\sum_{i=1}^{S^I} \gamma_k \geqslant 1-\epsilon.$$

Then, statement (b) above implies statement (a).

Proof. Fix $\boldsymbol{x} \in \mathbb{R}_+^B$, $\boldsymbol{g} \in \mathbb{R}^B$ and $\epsilon \in (0,1)$. Then, the probability density function of $\tilde{z}_i := \boldsymbol{x}_{\mathcal{B}_i}^{\mathrm{T}}\tilde{\boldsymbol{\xi}}_{\mathcal{B}_i}$ is given by

$$\begin{aligned} f_{\tilde{z}_i} &= \sum_{s=1}^{S} \frac{w_{i,s}}{\sqrt{(2\pi)^{|\mathcal{B}_i|}|\det(\boldsymbol{x}_{\mathcal{B}_i}^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathcal{B}_i,s}\boldsymbol{x}_{\mathcal{B}_i})|}} \\ &\quad \times \exp\left(-\tfrac{1}{2}(z_i - \boldsymbol{x}_{\mathcal{B}_i}^{\mathrm{T}}\boldsymbol{\mu}_{\mathcal{B}_i,s})^{\mathrm{T}}(\boldsymbol{x}_{\mathcal{B}_i}^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathcal{B}_i,s}\boldsymbol{x}_{\mathcal{B}_i})^{-1}(z_i - \boldsymbol{x}_{\mathcal{B}_i}^{\mathrm{T}}\boldsymbol{\mu}_{\mathcal{B}_i,s})\right) \\ &= \sum_{s=1}^{S} w_{i,s}\mathcal{N}(\boldsymbol{x}_{\mathcal{B}_i}^{\mathrm{T}}\boldsymbol{\mu}_{\mathcal{B}_i,s}, \boldsymbol{x}_{\mathcal{B}_i}^{\mathrm{T}}\boldsymbol{\Sigma}_{\mathcal{B}_i,s}\boldsymbol{x}_{\mathcal{B}_i}). \end{aligned}$$

Since the $\tilde{z}_i$, $\forall i \in \mathcal{I}$, are independent, it follows the proof of Proposition 1 to reach the conclusion. Here we replace $\mu_{b,s}$ and $\sigma_{b,s}$ by $\boldsymbol{\mu}_{\mathcal{B}_i,s}$ and $\boldsymbol{\Sigma}_{\mathcal{B}_i,s}$, $\forall s = \{1, 2, \ldots, S\}$ to obtain $\hat{\boldsymbol{\mu}}_k$ and $\hat{\boldsymbol{\Sigma}}_k$, respectively. □

From Proposition 2, the following conservative approximation is obtained under Assumption 2 as a deterministic optimization problem

$$\begin{aligned} \min_{\boldsymbol{x}\in\mathcal{X},\gamma_{q,k},\alpha_q} \quad & \boldsymbol{r}^{\mathrm{T}}\boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{\mu}_{q,k}^{\mathrm{T}}\boldsymbol{x} + \Phi^{-1}(\tfrac{\gamma_{q,k}}{\delta_{q,k}})\sqrt{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{\Sigma}_{q,k}\boldsymbol{x}} \leqslant \boldsymbol{g}_q^{\mathrm{T}}\boldsymbol{x} \quad \forall q \in \mathcal{Q},\ \forall k \in \mathcal{S}^I, \\ & \overline{\gamma}_{q,k} \geqslant \gamma_{q,k} \geqslant \underline{\gamma}_{q,k}, \forall q \in \mathcal{Q},\ \forall k \in \mathcal{S}^I, \\ & \textstyle\sum_{k\in\mathcal{S}^I} \gamma_{q,k} \geqslant 1-\alpha_q,\ \forall q \in \mathcal{Q}, \\ & \textstyle\sum_{q\in\mathcal{Q}} \alpha_q = \epsilon. \end{aligned} \tag{$\mathcal{LCB}_{\mathrm{M}}$}$$

In the case $|\mathcal{B}_i| = 1$, $\forall i \in \mathcal{I}$, the formulation $(\mathcal{LCB}_{\mathrm{U}})$ is recovered. In the case $I = 1$, the formulation for general (not necessarily diagonal) covariance matrix $\boldsymbol{\Sigma}_q$ is recovered. Since $I < B$, $(\mathcal{LCB}_{\mathrm{M}})$ has fewer conic constraints compared with $(\mathcal{LCB}_{\mathrm{U}})$. Moreover, the number of $\gamma_{q,k}$ and associated bilinear terms in $(\mathcal{LCB}_{\mathrm{M}})$ is also smaller than that of $(\mathcal{LCB}_{\mathrm{U}})$. Because $(\mathcal{LCB}_{\mathrm{M}})$ represents more general cases, its solution method will be our focus hereafter.

The lower bound $\underline{\gamma}_{q,k}$ in $(\mathcal{LCB}_{\mathrm{M}})$ is required to be greater than or equal to $0.5\delta_{q,k}$ such that a second-order cone relaxation and outer approximation can be constructed in the next section. This restriction narrows the feasible region and thus may lead to a sub-optimal solution of the original joint CCP.

3. Global optimization method for $(\mathcal{LCB}_{\mathrm{M}})$

3.1. Outer approximation

One of difficulties for optimizing $(\mathcal{LCB}_{\mathrm{M}})$ is that $\Phi^{-1}(\cdot)$ does not have an analytical form. Note that $\frac{\gamma_{q,k}}{\delta_{q,k}} \mapsto \Phi^{-1}(\frac{\gamma_{q,k}}{\delta_{q,k}})$ is convex when $\frac{\gamma_{q,k}}{\delta_{q,k}} \in [0.5, 1]$. An outer approximation for $\Phi^{-1}(\cdot)$ can be developed by following Cheng et al. (2012) and previous work (Yang et al., 2017, 2020). A graphical illustration is shown in Fig. 1. For the cutting plane at a sampling point of $\gamma_{q,k}$, denoted as $\gamma_{q,k,l}$, its slope and intercept are:

$$h_{q,k,l} = \left.\frac{d\Phi^{-1}(\frac{\gamma_{q,k}}{\delta_{q,k}})}{d(\frac{\gamma_{q,k}}{\delta_{q,k}})}\right|_{\frac{\gamma_{q,k,l}}{\delta_{q,k}}} = \frac{1}{\phi(\Phi^{-1}(\frac{\gamma_{q,k,l}}{\delta_{q,k}}))},$$

$$t_{q,k,l} = \Phi^{-1}(\frac{\gamma_{q,k,l}}{\delta_{q,k}}) - h_{q,k,l}\frac{\gamma_{q,k,l}}{\delta_{q,k}}.$$

Then, the equation of cutting plans is shown in Eq. (3)

$$v_{q,k} \geqslant h_{q,k,l} \frac{\gamma_{q,k,l}}{\delta_{q,k}} + t_{q,k,l}, \ \forall l \in \mathcal{L}, \tag{3}$$

where $v_{q,k}$ is a newly introduced auxiliary variable; $\mathcal{L}$ is the set of sampling points for cutting plane construction.

Consequently, an approximation of ($\mathcal{LCB}_{\mathrm{M}}$) is:

$$\begin{aligned} \min_{\boldsymbol{x}\in\mathcal{X},\gamma_{q,k},v_{q,k},\alpha_q} \quad & \boldsymbol{r}^{\mathrm{T}}\boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{\mu}_{q,k}^{\mathrm{T}}\boldsymbol{x} + v_{q,k}\sqrt{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{\Sigma}_{q,k}\boldsymbol{x}} \leq \boldsymbol{g}_q^{\mathrm{T}}\boldsymbol{x} \quad \forall q \in \mathcal{Q}, \forall k \in \mathcal{S}^I, \\ & \textstyle\sum_{k\in\mathcal{S}^I} \gamma_{q,k} \geqslant 1-\alpha_q, \ \forall q \in \mathcal{Q}, \\ & \overline{\gamma}_{q,k} \geqslant \gamma_{q,k} \geqslant \underline{\gamma}_{q,k}, \forall q \in \mathcal{Q}, \forall k \in \mathcal{S}^I, \\ & v_{q,k} \geqslant h_{q,k,l}\frac{\gamma_{q,k,l}}{\delta_{q,k}} + t_{q,k,l}, \ \forall q \in \mathcal{Q}, \ \forall k \in \mathcal{S}^I, \ \forall l \in \mathcal{L}, \\ & \textstyle\sum_{q\in\mathcal{Q}} \alpha_q = \epsilon. \end{aligned} \tag{$\mathcal{LCB}_{\mathrm{A}}$}$$

As $|\mathcal{L}| \to \infty$, ($\mathcal{LCB}_{\mathrm{A}}$) approaches ($\mathcal{LCB}_{\mathrm{M}}$) with arbitrary accuracy. In practice, a number of sampling points are evenly distributed in the domain $[0.5, 1]$ initially. The solution points $\gamma_{q,k}^*$ obtained during the branch-and-bound will be chosen as the new sampling points for cutting plane generation to continuously tighten the outer approximation.

3.2. Convex relaxation

The formula ($\mathcal{LCB}_{\mathrm{A}}$) is still not convex because of the term $v_{q,k}\sqrt{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{\Sigma}_{q,k}\boldsymbol{x}}$. However, the McCormick method (McCormick, 1976) can be used to develop a convex relaxation. Let us introduce a new variable for the bilinear term $\boldsymbol{\beta}_{q,k} = v_{q,k}\boldsymbol{x}$, such that

$$v_{q,k}\sqrt{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{\Sigma}_{q,k}\boldsymbol{x}} = \sqrt{\boldsymbol{\beta}_{q,k}^{\mathrm{T}}\boldsymbol{\Sigma}_{q,k}\boldsymbol{\beta}_{q,k}}$$

According to the McCormick relaxation, a set of inequalities should be introduced to replace the bilinear term. Then, the resulting convex relaxation (SOCP form) is:

$$\begin{aligned} \min_{\boldsymbol{x}\in\mathcal{X},\gamma_{q,k},v_{q,k},\alpha_q,\boldsymbol{\beta}_{q,k}} \quad & \boldsymbol{r}^{\mathrm{T}}\boldsymbol{x} \\ \text{s.t.} \quad & \boldsymbol{\mu}_{q,k}^{\mathrm{T}}\boldsymbol{x} + \sqrt{\boldsymbol{\beta}_{q,k}^{\mathrm{T}}\boldsymbol{\Sigma}_{q,k}\boldsymbol{\beta}_{q,k}} \leqslant \boldsymbol{g}_q^{\mathrm{T}}\boldsymbol{x} \quad \forall q \in \mathcal{Q}, \ \forall k \in \mathcal{S}^I, \\ & \textstyle\sum_{k\in\mathcal{S}^I} \gamma_{q,k} \geqslant 1-\alpha_q, \ \forall q \in \mathcal{Q}, \\ & \overline{\gamma}_{q,k} \geqslant \gamma_{q,k} \geqslant \underline{\gamma}_{q,k}, \forall q \in \mathcal{Q}, \ \forall k \in \mathcal{S}^I, \\ & v_{q,k} \geqslant h_{q,k,l}\frac{\gamma_{q,k,l}}{\delta_{q,k}} + t_{q,k,l}, \ \forall q \in \mathcal{Q}, \ \forall k \in \mathcal{S}^I, \ \forall l \in \mathcal{L}, \\ & \textstyle\sum_{q\in\mathcal{Q}} \alpha_q = \epsilon, \\ & \boldsymbol{\beta}_{q,k} \geqslant \overline{v}_{q,k}\boldsymbol{x} + v_{q,k}\overline{\boldsymbol{x}} - \overline{v}_{q,k}\overline{\boldsymbol{x}}, \\ & \boldsymbol{\beta}_{q,k} \geqslant \underline{v}_{q,k}\boldsymbol{x} + v_{q,k}\underline{\boldsymbol{x}} - \underline{v}_{q,k}\underline{\boldsymbol{x}}, \\ & \boldsymbol{\beta}_{q,k} \leqslant \overline{v}_{q,k}\boldsymbol{x} + v_{q,k}\underline{\boldsymbol{x}} - \overline{v}_{q,k}\underline{\boldsymbol{x}}, \\ & \boldsymbol{\beta}_{q,k} \leqslant \underline{v}_{q,k}\boldsymbol{x} + v_{q,k}\overline{\boldsymbol{x}} - \underline{v}_{q,k}\overline{\boldsymbol{x}}, \end{aligned} \tag{$\mathcal{LCB}_{\mathrm{C}}$}$$

where the upper and lower bars represent the upper and lower bounds on variables, respectively. Solving ($\mathcal{LCB}_{\mathrm{C}}$) will yield a lower bound solution, denoted as LB.

This SOCP form is the essential step of the proposed global optimization scheme. It requires that constraints can be affinely relaxed and affinely dependent on the uncertain parameters. The linear mixing law meets these two criteria. If a nonlinear mixing law satisfies these conditions, such as the octane model in Yang et al. (2020), it is also solvable by the proposed approach.

3.3. Optimality-based bounds tightening

McCormick method generates the tightest convex envelop for bilinear terms. In general, a smaller variable interval implies tighter relaxation. Traditional interval analysis can be used to reduce the bound quickly. However, previous research shows that an optimality-based bounds tightening (OBBT) can be more efficient. In that scheme, two convex optimization subproblems are solved for each variable to yield its lower and upper bounds, respectively. In the blending problem, both $\boldsymbol{x}$ and $v_{q,k}$ are involved in bilinear terms. Thus, their OBBT formulas are shown in ($\mathcal{BT}$):

$$\begin{aligned} \min/\max_{\boldsymbol{x}\in\mathcal{X},\gamma_{q,k},v_{q,k},\alpha_q,\boldsymbol{\beta}_{q,k}} \quad & x_b \ \text{ or } \ v_{q,k} \\ \text{s.t.} \quad & \boldsymbol{\mu}_{q,k}^{\mathrm{T}}\boldsymbol{x} + \sqrt{\boldsymbol{\beta}_{q,k}^{\mathrm{T}}\boldsymbol{\Sigma}_{q,k}\boldsymbol{\beta}_{q,k}} \leq \boldsymbol{g}_q^{\mathrm{T}}\boldsymbol{x} \quad \forall q \in \mathcal{Q}, \ \forall k \in \mathcal{S}^I, \\ & \boldsymbol{r}^{\mathrm{T}}\boldsymbol{x} \leqslant \mathrm{UB}, \\ & \textstyle\sum_{k\in\mathcal{S}^I} \gamma_{q,k} \geq 1-\alpha_q, \ \forall q \in \mathcal{Q}, \\ & \overline{\gamma}_{q,k} \geqslant \gamma_{q,k} \geqslant \underline{\gamma}_{q,k}, \ \forall q \in \mathcal{Q}, \ \forall k \in \mathcal{S}^I, \\ & v_{q,k} \geqslant h_{q,k,l}\frac{\gamma_{q,k,l}}{\delta_{q,k}} + t_{q,k,l}, \ \forall q \in \mathcal{Q}, \ \forall k \in \mathcal{S}^I, \ \forall l \in \mathcal{L}, \\ & \textstyle\sum_{q\in\mathcal{Q}} \alpha_q = \epsilon, \\ & \boldsymbol{\beta}_{q,k} \geqslant \overline{v}_{q,k}\boldsymbol{x} + v_{q,k}\overline{\boldsymbol{x}} - \overline{v}_{q,k}\overline{\boldsymbol{x}}, \\ & \boldsymbol{\beta}_{q,k} \geqslant \underline{v}_{q,k}\boldsymbol{x} + v_{q,k}\underline{\boldsymbol{x}} - \underline{v}_{q,k}\underline{\boldsymbol{x}}, \\ & \boldsymbol{\beta}_{q,k} \leqslant \overline{v}_{q,k}\boldsymbol{x} + v_{q,k}\underline{\boldsymbol{x}} - \overline{v}_{q,k}\underline{\boldsymbol{x}}, \\ & \boldsymbol{\beta}_{q,k} \leqslant \underline{v}_{q,k}\boldsymbol{x} + v_{q,k}\overline{\boldsymbol{x}} - \underline{v}_{q,k}\overline{\boldsymbol{x}}, \end{aligned} \tag{$\mathcal{BT}$}$$

where UB is the upper bound solution of ($\mathcal{LCB}_{\mathrm{M}}$). The constraint $\boldsymbol{r}^{\mathrm{T}}\boldsymbol{x} \leqslant \mathrm{UB}$ requires that any feasible values of x_b or $v_{q,k}$ lead to a lower solution than UB. Because ($\mathcal{BT}$) is still a SCOP, x_b and $v_{q,k}$ can be solved to their valid bounds quickly.

3.4. Reformulation-linearization technique

The reformulation-linearization technique (RLT) is a way to create extra linear constraints for tightening the convex relaxation. Let us consider a valid, linear, and deterministic constraint for $\boldsymbol{x}$, shown in Eq. (4):

$$\boldsymbol{u}^{\mathrm{T}}\boldsymbol{x} \leqslant U, \tag{4}$$

where $\boldsymbol{u}$ and U are deterministic coefficients. Eq. (4) can be multiplied with $v_{q,k}$ to form a series of new constraints:

$$\sum_{b\in\mathcal{B}} \boldsymbol{u}^{\mathrm{T}}\boldsymbol{x}v_{q,k} \leqslant Uv_{q,k} \to \boldsymbol{u}^{\mathrm{T}}\boldsymbol{\beta}_{q,k} \leqslant Uv_{q,k}, \ \forall k \in \mathcal{S}^I, \forall q \in \mathcal{Q}. \tag{5}$$

Note that (5) does not introduce any new variables. It only constrains auxiliary variable $\boldsymbol{\beta}$ used in the McCormick relaxation. Thus, (5) can be integrated into ($\mathcal{LCB}_{\mathrm{C}}$) and ($\mathcal{BT}$) to tighten the convex relaxation gap, and speed up the global optimization.

3.5. Branch-and-bound

In this sub-section, a global optimization framework for the conservative approximation formula ($\mathcal{LCB}_{\mathrm{M}}$) is developed. Even though solving ($\mathcal{BT}$) can tighten the convex relaxation, a branch-and-bound scheme is more crucial and efficient to reduce the relaxation gap continuously. To this end, the root node of a searching tree is initialized with the entire variable intervals. During the tree traversal, ($\mathcal{LCB}_{\mathrm{C}}$) is solved at each node with associated intervals, and then branching is conducted to generate new nodes on the searching tree. According to previous research (Yang et al., 2017), branching $\boldsymbol{x}$ is more efficient than branching $v_{q,k}$ for global optimization. Hence, the following criterion is proposed for branching variable selection.

$$b' = \arg\max_{b\in\mathcal{B}} \sum_{q\in\mathcal{Q}} \sum_{k\in\mathcal{S}^I} |v_{q,k}^* x_b^* - \beta_{q,k,b}^*|, \tag{6}$$

where $v_{q,k}^*$, x_b^* and $\beta_{q,k,b}^*$ are the solution of ($\mathcal{LCB}_{\mathrm{C}}$) at a node. Once $x_{b'}$ is chosen, its interval is partitioned into two parts: $[\underline{x}_{b'}, x_{b'}^*]$ and $[x_{b'}^*, \overline{x}_{b'}]$,

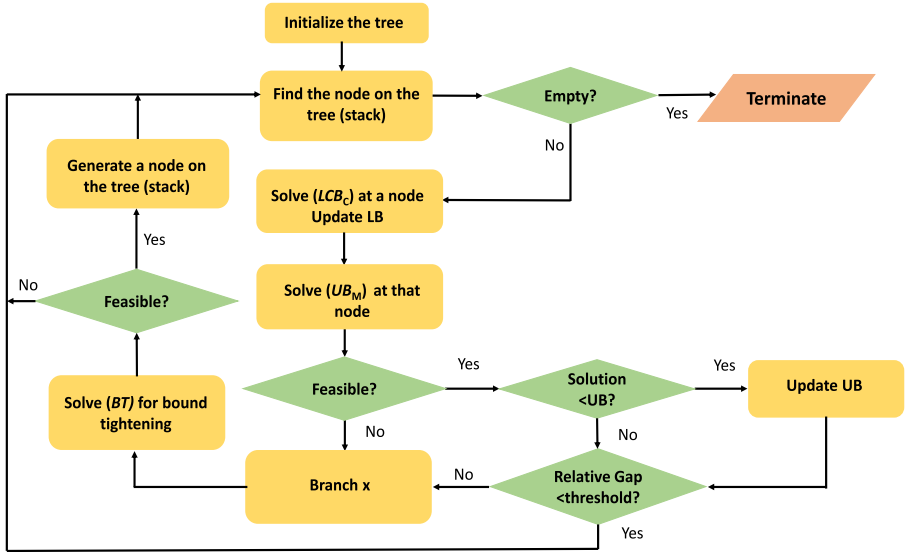

Fig. 2. The algorithm flowchart for solving ($\mathcal{LCB}_{\mathrm{M}}$).

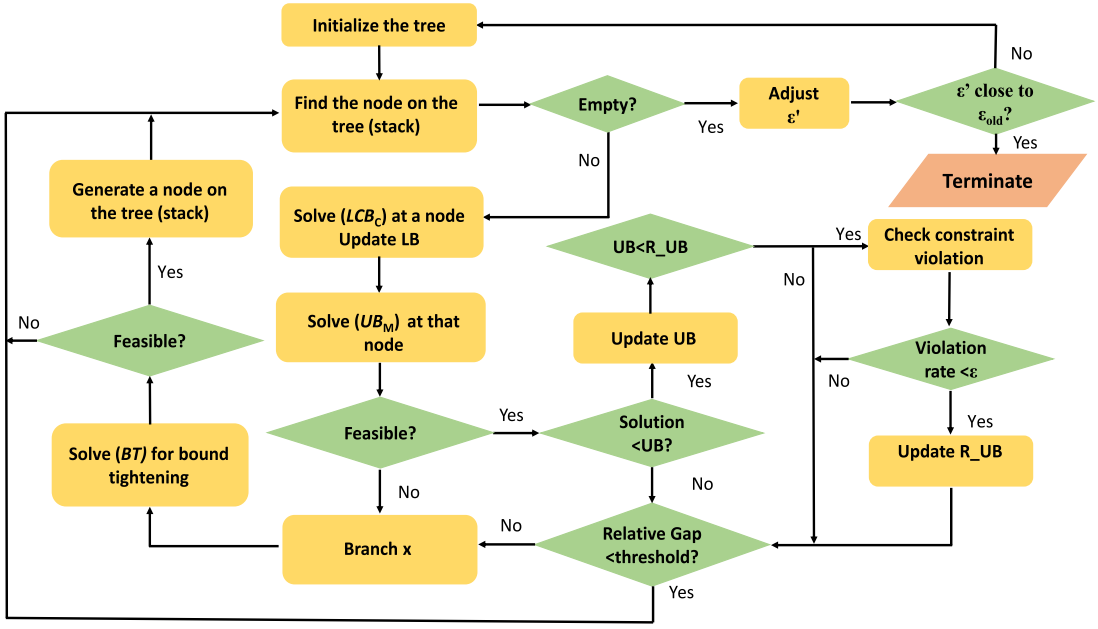

Fig. 3. The algorithm flowchart for solving ($\mathcal{LCB}_{\mathrm{R}}$) with ϵ' adjustment. UB is the upper bound solution given a specified ϵ' in ($\mathcal{LCB}_{\mathrm{R}}$). LB is the lower bound solution. R_UB is the true optimal solution meeting chance constraints with the desired risk level ϵ.

while other variables' bounds are unchanged. The resulting two exclusive feasible regions will be saved as two nodes in the searching tree for further processing.

Once the risk level $\gamma^*_{q,k}$ in the solution of ($\mathcal{LCB}_{\mathrm{C}}$) at a node is obtained, it can be substituted into ($\mathcal{LCB}_{\mathrm{M}}$) to yield an upper bounding problem:

$$\begin{aligned} &\min_{\boldsymbol{x}\in\mathcal{X}} \ \boldsymbol{r}^{\mathrm{T}}\boldsymbol{x} \\ &\text{s.t.} \ \ \boldsymbol{\mu}_{q,k}^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{\Phi}^{-1}\left(\frac{\gamma^*_{q,k}}{\delta_{q,k}}\right)\sqrt{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{\Sigma}_{q,k}\boldsymbol{x}} \leqslant \boldsymbol{g}_q^{\mathrm{T}}\boldsymbol{x} \quad \forall q \in \mathcal{Q},\ \forall k \in \mathcal{S}^I. \end{aligned} \tag{$\mathcal{UB}_{\mathrm{M}}$}$$

If ($\mathcal{UB}_{\mathrm{M}}$) is infeasible, it implies that the convex envelop made from outer approximation and McCormick relaxation is not tight enough. Then, the sampling point $\gamma^*_{q,k}$ should be included in set $\mathcal{L}$ to yield an additional cutting plane, and the variable bounds should be further reduced by branching. If ($\mathcal{UB}_{\mathrm{M}}$) is feasible and its objective value is smaller than the existing UB, then UB can be updated. The entire algorithm will terminate if the relative gap, defined in Eq. (7), is smaller than a threshold.

$$\text{Relative gap} = \frac{\text{UB} - \text{LB}}{|\text{LB}|} \tag{7}$$

A flowchart of the global optimization algorithm for the deterministic approximation of joint CCP is shown in Fig. 2.

3.6. Risk level adjustment

($\mathcal{LCB}_{\mathrm{M}}$) serves as a conservative approximation of ($\mathcal{CBP}$) based on the Boole's inequality for decomposing joint chance constraints and enforced constraint $\gamma_{q,k} \geqslant \underline{\gamma}_{q,k} = 0.5\delta_{q,k}$. The global optimal solution of ($\mathcal{LCB}_{\mathrm{M}}$) can be further improved by adjusting the risk level if posterior evaluation is allowed. Two approaches can be applied:

Method I Replace some conic constraints by their deterministic linear part.

Method II Directly increase ϵ in $(\mathcal{LCB}_{\mathrm{M}})$.

Both methods can be synthesized into a new optimization formula:

$$
\begin{aligned}
\min_{\boldsymbol{x}\in\mathcal{X},\gamma_{q,k},\alpha_q}\quad & \boldsymbol{r}^{\mathrm{T}}\boldsymbol{x} \\
\text{s.t.}\quad & \boldsymbol{\mu}_{q,k}^{\mathrm{T}}\boldsymbol{x}+\boldsymbol{\Phi}^{-1}\left(\tfrac{\gamma_{q,k}}{\delta_{q,k}}\right)\sqrt{\boldsymbol{x}^{\mathrm{T}}\boldsymbol{\Sigma}_{q,k}\boldsymbol{x}}\leqslant \boldsymbol{g}_q^{\mathrm{T}}\boldsymbol{x} \quad \forall q\in\mathcal{Q},\ \forall k\in\mathcal{S}^I\setminus\tilde{\mathcal{S}}_q^I, \\
& \boldsymbol{\mu}_{q,k}^{\mathrm{T}}\boldsymbol{x}\leqslant \boldsymbol{g}_q^{\mathrm{T}}\boldsymbol{x} \quad \forall q\in\mathcal{Q},\ \forall k\in\tilde{\mathcal{S}}_q^I, \\
& \overline{\gamma}_{q,k}\geqslant\gamma_{q,k}\geqslant\underline{\gamma}_{q,k}, \quad \forall q\in\mathcal{Q},\ \forall k\in\mathcal{S}^I, \\
& \textstyle\sum_{k\in\mathcal{S}^I}\gamma_{q,k}\geqslant 1-\alpha_q,\ \forall q\in\mathcal{Q}, \\
& \textstyle\sum_{q\in\mathcal{Q}}\alpha_q=\epsilon',
\end{aligned}
\tag{$\mathcal{LCB}_{\mathrm{R}}$}
$$

where the index set $\tilde{\mathcal{S}}_q^I$ incorporates eliminated second-order cone constraints for quality q, defined as $\tilde{\mathcal{S}}_q^I := \{k|\delta_{q,k}<R\}, \forall q\in\mathcal{Q}$. R is a small positive threshold to limit the number of dropped second-order cone constraints. $(\mathcal{LCB}_{\mathrm{R}})$ is a further relaxation of $(\mathcal{LCB}_{\mathrm{M}})$ because the square root part, as a positive protection term, is removed to recover the deterministic linear constraint. Moreover, this relaxation can reduce the number of bilinear terms, and thus may speed up the global optimization algorithm. By increasing the allowable risk level $\epsilon'>\epsilon$ in $(\mathcal{LCB}_{\mathrm{R}})$, the limitation of Boole's inequality in solving joint chance-constrained programs can be further hedged.

Both methods I and II need posterior evaluation. To this end, a large number of independent scenarios for uncertain parameters should be generated as a testbed to assess the constraints violation rate, given a solution $\boldsymbol{x}$. Then, an one-dimensional bi-section search scheme can be used to adjust ϵ'. Let us pre-specify a step size Δ such that $\epsilon'=\epsilon+\Delta$ and $\epsilon_{old}=\epsilon$. When a better upper bound (feasible) solution of the resulting $(\mathcal{LCB}_{\mathrm{R}})$ is found during the global optimization, it will be evaluated on the test scenarios to determine its probabilistic feasibility in $(\mathcal{CBP})$. If the evaluated constraint violation rate is greater than ϵ, then $\Delta\leftarrow\Delta/2$; Otherwise $\epsilon_{old}=\epsilon'$. Let $\epsilon'=\epsilon_{old}+\Delta$ and solve $(\mathcal{LCB}_{\mathrm{R}})$ again. The entire procedure of the risk level adjustment is shown in Fig. 3.

It is worthwhile to note that even though the risk level adjustment can reduce the conservativeness of the solution, the proposed method still cannot guarantee the global optimality of the original joint CCP due to Boole's inequality and approximation nature of GMM.

4. Case studies

Two blending problems with non-Gaussian distributed uncertainties are solved in this section. By observing the histogram of uncertain parameters shown in Figs. 4–13, it is clear that GMM is needed. We start from 2-component and gradually increase the number of Gaussian components to improve the likelihood of data fitting. This procedure terminates when the likelihood cannot be enhanced significantly. Because the number of conic constraints and bilinear terms are dependent on the quantity of Gaussian components, less component GMM is preferred to enable fast calculation. In case studies, we find that 2-component GMM is sufficient to approximate the true distribution and does not incur considerably long computational time. The posterior evaluation of the CCP solution consisting of 10 000 independent samples of uncertain parameters is conducted to mitigate the approximation error.

The software platform is GAMS 32.2.0, with SOCP solver CPLEX 12.10. The hardware platform is a laptop with Intel Core i7-7500U CPU 2.70 GHZ and 8GM RAM. The risk level ϵ is set as 5%, as previous research (Yang et al., 2017). When the relative gap reaches 1%, the global optimization for $(\mathcal{LCB}_{\mathrm{M}})$ terminates. There are 200 sampling points evenly distributed in the domain $[0.5, 1]$ of $\gamma_{q,k}$ during the initial stage of outer approximation for both cases. More sampling points and cutting planes will be generated on-the-fly based on the solutions of branch-and-bound. For comparison, SA is employed to solve two cases with decreasing number of scenarios. Because SAA requires special algorithms to tune step size, smoothing parameters, and initial guesses (Kannan and Luedtke, 2021), it is not considered here.

Table 1
Product quality standard of steel.

Quality	Upper limit	Lower limit
Carbon content	3.50%	2.50%
Chrome Content	0.45%	0.10%
Manganese Content	1.65%	1.00%
Silicon Content	4.00%	2.00%

Table 2
Raw materials nominal quality, cost, and available amount.

Raw materials	Cost per Pound	Carbon percent	Chrome percent	Manganese percent	Silicon percent	Amount available
Pig Iron 1	0.0300	4.00%	0	0.90%	2.25%	1500
Pig Iron 2	0.0645	0	10.00%	4.50%	15.00%	1500
Ferro-Silicon 1	0.0650	0	0	0	45.00%	1500
Ferro-Silicon 2	0.0610	0	0	0	42.00%	1500
Alloy 1	0.1000	0	0	60.00%	18.00%	1500
Alloy 2	0.1300	0	20.00%	9.00%	30.00%	1500
Alloy 3	0.1190	0	8.00%	33.00%	25.00%	1500
Carbide	0.0800	15.00%	0	0	30.00%	20
Steel 1	0.0210	0.40%	0	0.90%	0	200
Steel 2	0.0200	0.10%	0	0.30%	0	200
Steel 3	0.0195	0.10%	0	0.30%	0	200

Table 3
Mean, standard deviation, and weight of GMM component in case 1.

Quality	Mean 1	Mean 2	Standard deviation 1	Standard deviation 2	Weight 1	Weight 2
Steel 2 Carbon	0.0851	0.1487	0.0008	0.0036	0.7653	0.2347
Steel 3 Carbon	0.0648	0.1529	0.0007	0.0054	0.6371	0.3629
Steel 2 Manganese	0.3583	0.2612	0.0091	0.0035	0.3546	0.6454
Steel 3 Manganese	0.2718	0.3515	0.0021	0.0049	0.6424	0.3576
Pig Iron 1 Silicon	2.0346	2.8375	0.1898	0.5100	0.6728	0.3272

4.1. Case 1: The pittsburgh steel company blending problem

A manufacturer plans to produce a new type of steel with the following quality standard, shown in Table 1, and minimize the production cost (Schrage, 2006). The nominal quality parameters $\boldsymbol{\xi}_q$ and cost of raw materials are shown in Table 2. All parameters are not correlated. Except chrome content, all other three quality parameters are subject to uncertainties. Steel 2 carbon and manganese percent, Steel 3 carbon and manganese percent, and Pig Iron 1 silicon percent are assumed to satisfy lognormal distribution. Their PDFs are non-symmetric, and approximated by 2-component GMMs with 1000 data samples, shown in Figs. 4–8. The mean, standard deviation, and weights of each Gaussian component for these properties are shown in Table 3. Other non-zero parameters of carbon, manganese, and silicon are Gaussian distributed, $\tilde{\boldsymbol{\xi}}_q\sim\mathcal{N}(\boldsymbol{\xi}_q, \mathrm{diag}(0.01\boldsymbol{\xi}_q)^2)$.

From the last two columns of Table 3, the $\delta_{q,k}$ can be calculated. Because the minimal value of δ for carbon $0.2347\times0.3629=0.0852$ is still larger than ϵ, we decide not to remove any chance constraints, and set $\epsilon'=\epsilon=5\%$. As a result, $(\mathcal{LCB}_{\mathrm{R}})$ recovers to $(\mathcal{LCB}_{\mathrm{M}})$, and it can be solved to the global optimum. The solution time, objective value, constraints violation rate in posterior evaluation, and $1-\gamma_{q,k}/\delta_{q,k}$ are shown in Table 4. Through the sampling-based posterior evaluation, the solution results in 4.73% constraint violation rate, which is very close to the desired 5%. After updating the ϵ', a better solution cannot be found if the relative gap is set as 1% to terminate the global optimization of $(\mathcal{LCB}_{\mathrm{R}})$ given a specified ϵ'. The final solution of blending recipe is shown in Table 5.

For comparison, the sampling complexity in Alamo et al. (2015) is adopted to implement SA. With 11 variables, confidence 10^{-6}, and $\epsilon=5\%$, the number of scenarios should be at least 738 (Alamo et al., 2015).

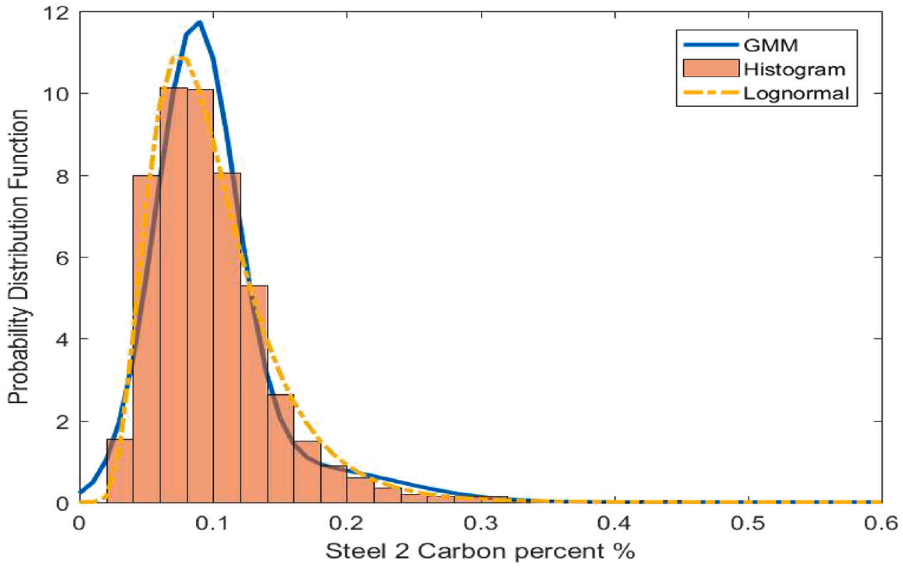

Fig. 4. Steel 2 carbon percent lognormal PDF, GMM approximation, and histogram (1000 data points).

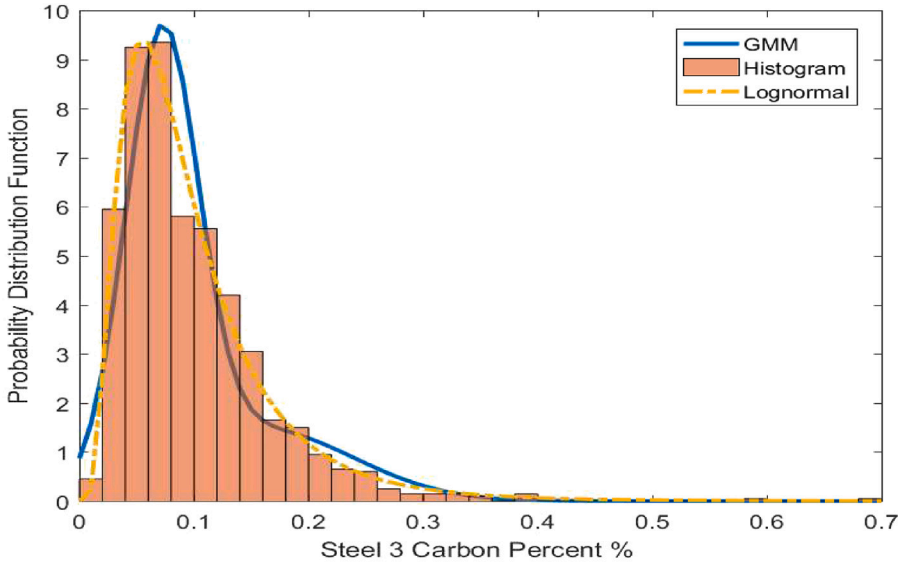

Fig. 5. Steel 3 carbon percent lognormal PDF, GMM approximation, and histogram (1000 data points).

To mitigate the randomness of sampled scenarios, SA is repeated 200 times with different batches of scenarios, and only the best solution is reported in Table 4. Unfortunately, no feasible solution is found within 200 replicates, implying conservativeness of the sample complexity. By allowing the posterior evaluation, we can reduce the number of scenarios and solve the SA many times until a satisfactory solution is obtained. Via trial-and-error, a comparable solution is found using 80 scenarios, and 200 replicates. Please note that GMM-CCP finds such a solution without using posterior evaluation (no risk level adjustment), whereas SA needs a significantly large number of samples to build the formula and check the probabilistic feasibility.

Table 4
Results of case 1. Total constraint violation rate is the synthesis (not summation) of all qualities.

	Solution time (s)	Objective value (Cost)
Proposed method	73.9	28.524
SA 738	69.3	Infeasible
SA 80	67.5	28.528
	Violation rate	$1-\gamma_{q,k}/\delta_{q,k}$
Carbon upper limit	0	0.01%, 0.01%, 0.01%, 0.01%
Carbon lower limit	2.35%	2.64%, 1.94%, 2.64%, 1.94%
Manganese upper limit	0	0.01%, 0.01%, 0.01%, 0.01%
Manganese lower limit	0.37%	0.39%, 0.20%,0.40%, 0.20%
Silicon upper limit	0.41%	0.01%, 0.71%
Silicon lower limit	1.69%	2.51%, 1.02%
Total	4.73%	N/A

4.2. Case 2: Gasoline blending problem

An oil company plans to produce 3 types of gasoline to meet quality specification with 95% and yield high profit. The price and quality specifications of 3 types gasoline are shown in Table 6. Each type of gasoline has maximal production 50 and minimal production 1. There are 10 types of intermediate blendstocks with nominal quality parameters shown in Table 7. We assume that RVP, octane rating, and sulfur are subject to uncertainties. The RVP of x_3, x_5, x_{10}, sulfur of x_4, and octane of x_5 satisfy lognormal distribution, approximated by

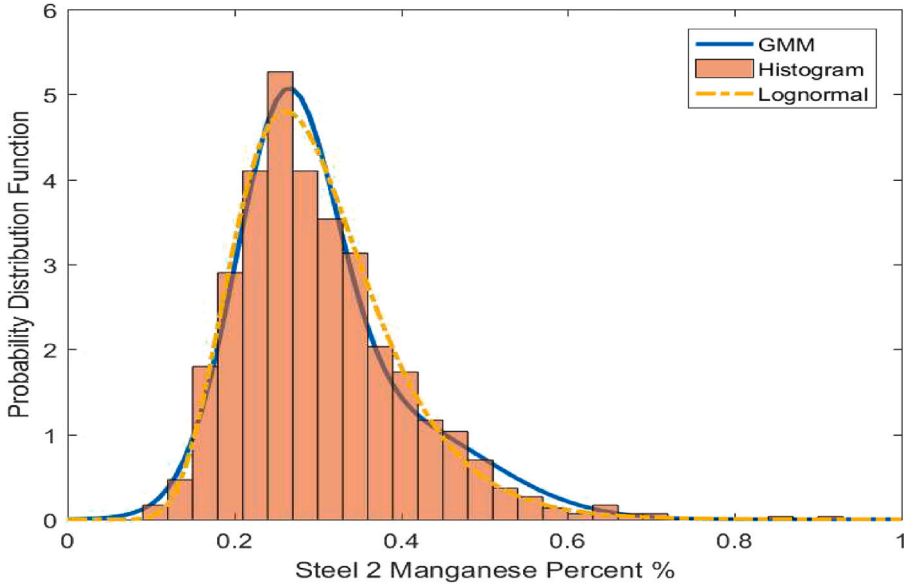

Fig. 6. Steel 2 manganese percent lognormal PDF, GMM approximation, and histogram (1000 data points).

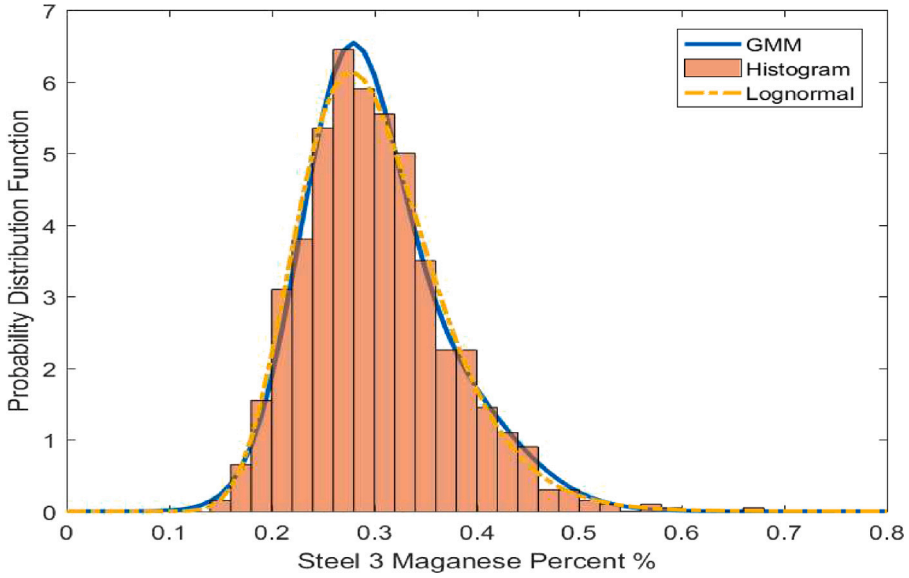

Fig. 7. Steel 3 manganese percent lognormal PDF, GMM approximation, and histogram (1000 data points).

Table 5
Blending recipe for case 1.

Raw materials	Amount	Percent
Pig Iron 1	586.601	58.66%
Pig Iron 2	10.000	1.00%
Ferro-Silicon 1	0.000	0
Ferro-Silicon 2	10.601	1.06%
Alloy 1	6.234	0.62%
Alloy 2	0	0
Alloy 3	0	0
Carbide	20.000	2%
Steel 1	175.181	17.52%
Steel 2	0.000	0
Steel 3	191.377	19.14%

2-component GMM with 1000 independent samples. Their PDFs and histogram are shown in Figs. 9–13. The GMMs parameters are listed in Table 8. Other non-zero parameters of RVP, octane rating, and sulfur

Table 6
Gasoline specifications and prices.

Gasoline	Density (max)	RVP (max)	Octane (min)	Sulfur (max)	Benzene (max)	Price
Type I	0.79	7.8	85	10	0.6	49.7
Type II	0.79	7.8	87	10	0.6	52.0
Type III	0.79	7.8	91	10	0.6	54.6

are Gaussian distributed, $\tilde{\boldsymbol{\xi}}_q \sim \mathcal{N}(\boldsymbol{\xi}_q, \boldsymbol{\Sigma}_q)$. The diagonal terms of $\boldsymbol{\Sigma}_q$ are $(0.01\boldsymbol{\xi}_q)^2$ and the octane rating of x_8 and x_9 are correlated with covariance 0.4295.

Given weight $w_{b,s}$ in Table 8, there are $2^3 = 8$ combinations of Gaussian components (conic constraints) for RVP. Let us set threshold $R = 0.05$. Then, three conic constraints, corresponding to $\delta_{\text{RVP},8} = 0.1891\times0.2797\times0.3599 = 0.0190$, $\delta_{\text{RVP},6} = 0.1891\times0.7203\times0.3599 = 0.0490$, and $\delta_{\text{RVP},7} = 0.1891 \times 0.2797 \times 0.6401 = 0.0339$, can be replaced by their linear part. We set the initial $\epsilon' = \epsilon = 5\%$ and compare the

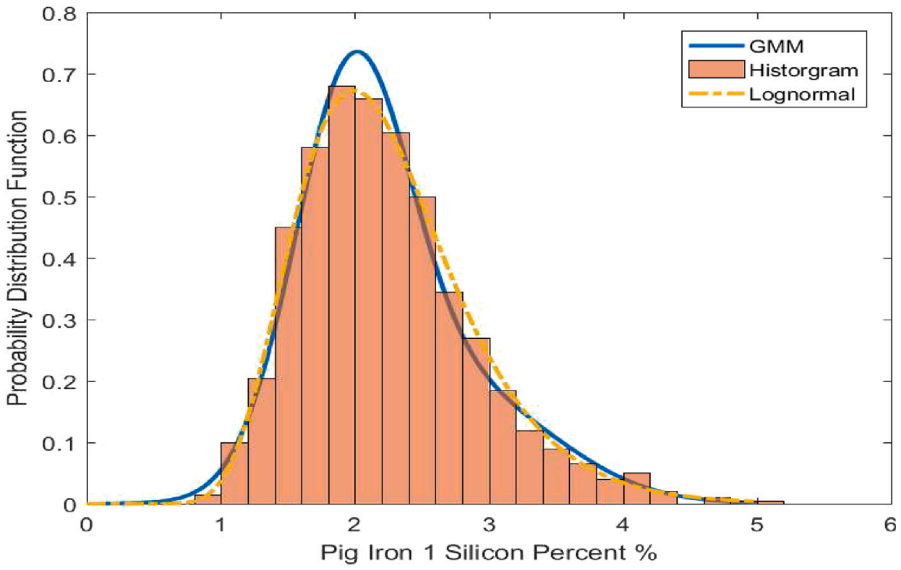

Fig. 8. Pig Iron 1 silicon percent lognormal PDF, GMM approximation, and histogram (1000 data points).

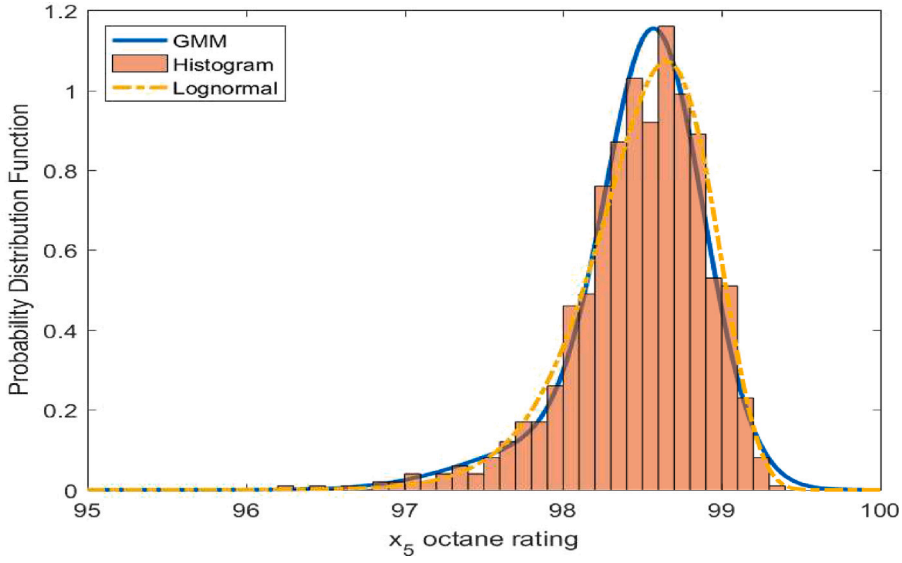

Fig. 9. x_5 octane rating lognormal PDF, GMM approximation, and histogram (1000 data points).

Table 7
Blendstocks nominal parameters.

Blendstock	Max flow	Density	RVP	Octane	Sulfur	Benzene	Price	Stock
x_1	8	0.565	60	84.8	10	0	36	8.0
x_2	2.6	0.656	10.4	69.8	0	3.8	40	2.6
x_3	12.0	0.772	1.6	71.9	6	0.4	39	12
x_4	20.1	0.618	11.2	82.1	1	0	41	20.1
x_5	15.6	0.855	2.7	99	0	0	52	15.6
x_6	2.3	0.693	4.6	68	0	0	42	2.3
x_7	26.3	0.679	10.6	91.6	41	0.6	47	26.3
x_8	17.5	0.757	4.4	86.7	111	0	44	17.5
x_9	9.2	0.803	3.3	85.1	30	0	45	9.2
x_{10}	18	0.713	2.1	93.7	20	0	55	18.0

Table 8
Mean, standard deviation, and weight of GMM components in case 2.

Quality	Mean 1	Mean 2	Standard deviation 1	Standard deviation 2	Weight 1	Weight 2
x_3 RVP	1.4932	1.9487	0.0569	0.1202	0.8109	0.1891
x_5 RVP	2.5357	3.1574	0.2108	0.4262	0.7203	0.2797
x_{10} RVP	0.3583	0.2612	0.0091	0.0035	0.6401	0.3599
x_4 sulfur	1.1332	0.9433	0.0530	0.0228	0.2950	0.7050
x_5 octane	98.2102	98.7116	0.1807	0.0602	0.4459	0.5541

solution with and without conic constraints relaxation, in Table 9. Their objective values are similar, but the solution time is significantly reduced by relaxing several conic constraints. This relaxation yields the less conservative formula ($\mathcal{LCB}_{\mathrm{R}}$) and makes the on-spec rate of the solution closer to the desired level, 95%. It is worthwhile to note that the resulting solution is infeasible to the conservative approximation ($\mathcal{LCB}_{\mathrm{M}}$), but can be feasible to the original joint CCP if posterior evaluation can validate it.

To speed up the global optimization, we can introduce RLT based on any linear and deterministic constraints by multiplying $v_{q,k}$ with them. Besides existing density and Benzene quality constraints, the following

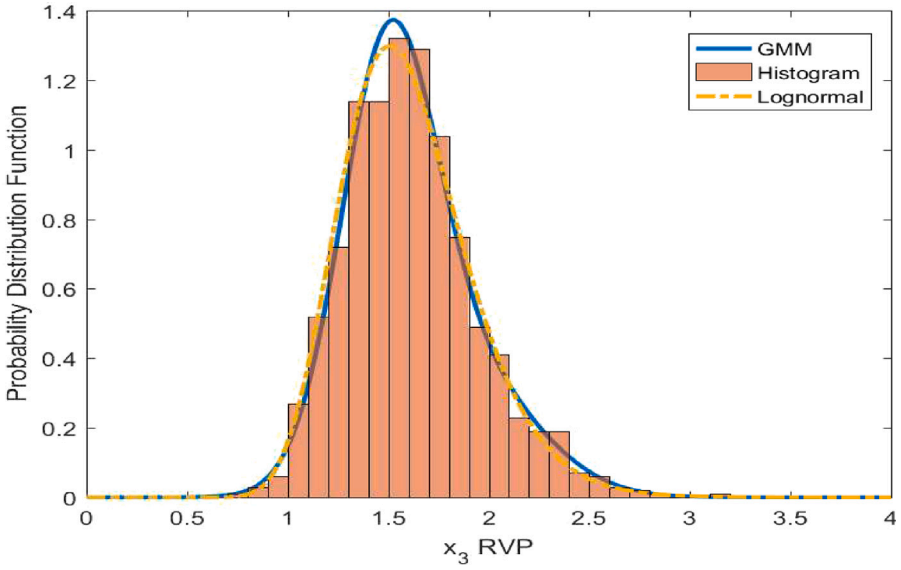

Fig. 10. x_3 RVP lognormal PDF, GMM approximation, and histogram (1000 data points).

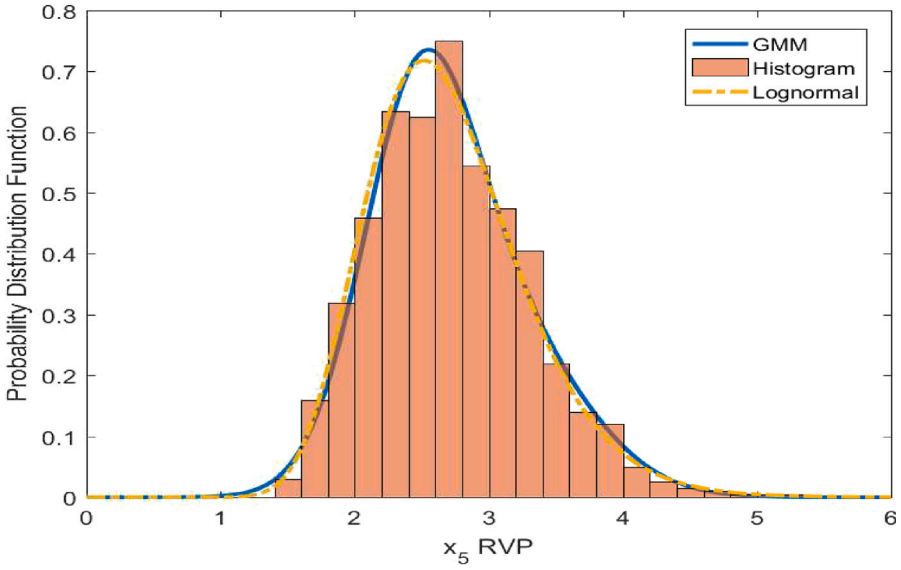

Fig. 11. x_5 RVP lognormal PDF, GMM approximation, and histogram (1000 data points).

total flow constraint is also considered for each product:

$$\underline{U}_p \leqslant \sum_i x_{i,p} \leqslant \overline{U}_p, \ \ p = 1, 2, 3, \tag{8}$$

where the subscript p represents each product. Although the default upper bound value $\overline{U}_p$ is 50 and lower bound value $\underline{U}_p$ is zero, they can be further tightened through OBBT. Then, two extra RLTs are generated for each gasoline. Table 9 shows that introducing these extra RLTs can substantially reduce the solution time.

To further improve the objective value, we use Algorithm 2 to adjust ϵ' and show the final solutions in Table 10. The conic constraint relaxation leads to a slightly better solution and shorter computation time again. However, even though ϵ' is increased, the objective value is only slightly improved. This is reasonable because the on-spec rate is already nearly 95% when $\epsilon' = 5\%$.

To compare the proposed method with SA, we still employ the sample complexity proposed by Alamo et al. (2015). Even though there are 30 variables in total, the problem requires each type of gasoline to meet specifications with 95% chance, respectively. Thus, there should be 702 scenarios based on 10 variables, confidence 10^{-6}, and $\epsilon = 5\%$. Then, the number of scenarios is gradually reduced if the posterior evaluation shows the probabilistic feasibility of the resulting solution. To mitigate the randomness of sampled scenarios, SA is repeated 200 times with different batches of scenarios, and only the best solution is reported in Table 10. These results show that the sample complexity bound proposed in Alamo et al. (2015) is valid but conservative. By reducing the number of scenarios to 50, a near-optimal solution is achieved. However, SA does not provide the theoretical bound on optimality. There is no guarantee that the solution can be improved monotonically as the number of scenarios is reduced. Thus, a trial-and-error scheme should be used to determine the number of scenarios. Moreover, the number of replicates (200 in our study) for SA is purely determined heuristically without any justification, which cannot be generalized.

The calculated blending recipes for 3 types of gasoline are shown in Fig. 14. Clearly, type II is the most profitable gasoline, and thereby it reaches the maximum production 50. Type III is the least profitable gasoline due to its high octane rating, and thus its production is nearly

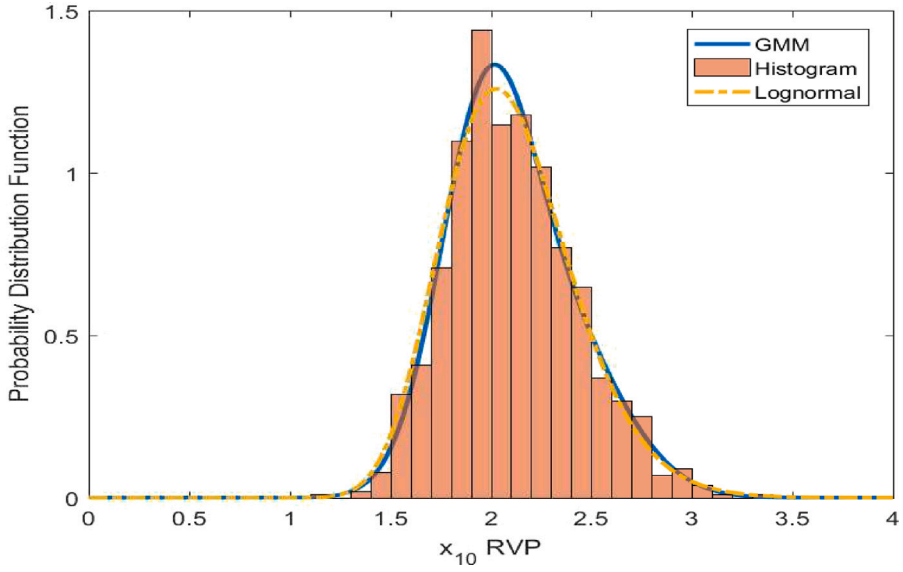

Fig. 12. x_{10} RVP lognormal PDF, GMM approximation, and histogram (1000 data points).

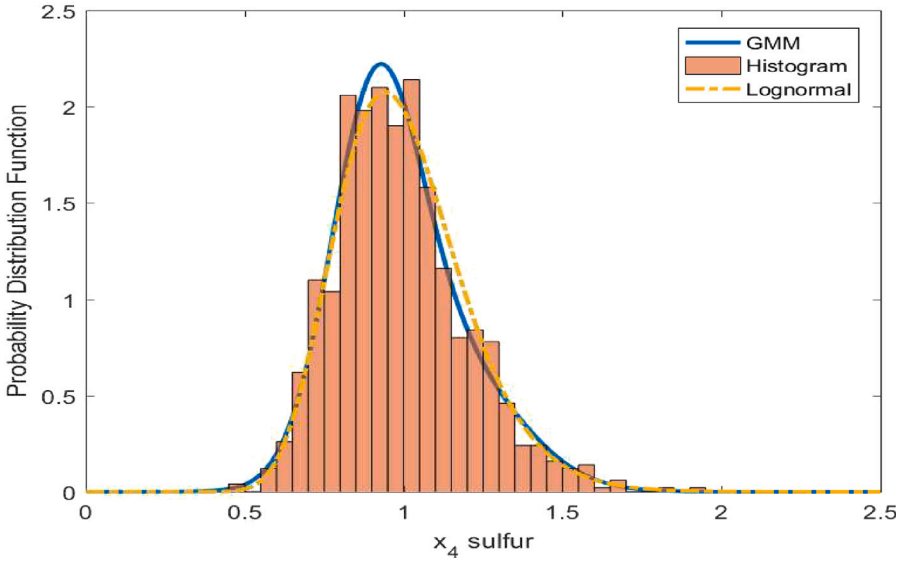

Fig. 13. x_4 sulfur lognormal PDF, GMM approximation, and histogram (1000 data points).

Table 9
Results of case 2 with $\epsilon' = \epsilon = 5\%$.

	With constraint relaxation	Without constraint relaxation
Time with flow RLT (s)	1514.0	2427.5
Time without flow RLT (s)	2108.1	3447.9
Objective value (Profit)	378.00	377.26
On-spec rate (I, II, III)	95.48%, 95.22%, 96.11%	95.80%, 95.54%, 96.15%

Table 10
Final results of case 2 by increasing ϵ'.

	With constraint relaxation	Without constraint relaxation	SA 702	SA 50
Solution time (s)	6668.5	110 49.7	101.9	69.7
Objective value (Profit)	378.49	378.44	354.6	377.86
On-spec rate (I)	95.36%	95.42%	98.34%	95.56%
On-spec rate (II)	95.05%	95.12%	98.75%	95.84%
On-spec rate (III)	96.32%	97.75%	99.29%	95.66%
ϵ'	5.50%	6.25%	N/A	N/A

zero. In the blending recipe, x_3 is selected because of its low RVP and price. x_4 is selected because of its low sulfur content and density. x_5 has low RVP, low sulfur content, and high octane, but its sale price is high. x_9 has low RVP, but its sulfur is high, and thus cannot be heavily used. Finally, x_{10} is chosen to lower the density and keep the octane rating high.

5. Conclusion

This paper presents a chance-constrained program and its optimization method to plan the blending process under general uncertainties. The distribution of uncertain parameter is estimated by Gaussian mixture models using sampled data. Then, we show that a probabilistic constraint can be converted into a set of conic constraints for each combination of Gaussian components. After decomposing joint constraints based on Boole's inequality and building cutting planes of the inverse cumulative distribution function, the resulting conservative approximation formula is bi-convex. The McCormick relaxation, branch and bound, optimality-based bound tightening, and

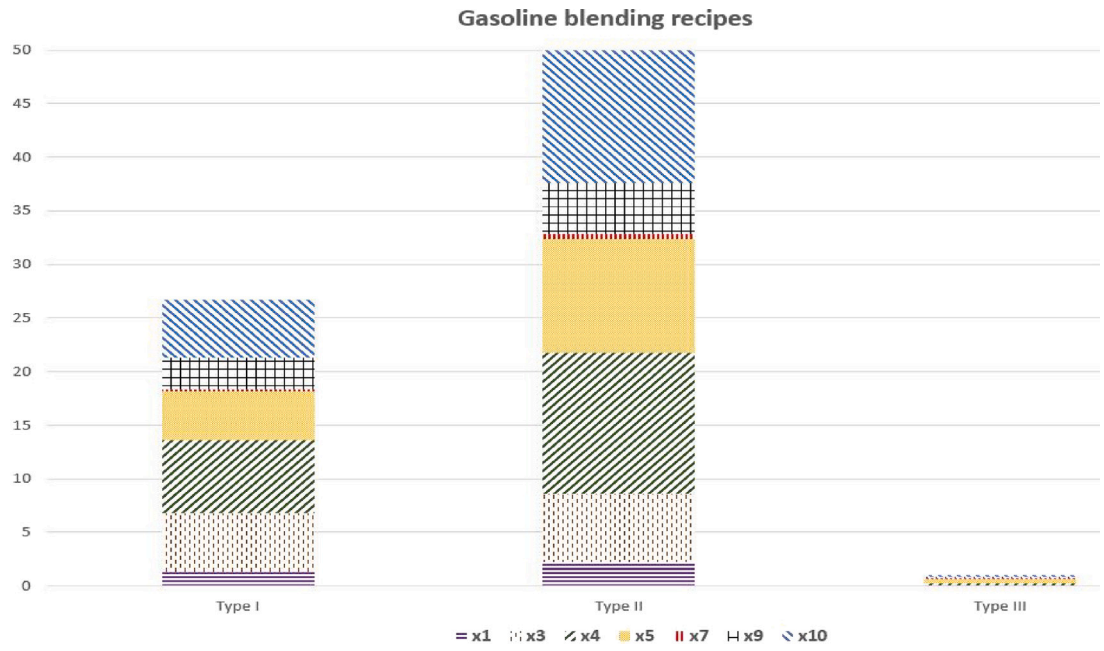

Fig. 14. Recipes of three types of gasoline.

reformulate-linearization techniques are employed to continuously reduce the gap between upper and lower bounds until a global optimum is achieved. The risk level is adjusted through bi-section search and posterior evaluation to further reduce the conservativeness of approximation. Finally, two case studies demonstrate the moderate solution time and optimality of the proposed method, which achieves better objective value than SA even without posterior evaluation.

CRediT authorship contribution statement

Yu Yang: Conceptualization, Methodology, Software, Data curation, Validation, Investigation, Writing – original draft, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The author is grateful to helpful discussions with Dr. Phebe Vayanos and financial support from National Science Foundation (NSF), award number: 2151497.

References

Alamo, T., Tempo, R., Luque, A., Ramirez, D.R., 2015. Randomized methods for design of uncertain systems: Sample complexity and sequential algorithms. Automatica 5, 160–172.
Ben-Tal, A., Nemirovski, A., 2002. Robust optimization-methodology and applications. Math. Program. 92, 453–480.
Calafiore, C.C., Campi, M.C., 2006. The scenario approach to robust control design. IEEE Trans. Automat. Control 51, 742–753.
Calfa, A., Grossmann, I.E., Agarwal, A., Bury, S.J., Wassick, J.M., 2015. Data-driven individual and joint chance-constrained optimization via kernel smoothing. Comput. Chem. Eng. 78, 51–69.
Campi, M.C., Garatti, S., 2008. The exact feasibility of randomized solutions of uncertain convex programs. SIAM J. Optim. 19, 1211–1230.
Campi, M.C., Garatti, S., 2011. A sampling-and-discarding approach to chance-constrained optimization: feasibility and optimality. J. Optim. Theory Appl. 148, 257–280.
Charnes, A., Cooper, W.W., 1959. Chance-constrained programming. Manage. Sci. 6, 73–79.
Cheng, J., Gicquel, C., Lisser, A., 2012. A second-order cone programming approximation to joint chance-constrained linear programs. Lecture Notes in Comput. Sci. 7422, 71–80.
Esfahani, P.M., Sutter, T., Lygeros, J., 2015. Performance bounds for the scenario approach and an extension to a class of non-convex programs. IEEE Trans. Automat. Control 60, 46–58.
Fathabad, A.M., Cheng, J., Pan, K., Yang, B., 2023. Asymptotically tight conic approximations for chance-constrained AC optimal power flow. European J. Oper. Res. 305, 738–753.
Grossmann, I.E., Apap, R.M., Calfa, B.A., García-Herreros, P., Zhang, Q., 2016. Recent advances in mathematical programming techniques for the optimization of process systems under uncertainty. Comput. Chem. Eng. 91, 3–14.
Hong, L.J., Yang, Y., Zhang, L., 2011. Sequential convex approximations to joint chance constrained programs: A monte carlo approach. Oper. Res. 59, 617–630.
Hu, Z., Sun, W., Zhu, S., 2022. Chance constrained programs with Gaussian mixture models. IISE Trans. 54, 1117–1130.
Jiang, R., Guan, Y., 2016. Data-driven chance constrained stochastic program. Math. Program. 158, 291–327.
Kannan, R., Luedtke, J.R., 2021. A stochastic approximation method for approximating the efficient frontier of chance-constrained nonlinear programs. Math. Program. Comput. 13, 705–751.
Li, P., Arellano-Garcia, H., Wozny, G., 2008. Chance constrained programming approach to process optimization under uncertainty. Comput. Chem. Eng. 32, 25–45.
Luedtke, J., 2014. A branch-and-cut decomposition algorithm for solving chance-constrained mathematical programs with finite support. Math. Program. 146, 219–244.
Luedtke, J., Ahmed, S., 2008. A sample approximation approach for optimization with probabilistic constraints. SIAM J. Optim. 19, 674-69.
McCormick, G.P., 1976. Computation of global solutions to factorable nonconvex programs: Part I convex underestimating problems. Math. Program. 10, 147–175.
Nemirovski, A., Shapiro, A., 2006. Convex approximation of chance constrained programs. SIAM J. Optim. 17, 969–996.
Ning, C., You, F., 2018. Data-driven stochastic robust optimization: General computational framework and algorithm leveraging machine learning for optimization under uncertainty in the big data era. Comput. Chem. Eng. 111, 115–133.
Peña-Ordieres, A., Luedtke, J., Wächter, A., 2020. Solving chance-constrained problems via a smooth sample-based nonlinear approximation. SIAM J. Optim. 30, 2221–2250.
Peng, S., Maggioni, F., Lisser, A., 2022. Bounds for probabilistic programming with application to a blend planning problem. European J. Oper. Res. 297, 964–976.
Prékoba, A., 1995. Stochastic Programming. Kluwer Academic Publishers, Netherlands.

Sahinidis, N.V., 2004. Optimization under uncertainty: state-of-the-art and opportunities. Comput. Chem. Eng. 28, 971–983.
Schrage, L.E., 2006. Optimization Modeling with LINGO (Six Edition). LINGO Systems Inc, Chicago, Illinois.
Shan, F., Zhang, L., Xiao, X., 2014. A smoothing function approach to joint chance constrained programs. J. Optim. Theory Appl. 163, 181–199.
Sun, W., Huang, G., Lv, Y., Li, G., 2013. Inexact joint-probabilistic chance-constrained programming with left-hand-side randomness: An application to solid waste management. European J. Oper. Res. 228, 217–225.
Tovar-Facio, J., Cao, Y., Ponce-Ortega, J., Zavala, V.M., 2018. Scalable solution strategies for chance-constrained nonlinear programs. Ind. Eng. Chem. Res. 57 (2018), 7987–7998.
Yang, Y., Phebe, V., Barton, P., 2017. Chance-constrained optimization for refinery blend planning under uncertainty. Ind. Eng. Chem. Res. 56, 12139–12150.
Yang, Y., dela Rosa, L., Chow, T., 2020. Non-convex chance-constrained optimization for blending recipe design under uncertainties. Comput. Chem. Eng. 139, 106868.