Computers and Chemical Engineering 184 (2024) 108632

Contents lists available at ScienceDirect

Computers and Chemical Engineering

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

Two-stage chance-constrained programming based on Gaussian mixture model and piecewise linear decision rule for refinery optimization

Yu Yang

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

ARTICLE INFO

Keywords:
Stochastic programming
Chance-constrained optimization
Gaussian mixture model
Piecewise linear decision rule

ABSTRACT

The two-stage chance-constrained program (CCP) is studied for a refinery optimization problem. In stage-I, the refinery decision-makers determine the type and quantity of crude oil procurement under operational uncertainties to maximize the expected profit under all possibilities. In stage-II, process unit flowrates are adjusted based on the realized uncertainties and available crude oil, while introducing probabilistic constraints to manage the off-spec risk. To solve such a two-stage optimization problem, we propose a novel approach using Gaussian mixture model (GMM) to characterize uncertainties, and piecewise linear decision rule for stage-II operations. Comparing to the conventional scenario-based mixed-integer linear program (MILP), our new approach offers three advantages. First, it leverages a well-developed global optimization scheme for joint CCP to avoid scenario generation and potential bias. Second, the data-driven GMM enables CCP to handle uncertainties with general distributions. Third, the stage-II variables are parameterized via Gaussian component induced piecewise linear decision rule to strike an excellent trade-off between optimality and computational time. A simplified refinery plant, consisting of distillation, cracker, reformer, isomerization, and desulfurization units, is used as a test bed to demonstrate the superiority of the proposed optimization method in solution time, probabilistic feasibility, and optimality over the large-scale scenario-based MILP.

1. Introduction

Attaining a globally optimal solution for the multi-stage process scheduling and operations in the presence of parametric uncertainties may enhance the profitability and safety of a refinery. Explicitly accounting for model uncertainty in such optimization problems is of importance to the capital-intensive industry and many successful applications can be found in some comprehensive reviews (Grossmann et al., 2016; Li and Ierapetritou, 2008; Sahinidis, 2004). One of the popular methodologies for multi-stage decision-making is the stochastic programming (SP). In a typical two-stage case, the stage-I decision is made before uncertainty realization to optimize the expected objective value. Once the uncertainty is revealed, then a stage-II optimization can be performed for a specific scenario. The SP framework integrates the data-driven stochastic modeling and optimization techniques, which could address a wide range of engineering problems (Chen et al., 2008). The chance-constrained program (CCP) stems from an alternative perspective on the optimization under uncertainty. CCP leverages uncertainty distributions and allows constraints to be violated with a small risk level, denoted as ϵ. This approach offers greater generality and lower conservatism compared to robust optimization (Grossmann et al., 2016; Li et al., 2008).

Nevertheless, both SP and CCP pose significant computational challenges. Even in the case of two-stage SP, it typically takes a large number of scenarios into account and the resulting large-scale formula is hard to solve directly. In most instances, except for special cases like an individual linear chance constraint with normal distribution or right-hand-side uncertainty with log-concave distribution (Prékoba, 1995), general CCP problems are nonconvex. If a program involves multiple chance constraints, the resulting joint CCP is even more difficult to tackle. We first review several widely-adopted methods for addressing SP and CCP, respectively. Subsequently, a novel approach for solving two-stage chance-constrained program will be introduced.

Stochastic Programming Methods: SP relies on a scenario tree to capture the true distribution of uncertain parameters at each decision stage. In general, incorporating more scenarios enhances its approximation accuracy, but the problem complexity and solution time will also increase. To resolve this issue, the Benders (Geoffrion, 1972; Benders, 1962) or Lagrangian (Mouret et al., 2011; Karuppiah and Grossmann, 2008) decomposition must be employed for convex problem to sequentially or parallelly solve each scenario. Further research also extended the Benders decomposition from convex to non-convex cases (Li et al., 2012, 2011). Computational studies have demonstrated

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

https://doi.org/10.1016/j.compchemeng.2024.108632
Received 29 September 2023; Received in revised form 1 February 2024; Accepted 15 February 2024
Available online 21 February 2024

Nomenclature

CCP	Chance-constrained Program
CDF	Cumulative Distribution Function
CDU	Crude Distillation Unit
CGO	Coker Gas Oil
CN	Cracked Gasoline
GMM	Gaussian Mixture Model
GO	Gas Oil
HF	Heavy Fuel Oil
HN	Heavy Naphtha
JF	Jet Fuel
KE	Kerosene
LG	Liquefied Petroleum Gas
LN	Light Naphtha
MILP	Mixed-integer Linear Program
PDF	Probability Density Function
RG	Refinery Gas
RON	Research Octane Number
SOCP	Second-order cone program
SP	Stochastic Programming
VGO	Vacuum distillate
VR	Vacuum Residual

that such decomposition approaches offer improved scalability compared to directly solving all scenarios simultaneously (Yang and Barton, 2016). Another methodology for shortening the computational time of SP is the scenario reduction. It selects support scenarios from a large set (Kammammettu and Li, 2023; Li and Floudas, 2014) based on specific metric to minimize the error between the original and reduced sets. Alternatively, clustering methods can generate or aggregate existing scenarios to enhance their representativeness (Bounitsis et al., 2022; Xu et al., 2012).

The decision rule can be applied to solve multi-stage optimization efficiently. It assumes the recourse variable (reactive actions) as a function of uncertain parameters. The SP can be reformulated as a robust counterpart through linear decision rule (Ben-Tal et al., 2004). A scheduling model for power-intensive processes was developed and solved by the adjustable robust optimization approach with linear decision rule embedded (Zhang et al., 2016). Multi-parametric programming has recently been applied to derive a globally optimal linear decision rule for the multilevel decision process (Avraamidou and Pistikopoulos, 2020). While the linear structure may sacrifice some generality, it allows for efficient calculations under the affine recourse hypothesis (Calafiore, 2008). To obtain more flexibility in stage-II, the lifting method was proposed to unify different types of decision rules (Georghiou et al., 2015). Furthermore, polynomial decision rule for multi-stage SP can be efficiently determined by solving two tractable semi-definite programs (Bampou and Kuhn, 2011). Their computational study showed that the cubic decision rule significantly reduced the optimality gap compared with the linear counterpart. The uncertainty set can be partitioned into several parts for the development of adaptive piecewise linear decision rule to balance optimality and complexity (Nasab and Li, 2021; Bertsimas and Dunning, 2016). This concept has also been found in several other literature (Rahal et al., 2022; Hanasusanto et al., 2015).

Chance-Constrained Programming Methods: The CCP also can be solved through scenario-based approaches. A straightforward manner is to require that a fraction $1-\epsilon$ of the total scenarios are feasible within the optimization framework (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, this approach typically introduces binary variables for each scenario, potentially hindering its scalability. A decomposition approach can be applied to speedup the solving process of scenario-based formula (Liu et al., 2016). To avoid additional binary variables, a sample complexity bound was studied for convex problem in several works (Alamo et al., 2015; Campi and Garatti, 2011; Calafiore and Campi, 2006; Campi and Garatti, 2008), which links the optimality of full-scenario problem with the probabilistic feasibility. Recent efforts have extended this methodology to nonconvex problems via the posterior evaluation (Esfahani et al., 2015). Since the expectation on indicator function can be used to represent the probability of constraint satisfaction, many works focused on approximating indicator function through sample average schemes. The conditional value-at-risk (CVaR) approximation and the Bernstein approximation (Nemirovski and Shapiro, 2006) have gained prominence. The sigmoidal approximation offers another choice to replace the indicator function in chance constraints (Tovar-Facio et al., 2018). One of the challenges in those methods is the inaccurate gradient information (Peña-Ordieres et al., 2020). A recent work adopted the projected stochastic subgradient algorithm to solve a convergent sequence of smooth approximation for CCP and showed excellent results (Kannan and Luedtke, 2021).

Another promising methodology for CCP leverages historical data to establish the probability density function (PDF), cumulative distribution function (CDF), or quantile function of uncertain parameters. These functions are utilized in chance constraints to facilitate deterministic reformulations. The kernel smoothing method can estimate the PDF, CDF, or quantile function to solve chance constraints with right-hand side uncertainty (Calfa et al., 2015). A related work (Jiang and Guan, 2016) described the confidence set of an estimated PDF via ψ-divergence with a perturbed risk level to guarantee the robustness of the reformulated CCP when dealing with uncertain distributions. In author's recent work, the Gaussian Mixture Models (GMM) were employed to approximate general uncertainty distributions in the single-stage CCP and the resulting bi-convex formulation can be solved to global optimality through second-order cone relaxation (Yang, 2023). That work significantly broadened the application scope of non-sampling-based CCP methodologies.

This paper aims to solve two-stage CCP (TCCP) that integrates chance constraints with two-stage SP to enhance the flexibility of refinery optimization. We employ GMM to model uncertainty distributions due to the following merits:

- GMM is a parametric approach to approximate complex distributions with arbitrary shapes by combining multiple Gaussian components.
- As a clustering method, GMM not only approximates the true distribution of uncertainty, but also enables cluster-dependent linear decision rule for two-stage optimization.
- GMM can be easily built from data through the well-developed expectation–maximization (EM) algorithm.

In view of these advantages, GMM derived from uncertainty data is naturally suitable for solving TCCP efficiently. Building upon on our prior work (Yang, 2023), where we have established a global optimization algorithm for single-stage CCP with GMM embedded, our current research demonstrates that GMM and piecewise linear decision rule integrated formula can be optimized through the adaptive outer approximation, branch-and-bound, and bound tightening techniques. The proposed approach outperforms sampling-based methods in terms of solution time and convergence. Nevertheless, we acknowledge that the proposed approach still can be improved to allow for more flexible decision rule and characterize the impact of GMM estimation error on solution optimality.

The rest of this paper is organized as follows. The refinery model and TCCP reformulation under GMM and piecewise linear decision rule are described in Section 2. The global optimization framework is presented in Section 3. The comparison between GMM with piecewise

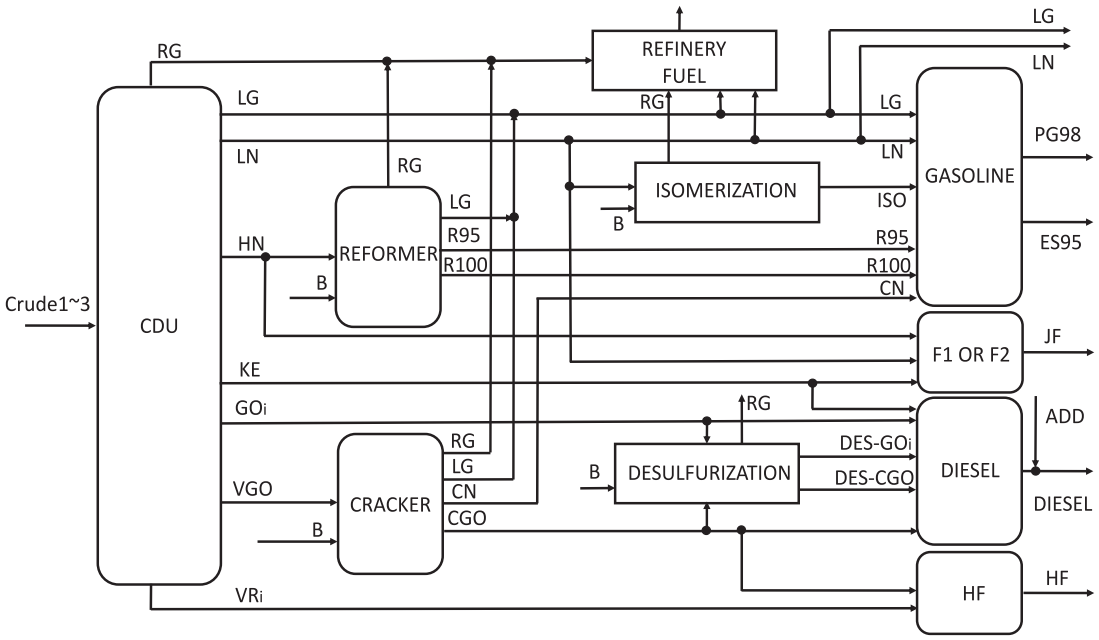

Fig. 1. The refinery flowchart (Favennec, 2001).

linear decision rule and sample average approximation on refinery optimization are conducted in Section 4 to highlight the superiority of the proposed scheme. Finally, conclusions are drawn in Section 5.

Notation. Throughout this paper, vectors and matrices are denoted by boldface letters. 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 $\boldsymbol{\theta} \sim \mathcal{N}(\boldsymbol{\mu}, \boldsymbol{\Sigma})$ to express that $\boldsymbol{\theta}$ is normally distributed with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. Similarly, for $\boldsymbol{w} \in \mathbb{R}_+^S$ such that $\sum_{s=1}^S w_s = 1$, $\boldsymbol{\mu}_s \in \mathbb{R}^n$ and $\boldsymbol{\Sigma}_s \in \mathbb{S}_+^n$, $\forall s \in \{1, \ldots, S\}$, we write $\boldsymbol{\theta} \sim \sum_{s=1}^S w_s \mathcal{N}(\boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s)$ to express that $\boldsymbol{\theta}$ follows the S-component Gaussian mixture distribution.

2. Methodology

2.1. Refinery model

A simplified refinery model is applied for this study (Favennec, 2001). The plant monthly planning and operations can be conceptualized as a two-stage decision process (Yang and Barton, 2016). In stage-I, the optimal crude combination is determined to maximize the expected profit. In stage-II, the selected crude oils undergo a series of processing steps outlined in Fig. 1. The crude distillation unit (CDU) separates the inflow into different fractions based on their boiling points, including refinery gas (RG), liquefied petroleum gas (LG), light naphtha (LN), heavy naphtha (HN), kerosene (KE), gas oil (GO), vacuum distillate (VGO), and vacuum residual (VR). These intermediate streams are subsequently directed to the reformer, cracker, isomeration, and desulferization units for further processing to enhance their values or reduce undesirable components. The processed streams are ultimately blended in the final product tanks to yield gasoline PG98, ES95, diesel, heavy fuel oil (HF) and jet fuel (JF). It is worth noting that, in an effort to reduce sulfur content in diesel, a proportion of low-sulfur (15 ppm) KE is mixed into the final product, denoted as ADD in the flowchart.

In stage-I, the procurement of three crude oil is decided according to their properties, price, refinery capacity, and product requirement. The crude oil is usually traded in integer lots that can be represented in base 2 by using binary variables $\boldsymbol{x}_j = [x_{j,1}, x_{j,2}, \ldots]^{\mathrm{T}}$, where j is the type index of crude oil. Here we assume the lot size to be 5000 barrels. Eqs. (1) and (2) are enforced for crude oil procurement. If type j crude oil is selected, namely $e_j = 1$, the lower bound on that order is specified as 10 lots. The upper bound of an order is specified as 360 lots due to the stock availability. As a result, the dimension of $\boldsymbol{x}_j$ should be 9 and each dimension is denoted by index t. If a type of crude oil is not selected, namely $e_j = 0$, (2) becomes active and $x_{j,t}, \forall t = 1, 2, \ldots, 9$ are zero.

$$10e_j \leqslant \sum_{t=1}^{9} x_{j,t} \times 2^{t-1} \leqslant 360, \ j = 1, 2, 3, \tag{1}$$

$$x_{j,t} \leqslant e_j, \ x_{j,t} \in \{0, 1\}, \ e_j \in \{0, 1\}, \ j = 1, 2, 3, \ t = 1, 2, \ldots, 9. \tag{2}$$

Constraints listed in Eqs. (3)–(8) are required to satisfy in stage-II, including the sulfur contents of PG98, ES95, and diesel, research octane number (RON) of PG98 and ES95, HF viscosity, PG98 and diesel demands. However, the operational uncertainties, denoted as $\boldsymbol{\theta} = [\theta_{\mathrm{Cr}}, \theta_{\mathrm{RON}}, \theta_{\mathrm{S}}]$, including CN yield increment, ISO RON reduction, and sulfur residual increment, may render the problem infeasible under specific scenarios. Therefore, those quality or demand constraints will be relaxed such that a joint violation chance ϵ is allowed. In addition, the catalytic cracked gasoline (CN) and coker gas oil (CGO) mass balance are also impacted by the cracker yield uncertainty, shown in (10) and (11) respectively. Other capacity inequalities and mass balance equations are the same with Yang (2019).

Sulfur in PG98:

$$\begin{aligned} &y_{\mathrm{PG98,ISO}}\mathrm{Sul}_{\mathrm{ISO}} + y_{\mathrm{PG98,LN}}\mathrm{Sul}_{\mathrm{LN}} + y_{\mathrm{PG98,R95}}\mathrm{Sul}_{\mathrm{R95}} \\ &\quad + y_{\mathrm{PG98,LG}}\mathrm{Sul}_{\mathrm{LG}} + y_{\mathrm{PG98,R100}}\mathrm{Sul}_{\mathrm{R100}} \\ &+ y_{\mathrm{PG98,CN}}\mathrm{Sul}_{\mathrm{CN}} \leqslant 0.0010\%(y_{\mathrm{PG98,LG}} + y_{\mathrm{PG98,LN}} + y_{\mathrm{PG98,ISO}} \\ &\quad + y_{\mathrm{PG98,R95}} + y_{\mathrm{PG98,R100}} + y_{\mathrm{PG98,CN}}), \end{aligned} \tag{3}$$

where Sul represents the sulfur content in each flow.

Sulfur in ES95:

$$\begin{aligned} &y_{\mathrm{ES95,ISO}}\mathrm{Sul}_{\mathrm{ISO}} + y_{\mathrm{ES95,LN}}\mathrm{Sul}_{\mathrm{LN}} + y_{\mathrm{ES95,R95}}\mathrm{Sul}_{\mathrm{R95}} \\ &\quad + y_{\mathrm{ES95,LG}}\mathrm{Sul}_{\mathrm{LG}} + y_{\mathrm{ES95,R100}}\mathrm{Sul}_{\mathrm{R100}} \\ &+ y_{\mathrm{ES95,CN}}\mathrm{Sul}_{\mathrm{CN}} \leqslant 0.0020\%(y_{\mathrm{ES95,ISO}} + y_{\mathrm{ES95,LN}} + y_{\mathrm{ES95,LG}} \\ &\quad + y_{\mathrm{ES95,R95}} + y_{\mathrm{ES95,R100}} + y_{\mathrm{ES95,CN}}), \end{aligned} \tag{4}$$

Sulfur in Diesel:

$$\sum_{j=1}^{3} y_{\mathrm{DIESEL,GO}_j}\mathrm{Sul}_{\mathrm{GO}_j} + \sum_{j=1}^{3} y_{\mathrm{DIESEL,DESGO}_j}\mathrm{Sul}_{\mathrm{GO}_j}(0.01+\theta_S) + y_{\mathrm{DIESEL,DESCGO}}\mathrm{Sul}_{\mathrm{CGO}}(0.01+\theta_S) + y_{\mathrm{DIESEL,CGO}}\mathrm{Sul}_{\mathrm{CGO}} + y_{\mathrm{DIESEL,Add}}\mathrm{Sul}_{\mathrm{Add}} + y_{\mathrm{DIESEL,KE}}\mathrm{Sul}_{\mathrm{KE}} \leqslant 0.0018\%\left(y_{\mathrm{DIESEL,KE}} \sum_{j=1}^{3}\left(y_{\mathrm{DIESEL,GO}_j} + y_{\mathrm{DIESEL,DESGO}_j}\right) + y_{\mathrm{DIESEL,DESCGO}} + y_{\mathrm{DIESEL,CGO}} + y_{\mathrm{DIESEL,Add}}\right), \tag{5}$$

RON of PG98:

$$\frac{y_{\mathrm{PG98,ISO}}(\mathrm{RON}_{\mathrm{ISO}}-\theta_{\mathrm{RON}})}{\rho_{\mathrm{ISO}}} + \frac{y_{\mathrm{PG98,LN}}\mathrm{RON}_{\mathrm{LN}}}{\rho_{\mathrm{LN}}} + \frac{y_{\mathrm{PG98,R95}}\mathrm{RON}_{\mathrm{R95}}}{\rho_{\mathrm{R95}}} + \frac{m_{\mathrm{PG98,R100}}\mathrm{RON}_{\mathrm{R100}}}{\rho_{\mathrm{R100}}} + \frac{y_{\mathrm{PG98,CN}}\mathrm{RON}_{\mathrm{CN}}}{\rho_{\mathrm{CN}}} + \frac{y_{\mathrm{PG98,LG}}\mathrm{RON}_{\mathrm{LG}}}{\rho_{\mathrm{LG}}} \geqslant 98V_{\mathrm{PG98}}, \tag{6}$$

where V is the volume of products and ρ is the density.

RON of ES95:

$$\frac{y_{\mathrm{ES95,ISO}}(\mathrm{RON}_{\mathrm{ISO}}-\theta_{\mathrm{RON}})}{\rho_{\mathrm{ISO}}} + \frac{y_{\mathrm{ES95,LN}}\mathrm{RON}_{\mathrm{LN}}}{\rho_{\mathrm{LN}}} + \frac{y_{\mathrm{ES95,R95}}\mathrm{RON}_{\mathrm{R95}}}{\rho_{\mathrm{R95}}} + \frac{y_{\mathrm{ES95,R100}}\mathrm{RON}_{\mathrm{R100}}}{\rho_{\mathrm{R100}}} + \frac{y_{\mathrm{ES95,CN}}\mathrm{RON}_{\mathrm{CN}}}{\rho_{\mathrm{CN}}} + \frac{y_{\mathrm{ES95,LG}}\mathrm{RON}_{\mathrm{LG}}}{\rho_{\mathrm{LG}}} \geqslant 95V_{\mathrm{ES95}}. \tag{7}$$

Viscosity of HF:

$$\frac{\sum_{j=1}^{3} y_{\mathrm{HF,VR}_j}\mathrm{Vis}_{\mathrm{VR}_j}}{\rho_{\mathrm{VR}_j}} + \frac{y_{\mathrm{HF,CGO}}\mathrm{Vis}_{\mathrm{CGO}}}{\rho_{\mathrm{CGO}}} \leqslant 35V_{\mathrm{HF}},$$

where Vis denotes the viscosity of streams.

PG98 demand:

$$y_{\mathrm{PG98,LG}} + y_{\mathrm{PG98,ISO}} + y_{\mathrm{PG98,LN}} + y_{\mathrm{PG98,R100}} + y_{\mathrm{PG98,R95}} + y_{\mathrm{PG98,CN}} \geqslant 15. \tag{8}$$

DIESEL demand:

$$y_{\mathrm{DIESEL,Add}} + y_{\mathrm{DIESEL,KE}} + \sum_{j=1}^{3}\left(y_{\mathrm{DIESEL,GO}_j} + y_{\mathrm{DIESEL,DESGO}_j}\right) + y_{\mathrm{DIESEL,CGO}} + y_{\mathrm{DIESEL,DESCGO}} \geqslant 100, \tag{9}$$

CN balance:

$$y_{\text{Cr-M,VGO}}(\mathrm{P}_{\text{Cr-M,CN}}+\theta_{\mathrm{Cr}}) + y_{\text{Cr-A,VGO}}(\mathrm{P}_{\text{Cr-A,CN}}+\theta_{\mathrm{Cr}}) = y_{\mathrm{PG98,CN}} + y_{\mathrm{ES95,CN}} \tag{10}$$

where P is the production percentage and cracker may work on two modes, Mogas: Cr-M and AGO: Cr-A.

CGO balance:

$$y_{\text{Cr-M,VGO}}(\mathrm{P}_{\text{Cr-M,CGO}}-\theta_{\mathrm{Cr}}) + y_{\text{Cr-A,VGO}}(\mathrm{P}_{\text{Cr-A,CGO}}-\theta_{\mathrm{Cr}}) = y_{\mathrm{DIESEL,CGO}} + y_{\mathrm{Des,CGO}} + y_{\mathrm{HF,CGO}}. \tag{11}$$

2.2. Two-stage chance-constrained program

A generic formula of two-stage chance-constrained program is developed in ($\mathcal{TCCP}$) to minimize the cost of refining process shown in Fig. 1:

$$\min_{\boldsymbol{x},\boldsymbol{e}} \ \boldsymbol{p}_0^{\mathrm{T}}\boldsymbol{x} + \mathbb{E}_{\boldsymbol{\theta}}(\mathcal{F}(\boldsymbol{x},\boldsymbol{\theta})) \qquad (\mathcal{TCCP})$$

s.t. Eqs. (1), (2),

$$\boldsymbol{x}\in\{0,1\}, \boldsymbol{e}\in\{0,1\},$$

where $\mathbb{E}$ is the expectation operator and stage-II formula is

$$\mathcal{F}(\boldsymbol{x},\boldsymbol{\theta}) = \min_{\boldsymbol{y}_1,\boldsymbol{y}_2} \boldsymbol{p}_1^{\mathrm{T}}\boldsymbol{y}_1 + \boldsymbol{p}_2^{\mathrm{T}}\boldsymbol{y}_2$$

$$\text{s.t. } \mathbb{P}\left\{\boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x} + \sum_{k=1}^{K}\boldsymbol{b}_{k,i}^{\mathrm{T}}\theta_k\boldsymbol{y}_1 + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{y}_2 \leqslant \boldsymbol{g}_i,\ \forall i=1,2,\ldots,I\right\} \geqslant 1-\epsilon, \tag{12}$$

$$\boldsymbol{A}'\boldsymbol{x} + \sum_{k=1}^{K}\boldsymbol{B}_k'\theta_k\boldsymbol{y}_1 + \boldsymbol{C}'\boldsymbol{y}_2 = 0, \tag{13}$$

$$\boldsymbol{0}\leqslant\boldsymbol{y}_1, \boldsymbol{0}\leqslant\boldsymbol{y}_2,$$

where K is the dimension of uncertain parameters. Constant vectors and matrices include $\boldsymbol{p}_0$, $\boldsymbol{p}_1$, $\boldsymbol{p}_2$, $\boldsymbol{a}_i$, $\boldsymbol{A}'$, $\boldsymbol{b}_{k,i}$, $\boldsymbol{B}_k'$, $\boldsymbol{c}_i$, $\boldsymbol{C}'$ and $\boldsymbol{g}_i$. $\boldsymbol{y}_1\in\mathcal{R}^{N_1}$ and $\boldsymbol{y}_2\in\mathcal{R}^{N_2}$ are two mutually exclusive vectors representing stage-II recourse variables, such as flowrate of each stream. Their coefficients $\boldsymbol{p}_1$ and $\boldsymbol{p}_2$ are negative to represent the revenue of productions. $\boldsymbol{x}$ is the crude oil procurement, and thus its coefficient $\boldsymbol{p}_0$ is positive to represent the oil price. In this formulation, we assume that each uncertainty parameter can be modeled by the independent GMM with S components: $\theta_k \sim \sum_{s=1}^{S} w_{k,s}\mathcal{N}\left(\mu_{k,s},\sigma_{k,s}^2\right)$, $\forall k\in\{1,2,\ldots,K\}$. Here S is a pre-determined number of Gaussian components in the GMM to control the approximation accuracy. We recommend fitting the uncertainty data from $S=2$ because a larger number of S may lead to more complexities in optimization. Striking a balance between accuracy and complexity is still an ongoing research. ϵ is a small risk level that can exclude extreme cases in the optimization. According to Eqs. (3)–(8), totally $I=8$ joint chance constraints should be considered in stage-II to allow the incomplete recourse. In addition, Eq. (13) represents the mass balance in the refinery model, such as (10) and (11), in which uncertain parameter θ_{Cr} multiples with variables $y_{\text{Cr-M,VGO}}$ and $y_{\text{Cr-A,VGO}}$ to determine the CN and CGO productions.

2.3. Piecewise linear decision rule

In this subsection, the vector $\boldsymbol{y}_2$ is specified as a function of uncertain parameters whereas $\boldsymbol{y}_1$ is kept as constant. Otherwise, quadratic terms on $\boldsymbol{\theta}$ will be introduced into Eqs. (12) and (13), rendering the optimization intractable. This restriction may limit the flexibility of the proposed approach applied in processes with many uncertain parameters.

Because GMM has represented the distribution of each uncertain parameter by S Gaussian components, the entire uncertainty space with K parameters can be partitioned into $L=S^K$ clusters $\Omega_1,\Omega_2,\ldots,\Omega_L$. A linear decision rule can be equipped with each cluster to parameterize $\boldsymbol{y}_2$:

$$\boldsymbol{y}_2 = \tilde{\boldsymbol{H}}\boldsymbol{\theta}+\boldsymbol{r} = \begin{cases} \boldsymbol{H}_1\boldsymbol{\theta}+\boldsymbol{r} & \text{if } \boldsymbol{\theta}\in\Omega_1, \\ \boldsymbol{H}_2\boldsymbol{\theta}+\boldsymbol{r} & \text{if } \boldsymbol{\theta}\in\Omega_2, \\ \vdots & \vdots, \\ \boldsymbol{H}_L\boldsymbol{\theta}+\boldsymbol{r} & \text{if } \boldsymbol{\theta}\in\Omega_L, \end{cases} \tag{14}$$

where $\boldsymbol{r}$ and $\boldsymbol{H}_l = [\boldsymbol{h}_{l,1}^{\mathrm{T}};\boldsymbol{h}_{l,2}^{\mathrm{T}};\ldots;\boldsymbol{h}_{l,N_2}^{\mathrm{T}}]\in\Re^{N_2\times K}$ are decision variables calculated in the optimization. Here $\boldsymbol{r}$ should be constant across all clusters. To show that, let us substitute (14) into (13):

$$\boldsymbol{A}'\boldsymbol{x} + \sum_{k=1}^{K}\boldsymbol{B}_k'\theta_k\boldsymbol{y}_1 + \boldsymbol{C}'(\tilde{\boldsymbol{H}}\boldsymbol{\theta}+\boldsymbol{r}) = 0.$$

To ensure this equality under any scenarios, there should be:

$$\boldsymbol{A}'\boldsymbol{x} + \boldsymbol{C}'\boldsymbol{r} = 0,$$

$$\sum_{k=1}^{K} \boldsymbol{B}'_k \theta_k \boldsymbol{y}_1 + \boldsymbol{C}' \tilde{\boldsymbol{H}} \boldsymbol{\theta} = 0.$$

Here we can see that $\boldsymbol{r}$ has no flexibility to vary across different pieces.

There is a mapping function from cluster to the dominant Gaussian component index set: $\mathcal{M} : \Omega_l \mapsto \{d_{l,1}, d_{l,2}, \dots, d_{l,K}\}$. Given a sampled uncertainty vector $\boldsymbol{\theta}^*$, the dominant component of each uncertain parameter $\{\mathcal{D}_1(\theta_1^*), \mathcal{D}_2(\theta_2^*), \dots, \mathcal{D}_K(\theta_K^*)\}$ will determine the assigned cluster through the inverse mapping $\mathcal{M}^{-1}$. Here $\mathcal{D}_k(\theta_k^*) \in \{1, 2, \dots, S\}, \forall k \in \{1, 2, \dots, K\}$ is defined as:

$$D_k(\theta_k^*) = \arg\max_{s'} \frac{w_{k,s'} \phi(\theta_k^*; \mu_{s'}, \Sigma_{k,s'})}{\sum_{s=1}^{S} w_{k,s} \phi(\theta_k^*; \mu_{k,s}, \Sigma_{k,s})}. \tag{15}$$

Because all uncertainty variables are independent, the probability of a sampled uncertainty vector belonging to cluster Ω_l is given by $\delta_l = \prod_{k=1}^{K} w_{d_{l,k}}$. Furthermore, we define the mean vector and covariance matrix corresponding to the cluster Ω_l as $\hat{\boldsymbol{\mu}}_l = [\mu_{d_{l,1}}, \mu_{d_{l,2}}, \dots, \mu_{d_{l,K}}]^{\mathrm{T}}$ and $\hat{\boldsymbol{\Sigma}}_l = \mathrm{diag}(\sigma^2_{d_{l,1}}, \dots, \sigma^2_{d_{l,K}})$, respectively.

Let us substitute (14) into ($\mathcal{TCCP}$). The objective function becomes:

$$\begin{aligned} &\boldsymbol{p}_0^{\mathrm{T}} \boldsymbol{x} + \mathbb{E}_{\boldsymbol{\theta}}(\boldsymbol{p}_1^{\mathrm{T}} \boldsymbol{y}_1 + \boldsymbol{p}_2^{\mathrm{T}} \boldsymbol{y}_2) = \boldsymbol{p}_0^{\mathrm{T}} \boldsymbol{x} + \boldsymbol{p}_1^{\mathrm{T}} \boldsymbol{y}_1 + \mathbb{E}_{\boldsymbol{\theta}}(\boldsymbol{p}_2^{\mathrm{T}} \boldsymbol{y}_2) \\ &= \boldsymbol{p}_0^{\mathrm{T}} \boldsymbol{x} + \boldsymbol{p}_1^{\mathrm{T}} \boldsymbol{y}_1 + \sum_{l=1}^{L} \boldsymbol{p}_2^{\mathrm{T}} \boldsymbol{r} + \sum_{l=1}^{L} \boldsymbol{p}_2^{\mathrm{T}} \boldsymbol{H}_l \hat{\boldsymbol{\mu}}_l \delta_l \end{aligned} \tag{16}$$

The resulting single stage optimization is:

$$\min_{\boldsymbol{x}, \boldsymbol{e}, \boldsymbol{y}_1, \boldsymbol{H}_l, \boldsymbol{r}_l} \ \boldsymbol{p}_0^{\mathrm{T}} \boldsymbol{x} + \boldsymbol{p}_1^{\mathrm{T}} \boldsymbol{y}_1 + \sum_{l=1}^{L} \boldsymbol{p}_2^{\mathrm{T}} \boldsymbol{r} + \sum_{l=1}^{L} \boldsymbol{p}_2^{\mathrm{T}} \boldsymbol{H}_l \hat{\boldsymbol{\mu}}_l \delta_l \tag{$\mathcal{CCP}$}$$

$$\text{s.t. } \mathbb{P}\{\boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x} + (\boldsymbol{y}_1^{\mathrm{T}} \boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}} \tilde{\boldsymbol{H}}) \boldsymbol{\theta} + \boldsymbol{c}_i^{\mathrm{T}} \boldsymbol{r} \leqslant g_i, \ \forall i = 1, 2, \dots, I, \} \geqslant 1 - \epsilon + \lambda N_2, \tag{17}$$

$$\boldsymbol{A}' \boldsymbol{x} + \sum_{k=1}^{K} \boldsymbol{B}'_k \theta_k \boldsymbol{y}_1 + \boldsymbol{C}'(\boldsymbol{H}_l \boldsymbol{\theta} + \boldsymbol{r}) = \boldsymbol{0}, \ \forall l = 1, 2, \dots, L, \tag{18}$$

$$\mathbb{P}\{0 \leqslant y_{2,j}\} \geqslant 1 - \lambda, \forall j = 1, 2, \dots, N_2, \tag{19}$$

$$\boldsymbol{x} \in \{0, 1\}, \boldsymbol{e} \in \{0, 1\}, \text{Eqs. (1), (2), } \boldsymbol{0} \leqslant \boldsymbol{y}_1,$$

where $\boldsymbol{B}_i^{\mathrm{T}} = [\boldsymbol{b}_{1,i}, \boldsymbol{b}_{2,i}, \dots, \boldsymbol{b}_{N,i}]$; λ is a small constant risk level for non-negative constraints such that $\lambda N_2 \ll \epsilon$. This configuration is justifiable in that the quality or demand constraint violation can be mitigated via some recovery operations whereas the negative flowrates are not physically plausible. Because $\boldsymbol{y}_2$ is linearly dependent on uncertainty $\boldsymbol{\theta}$, the stringent constraint $\boldsymbol{0} \leqslant \boldsymbol{y}_2$ is transformed into a series of high-probability constraints (19). In addition, the joint risk level of (17) and (19) is smaller than ϵ according to Boole's inequality, which ensuring the solution of ($\mathcal{CCP}$) to be feasible in ($\mathcal{TCCP}$).

2.4. Reformulation

Eqs. (17) and (19) are all chance-constrained with GMM uncertainty. We present Proposition 1 to derive their deterministic reformulation.

Proposition 1. *For GMM uncertainty $\boldsymbol{\theta} \sim \sum_{s=1}^{S} w_s \mathcal{N}(\boldsymbol{\mu}_s, \boldsymbol{\Sigma}_s)$, piecewise linear decision rule (14), and $\lambda \in (0, 1)$, if there exists a series of positive number $\gamma_l < \delta_l, \forall l = 1, 2, \dots, L$, satisfying*

$$\sum_{l=1}^{L=S^K} \gamma_l \geqslant 1 - \lambda, \tag{20}$$

then statement (b) below implies statement (a).

(a) $\mathbb{P}(0 \leqslant y_{2,j}) \geqslant 1 - \lambda$;

(b) $\boldsymbol{\Phi}^{-1}(\frac{\gamma_l}{\delta_l}) \sqrt{\boldsymbol{h}_{l,j}^{\mathrm{T}} \hat{\boldsymbol{\Sigma}}_l \boldsymbol{h}_{l,j}} \leqslant \hat{\boldsymbol{\mu}}_l^{\mathrm{T}} \boldsymbol{h}_{l,j} + r_j, \ \forall l \in \{1, 2, \dots, L\}$.

Proof. Because clusters $\Omega_l, \forall l \in \{1, 2, \dots, L\}$ form a finite partition of the entire uncertainty space, the law of total probability shows that:

$$\begin{aligned} \mathbb{P}(0 \leqslant y_{2,j}) &= \sum_{l=1}^{L} \mathbb{P}(\boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{h}_{l,j} + r_j \geqslant 0 |\ \boldsymbol{\theta} \in \Omega_l) \mathbb{P}(\boldsymbol{\theta} \in \Omega_l) \\ &= \sum_{l=1}^{L} \mathbb{P}(\boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{h}_{l,j} + r_j \geqslant 0 |\ \boldsymbol{\theta} \in \Omega_l) \delta_l \end{aligned} \tag{21}$$

The conditional probability of the GMM is Gaussian: $\mathbb{P}(\boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{h}_{l,j} + r_j | \ \boldsymbol{\theta} \in \Omega_l) \sim \mathcal{N}(\hat{\boldsymbol{\mu}}_l \boldsymbol{h}_{l,j} + r_j, \boldsymbol{h}_{l,j}^{\mathrm{T}} \hat{\boldsymbol{\Sigma}}_l \boldsymbol{h}_{l,j})$. Therefore, the classical results of Gaussian-distributed linear stochastic program (Prékoba, 1995) can be applied:

$$\begin{aligned} & \boldsymbol{\Phi}^{-1}(\tfrac{\gamma_l}{\delta_l}) \sqrt{\boldsymbol{h}_{l,j}^{\mathrm{T}} \hat{\boldsymbol{\Sigma}}_l \boldsymbol{h}_{l,j}} \leqslant \hat{\boldsymbol{\mu}}_l^{\mathrm{T}} \boldsymbol{h}_{l,j} + r_j, \ \forall l \in \{1, 2, \dots, L\} \\ \Leftrightarrow \quad & \delta_l \, \boldsymbol{\Phi}\left(\boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{h}_{l,j} + r_j \geqslant 0 |\ \boldsymbol{\theta} \in \Omega_l; \hat{\boldsymbol{\mu}}_l \boldsymbol{h}_{l,j} + r_{l,j}, \boldsymbol{h}_{l,j}^{\mathrm{T}} \hat{\boldsymbol{\Sigma}}_l \boldsymbol{h}_{l,j}\right) \\ & \quad \geqslant \gamma_l, \ \forall l \in \{1, 2, \dots, L\} \\ \Rightarrow \quad & \textstyle\sum_{l=1}^{L} \delta_l \, \boldsymbol{\Phi}\left(\boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{h}_{l,j} + r_j \geqslant 0 |\ \boldsymbol{\theta} \in \Omega_l; \hat{\boldsymbol{\mu}}_l \boldsymbol{h}_{l,j} + r_j, \boldsymbol{h}_{l,j}^{\mathrm{T}} \hat{\boldsymbol{\Sigma}}_l \boldsymbol{h}_{l,j}\right) \\ & \quad \geqslant \textstyle\sum_{l=1}^{L} \gamma_l \\ \Rightarrow \quad & \textstyle\sum_{l=1}^{L} \mathbb{P}(\boldsymbol{\theta}^{\mathrm{T}} \boldsymbol{h}_{l,j} + r_j \geqslant 0 |\ \boldsymbol{\theta} \in \Omega_l) \delta_l \geqslant 1 - \lambda \\ \Leftrightarrow \quad & \mathbb{P}(0 \leqslant y_{2,j}) \geqslant 1 - \lambda \end{aligned}$$

This concludes the proof. □

Even though Proposition 1 is described based on constraint (19), it can be further applied to constraint (17) with piecewise linear decision rule $\tilde{\boldsymbol{H}}$ and $\boldsymbol{r}$ embedded. Consequently, ($\mathcal{CCP}$) can be conservatively approximated by the following formula:

$$\min_{\boldsymbol{x}, \boldsymbol{e}, \boldsymbol{y}_1, \boldsymbol{H}_l, \boldsymbol{r}, \gamma_{i,l}} \ \boldsymbol{p}_0^{\mathrm{T}} \boldsymbol{x} + \boldsymbol{p}_1^{\mathrm{T}} \boldsymbol{y}_1 + \sum_{l=1}^{L} \boldsymbol{p}_2^{\mathrm{T}} \boldsymbol{r} \delta_l + \sum_{l=1}^{L} \boldsymbol{p}_2^{\mathrm{T}} \boldsymbol{H}_l \hat{\boldsymbol{\mu}}_l \delta_l \tag{$\mathcal{CCP}2$}$$

$$\text{s.t. } \boldsymbol{\Phi}^{-1}(\frac{\gamma_{i,l}}{\delta_l}) \sqrt{(\boldsymbol{y}_1^{\mathrm{T}} \boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}} \boldsymbol{H}_l) \hat{\boldsymbol{\Sigma}}_l (\boldsymbol{y}_1^{\mathrm{T}} \boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}} \boldsymbol{H}_l)^{\mathrm{T}}} \leqslant g_i - (\boldsymbol{y}_1^{\mathrm{T}} \boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}} \boldsymbol{H}_l) \hat{\boldsymbol{\mu}}_l - \boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x} - \boldsymbol{c}_i^{\mathrm{T}} \boldsymbol{r}, \ \forall i = 1, 2, \dots, I, \ \forall l = 1, 2, \dots, L, \tag{22}$$

$$\sum_{i=1}^{I} \sum_{l=1}^{L} \gamma_{i,l} = 1 - \epsilon + \lambda N_2, \tag{23}$$

$$\boldsymbol{A}' \boldsymbol{x} + \boldsymbol{C}' \boldsymbol{r} = \boldsymbol{0}, \tag{24}$$

$$\boldsymbol{B}'_k \boldsymbol{y}_1 + (\boldsymbol{C}' \boldsymbol{H}_l)_{:,k} = \boldsymbol{0}, \ \forall l = 1, 2, \dots, L, \ \forall k = 1, 2, \dots, K, \tag{25}$$

$$\boldsymbol{\Phi}^{-1}(1 - \lambda) \sqrt{\boldsymbol{h}_{l,j}^{\mathrm{T}} \hat{\boldsymbol{\Sigma}}_l \boldsymbol{h}_{l,j}} \leqslant \boldsymbol{h}_{l,j}^{\mathrm{T}} \hat{\boldsymbol{\mu}}_l + r_j, \forall j = 1, 2, \dots, N_2, \ \forall l = 1, 2, \dots, L, \tag{26}$$

$$\boldsymbol{x} \in \{0, 1\}, \boldsymbol{e} \in \{0, 1\}, \text{Eqs. (1), (2), } \boldsymbol{0} \leqslant \boldsymbol{y}_1.$$

Eqs. (24) and (25) are derived from Eq. (18) such that the constant term and coefficient of uncertainties are equal to zero, respectively. Because λ for non-negative constraints should be small, there is not much space to tune their risk level. Hence, each GMM cluster induced constraint for $0 \leqslant y_{2,j}$ specifies $\gamma_{j,l}/\delta_l = (1 - \lambda)$ in (26) to meet (20) and avoid unnecessary computational burden. In case study, we set $1 - \lambda = 0.9999$. Eqs. (22) and (23) are designed for quality constraint (17) by applying Boole's inequality and Proposition 1. The risk level $\gamma_{i,l}$ for each cluster-induced constraint (22) is a decision variable.

3. Optimization methods

In this section, several optimization methods, including outer approximation, convex relaxation, branch-and-bound, and optimality-based bound tightening, are developed and integrated to solve ($\mathcal{CCP}2$). Given that ($\mathcal{CCP}2$) is inherently non-convex, the proposed algorithm constructs and solves a lower bounding mixed-integer convex problem and an upper bounding mixed-integer convex problem iteratively.

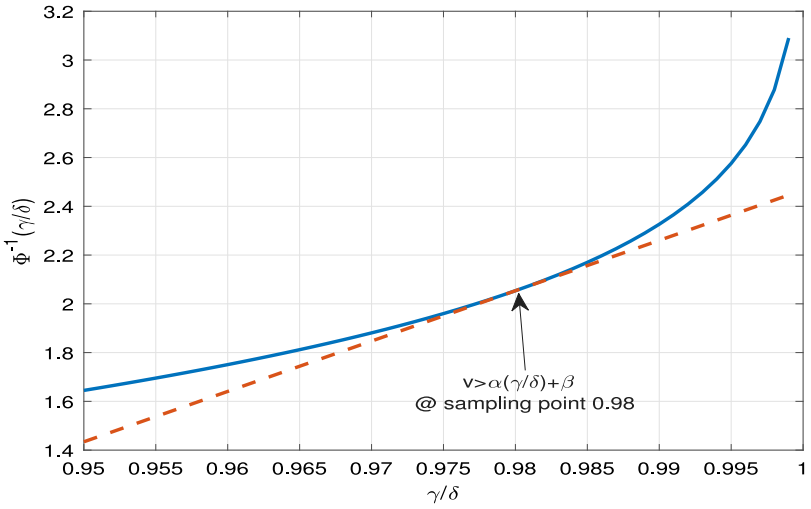

Fig. 2. The outer approximation of $\Phi^{-1}(\frac{\gamma}{\delta})$.

The gap between lower and upper bounds gradually converges and ultimately a near-optimal solution is achieved. The conservativeness of the proposed method mainly arises from the Boole's inequality and pre-specified functional structure of stage-II variables.

3.1. Adaptive outer approximation

The function Φ^{-1} should be numerically approximated because it does not have an analytical form. The outer approximation method developed by Cheng et al. (2012) can be applied because $\frac{\gamma_{i,l}}{\delta_l} \mapsto \Phi^{-1}(\frac{\gamma_{i,l}}{\delta_l})$ is convex when $\frac{\gamma_{i,l}}{\delta_l} \in [0.5, 1)$. A graphical demonstration of the outer approximation is shown in Fig. 2 with qth cutting plane $v \geqslant \alpha_q \frac{\gamma_{i,l}}{\delta_l} + \beta_q$. The tangent α_q and interception β_l are:

$$\alpha_q = \left.\frac{d\Phi^{-1}(\frac{\gamma_{i,l}}{\delta_l})}{d(\frac{\gamma_{i,l}}{\delta_l})}\right|_{\frac{\gamma_{i,l}}{\delta_l}=\pi_q} = \frac{1}{\phi(\Phi^{-1}(\pi_q))},$$

$$\beta_q = \Phi^{-1}(\pi_q) - \alpha_q \pi_q.$$

where π_q is qth sampled value of $\frac{\gamma_{i,l}}{\delta_l}$ and ϕ is the standard probability distribution function of a normal distribution.

Consequently, an outer approximation of (*CCP*2) is:

$$\min_{\boldsymbol{x},\boldsymbol{e},\boldsymbol{y}_1,\boldsymbol{H}_l,\boldsymbol{r}_l,\gamma_{i,l}} \quad \boldsymbol{p}_0^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{p}_1^{\mathrm{T}}\boldsymbol{y}_1 + \sum_{l=1}^{L}\boldsymbol{p}_2^{\mathrm{T}}\boldsymbol{r}_l\delta_l + \sum_{l=1}^{L}\boldsymbol{p}_2^{\mathrm{T}}\boldsymbol{H}_l\hat{\boldsymbol{\mu}}_l\delta_l \tag{*CCP*3}$$

$$\text{s.t. } v_{i,l}\sqrt{(\boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l)\hat{\boldsymbol{\Sigma}}_l(\boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l)^{\mathrm{T}}} \leqslant \tag{27}$$

$$g_i - (\boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l)\hat{\boldsymbol{\mu}}_l - \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x} - \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{r},\ \forall i = 1,2,\dots,I,$$

$$\forall l = 1,2,\dots,L,$$

$$\sum_{i=1}^{I}\sum_{l=1}^{L}\gamma_{i,l} = 1 - \epsilon + \lambda N_2,$$

$$0.5 \leqslant \gamma_{i,l}/\delta_l \leqslant 0.9999, \tag{28}$$

$$v_{i,l} \geqslant \alpha_q\frac{\gamma_{i,l}}{\delta_l} + \beta_q, \forall q = 1,2,\dots,Q,\ \forall l = 1,2,\dots,L,$$

$$\forall i = 1,2,\dots,I,$$

$$\boldsymbol{A}'\boldsymbol{x} + \boldsymbol{C}'\boldsymbol{r} = \boldsymbol{0},$$

$$\boldsymbol{B}_k'\boldsymbol{y}_1 + (\boldsymbol{C}'\boldsymbol{H}_l)_{:,k} = \boldsymbol{0},\ \forall l = 1,2,\dots,L,\ \forall k = 1,2,\dots,K,$$

$$\Phi^{-1}(1-\lambda)\sqrt{\boldsymbol{h}_{l,j}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_l\boldsymbol{h}_{l,j}} \leqslant \boldsymbol{h}_{l,j}^{\mathrm{T}}\hat{\boldsymbol{\mu}}_l + r_j, \forall j = 1,2,\dots,N_2,$$

$$\forall l = 1,2,\dots,L,$$

$$\boldsymbol{x} \in \{0,1\}, \boldsymbol{e} \in \{0,1\}, \text{Eqs. (1), (2), } \boldsymbol{0} \leqslant \boldsymbol{y}_1,$$

where Q is the number of sampling points on γ/δ. As $Q \to \infty$, (*CCP*3) approaches (*CCP*2) with arbitrary accuracy. However, because (*CCP*2) has $I \times L \times Q$ cutting planes, increasing Q implies more constraints to be considered during the optimization. To balance the computational demand and approximation accuracy, an adaptive outer approximation scheme developed in Yang et al. (2017) is adopted to maintain Q at a manageable level. Starting from a small set of sampling points, the lower bounding problem is solved repeatedly at each iteration, and the resulting solution $\frac{\gamma_{i,l}^*}{\delta_l}$ is used as the sampling point to add new cutting planes in real time. Through this manner, the solution v_q can be continuously elevated to reduce the outer approximation error at interested feasible region and tighten (27). Moreover, some existing cutting planes can be eliminated if their associated sampling points are out of the interested feasible region. Because of $\Phi^{-1}(1) \to \infty$ and to ensure convexity, Eq. (28) sets the upper and lower bounds on $\gamma_{i,l}/\delta_l$. The upper bound 0.9999 is consistent with $1 - \lambda = 0.9999$ in the non-negative constraints.

3.2. Convex relaxation

The formula (*CCP*3) has non-convex terms:

$$v_{i,l}\sqrt{(\boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l)\hat{\boldsymbol{\Sigma}}_l(\boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l)^{\mathrm{T}}},\ \forall i = 1,2,\dots,I,\ \forall l = 1,2,\dots,L.$$

The McCormick method (McCormick, 1976) can be used to develop a convex relaxation of those non-convex terms. Let us define $\boldsymbol{u}_{i,l} = \boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l \in \mathcal{R}^K$ and introduce an auxiliary variable: $\boldsymbol{z}_{i,l} = v_{i,l}\boldsymbol{u}_{i,l} \in \mathcal{R}^K$, such that

$$v_{i,l}\sqrt{(\boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l)\hat{\boldsymbol{\Sigma}}_l(\boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l)^{\mathrm{T}}} = v_{i,l}\sqrt{\boldsymbol{u}_{i,l}\hat{\boldsymbol{\Sigma}}_l\boldsymbol{u}_{i,l}^{\mathrm{T}}} = \sqrt{\boldsymbol{z}_{i,l}^{\mathrm{T}}\boldsymbol{\Sigma}_l\boldsymbol{z}_{i,l}}$$

The McCormick relaxation utilizes a set of inequalities (29)–(32) to build a convex hull of $\boldsymbol{z}_{i,l} = v_{i,l}\boldsymbol{u}_{i,l}$. The resulting formula is:

$$\min_{\boldsymbol{x},\boldsymbol{e},\boldsymbol{y}_1,\boldsymbol{H}_l,\boldsymbol{r},\gamma_{i,l}} \quad \boldsymbol{p}_0^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{p}_1^{\mathrm{T}}\boldsymbol{y}_1 + \sum_{l=1}^{L}\boldsymbol{p}_2^{\mathrm{T}}\boldsymbol{r}_l\delta_l + \sum_{l=1}^{L}\boldsymbol{p}_2^{\mathrm{T}}\boldsymbol{H}_l\hat{\boldsymbol{\mu}}_l\delta_l \tag{*LBP*}$$

$$\text{s.t. } \sqrt{\boldsymbol{z}_{i,l}\hat{\boldsymbol{\Sigma}}_l\boldsymbol{z}_{i,l}^{\mathrm{T}}} \leqslant g_i - \boldsymbol{u}_{i,l}\hat{\boldsymbol{\mu}}_l - \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x} - \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{r},\ \forall i = 1,2,\dots,I,$$

$$\forall l = 1,2,\dots,L,$$

$$\boldsymbol{u}_{i,l} = \boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l,$$

$$\sum_{i=1}^{I}\sum_{l=1}^{L}\gamma_{i,l} = 1 - \epsilon + \lambda N_2,$$

$$0.5 \leqslant \gamma_{i,l}/\delta_l \leqslant 0.9999,$$

$$v_{i,l} \geqslant \alpha_q\frac{\gamma_{i,l}}{\delta_l} + \beta_q, \forall q = 1,2,\dots,Q,\ \forall l = 1,2,\dots,L,$$

$$\forall i = 1,2,\dots,I,$$
$$\boldsymbol{A}'\boldsymbol{x} + \boldsymbol{C}'\boldsymbol{r} = \boldsymbol{0},$$
$$\boldsymbol{B}'_k\boldsymbol{y}_1 + (\boldsymbol{C}'\boldsymbol{H}_l)_{:,k} = \boldsymbol{0},\ \forall l = 1,2,\dots,L,\ \forall k = 1,2,\dots,K,$$
$$\Phi^{-1}(1-\lambda)\sqrt{\boldsymbol{h}_{l,j}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_l\boldsymbol{h}_{l,j}} \leqslant \boldsymbol{h}_{l,j}^{\mathrm{T}}\hat{\boldsymbol{\mu}}_l + r_j, \forall j = 1,2,\dots,N_2,$$
$$\forall l = 1,2,\dots,L,$$
$$\boldsymbol{z}_{i,l} \geqslant \overline{v}_{i,l}\boldsymbol{u}_{i,l} + v_{i,l}\overline{\boldsymbol{u}}_{i,l} - \overline{v}_{i,l}\overline{\boldsymbol{u}}_{i,l},\ \forall l = 1,2,\dots,L, \forall i = 1,2,\dots,I, \tag{29}$$
$$\boldsymbol{z}_{i,l} \geqslant \underline{v}_{i,l}\boldsymbol{u}_{i,l} + v_{i,l}\underline{\boldsymbol{u}}_{i,l} - \underline{v}_{i,l}\underline{\boldsymbol{u}}_{i,l},\ \forall l = 1,2,\dots,L, \forall i = 1,2,\dots,I, \tag{30}$$
$$\boldsymbol{z}_{i,l} \leqslant \overline{v}_{i,l}\boldsymbol{u}_{i,l} + v_{i,l}\underline{\boldsymbol{u}}_{i,l} - \overline{v}_{i,l}\underline{\boldsymbol{u}}_{i,l},\ \forall l = 1,2,\dots,L, \forall i = 1,2,\dots,I, \tag{31}$$
$$\boldsymbol{z}_{i,l} \leqslant \underline{v}_{i,l}\boldsymbol{u}_{i,l} + v_{i,l}\overline{\boldsymbol{u}}_{i,l} - \underline{v}_{i,l}\overline{\boldsymbol{u}}_{i,l},\ \forall l = 1,2,\dots,L, \forall i = 1,2,\dots,I, \tag{32}$$
$$\boldsymbol{x} \in \{0,1\}, \boldsymbol{e} \in \{0,1\}, \text{Eqs. (1), (2), } \boldsymbol{0} \leqslant \boldsymbol{y}_1,$$

where the upper and lower bars represent the upper and lower bounds on variables, respectively. Solving ($\mathcal{LBP}$) will generate a lower bounding solution of ($\mathcal{CCP}2$) and ($\mathcal{CCP}3$), denoted as LB. Note that ($\mathcal{LBP}$) is a mixed-integer second-order cone program (MI-SOCP), and thus can be solved to the global optimum with pre-specified relative gap through off-the-shelf solvers.

In the proposed scheme, totally $I \times L \times K$ bilinear terms in ($\mathcal{LBP}$) can be relaxed. Alternatively, one may consider to multiple $v_{i,l}$ into the matrix $\boldsymbol{H}_l$ and relax the term $v_{i,l}h_{l,j,k}$, $\forall j = 1,2,\dots,N_2, \forall k = 1,2,\dots,K$. While this approach may result in a tighter relaxation, it has to consider $I \times L \times N_2 \times K$ bilinear terms, which may impede the efficiency of subsequent branch-and-bound and bound tightening. Moreover, as both $\boldsymbol{h}_{i,j,k}$ and $\boldsymbol{u}_{i,l}$ involve decision rule coefficients, their upper and lower bounds should be guessed without prior information. Therefore, a relaxation scheme with less bilinear terms is preferred.

3.3. Branch-and-bound

In this sub-section, we develop a global optimization framework for ($\mathcal{CCP}2$). The relaxation gap of ($\mathcal{LBP}$) depends on the distance between upper and lower bounds on variables $\boldsymbol{v}$ and $\boldsymbol{u}$. Hence, a branch-and-bound scheme can be employed to artificially reduce the feasible interval, and thereby the relaxation gap. To this end, a searching tree is initialized with the entire $\boldsymbol{u}$ intervals. According to our previous research (Yang et al., 2017), branching $\boldsymbol{u}$ is more efficient than branching v for global optimization. Hence, ($\mathcal{LBP}$) is solved at each node with associated intervals during the tree traversal, and then the selected variable in vector $\boldsymbol{u}$ is branched to create two new nodes on the searching tree. The following criterion is applied for branching variable selection.

$$\{i', l', k'\} = \arg \max_{i\in\{1,2,\dots,I\}, l\in\{1,2,\dots,L\}, k\in\{1,2,\dots,K\}} |v_{i,l}^* u_{i,l,k}^* - z_{i,l,k}^*|, \tag{33}$$

where $v_{i,l}^*$, $u_{i,l,k}^*$ and $z_{i,l,k}^*$ are the solution of ($\mathcal{LBP}$) at a node. Once $u_{i',l',k'}$ is chosen, its interval is divided into two disjoint sub-intervals: $[\underline{u}_{i',l',k'}, u_{i',l',k'}^*]$ and $[u_{i',l',k'}^*, \overline{u}_{i',l',k'}]$, while other variables' bounds are unchanged. The resulting two exclusive feasible regions together with their parent ($\mathcal{LBP}$) solution value will be used to create two children nodes in the searching tree.

At each iteration, we always choose a unsolved node associated with the smallest parent ($\mathcal{LBP}$) solution value. By solving ($\mathcal{LBP}$) at that node, we obtain the risk level $\gamma_{i,l}^*$ assignment, which can be substituted into ($\mathcal{CCP}2$) to yield an upper bounding problem:

$$\min_{\boldsymbol{x},\boldsymbol{e},\boldsymbol{y}_1,\boldsymbol{H}_l,\boldsymbol{r}} \ \boldsymbol{p}_0^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{p}_1^{\mathrm{T}}\boldsymbol{y}_1 + \sum_{l=1}^{L}\boldsymbol{p}_2^{\mathrm{T}}\boldsymbol{r}\delta_l + \sum_{l=1}^{L}\boldsymbol{p}_2^{\mathrm{T}}\boldsymbol{H}_l\hat{\boldsymbol{\mu}}_l\delta_l \tag{$\mathcal{UBP}$}$$
$$\text{s.t. } \Phi^{-1}(\frac{\gamma_{i,l}^*}{\delta_l})\sqrt{(\boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l)\hat{\boldsymbol{\Sigma}}_l(\boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l)^{\mathrm{T}}} \leqslant$$
$$g_i - (\boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l)\hat{\boldsymbol{\mu}}_l - \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x} - \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{r},\ \forall i = 1,2,\dots,I,$$
$$\forall l = 1,2,\dots,L,$$
$$\boldsymbol{A}'\boldsymbol{x} + \boldsymbol{C}'\boldsymbol{r} = \boldsymbol{0},$$
$$\boldsymbol{B}'_k\boldsymbol{y}_1 + (\boldsymbol{C}'\boldsymbol{H}_l)_{:,k} = \boldsymbol{0},\ \forall l = 1,2,\dots,L,\ \forall k = 1,2,\dots,K,$$
$$\Phi^{-1}(1-\lambda)\sqrt{\boldsymbol{h}_{l,j}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_l\boldsymbol{h}_{l,j}} \leqslant \boldsymbol{h}_{l,j}^{\mathrm{T}}\hat{\boldsymbol{\mu}}_l + r_j, \forall j = 1,2,\dots,N_2,$$
$$\forall l = 1,2,\dots,L,$$
$$\boldsymbol{x} \in \{0,1\}, \boldsymbol{e} \in \{0,1\}, \text{Eqs. (1), (2), } \boldsymbol{0} \leqslant \boldsymbol{y}_1.$$

Note that ($\mathcal{UBP}$) is still an MI-SOCP that can be solved to optimum rapidly. If ($\mathcal{UBP}$) is infeasible, it implies that the convex envelop made from outer approximation and McCormick relaxation is not tight enough. To resolve this issue, the adaptive outer approximation scheme always samples at the solution point of ($\mathcal{LBP}$) $\frac{\gamma_{i,l}^*}{\delta_{i,l}}$ to construct an additional cutting plane, and the variable bounds should be further reduced by branching and ($\mathcal{OBBT}$). If ($\mathcal{UBP}$) is feasible and its objective value is less than the existing upper bound solution, denoted as UB, then we can update UB accordingly. The entire algorithm, outlined in Fig. 3, will terminate if the relative gap, defined in Eq. (34), is below a predefined threshold.

$$\text{Relative gap} = \frac{\text{UB} - \text{LB}}{|\text{LB}|} \tag{34}$$

3.4. Optimality-based bounds tightening

The McCormick method generates the tightest convex envelop for bilinear terms given a specific variable interval. Typically, a narrower interval results in a tighter relaxation. While conventional interval analysis techniques are effective in reducing bounds with minimum computational overhead, previous research has showed that optimality-based bounds tightening (OBBT) offers superior efficiency, especially when lower and upper bounding problems are solved repeatedly. Even though the bilinear terms of ($\mathcal{CCP}3$) involve two variables, $v_{i,l}$ only lies in a small range: $[\Phi^{-1}(0.5), \Phi^{-1}(0.9999)]$ whereas $u_{i,l,k}$ may have a much wider interval. Thus, the OBBT formula shown in ($\mathcal{OBBT}$) only focuses on $u_{i,l,k}, \forall i = 1,2,\dots,I, \forall l = 1,2,\dots,L, \forall k = 1,2,\dots,K$:

$$\min_{\boldsymbol{x},\boldsymbol{e},\boldsymbol{y}_1,\boldsymbol{H}_l,\boldsymbol{r}} \ u_{i,l,k} \tag{$\mathcal{OBBT}$}$$
$$\text{s.t. } \sqrt{\boldsymbol{z}_{i,l}\hat{\boldsymbol{\Sigma}}_l\boldsymbol{z}_{i,l}^{\mathrm{T}}} \leqslant g_i - \boldsymbol{u}_{i,l}\hat{\boldsymbol{\mu}}_l - \boldsymbol{a}_i^{\mathrm{T}}\boldsymbol{x} - \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{r},\ \forall i = 1,2,\dots,I,$$
$$\forall l = 1,2,\dots,L,$$
$$\boldsymbol{u}_{i,l} = \boldsymbol{y}_1^{\mathrm{T}}\boldsymbol{B}_i^{\mathrm{T}} + \boldsymbol{c}_i^{\mathrm{T}}\boldsymbol{H}_l,$$
$$\sum_{i=1}^{I}\sum_{l=1}^{L}\gamma_{i,l} = 1 - \epsilon + \lambda N_2,$$
$$0.5\delta_l \leqslant \gamma_{i,l} \leqslant 0.9999\delta_l,$$
$$v_{i,l} \geqslant \alpha_q\frac{\gamma_{i,l}}{\delta_l} + \beta_q, \forall q = 1,2,\dots,Q,\ \forall l = 1,2,\dots,L,$$
$$\forall i = 1,2,\dots,I,$$
$$\boldsymbol{A}'\boldsymbol{x} + \boldsymbol{C}'\boldsymbol{r} = \boldsymbol{0},$$
$$\boldsymbol{B}'_k\boldsymbol{y}_1 + (\boldsymbol{C}'\boldsymbol{H}_l)_{:,k} = \boldsymbol{0},\ \forall l = 1,2,\dots,L,\ \forall k = 1,2,\dots,K,$$
$$\Phi^{-1}(1-\lambda)\sqrt{\boldsymbol{h}_{l,j}^{\mathrm{T}}\hat{\boldsymbol{\Sigma}}_l\boldsymbol{h}_{l,j}} \leqslant \boldsymbol{h}_{l,j}^{\mathrm{T}}\hat{\boldsymbol{\mu}}_l + r_j, \forall j = 1,2,\dots,N_2,$$
$$\forall l = 1,2,\dots,L,$$
$$\boldsymbol{z}_{i,l} \geqslant \overline{v}_{i,l}\boldsymbol{u}_{i,l} + v_{i,l}\overline{\boldsymbol{u}}_{i,l} - \overline{v}_{i,l}\overline{\boldsymbol{u}}_{i,l},\ \forall l = 1,2,\dots,L, \forall i = 1,2,\dots,I,$$
$$\boldsymbol{z}_{i,l} \geqslant \underline{v}_{i,l}\boldsymbol{u}_{i,l} + v_{i,l}\underline{\boldsymbol{u}}_{i,l} - \underline{v}_{i,l}\underline{\boldsymbol{u}}_{i,l},\ \forall l = 1,2,\dots,L, \forall i = 1,2,\dots,I,$$
$$\boldsymbol{z}_{i,l} \leqslant \overline{v}_{i,l}\boldsymbol{u}_{i,l} + v_{i,l}\underline{\boldsymbol{u}}_{i,l} - \overline{v}_{i,l}\underline{\boldsymbol{u}}_{i,l},\ \forall l = 1,2,\dots,L, \forall i = 1,2,\dots,I,$$
$$\boldsymbol{z}_{i,l} \leqslant \underline{v}_{i,l}\boldsymbol{u}_{i,l} + v_{i,l}\overline{\boldsymbol{u}}_{i,l} - \underline{v}_{i,l}\overline{\boldsymbol{u}}_{i,l},\ \forall l = 1,2,\dots,L, \forall i = 1,2,\dots,I,$$
$$\boldsymbol{x} \in \{0,1\}, \boldsymbol{e} \in \{0,1\}, \text{Eqs. (1), (2), } \boldsymbol{0} \leqslant \boldsymbol{y}_1,$$
$$\boldsymbol{p}_0^{\mathrm{T}}\boldsymbol{x} + \boldsymbol{p}_1^{\mathrm{T}}\boldsymbol{y}_1 + \sum_{l=1}^{L}\boldsymbol{p}_2^{\mathrm{T}}\boldsymbol{r}\delta_l + \sum_{l=1}^{L}\boldsymbol{p}_2^{\mathrm{T}}\boldsymbol{H}_l\hat{\boldsymbol{\mu}}_l\delta_l \leqslant \text{UB}. \tag{35}$$

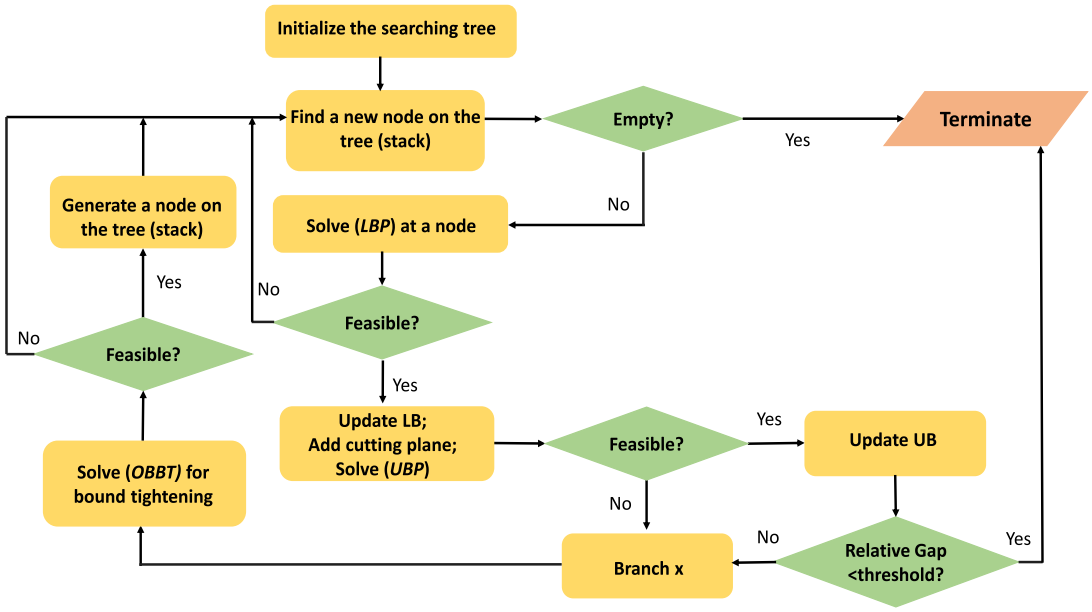

Fig. 3. The algorithm flowchart for solving (*CCP2*).

The constraint (35) requires that any feasible values of $u_{i,l,k}$ in ($\mathcal{OBBT}$) yields lower value than UB. Because ($\mathcal{OBBT}$) is solved at least $I \times L \times K$ times at each node of searching tree, its implementation should be well-designed to reduce the overall computational time. Several strategies are shown below:

- The integer constraint can be relaxed to convert ($\mathcal{OBBT}$) into a SOCP, which can be solved to global optimum quickly.
- One of options is to repeat the OBBT several rounds until none of any variable interval can be further reduced. However, this may substantially increase the solving time, and thus is not applied in the case study.
- If the interval of a variable is significantly small, we may omit its OBBT to save the computational time.
- Constraint (35) can significantly reduce the variable interval given the upper bound solution UB. In order to obtain UB, we may initialize $\frac{\gamma_{i,l}}{\delta_l}$, $\forall i = 1,2,\ldots,I$, $\forall l = 1,2,\ldots,L$ with equal value and solve ($\mathcal{UBP}$). If it is infeasible, we just initialize UB as zero by assuming the refining process is profitable. During the optimization, once a better UB is found, ($\mathcal{OBBT}$) should be solved for each component of $\boldsymbol{u}$ to refine variable intervals.

Finally, a flowchart of the global optimization algorithm is shown in Fig. 3.

4. Case study

In the case study, TCCP with non-Gaussian distributed uncertainties is solved for refinery optimization. The software platform is GAMS 41, with MI-SOCP solver CPLEX. The hardware platform is a laptop with Intel Core i5-8300U CPU 2.30 GHZ and 8GM RAM. The risk level ϵ is set as 5%, 6% and 7%, respectively. The proposed algorithm is executed to determine the optimal crude oil procurement and the decision rule for refining in stage-II. The algorithm terminates when the relative gap reaches 1%. Then, the resulting solution is evaluated in a test bed consisting of 500 independent samples of uncertain parameters. A scenario-based mixed-integer linear program (MILP) formula is also developed as a benchmark to compare with the proposed algorithm regarding the solution time and quality.

4.1. Refining process parameters

A refining process shown in Fig. 1 is studied. The mass balance, product demands, and quality constraints impacted by uncertainties are listed in Eqs. (3)–(11). The crude oil yields and their prices are shown in Table 1. The constraints, including product quality, demands, and unit capacity, are presented in Table 2.

Three operational properties are influenced by uncorrelated uncertainties, including sulfur residual, ISO RON, and CN yield in cracker. Specifically, their nominal values are 1% for sulfur residual, 94 for ISO RON, and 43.6% for CN yield in Mogas mode, or 38.1% in AGO mode. The uncertainties associated with these properties are characterized by inverse Gaussian, Weibull, and Gamma distributions, respectively, which are all approximated by 2-component GMMs based on a dataset comprising 1000 samples. Here we need to point out that when more GMM components are introduced for these uncertainties, the resulting weight w becomes significantly smaller than others. Our previous work (Yang, 2023) for GMM-based single-stage CCP showed that removing the second-order constraints associated with such small w and δ does not impact the solution feasibility and only slightly reduces optimality. In addition, we employ various types of asymmetric distribution to assess the approximation ability of GMM. The resulting GMM parameters obtained through the Expectation-Maximization algorithm are listed in Table 3. The GMM and histogram derived PDFs on training data, alongside the true distribution are shown in Figs. 4–6. The visual comparison reveals a close alignment between the GMM approximations and the true distribution, particularly in the distribution tails.

4.2. Sample average approximation

Alternative, the TCCP for refining process optimization can be solved via the sample average approximation (SAA) (Luedtke and Ahmed, 2008). Different from the proposed approach, SAA is a scenario-based method. Its formula is shown in 4.2:

$$\min_{\boldsymbol{x},\boldsymbol{e},\boldsymbol{o}} \ \boldsymbol{p}_0^{\mathrm{T}}\boldsymbol{x} + \frac{1}{M(1-\epsilon)} \sum_{m=1}^{M} \mathcal{F}(\boldsymbol{x}, \boldsymbol{\theta}_m, o_m) \qquad (\mathcal{SAA})$$

s.t. Eqs. (1), (2),

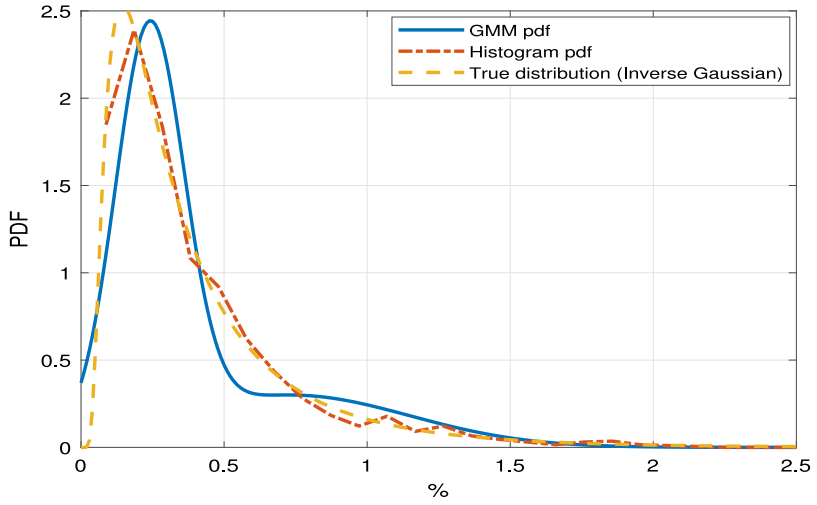

Fig. 4. The GMM, histogram derived, and true distribution (Inverse Gaussian) of sulfur residual increment.

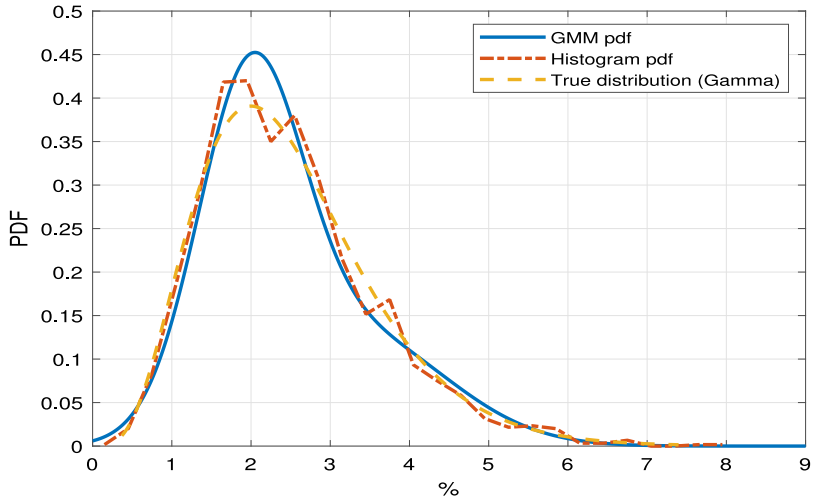

Fig. 5. The GMM, histogram derived, and true distribution (Gamma) of CN yield increment.

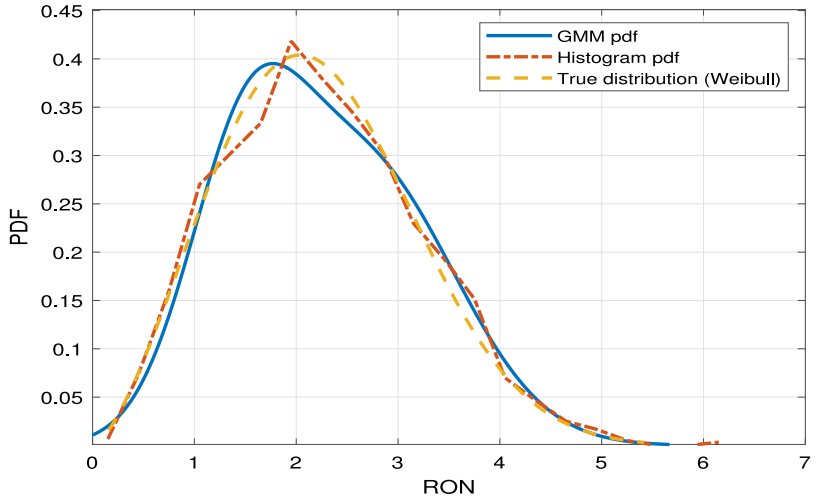

Fig. 6. The GMM, histogram derived, and true distribution (Weibull) of ISO RON reduction.

Table 1
Yields and price of three crude in 2017 ($ per barrel).

Crude	RG	LG	LN	HN	KE	GO	VGO	VR	Price ($)
Crude_1	0.002	0.0091	0.0698	0.1598	0.1003	0.2876	0.2682	0.1032	49.2
Crude_2	0.002	0.0080	0.061	0.1206	0.0861	0.2414	0.2646	0.2163	46.3
Crude_3	0.004	0.020	0.0851	0.1532	0.0947	0.2539	0.2535	0.1356	48.3

Table 2
Specifications, demands, and capacity.

Specifications	Max	Min
Gasoline-98 RON		98
Gasoline-95 RON		95
Gasoline-98 Sulfur (ppm)	15	
Gasoline-95 Sulfur (ppm)	15	
Heavy Fuel Oil Viscosity (index)	35	
Diesel Sulfur (ppm)	50	
Demand (KT)	Max	Min
Gasoline98		15
AGO		100
LG		2
Capacity (KT)	Max	Min
Cracker	135	
Desulfurization	130	

Table 3
Operational uncertainties. Here we build GMMs to approximate the distribution of CN yield increment, ISO RON reduction, and sulfur residual increment.

Uncertainty	Mean 1	Mean 2	Variance 1	Variance 2	Weight 1	Weight 2
θ_{Cr}	3.354%	2.021%	$1.434/10^4$	$0.510/10^4$	0.370	0.630
θ_{RON}	1.610	2.659	0.410	0.803	0.428	0.572
θ_S	0.240%	0.733%	$0.014/10^4$	$0.172/10^4$	0.688	0.312

$$\sum_{m=1}^{M} o_m = (1-\epsilon)M,$$
$$\boldsymbol{x} \in \{0,1\}, \boldsymbol{e} \in \{0,1\}, o_m \in \{0,1\}, \forall m = 1,2,\dots,M,$$

where $\boldsymbol{\theta}_m$ is the sampled uncertainty value and o_m indicates if that scenario can be infeasible or not. When $o_m = 0$, mth scenario will not be counted in the objective function. The stage-II formula of mth scenario is:

$$\begin{aligned}\mathcal{F}(\boldsymbol{x}, \boldsymbol{\theta}_m, o_m) = \min_{\boldsymbol{y}_1, \boldsymbol{y}_2} (\boldsymbol{p}_1^{\mathrm{T}} \boldsymbol{y}_1 + \boldsymbol{p}_2^{\mathrm{T}} \boldsymbol{y}_2) o_m \\ \text{s.t. } \boldsymbol{a}_i^{\mathrm{T}} \boldsymbol{x} + \sum_{k=1}^{K} \boldsymbol{b}_{k,i}^{\mathrm{T}} \theta_{m,k} \boldsymbol{y}_1 + \boldsymbol{c}_i^{\mathrm{T}} \boldsymbol{y}_2 \leqslant \boldsymbol{g}_i + (1-o_m)W, \\ \forall i = 1,2,\dots,I, \\ \boldsymbol{A}'\boldsymbol{x} + \sum_{k=1}^{K} \boldsymbol{B}'_k \theta_{m,k} \boldsymbol{y}_1 + \boldsymbol{C}' \boldsymbol{y}_2 = 0, \\ 0 \leqslant \boldsymbol{y}_1, \boldsymbol{0} \leqslant \boldsymbol{y}_2,\end{aligned}$$

where W is a big positive number to ensure that all inequality constraints of mth scenario are relaxed when $o_m = 0$. Although the objective function of $\mathcal{F}(\boldsymbol{x}, \boldsymbol{\theta}_m, o_m)$ has bilinear term, it can be equivalently converted into a MILP by using McCormick relaxation because of binary variable o_m. Here $\{1,2,\dots,M\}$ represents a training scenario set for SAA. The computational challenge of 4.2 lies in the M newly introduced binary variables to deal with chance constraints. Even though the decomposition approaches in Liu et al. (2016), Yang (2019) may speedup the solving process of 4.2, they usually require recovery operations in stage-II, which is not applicable in this case study. Thus, the MILP solver is directly applied in 4.2 to obtain the solution.

There are plenty of works (Yang and Sutanto, 2019; Alamo et al., 2015; Campi and Garatti, 2011, 2008; Calafiore and Campi, 2006) to investigate the sample complexity of single-stage CCP, but there has been relatively limited research addressing the number of required samples for two-stage SP or CCP. Therefore, it becomes imperative to vary the number of sampled scenarios, namely T in 4.2, to examine the computational time and solution quality. It is worthwhile to note that 4.2 can be converted into an MILP and solved using the training scenario set to a specified relative gap. However, that solution should be validated in an independent scenario set because there is no theoretical guarantee of optimality on unseen scenarios.

4.3. Results comparison

In this subsection, we show the optimization results under various risk levels (5%, 6%, 7%) and compare the proposed method with SAA on the validation set, including 500 independent samples of uncertain parameters. While we use the piecewise linear decision rule to solve $(CCP2)$ and obtain $\boldsymbol{x}^*$, the stage-II operations can be further optimized once the crude oil procurement is determined. Hence, given $\boldsymbol{x}^*$, we show the solution of $(CCP2)$, the decision rule on validation set, and scenario-based approximation on validation set, in Tables 4–6.

Proposed approach $(CCP2)$ and DR validation: To verify the decision rule, we first cluster the sampled uncertain parameters on the validation set using the GMM. Because this refinery model only has 3 uncertain parameters and 2-component for each GMM, there are only $L = 2^3 = 8$ clusters. Each cluster is equipped with a linear function for stage-II operations. When stage-I variable $\boldsymbol{x}^*$ is fixed, stage-II variables of each scenario can be determined according to the cluster it belongs to and associated decision rule function. The optimal objective value of $(CCP2)$ matches the result of decision rule on validation set (DR validation) very well. At risk level 5%, there is 0.1% difference in profit and 0.2% difference in probabilistic feasibility. At risk level 6%, there is 0.1% difference in profit and 1.2% in feasible chance. At risk level 7%, there is 0.12% difference in profit and 1.2% in feasible chance. The resulting feasible chance under all risk levels aligns with the desired reliability $1-\epsilon$ and is slightly higher on the validation. This further shows that GMM can capture the uncertainty distribution and our optimization algorithm indeed yields a satisfactory solution.

Proposed approach SAA(x^*): It is not surprised that when $\boldsymbol{x}^*$ is derived from $(CCP2)$, optimizing each scenario individually leads to a better profit than simply applying the decision rule. We denote such results as SAA(x^*). On validation set, the feasible chance of SAA($\boldsymbol{x}^*$) is moderately higher than the desired $1-\epsilon$ and that of decision rule. This result indicates that the proposed method is conservative (safe) due to the pre-specified piecewise linear structure of stage-II variables and the limitation of Boole's inequality. More complex functional structure may be needed in the future work.

SAA 200, SAA 500, SAA 1000: It is worthwhile to note that using SAA to search $\boldsymbol{x}$ may also attain good solutions. In general, a large number of scenarios are required to accurately estimate the expected objective function and ensure probabilistic feasibility. When only 200 scenarios are considered in the training set, the resulting solution cannot meet the reliability $1-\epsilon$ on the validation set. Due to the

Table 4
Optimization results at risk level 5%. DR Validation: Decision rule is applied on each validation scenario; SAA($\boldsymbol{x}^*$): $\boldsymbol{x}^*$ is the state-I solution of (*CCP*2) and each stage-II validation scenario is solved to optimum individual. The results of SAA 200, SAA 500, and SAA 1000 are all on validation set with the maximum solution time 4 h (14400 s). Note that SAA 200 does not meet the required probabilistic feasibility.

	Proposed approach (*CCP*2) & DR validation & SAA($\boldsymbol{x}^*$)	SAA 200	SAA 500	SAA 1000
Solution time (s)	5434	14 400	14 400	14 400
Relative gap	<1%	10.53%	12.16%	14.00%
Objective value (Profit $)	99,723,064 & 99,619,684 & 103,621,483	104,761,071	103,424,582	103,895,013
Probabilistic feasibility	95% & 95.2% & 97.4%	91.4%	97.6%	96.4%

Table 5
Optimization results at risk level 6%. DR Validation: Decision rule is applied on each validation scenario; SAA($\boldsymbol{x}^*$): $\boldsymbol{x}^*$ is the state-I solution of (*CCP*2) and each stage-II validation scenario is solved to optimum individual. The results of SAA 200, SAA 500, and SAA 1000 are all on validation set with the maximum solution time 4 h (14400 s). Note that SAA 200 does not meet the required probabilistic feasibility.

	Proposed approach (*CCP*2) & DR validation & SAA($\boldsymbol{x}^*$)	SAA 200	SAA 500	SAA 1000
Solution time (s)	3135	14 400	14 400	14 400
Relative gap	<1%	13.06%	14.94%	16.73%
Objective value (Profit $)	100,094,923 & 99,985,827 & 103,875,771	104,798,472	103,701,645	104,065,318
Probabilistic feasibility	94% & 95.2% & 96.4%	91.2%	97%	96%

Table 6
Optimization results at risk level 7%. DR Validation: Decision rule is applied on each validation scenario; SAA($\boldsymbol{x}^*$): $\boldsymbol{x}^*$ is the state-I solution of (*CCP*2) and each stage-II validation scenario is solved to optimum individual. The results of SAA 200, SAA 500, and SAA 1000 are all on validation set with the maximum solution time 4 h (14400 s). Note that SAA 200 does not meet the required probabilistic feasibility.

	Proposed approach (*CCP*2) & DR validation & SAA($\boldsymbol{x}^*$)	SAA 200	SAA 500	SAA 1000
Solution time (s)	6461	14 400	14 400	14 400
Relative gap	<1%	16.78%	18.25%	24.97%
Objective value (Profit $)	100,391,861 & 100,266,904 & 104,036,565	104,706,025	103,944,638	103,424,582
Probabilistic feasibility	93% & 94.2% & 96.0%	91.8%	96.2%	97.6%

scenario-dependence, the SAA-based stage-I solution may not enable optimal or even feasible stage-II operations in unseen scenarios. As the scenario number is increased to 1000, then SAA slightly outperforms our method at risk levels 5% and 6%. On the other hand, a large number of scenario also lowers the convergence rate. As we terminate the MILP solver for SAA after 4 h, the relative gap for 7% risk level is nearly 25% and the resulting profit is much lower than that of the piecewise linear decision rule SAA(x^*). In fact, when running the SAA with 1000 scenarios, the solving process consumes the laptop's memory resources after 20 h. In contrast, the proposed (*CCP*2) successfully converges to the desired relative gap (1%) within 3100–6500 s at different risk levels.

In Fig. 7, we present the results of crude oil procurement determined by the proposed formula under 5%, 6%, and 7% risk levels. As allowed risk level increases, more type-2 crude oil is purchased. Even though that oil has low KE and GO yields, which are essential for producing on-spec gasoline and diesel, it is cheaper than others and we are willing to accept some level of risks. Hence, the proposed method opts to purchase 54.350, 67.099, and 76.492 ton of type-2 crude oil for risk levels 5%, 6% and 7%, respectively. From the data in Tables 4–6, we can see that the expected profit boosts for nearly $200,000 when 1% more risk is tolerated.

5. Conclusion

This paper proposes a methodology to solve the two-stage chance-constrained program for refinery optimization. The piecewise linear decision rule is employed to parameterize the stage-II variable as a function of uncertainty whose distribution is approximated by GMM. We show that the resulting formula can be safely convexified to a lower bounding problem using outer approximation and McCormick relaxation techniques. Additionally, an upper bounding problem can be constructed by fixing the risk level of joint chance constraints. By iteratively solving the lower and upper bounding problems through

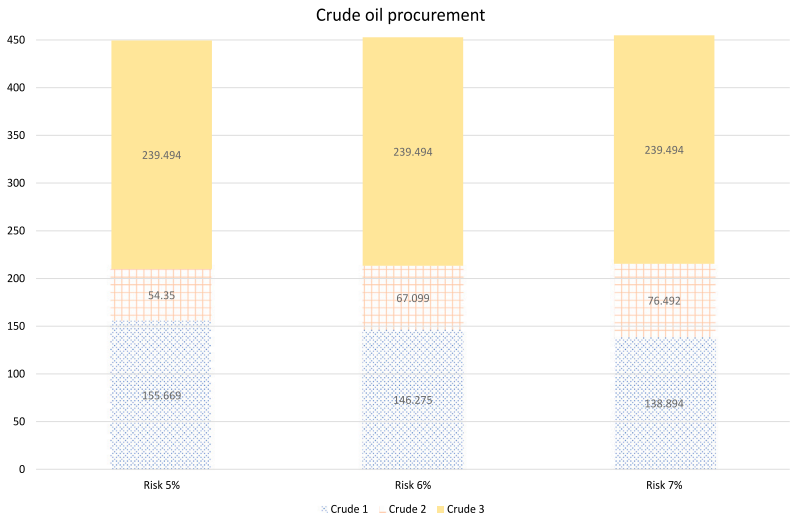

Fig. 7. The crude oil procurement determined by the proposed method under 5%, 6%, and 7% risk levels (Unit: ton).

branch-and-bound and optimality-based bound tightening, the gap converges to zero and a global optimum of two-stage chance-constrained program is found. The proposed method is compared with the sample average approximation in the optimization of crude oil procurement and plant operations for a simplified refinery model to demonstrate its effectiveness in solution time and optimality.

CRediT authorship contribution statement

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

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 the financial support from National Science Foundation (NSF), United States, 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.
Avraamidou, S., Pistikopoulos, E.N., 2020. Adjustable robust optimization through multi-parametric programming. Optim. Lett. 14, 873–887.
Bampou, D., Kuhn, D., 2011. Scenario-free stochastic programming with polynomial decision rules. In: IEEE Decision and Control and European Control Conference. pp. 7806–7812.
Ben-Tal, A., Goryashko, A., Guslitzer, E., Nemirovski, A., 2004. Adjustable robust solutions of uncertain linear programs. Math. Program. 99, 351–376.
Benders, J.F., 1962. Partitioning procedures for solving mixed-variables programming problems. Numer. Math. 4, 238–252.
Bertsimas, D., Dunning, I., 2016. Multistage robust mixed-integer optimization with adaptive partitions. Oper. Res. 64, 980–998.
Bounitsis, G.L., Papageorgiou, L.G., Charitopoulos, V.M., 2022. Data-driven scenario generation for two-stage stochastic programming. Chem. Eng. Res. Des. 187, 206–224.
Calafiore, G.C., 2008. Multi-period portfolio optimization with linear control policies. Automatica 44, 2463–2473.
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.
Chen, X., Sim, M., Sun, P., Zhang, J., 2008. A linear decision-based approximation approach to stochastic programming. Oper. Res. 56, 344–357.
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.
Favennec, J.P., 2001. Refinery Operation and Management. Editions TECHNIP, Paris.
Geoffrion, A.M., 1972. Generalized benders decomposition. J. Optim. Theory Appl. 10, 237–260.
Georghiou, A., Wiesemann, W., Kuhn, D., 2015. Generalized decision rule approximations for stochastic programming via liftings. Math. Program. 152, 301–338.
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.
Hanasusanto, G.A., Kuhn, D., Wiesemann, W., 2015. K-adaptability in two-stage robust binary programming. Oper. Res. 63, 877–891.
Jiang, R., Guan, Y., 2016. Data-driven chance constrained stochastic program. Math. Program. 158, 291–327.
Kammammettu, S., Li, Z., 2023. Scenario reduction and scenario tree generation for stochastic programming using sinkhorn distance. Comput. Chem. Eng. 170, 108122.
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.
Karuppiah, R., Grossmann, I.E., 2008. A Lagrangean based branch-and-cut algorithm for global optimization of nonconvex mixed-integer nonlinear programs with decomposable structures. J. Global Optim. 41, 163–186.
Li, P., Arellano-Garcia, H., Wozny, G., 2008. Chance constrained programming approach to process optimization under uncertainty. Comput. Chem. Eng. 32, 25–45.
Li, X., Armagan, E., Tomasgard, A., Barton, P.I., 2011. Stochastic pooling problem for natural gas production network design and operation under uncertainty. AIChE J. 57, 2120–2135.
Li, X., Chen, Y., Barton, P.I., 2012. Nonconvex generalized benders decomposition with piecewise convex relaxation for global optimization of integrated process design and operation problems. Ind. Eng. Chem. Res. 51, 7287–7299.
Li, Z., Floudas, C.A., 2014. Optimal scenario reduction framework based on distance of uncertainty distribution and output performance: I. Single reduction via mixed integer linear optimization. Comput. Chem. Eng. 70, 50–66.
Li, Z., Ierapetritou, M., 2008. Process scheduling under uncertainty: review and challenges. Comput. Chem. Eng. 32, 715–727.

Liu, X., Küçükyavuz, S., Luedtke, J., 2016. Decomposition algorithms for two-stage chance-constrained programs. Math. Program. 157, 219–243.
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, 674–699.
McCormick, G.P., 1976. Computation of global solutions to factorable nonconvex programs: Part I convex underestimating problems. Math. Program. 10, 147–175.
Mouret, S., Grossmann, I.E., Pestiaux, P., 2011. A new Lagrangian decomposition approach applied to the integration of refinery planning and crude-oil scheduling. Comput. Chem. Eng. 35, 2750–2766.
Nasab, F.M., Li, Z., 2021. Multistage adaptive stochastic mixed integer optimization through piecewise decision rule approximation. Comput. Chem. Eng. 149, 107286.
Nemirovski, A., Shapiro, A., 2006. Convex approximation of chance constrained programs. SIAM J. Optimiz. 17, 969–996.
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.
Prékoba, A., 1995. Stochastic Programming. Kluwer Academic Publishers, Netherlands.
Rahal, S., Li, Z., Papageorgiou, D.J., 2022. Deep lifted decision rules for two-stage adaptive optimization problems. Comput. Chem. Eng. 159, 107661.
Sahinidis, N.V., 2004. Optimization under uncertainty: state-of-the-art and opportunities. Comput. Chem. Eng. 28, 971–983.
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.
Xu, D., Chen, Z., Yang, L., 2012. Scenario tree generation approaches using K-means and LP moment matching methods. J. Comput. Appl. Math. 236, 4561–4579.
Yang, Y., 2019. Improved benders decomposition and feasibility validation for two-stage chance-constrained programs in process optimization. Ind. Eng. Chem. Res. 58, 4853–4865.
Yang, Y., 2023. Optimal blending under general uncertainties: A chance-constrained programming approach. Comput. Chem. Eng. 171, 108170.
Yang, Y., Barton, P., 2016. Integrated crude selection and refinery optimization under uncertainty. AIChE J. 62, 1038–1053.
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., Sutanto, C., 2019. Chance-constrained optimization for nonconvex programs using scenario-based methods. ISA Trans. 90, 157–168.
Zhang, Q., Morari, M.F., Grossmann, I.E., Sundaramoorthy, Pinto, A.J.M., 2016. An adjustable robust optimization approach to scheduling of continuous industrial processes providing interruptible load. Comput. Chem. Eng. 86, 106–119.