Petroleum Science 19 (2022) 2497–2517

Contents lists available at ScienceDirect

Petroleum Science

journal homepage: www.keaipublishing.com/en/journals/petroleum-science

Original Paper

Two-stage robust optimization of power cost minimization problem in gunbarrel natural gas networks by approximate dynamic programming

Yi-Ze Meng, Ruo-Ran Chen, Tian-Hu Deng*

Department of Industrial Engineering, Tsinghua University, Beijing, 100084, China

ARTICLE INFO

Article history:
Received 21 May 2021
Accepted 3 September 2021
Available online 13 January 2022

Edited by Xiu-Qiu Peng

Keywords:
Natural gas
Gunbarrel gas pipeline networks
Robust optimization
Approximate dynamic programming

ABSTRACT

In short-term operation of natural gas network, the impact of demand uncertainty is not negligible. To address this issue we propose a two-stage robust model for power cost minimization problem in gunbarrel natural gas networks. The demands between pipelines and compressor stations are uncertain with a budget parameter, since it is unlikely that all the uncertain demands reach the maximal deviation simultaneously. During solving the two-stage robust model we encounter a bilevel problem which is challenging to solve. We formulate it as a multi-dimensional dynamic programming problem and propose approximate dynamic programming methods to accelerate the calculation. Numerical results based on real network in China show that we obtain a speed gain of 7 times faster in average without compromising optimality compared with original dynamic programming algorithm. Numerical results also verify the advantage of robust model compared with deterministic model when facing uncertainties. These findings offer short-term operation methods for gunbarrel natural gas network management to handle with uncertainties.

1. Introduction

China has committed to peak carbon dioxide emissions before 2030 and achieve carbon neutrality before 2060. Natural gas is playing an important role in optimizing current energy consumption structure and it is a strategic replacement of crude oil (Li et al., 2016; Liang et al., 2019; Zhao et al., 2019; Huang et al., 2019) to reduce carbon emissions. To rationalize the natural gas availability, long-distance natural gas network is built across China through the West-East Gas Transmission Project, etc (Ma and Li, 2010). Natural gas network is one of the largest and most complex nonlinear systems in the world. Operating the natural gas network properly has attracted attention from industry and academia.

However, compared with crude oil, natural gas shortage problem is greater as the transportation cost is greater and as a result of abnormal weather conditions (Lin and Wang, 2012). Take the "Gas shortage" in winter of 2017 for example, acute natural gas shortages severely influenced residents' heating and enterprises' energy supply in Hebei Province of China. In short-term operation of natural gas network, other factors that result in customer demand varying in unpredictable pattern include uncertain market and oil prices (Hellemo et al., 2012; Ríos-Mercado and Borraz-Sánchez, 2015, Pan et al., 2020).

Operation management under the guidance of deterministic model will fail to perform well under uncertainties. The pipelines in the network usually have a long distance, a change at one end may need a certain amount of time to be transmitted to the other end. If the planning decision is made based on prediction value, when the real customer demand is revealed, previous planning decisions may be sub-optimal or even infeasible and the system takes time to react. Small disturbances may cause serious operation problems. Handling with uncertainties in this field is quite challenging (Ríos-Mercado and Borraz-Sánchez, 2015) and meaningful.

Robust optimization aims to find a solution that optimizes the worst-case cost. Thus no matter what the uncertainty realization is, the performance is acceptable. Acceptable here has two meanings: the decision should be feasible in all possible realizations and the worst-case cost is optimal. By this method we can respond to a variety of unexpected situations. What's more, compared with

* Corresponding author.
E-mail address: deng13@mail.tsinghua.edu.cn (T.-H. Deng).

https://doi.org/10.1016/j.petsci.2021.09.048

stochastic programming and chance-constraint programming, robust optimization needs not assuming distributional information of the uncertain parameters. It is more applicable in practice when the accurate distributions of uncertain parameters are not clear.

We establish a two-stage robust optimization model considering uncertain volumetric flow rate demands at each intermediate node. In the two-stage robust optimization problem, the first-stage decision is made before the uncertainty is revealed and cannot be adjusted within a certain time. However, the worst case is a rare occurrence. Considering it as no difference between other much common cases will result in over-conservatism. Methods to overcome this problem include utilizing the probability information to construct the uncertainty set, introducing a budget parameter to adjust the degree of conservatism. In this paper, a budget parameter is introduced to control the conservatism since it is rare that all the demands reach the maximal deviation simultaneously.

The topology of natural gas transmission network has three types: gunbarrel structure, tree structure and cyclic structure, as shown in Fig. 1. In our study, we assume gunbarrel network structure. Though it is the simplest type of the three, it is an important building block of tree structured and cyclic structured networks. Focusing on gunbarrel network is meaningful for natural gas network applications (Deng et al., 2019). What's more, many important natural gas networks are gunbarrel in China such as the transmission pipeline from central Asia to China (Hu, 2014). We focus on transmission network in which a large volume of gas is moved at high pressures from suppliers to regional demand points. Appropriate operation of compressor stations in the network is crucial to satisfy customer demands while at the same time optimizing the power cost.

During solving the two-stage robust model in gunbarrel network, we encounter a *maxmin* subproblem (SP), which is not common in natural gas optimization research field. Solving this non-convex bilevel programming problem is challenging. We take advantage of the separable structure of the problem and express it as a multi-dimensional dynamic programming formulation. To accelerate the calculation, we propose multi-dimensional approximate dynamic programming algorithms.

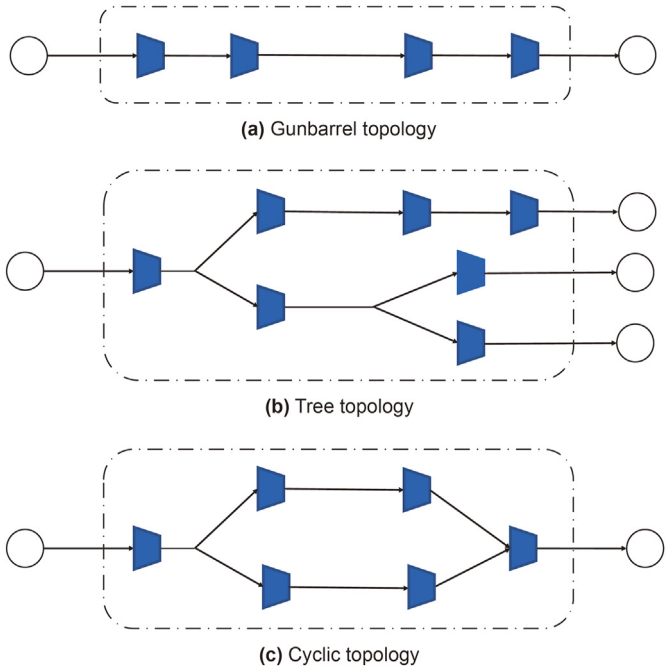

Fig. 1. Three types of pipeline network topologies.

Approximate dynamic programming has recently drawn attention from academia and industry. In artificial intelligence/computer science communities it is known as *Reinforcement Learning*, while in the control theory related field it is known as *Neuro-Dynamic Programming*. From the perspective of operation research, it emphasizes more on the high-dimensional problems (Powell, 2009). The method can focus on reducing the number of states or approximating the value function and policy function. In solving the multi-dimensional *maxmin* problem by approximate dynamic programming in our problem, two techniques are proposed: neighborhood search and aggregation.

The idea of neighborhood search is that adjacent states in dynamic programming should have similar optimal corresponding states in previous stage. Deng et al. (2019) and Zhang et al. (2020) successfully applied neighborhood search in solving deterministic steady-state and transient-state gunbarrel natural gas network power cost optimization problem. A significant speed improvement was obtained. However, the approximate dynamic programming algorithm and corresponding lemmas were based on one-dimensional case in Deng et al. (2019). We modify and extend them to multi-dimensional cases and give analytical bounds and theoretical justifications in any *N*-dimensional case. We also analyze conditions when the optimality is not compromised by multi-dimensional neighbourhood search.

Aggregation is another powerful technique in approximate dynamic programming with a review in Rogers et al. (1991). Instead of searching in the whole states, we can only consider a collection of "representative" states and apply neighborhood search mentioned above in these states. The new state space is a subset of the original states. It is essentially a piecewise constant approximation to the original cost function. Combining neighborhood search and aggregation, we can accelerate the calculation of dynamic programming within acceptable gaps.

The major contributions of this paper are:

1. From a management perspective, a hybrid method with both proactive and reactive action is provided to cope with uncertainties in short-term operation management of gunbarrel natural gas network, instead of passively reacting when change already happens. By applying two-stage robust optimization model proposed in this paper, the risk of natural gas transportation interruption caused by short-term accidents can be reduced, while the worst-case cost is optimized.
2. Previous research in this field utilizing robust optimization simplified the model by assuming linear relationship or considered only passive network without compressors (Aßmann et al., 2018, 2019). We consider detailed network model in both working domain and power cost function in natural gas compressor stations such that the solution can direct the operation (Deng, 2015). What's more, to reduce the over-conservatism, we introduce an uncertainty budget parameter to form the uncertainty set as the one proposed in Bertsimas and Sim (2004). Based on real gunbarrel network structure in China, we verify the robust model and conduct simulations to show the advantages compared with deterministic formulation model. The simulation results show that robust model ensures feasibility with uncertain demands while the deterministic model fails to be feasible in some simulated uncertainty realizations.
3. In solving the two-stage robust model, the SP is challenging since it is a bilevel problem which is also non-convex and nonlinear. We design appropriate and efficient algorithms to solve it starting with a reformulation to a dynamic programming problem. In multi-dimensional dynamic programming, *curses of dimensionality* are the major challenges which demand

heavily on computing resources, since the number of states increases exponentially as the dimension number increases. By applying the proposed approximate algorithms, the *maxmin* SP can be solved in reasonable time and we obtain a 7 times speed gain in calculation.

Section 2 reviews the relevant literature. Section 3 describes the model formulation. In Section 4, we discuss the algorithm to solve the problem. Section 5 presents numerical experiments to evaluate the algorithm performance and verify the advantages of proposed methods compared with deterministic models. In Section 6, we summarize our conclusions.

2. Literature review

The constraints of this power cost minimization problem contain three parts: the mass balance constraint, the relationship between the pressure drop and the flow represented by a nonlinear constraint and a nonlinear non-convex feasible set of the working region of the flow and pressure in each compressor station. Since the objective function is also non-convex in the flow and the pressure, the original problem is difficult to solve.

Taking into account the decision of switching the compressor brings integer variables. Nguyen et al. (2008) assumed that the switching occurred only at the beginning of each period or hour. Their experiments showed that Mixed Integer Nonlinear Programming (MINLP) model generated the most effective solution compared to the genetic algorithm and expert systems. In Wu et al. (2000) the amount of flow and pressure variables were determined at the first level and then working units were decided at the second level. Cobos-Zaleta and Ríos-Mercado (2002) considered all those variables at the same level and formulated the problem as an MINLP. They applied an outer approximation with equality relaxation and augmented penalty method. However, global optimality was not guaranteed due to the non-convexity. According to the conversations with managers from Chinese natural gas company, the on-off states of each compressor remain unchanged in short-term operations. The switching decisions of compressors are considered as first-stage decisions in this work.

To solve the deterministic fuel cost minimization problem, methods including dynamic programming, gradient-based algorithm, geometric programming approaches and linear approximation approaches are studied. We refer readers to Ríos-Mercado and Borraz-Sánchez (2015) for a comprehensive overview in this field. We mainly focus on formulations for steady networks and related methods.

Dynamic programming has been successfully applied to optimize operation of natural gas network (Borraz-Sánchez and Haugland, 2011; Wong and Larson, 1968). In power cost optimization problem of natural gas station operation, the problem typically has a separable structure. Optimization in single station (Deng, 2016) and in network (Deng et al., 2019) can both be solved by dynamic programming.

Approximate dynamic programming (ADP) is also known as reinforcement learning or neuro-dynamic programming. There are two main classes of ADP: approximation on policy function and approximation on cost/value function. In the first class, a set of functions map the states to the decisions (Sutton et al., 2000; Baxter and Bartlett, 2001). The second class is more common, in which a set of basis functions are utilized to approximate the cost/value functions by such as polynomial functions (Powell, 2007) and Fourier basis (Konidaris et al., 2011). ADP method is successfully applied in solving deterministic model of natural gas network operation. In Deng et al. (2019) and Zhang et al. (2020), approximate dynamic programming based on idea of neighborhood search was applied to operation management in steady state and transient state network.

Approximation with piecewise linear function was studied in De Wolf and Smeers (2000), who mainly handled the nonlinearity relations of flow rate and pressure. Geiβler et al. (2015) considered steady state network. They utilized piece-wise linear approximation and obtained a mix-integer formulation. Martin et al. (2006) studied non-separable functions approximation and proposed methods based on SOS Type 2 constraints. The method was compared with lambda-method and delta-method. In our work, we utilize the separable structure of the problem and combine lambda-method with SP to simplify the calculation.

Xue et al. (2016) and Han et al. (2019) established a natural gas pipeline transmission plan for China National Petroleum Corporation (CNPC). The model specified the amount of flow at each node to minimize the total cost including procurement costs and transportation costs. Their planning model was a multi-period problem with the minimum time unit of either a month or a year. In our model, shorter time unit, a week or a day, is considered. What's more, we consider the uncertainty in demand and supply making the volumetric flow rates uncertain variables.

Optimization with uncertainty adds complexity to the original deterministic formulation. Efforts have been made to consider uncertainty in the natural gas network operation problem (Gotzes et al., 2016; Wintergerst, 2017; Behrooz, 2016) with stochastic programming, chance-constraint programming, etc. Stochastic programming was widely utilized to consider uncertainties of natural gas price and demands (Vahid-Pakdel et al., 2017; Wang et al., 2020) in energy system operation and uncertainties were described and modeled by scenarios with specific predefined probability. Similar with stochastic programming, chance-constraint programming requires determination of the distribution information of uncertain parameters.

Unlike stochastic programming and chance-constraint programming, robust optimization only assumes that the uncertain parameter varies in a given uncertainty set. As for robust optimization related research, we find Aβmann et al. (2018) and Aβmann et al. (2019) who applied two-stage robust optimization in natural gas network operation. However, either the network examined was passive without compressors or compressor model was simplified by linear models in previous research. In this paper, we consider gunbarrel-network optimization with detailed compressor models.

To reduce the over-conservatism of robust optimization, three methods can be utilized. Firstly, budget parameter proposed by Bertsimas and Sim (2003) and Bertsimas and Sim (2004) is able to restrict the maximum number of uncertain parameters that simultaneously reach the worst case. Second, data-driven methods can be used to construct uncertainty sets. Third, adjustable robust optimization allows some decisions to adjust after the uncertainties are revealed. Typically the second stage decision variables were approximated by an affine rule (Ben-Tal et al., 2004) or extended affinely adjustable robust counterparts (Chen and Zhang, 2009). Other solution approximation technique is finite adaptability studied in Hanasusanto et al. (2015) where the decision maker pre-committed to K second-stage decisions. In this paper, we consider two-stage robust optimization with uncertainty budget, where the second stage decision variables keep all the details instead of being approximated by affine rules.

Two-stage robust optimization problems suffer from computation burden compared to the conventional static robust optimization problem. Ben-Tal et al. (2004) showed that general two-stage robust linear programs were NP-hard. The nonlinearity and non-convexity of our problem make it more complicated to compute the solution. We refer readers to Gabrel et al. (2014) and Gorissen et al. (2015) for an overview of the recent development in robust

optimization.

To solve the fully adjustable two-stage robust optimization problem, Benders-dual cutting plane and column-and-constraint generation (C&CG) algorithm are typically used. The C&CG algorithm generates constraints from uncertainty scenarios in a dynamic way, while the Benders method generates constraints based on the dual solution of the SP. See Zeng and Zhao (2013) for more details of this method. Their experiments showed that C&CG performed an order of magnitude faster than the Benders-dual algorithm in the sense that the solution was found with a smaller number of iterations. The method can apply to linear and nonlinear problems.

To solve the *maxmin* problem in second stage, applying dual of the inner problem and Karush-Kuhn-Tucker (KKT) optimal conditions are often used to transfer this bilevel problem into a single-level problem. The former is applicable for linear functions. The latter is only necessary and not sufficient for non-convex inner problems, which may result in local or even suboptimal solutions.

The work most closely related to us where two-stage robust optimization is applied is in Aβmann et al. (2019). They applied two-stage robust optimization to the fuel cost minimization problem considering uncertainty in demand and physical parameters and decided the square pressure difference of the compressors in the first stage, while squared pressure and flow became the second stage decisions. They used simplified compressor model that the cost was independent of the flow through the compressor and was linear in the square pressure difference, enabling them to exploit the specific problem structure and end up with a single stage problem. Compared to our work they used different first-stage variables, different uncertainty sets and much simpler objective functions. The designed approximate dynamic programming algorithm and applications in solving *maxmin* problem in robust optimization of gunbarrel natural gas network operation are also our innovative contributions.

3. Notation and model

In this section, we propose a two-stage robust optimization model of the compressor power cost minimization problem. In our settings, the first stage decisions are the discharge pressure of the compressor station and working status of each compressor. The rotational speed of each compressor and volumetric flow rate allocation can be adjusted once the uncertain flow condition is clear. The explanation of the operation process can be seen in Behrooz (2016). All the notations needed are listed in Table 1.

Moreover, we make following assumptions in the natural gas transmission network:

(A1) The transmission network has N stations overall, with K_n non-identical compressors installed in parallel at station n. Intermediate nodes may exist between stations and uncertain demands are extracted from these nodes.

(A2) The total gas inflow of a compressor station is equal to the total outflow since the natural gas is not used to power the compressors.

(A3) The network is in a steady state, i.e., velocity, pressure and the stream's cross-section vary from point to point but are time-independent.

In all the intermediate nodes, there is gas flow into or out of the nodes, which is uncertain as well as the original flow into the network dQ_0. Assuming that there are L pipelines, we use $\boldsymbol{Q} = (Q_1, Q_2, \cdots, Q_L)^{\mathrm{T}}$ to denote the uncertain variable volumetric flow rates passing through each pipeline. Let $l(n^+)$, $l(n^-)$ be the pipeline connecting the compressor n outlet and inlet, respectively. If there is no intermediate node between station $n-1$ and station n, we have $Q_{l((n-1)^+)} = Q_{l(n^-)}$, otherwise, $Q_{l((n-1)^+)}$ and $Q_{l(n^-)}$ may be different due to the uncertain flow in the intermediate node. Let dQ_m denote the flow rate withdrew by intermediate node m, then $Q_{l((n-1)^+)} - \sum_{m\in M(n-1,n)} dQ_m = Q_{l(n^-)}$, where $M(n-1,n)$ is the set of all intermediate nodes between station $n-1$ and n.

Since the demands are uncertain, dQ_m varies around the nominal value in the range of $\left|\overline{dQ}_m - dQ_m\right| \le \delta d\breve{Q}_m$. The maximum variation is expressed as the fraction δ of $d\breve{Q}_m$, where $0 \le \delta \le 1$. We denote the uncertainty budget parameter as Γ, where Γ is a predefined integer denoting the maximum number of nodes reaching the maximal deviation simultaneously. Following the formulation in Bertsimas and Sim (2004), the uncertainty set can be represented as

$$\boldsymbol{U} = \left\{ d\boldsymbol{Q} = (dQ_0, dQ_1 \ldots dQ_m) : \left|\overline{dQ}_m - dQ_m\right| \le \delta d\breve{Q}_m,\ \sum_{m\in[M]} \frac{\left|\overline{dQ}_m - dQ_m\right|}{d\breve{Q}_m} \le \Gamma,\ \forall m \in [M] \right\}.$$

To construct uncertainty set $\boldsymbol{U}$ in practice, the varying range of volumetric flow rate can be obtained based on prediction. The predicted value can be considered as nominal value $\overline{dQ}_m$. Based on average historical prediction accuracy, we can define the possible range of demand volumetric flow rate and uncertainty budget parameter.

The details of power cost minimization problem of a compressor station are shown in Appendix A. An example of working domain of a compressor is shown in Fig. 2. Objective function and constraints of the deterministic model in a compressor are highly non-convex and nonlinear. What's more, in natural gas network, the pressure drop in a pipeline is related to the volumetric flow rate in complex form (Xue et al., 2016).

As for the robust optimization, the first-stage decisions are the discharge pressure of each compressor station and on-off states of compressors in each station. Once the uncertain volumetric flow rate is revealed, suction pressure can be determined and allocation of flow rate in each compressor station can be adjusted by controlling the rotational speed of each compressor, which act as second-stage decision variables. The second stage decisions should optimize cost under given first-stage decisions and uncertainty.

Denote by $\varphi_{n,k}$ the mapping from each pair of suction pressure, suction temperature of station n $(p_n^{\mathrm{s}}, T_n^{\mathrm{s}})$ to the set of feasible discharge pressure, discharge temperature of station n and volumetric flow rate allocation in station n $(p_n^{\mathrm{d}}, T_n^{\mathrm{d}}, Q_{n,k})$. Then the working domain of compressor unit k in station n can be expressed as:

Table 1
Notation and symbols.

Exogenous parameters		
N	=	Total number of compressor stations
K_n	=	Total number of compressors at compressor station n, and let $[K_n] = \{1, 2, \ldots, K_n\}$ be the index set of compressors at station n, $n \in [N]$
$\overline{Q}_n$	=	Nominal value of the volumetric flow rate at station n, $n \in [N]$, kg/s
T_n^{s}/ T_n^{d}	=	Suction/discharge temperature at compressor station n, $n \in [N]$, K
$\mathscr{P}_0^{\mathrm{d}}$	=	Suction pressure at the initial pipeline segment, MPa
$p_n^{\mathrm{d,max}}$	=	Maximum required suction pressure at station n, $n \in [N]$, MPa
$p_n^{\mathrm{d,min}}$	=	Minimum required suction pressure at station n, $n \in [N]$, MPa
$\widehat{a}_{j,n,k}$	=	Constant values for modeling the surge and stonewall lines of compressor k in station n, $1 \le j \le 6$, $k \in [K_n]$, $n \in [N]$
$\widehat{b}_{j,n,k}$	=	Constant values for modeling isentropic head and efficiency of compressor k in station n, $1 \le j \le 6$, $k \in [K_n]$, $n \in [N]$
R	=	Gas constant, MJ/kg · K
Z_n	=	Compressibility factor at the suction side of compressor station n, $n \in [N]$
σ	=	Isentropic exponent
δ	=	The maximum variation fraction of the volumetric flow rate
Γ	=	The uncertainty budget parameter
Intermediate variables		
$f_{n,k}$	=	Mass flow rate of compressor k at station n, $k \in [K_n]$, $n \in [N]$, kg/s
$H_{n,k}$	=	Isentropic head of compressor k at station n, $k \in [K_n]$, $n \in [N]$, MJ/kg
$s_{n,k}$	=	Rotational speed of compressor k at station n, $k \in [K_n]$, $n \in [N]$, r/s
$\eta_{n,k}$	=	Isentropic efficiency of compressor k at station n, $k \in [K_n]$, $n \in [N]$
p_n^{s}	=	Suction pressure at compressor station n, $n \in [N]$, MPa
First-stage decision variables		
$\mathscr{S}_n$	=	The set of switched-on compressors in compressor station n, $n \in [N]$
p_n^{d}	=	Discharge pressure at compressor station n, $n \in [N]$, MPa
Second-stage decision variables		
$Q_{n,k}$	=	Actual volumetric flow rate passing through compressor k at station n, $k \in [K_n]$, $n \in [N]$, m^3/s

$$(p_n^{\mathrm{s}}, T_n^{\mathrm{s}}) \in \varphi_{n,k}\left(p_n^{\mathrm{d}}, T_n^{\mathrm{d}}, Q_{n,k}\right).$$

Next we consider the relationships between station $n-1$ and n. Let vector $\boldsymbol{Q}_{(\boldsymbol{n}-1,\boldsymbol{n})}$ denotes volumetric flow rates in all pipelines between station $n-1$ and n. Let g_n be the function that computes suction pressure and temperature of compressor station n from discharge pressure and temperature of compressor $n-1$ in gunbarrel network. Thus g_n calculates $(p_n^{\mathrm{s}}, T_n^{\mathrm{s}})$ from $(p_{n-1}^{\mathrm{d}}, T_{n-1}^{\mathrm{d}})$. Then, we have:

$$(p_n^{\mathrm{s}}, T_n^{\mathrm{s}}) = g_n\left(p_{n-1}^{\mathrm{d}}, T_{n-1}^{\mathrm{d}}, \boldsymbol{Q}_{(\boldsymbol{n}-1,\boldsymbol{n})}\right).$$

We assume that the network starts with a pipeline. The suction pressure $\mathscr{P}_0^{\mathrm{d}}$ and temperature $\mathscr{T}_0^{\mathrm{d}}$ of the first pipeline are given, then we have:

$$(p_1^{\mathrm{s}}, T_1^{\mathrm{s}}) = g_1\left(\mathscr{P}_0^{\mathrm{d}}, \mathscr{T}_0^{\mathrm{d}}, \boldsymbol{Q}_{(\boldsymbol{0},1)}\right).$$

Denote the set of switched-on compressors in compressor station n as $\mathscr{S}_n$, where $\mathscr{S}_n \in \mathscr{S}$, then the two-stage robust optimization model for the fuel cost minimization with N compressor stations is:

$$\min_{p_n^{\mathrm{d}}, \mathscr{S}_n} \max_{d\boldsymbol{Q} \in \boldsymbol{U}p_n^{\mathrm{s}}, Q_{n,k}} \min : \sum_{n=1}^{N} \sum_{k \in \mathscr{S}_n} P_{n,k}\left(p_n^{\mathrm{s}}, T_n^{\mathrm{s}}, p_n^{\mathrm{d}}, Q_{n,k}, \mathscr{S}_n\right) \quad \text{s.t.} \left(p_n^{\mathrm{s}}, T_n^{\mathrm{s}}\right) \in \varphi_{n,k}\left(p_n^{\mathrm{d}}, T_n^{\mathrm{d}}, Q_{n,k}\right), \quad \forall n \in \left[N\right], k \in \mathscr{S}_n,$$
$$\left(p_n^{\mathrm{s}}, T_n^{\mathrm{s}}\right) = g_n\left(p_{n-1}^{\mathrm{d}}, T_{n-1}^{\mathrm{d}}, \boldsymbol{Q}_{(\boldsymbol{n}-1,\boldsymbol{n})}\right), \quad \forall n \in \left[N\right] \backslash \left\{1\right\},$$
$$Q_{l((n-1)^+)} - \sum_{m \in M(n-1,n)} dQ_m = Q_{l(n^-)}, \quad \forall n \in \left[N\right] \backslash \left\{1\right\}, \sum_k Q_{n,k} = Q_{l(n^-)} = Q_{l(n^+)}, \quad \forall n \in \left[N\right], k \in \mathscr{S}_n, \left(p_1^{\mathrm{s}}, T_1^{\mathrm{s}}\right) = g_1\left(\mathscr{P}_0^{\mathrm{s}}, \mathscr{T}_0^{\mathrm{s}}, \boldsymbol{Q}_{(\boldsymbol{0},1)}\right) \tag{1}$$

4. Methods

In this part, before we demonstrate the C&CG algorithm used to solve the two-stage robust optimization problem and present the formulation of master problem (MP) and SP to be solved, we first introduce the same assumption in Deng et al. (2019) which can reduce the dimensions of the decision variables. The assumption asserts that there is only one feasible discharge temperature corresponding to each combination of suction pressure, discharge

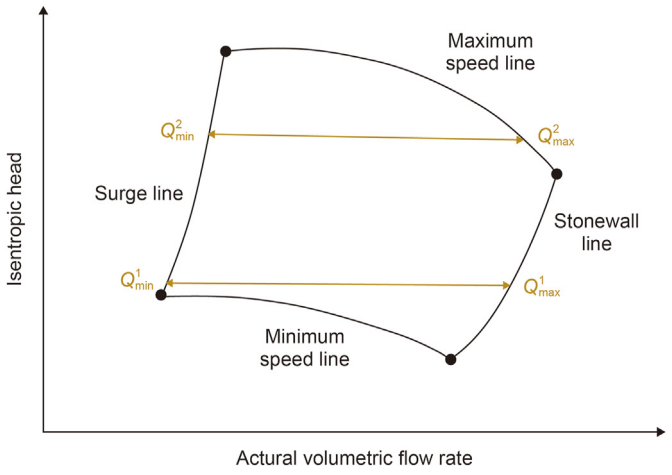

Fig. 2. The working domain of a compressor.

they both consist of a MP and SPs. However, they are different in the cutting strategy. The basic idea of the algorithm is that only a small subset of all the possible realizations of uncertainty would play a significant role in defining an optimal solution. The C&CG algorithm generates new constraints related to individual worst-case scenarios and adds them into the MP iteratively, thus successively computing stronger lower bounds of the optimal objective function value.

We index the selected realizations of the demand uncertainty in each iteration by j. The same index is also applied to corresponding second stage decisions. Here selected uncertainty scenarios are given. Then the formulation of MP is:

$$
\begin{aligned}
&\min_{p_n^{\mathrm{d}},\mathscr{S}_n,\eta}\ \eta\\
&\text{s.t. } \eta \geq \sum_{n=1}^{N}\sum_{k\in\mathscr{S}_n} P_{n,k}\left(p_n^{\mathrm{s},j},T_n^{\mathrm{s}},p_n^{\mathrm{d}},Q_{n,k}^{j},\mathscr{S}_n\right),\ \forall j\in J, n\in\left[N\right], k\in\mathscr{S}_n,\ \left(p_n^{\mathrm{s},j},T_n^{\mathrm{s}}\right)\in\varphi_{n,k}\left(p_n^{\mathrm{d}},T_n^{\mathrm{d}*}\left(p_n^{\mathrm{d}}\right),Q_{n,k}^{j}\right),\ \forall j\in J, n\in\left[N\right], k\in\mathscr{S}_n,\\
&\left(p_n^{\mathrm{s},j},T_n^{\mathrm{s}}\right)=g_n\left(p_{n-1}^{\mathrm{d}},T_{n-1}^{\mathrm{d}*}\left(p_{n-1}^{\mathrm{d}}\right),\widehat{\boldsymbol{Q}}^{\boldsymbol{j}}_{(\boldsymbol{n}-1,\boldsymbol{n})}\right),\ \forall j\in J, n\in\left[N\right]\backslash\left\{1\right\}, \widehat{Q}^{j}_{l((n-1)^{+})}\\
&-\sum_{m\in M(n-1,n)} d\widehat{Q}_m^{j}=\widehat{Q}^{j}_{l(n^{-})},\ \forall j\in J, n\in\left[N\right]\backslash\left\{1\right\},\ \sum_k Q_{n,k}^{j}=\widehat{Q}^{j}_{l(n^{-})}=\widehat{Q}^{j}_{l(n^{+})},\ \forall j\in J, n\in\left[N\right]\backslash\left\{1\right\},\ \left(p_1^{\mathrm{s},j},T_1^{\mathrm{s}}\right)=g_1\left(\mathscr{P}_0^{\mathrm{s}},\mathscr{T}_0^{\mathrm{s}},\widehat{\boldsymbol{Q}}^{\boldsymbol{j}}_{(\boldsymbol{0},1)}\right),\ \forall j\in J.
\end{aligned}
\tag{2}
$$

pressure and suction temperature: **Assumption 1.** *For any compressor station n, $1 \leq n \leq N$, the discharge temperature T_n^{d} can be uniquely determined from suction pressure p_n^{s}, suction temperature T_n^{s} and discharge pressure p_n^{d}.* Thus we can approximate T_n^{d} to reduce the dimension of the decision variables. Since the suction temperature of each station is considered as given in this work, we only need to consider decisions of pressure value and working status of compressors as first-stage variables.

4.1. C&CG algorithm to solve the two-stage robust model

The C&CG method is similar to the Benders decomposition since

The MP formulation (2) is non-convex and nonlinear, which can be approximated by Mixed Integer Linear Programming (B2) in Appendix B. The solution of MP will be the first stage decisions which are fixed in the subsequent SP. The SP finds the corresponding upper bound over all scenarios in uncertainty sets. If the first stage decision is infeasible, the SP returns a positive infinite value. If it is feasible, then the SP returns the worst objective value under uncertainty and the corresponding solutions and new constraints are added to the MP that should be considered in the next iteration. The algorithm terminates if the gap between the upper and lower bounds is within a predefined tolerance ε. The SP formulation $y(\widehat{p}_n^{\mathrm{d}},\widehat{\mathscr{S}}_n)$ with given $\widehat{p}_n^{\mathrm{d}},\widehat{\mathscr{S}}_n$ is:

$$
\begin{aligned}
&y\left(\widehat{\boldsymbol{p}}^{\mathrm{d}},\widehat{\mathscr{S}}_n\right)=\max_{d\boldsymbol{Q}\in\boldsymbol{U}p_n^{\mathrm{s}},Q_{n,k}}\min\sum_{n=1}^{N}\sum_{k\in\widehat{\mathscr{S}}_n} P_{n,k}\left(p_n^{\mathrm{s}},T_n^{\mathrm{s}},\widehat{p}_n^{\mathrm{d}},Q_{n,k},\widehat{\mathscr{S}}_n\right), s.t.\ \left(p_n^{\mathrm{s}},T_n^{\mathrm{s}}\right)\in\varphi_{n,k}\left(\widehat{p}_n^{\mathrm{d}},T_n^{\mathrm{d}*}\left(\widehat{p}_n^{\mathrm{d}}\right),Q_{n,k}\right),\ \ \forall n\in\left[N\right], k\in\widehat{\mathscr{S}}_n,\\
&\left(p_n^{\mathrm{s}},T_n^{\mathrm{s}}\right)=g_n\left(\widehat{p}_{n-1}^{\mathrm{d}},T_{n-1}^{\mathrm{d}*}\left(\widehat{p}_{n-1}^{\mathrm{d}}\right),\mathbf{Q}_{(\boldsymbol{n}-1,\boldsymbol{n})}\right),\ \ \forall n\in\left[N\right]\backslash\left\{1\right\},\\
&Q_{l((n-1)^{+})}-\sum_{m\in M(n-1,n)} dQ_m=Q_{l(n^{-})},\ \ \forall n\in\left[N\right]\backslash\left\{1\right\},\ \ \sum_k Q_{n,k}=Q_{l(n^{-})}=Q_{l(n^{+})},\ \ \forall n\in\left[N\right], k\in\widehat{\mathscr{S}}_n,\ \ \left(p_1^{\mathrm{s}},T_1^{\mathrm{s}}\right)=g_1\left(\mathscr{P}_0^{\mathrm{s}},\mathscr{T}_0^{\mathrm{s}},\mathbf{Q}_{(\boldsymbol{0},1)}\right)
\end{aligned}
\tag{3}
$$

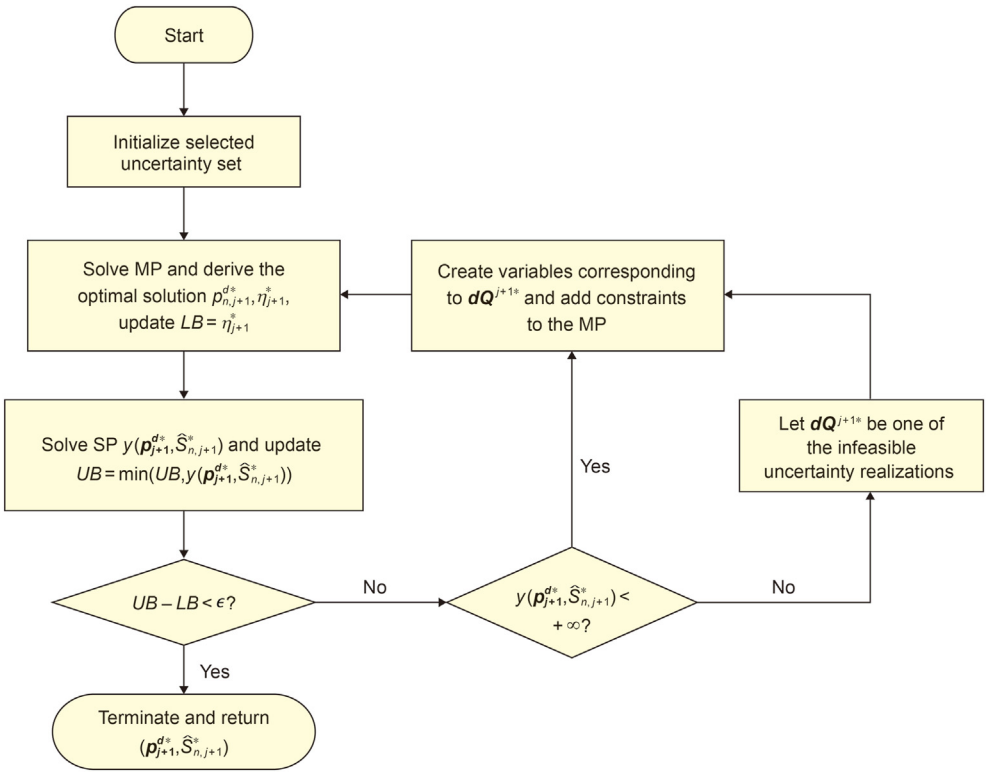

Fig. 3. The flowchart of C&CG algorithm.

Next, we explain the main steps of C&CG algorithm as shown in Fig. 3.

1. In the initialization, set $LB = -\infty$, $UB = +\infty$, iteration index $j = 0$, initialize J.
2. The optimal solutions $p^{d*}_{n,j+1}$, η^*_{j+1} of MP are decided, which are candidate values of first-stage decisions. Only finite realizations of uncertainties are considered in set J. The corresponding recourse decisions $Q^{j*}_{n,k}$, $p^{s,j}_n \cdots$ for all j in the set J are also calculated.
3. Solutions of MP are passed to SP in which the worst-case under given first-stage decisions are defined. The solution of any SP is an upper bound of the robust model. To ensure the upper bound is non-increasing, the upper bound only needs to be updated when the current optimal value of SP is smaller.
4. If gap between upper bound and lower bound is small enough, the algorithm terminates. Otherwise, create recourse decision related variables and add all the related constraints to the MP with the corresponding uncertainty realization $\boldsymbol{dQ}^{j+1*}$ being the optimal scenario in solution of $y(\boldsymbol{p}^{d*}_{j+1}, \widehat{\mathscr{T}}^*_{n,j+1})$. In addition, add feasibility/optimality cuts in Appendix C to MP. Update $j = j+1$, $J = J \cup \{j+1\}$ and go to step 2.

It should be noticed that the solution of simple linear bilevel programming problem is NP-hard (Bard, 1991; Hansen et al., 1992). As proved by Zeng and Zhao (2013), Zhao and Zeng (2012), the C&CG algorithm converges to the optimal solution in finite

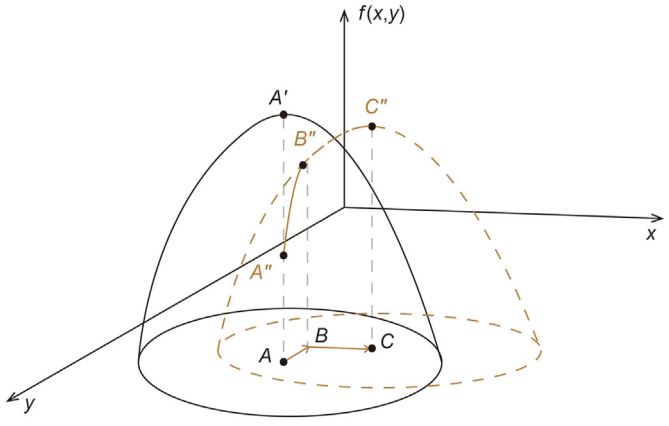

Fig. 4. The neighborhood search in adjacent states.

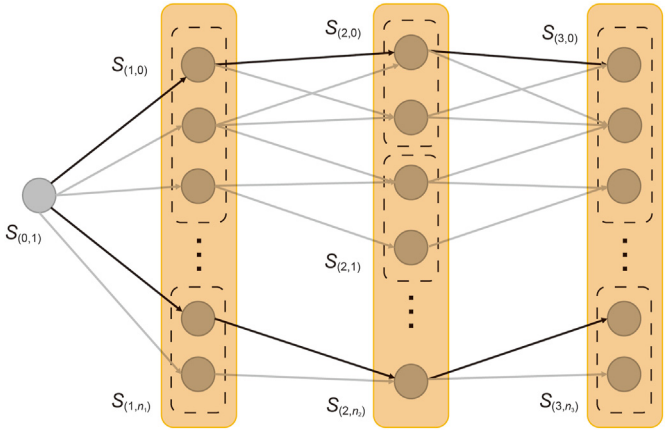

Fig. 5. The state aggregation in dynamic programming.

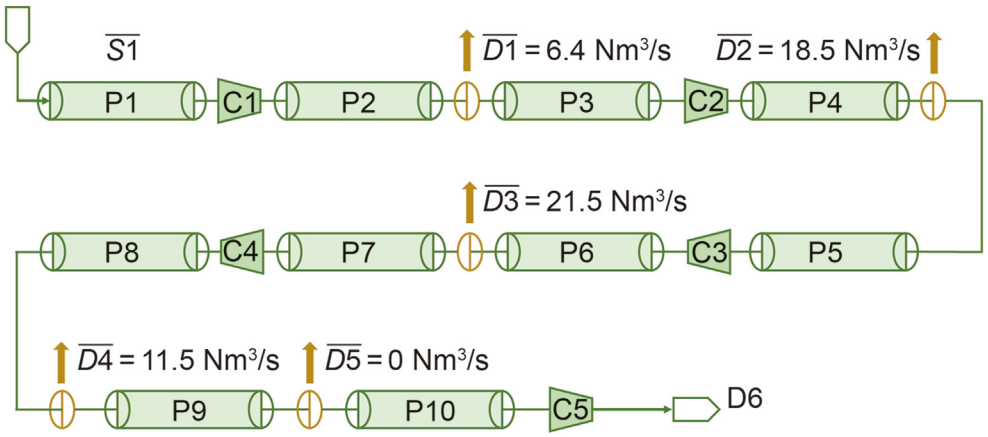

Fig. 6. The structure of selected network and nominal demands.

iterations for the polyhedral uncertainty sets as long as the MP and SP can be solved to their global optimal solutions when the relatively complete recourse assumption holds and the model is linear with both linear programming recourse and mixed-integer programming recourse problems.

4.2. Approximate dynamic programming algorithm to solve the SP

In SP of the C&CG algorithm, the discharge state of each stage $\widehat{p}_n^{\mathrm{d}}$ and working status of each compressor $\widehat{\mathscr{T}}_n$ are given, the objective is to search for the volumetric flow rate realizations such that the total optimization results are maximized. We first reformulate it as dynamic programming problem in Appendix C.

Next, we propose approximate dynamic programming methods to accelerate the calculation of SP. The approximate dynamic programming approach contains two main ideas: neighborhood search and aggregation. We illustrate the two approximate methods respectively, starting from the neighborhood search.

The neighborhood search idea is that when searching the optimal corresponding state in stage $n-1$ for a given state in stage n, instead of searching the whole region of states in stage $n-1$, we only need to search in a smaller range, since we have already known the optimal state for an adjacent state of the given state in stage n. Adjacent states of stage n should have optimal corresponding previous states close to each other. See Fig. 4 for an illustration of this idea. Take two-dimensional maximal problem with quasi-concave objective function as an example. The solid line is the function searching for optimal corresponding state $n-1$ for a given state S_n of stage n. The solution is point A with objective point A'. The dotted line is the corresponding function for an adjacent state S_n'. The search starts from also point A with objective point A''. It first searches in a direction parallel to the y axis and find a local optimal point B, and then searches in a direction parallel to the x axis. The change of direction may repeat until no updates are gained. By this scheme, the search only takes a few steps to find the optimal point C. We provide the details of this neighborhood search in Appendix D.

Another approximation method is aggregation. The idea is shown in Fig. 5. Each grey node denotes a state in the dynamic programming. The grey line with arrow denotes the state transition between nodes. An original dynamic programming with full states calculates all the grey nodes. The idea of aggregation is that we only search a subset of the original states. All the states in each subset are considered to have the same cost function and some of the transition between states is discarded. We list one possible aggregation in Fig. 5 denoted by black dotted frame. Each frame aggregates all the states included and simply replaces the cost function and state transition function by those of first stage in the subset. For example, $S_{(2,0)}$ denotes the first aggregated state in stage 2. The black lines denote all the left state transition between aggregated states. The aggregation method may also be modified to include more states, transitions and to get more accurate approximation. The states number and state transition number thus are reduced significantly. The aggregation should put similar states in a subset, otherwise the solution accuracy might be affected.

Noticing that the state in our problem is composed of volumetric flow rate vector $\tilde{\mathbf{Q}}_n$ and current total uncertainty deviation ratio γ_n. We choose to aggregate the states in the dimension of γ_n. However, we can not directly aggregate all the states with same $\tilde{\mathbf{Q}}_n$ and different γ_n and keep only one state. In that case too much possible transition is discarded. We still need to keep several states with the same $\tilde{\mathbf{Q}}_n$. The size of the subset should be controlled and tested carefully to keep the gap small and acceptable.

We provide details of the proposed DP algorithm pseudocode in appendix E. The core approximate dynamic programming Algorithm 1 utilizes the idea of PROPOSITION 1. The idea is that if for a state S_n in stage n, the corresponding optimal state $S_{n-1}^*(S_n)$ in stage $n-1$ is known. Then for a neighboring state S_n' in stage n, the corresponding optimal state may be in the neighboring region of $S_{n-1}^*(S_n)$. In multi-dimensional cases, for k_{th} dimension, the search direction can be set as the direction parallel to the k_{th}-dimensional coordinate axis. After searching forward and backward in k_{th} dimension and find an optimal point, the search starts from this optimal point and continue in the $k+1_{\mathrm{th}}$ direction. After searching in the last dimension, the search direction is changed to the direction parallel to the first dimensional coordinate axis again and a new cycle is started. The search ends when the gap of two optimal values of adjacent directions is less than ε.

5. Numerical experiment

In this part, we conduct numerical experiments based on real pipeline networks in China with intermediate nodes between two adjacent stations and five compressor stations. The structure is extracted from Deng et al. (2019). We do not use real data due to confidentiality reasons. The gunbarrel structure and nominal demand at each intermediate node are shown in Fig. 6.

The original subnetwork from Deng et al. (2019) contains one compressor in each station, the structure of which is denoted as setting 1. To illustrate that the proposed methods can be applied to

Table 2
Values of the suction temperature and first-stage decision variables in test 1.

Setting	Index	Suction Temperature, K	Discharge Pressure, MPa
1	1	[299.27, 288.36, 304.98, 291.05, 300.07, 295.58]	[9.8, 8.6, 9.6, 8.8, 9.1, 8.1]
	2	[299.27, 288.36, 304.98, 291.05, 300.07, 295.58]	[9.8, 8.6, 9.4, 8.6, 9.3, 8.4]
	3	[296.31, 284.14, 297.23, 289.12, 298.86, 292.24]	[9.8, 8.6, 9.6, 8.8, 9.1, 8.1]
	4	[296.31, 284.14, 297.23, 289.12, 298.86, 292.24]	[9.8, 8.6, 9.4, 8.6, 9.3, 8.4]
2	5	[299.27, 288.36, 304.98, 291.05, 300.07, 295.58]	[9.8, 8.4, 9.7, 8.7, 9.5, 7.6]
	6	[299.27, 288.36, 304.98, 291.05, 300.07, 295.58]	[9.8, 8.4, 9.7, 8.6, 9.3, 7.8]
	7	[296.31, 284.14, 297.23, 289.12, 298.86, 292.24]	[9.8, 8.4, 9.7, 8.7, 9.5, 7.6]
	8	[296.31, 284.14, 297.23, 289.12, 298.86, 292.24]	[9.8, 8.4, 9.7, 8.6, 9.3, 7.8]

Table 3
Computation time and objective values with varying first-stage decisions.

Index	CPU, s				Objective			
	DP	NS-ONLY	AG-ONLY	AG-NS	DP	NS-ONLY	AG-ONLY	AG-NS
1	357.38	163.30	179.88	97.85	59.11	59.11	59.11	59.11
2	94.98	48.57	51.10	29.41	60.50	60.50	60.50	60.50
3	95.59	48.84	53.28	26.93	57.39	57.39	57.39	57.39
4	95.28	48.16	51.49	26.36	59.37	59.37	59.37	59.37
5	493.48	255.09	275.78	141.90	78.12	78.12	78.12	78.12
6	550.35	283.39	309.46	156.62	84.31	84.31	84.31	84.31
7	497.86	256.00	278.60	151.12	76.29	76.29	76.29	76.29
8	568.07	280.35	307.06	158.27	82.43	82.43	82.43	82.43

Table 4
Computation time and objective values with varying uncertainty budget parameters.

Setting	Γ	CPU, s				Objective			
		DP	NS-ONLY	AG-ONLY	AG-NS	DP	NS-ONLY	AG-ONLY	AG-NS
1	1	106.01	47.99	52.12	26.78	58.52	58.52	58.52	58.52
	2	357.38	163.3	179.88	97.85	59.11	59.11	59.11	59.11
	3	697.66	458.48	367.58	240.29	59.67	59.67	59.67	59.67
	4	819.52	799.16	517.97	416.58	60.18	60.18	60.18	60.18
2	5	151.59	75.57	86.02	46.71	77.31	77.31	77.31	77.31
	6	493.48	255.09	275.78	141.90	78.12	78.12	78.12	78.12
	7	890.10	645.58	503.22	352.06	78.63	78.63	78.63	78.63
	8	1103.49	990.77	648.10	570.12	79.26	79.26	79.26	79.26

Table 5
Values of the volumetric flow rate ranges of four settings.

Index	volumetric flow rate ranges, Nm^3/s
a	[4, 4, 4, 6, 8, 8]
b	[4, 4, 6, 10, 6, 8]
c	[4, 2, 4, 6, 4, 4]
d	[6, 2, 6, 4, 4, 2]

more general case, we also modify the component of each compressor station to include nonidentical compressors in each station, the structure of which is denoted as setting 2. The details of the two settings are listed in Appendix G. The source flow rate for setting 1 is 599.6 Nm^3/s and the source flow rate for setting 2 is 660 Nm^3/s, where Nm^3/s is unit of the normal volumetric flow rate, that is, the mass flow rate over the density under standard conditions. The discretization length of pressure is 0.1 MPa. We include the following numerical experiments: (1) benefits of the approximate dynamic programming algorithm; (2) results of robust model and comparisons with deterministic model results and comparison of results between different uncertainty budgets. The codes were developed using C++. The experiments were performed on an Intel Core i90-9900K 3.60 GHz CPU. Models were solved by commercial software Gurobi 9.1.1.

Table 6
Computation time and objective values with varying volumetric flow rate ranges.

Setting	Index	CPU, s				Objective			
		DP	NS-ONLY	AG-ONLY	AG-NS	DP	NS-ONLY	AG-ONLY	AG-NS
1	a	357.38	163.3	179.88	97.85	59.11	59.11	59.11	59.11
	b	4925	1960.37	333.68	146.55	59.74	59.74	59.74	59.74
	c	2789.66	1157.55	207.68	85.66	59.64	59.64	59.64	59.64
	d	3487.37	1484.43	277	105.1	59.56	59.56	59.56	59.56
2	a	493.48	255.09	275.78	141.90	78.12	78.12	78.12	78.12
	b	7047.13	2540.68	608.33	285.34	78.84	78.84	78.84	78.84
	C	437.51	213.26	220.07	110.40	77.83	77.83	77.83	77.83
	d	458.28	202.63	271.43	124.32	78.30	78.30	78.30	78.30

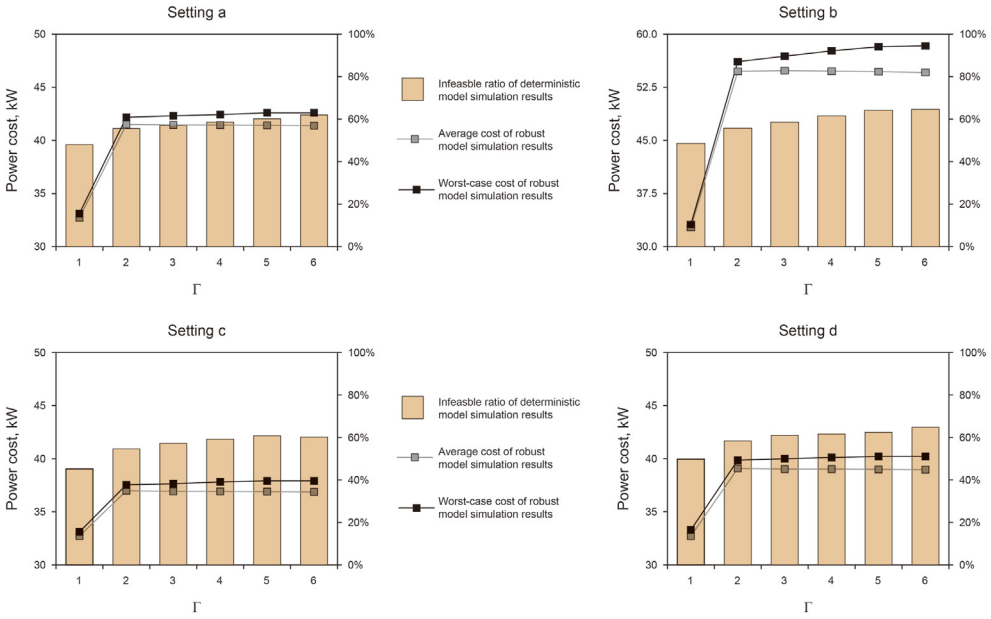

Fig. 7. The simulation results under different Γ in setting 1.

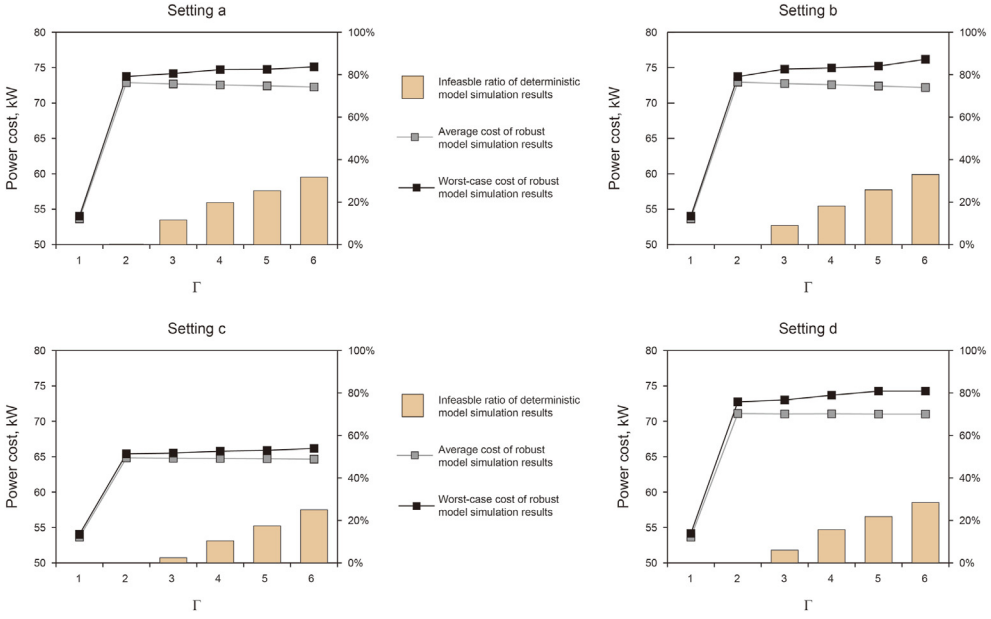

Fig. 8. The simulation results under different Γ in setting 2.

5.1. Benefits of the approximate dynamic programming algorithm

In this section, we compare the performance of the proposed multi-dimensional hybrid search algorithm with multi-dimensional global search algorithm in terms of computation times and objective values. Results with aggregation technique applied to these two algorithms are also recorded. Since the acceleration algorithm is designed for SP, we conduct three tests with different types of inputs for the SP, which are combination of suction temperature and first-stage decisions, uncertainty budget parameters and volumetric flow rate varying range, respectively.

In each test we record four results. The first result DP denotes the result of original dynamic programming algorithm. The second result NS-ONLY denotes the result of approximate dynamic programming with only neighborhood search approximate method, which is algorithm 3. The third result AG-ONLY denotes the result of aggregated version of Algorithm 2, where only aggregation method is applied. The last result AG-NS denotes the result of aggregated version of algorithm 3, in which both the two approximate techniques are used.

In these tests, the default suction pressure of the first pipeline and discharge pressure values of five compressor stations are [9.8, 8.6, 9.6, 8.8, 9.1, 8.1] MPa for setting 1 and [9.8, 8.4, 9.7, 8.7, 9.5, 7.6] MPa for setting 2 with suction temperature being [299.27, 288.36, 304.98, 291.05, 300.07, 295.58] K. The uncertainty budget parameter Γ is 3, the discretization length of volumetric flow rate is 1 Nm^3/s and the varying range of volumetric flow rate demand of each intermediate node is [4, 4, 4, 6, 8, 8] Nm^3/s unless otherwise specified.

The first test considers varying values of suction temperatures and first-stage decisions of discharge pressure. The settings are listed in Table 2 and results are in Table 3. The second test considers varying values of uncertainty budget parameter Γ and results are in Table 4. The third test records results under varying ranges of volumetric flow rate. The settings are listed in Table 5 and results are in Table 6. It is worth mentioning that heterogeneous intermediate nodes are considered, that is the varying range of demand in each node is different. Since we do not assume that all the customer demands vary in the same pattern.

The results firstly show that in our numerical test, all the approximation methods has the same optimal value as original DP, that is the gap is 0 in all cases. In addition, in both setting 1 and setting 2, the calculation time and objective value increase as the uncertainty budget parameter increases. This is reasonable since the feasible domain of larger Γ contains that of the smaller one. As for the speed gain, we introduce an assessment metric: $\frac{t_0-t_1}{t_1}$, where t_0 is the calculation time of DP and t_1 is the calculation time of algorithm to be compared. The results show that NS-ONLY has time improvements of 99.3% on average and 151.2% at maximum, AG-ONLY has time improvements of 271.1% on average and 1376.0% at maximum, AG-NS has the best performance with time improvements of 692.5% on average and 3260.6% at maximum. It implies that in our numerical tests the aggregation method is more efficient than neighborhood search. Combining these two methods together returns the best performance with a speed gain of about 7 times faster than original dynamic programming algorithm without compromising the optimality.

The speed gain verified by test results in this section is crucial since when solving the robust model in gunbarrel network, there is a high demand on computer resources due to the *curses of dimensionality* in dynamic programming. Without proposed approximation methods, the problem with realistic scale may not be able to be solved within a reasonable time.

5.2. Results of robust model and comparisons with deterministic model

In this section, we solve the two-stage robust model and deterministic model, respectively. Then we conduct simulations for 500 times to compare the performance of the two models under simulated uncertainty realizations. The uncertainty budget parameter Γ ranges from 1 to 6 in this subsection. Similar with Gorissen et al. (2015), the simulated demand is drawn from uniform distribution without compromising the uncertainty budget parameter. To better illustrate the results under different settings of volumetric flow rate ranges, we conduct simulations for four settings in Table 5, respectively.

The results are shown in Fig. 7 and Fig. 8. We introduce an assessment metric infeasible ratio, which is defined as the times when deterministic model result is infeasible divided by the total simulation times. In average, the deterministic model has an infeasible ratio of 57.91% in setting 1 and 12.58% in setting 2, while the robust model ensures feasibility in all cases. What's more, when the value of Γ is small, the robust model results are similar to deterministic model results. In setting 1 and 2, when Γ equals to 1, some of the compressor stations can be shut down to optimize the total costs. However, when Γ increases, additional costs are required to gain robustness.

The results in this subsection verify that the robust model can reduce the possibility of infeasibility when facing uncertainties. To gain that benefit, additional costs act as the price of robustness. For all the four settings, when Γ exceeds 2, the impact of increasing value of Γ is not obvious. However, when Γ increases, the calculation time increases as well, since the search space is larger, thus it is recommended to set the uncertainty budget parameter a moderate value.

What's more, simulation results verify that robust model can reduce the risk of interruption of natural gas network management. The interruption here means that operation can be infeasible due to inappropriate first-stage decisions. The risk can be quite high when fluctuation is not negligible. Considering robust optimization in natural gas network operation can prevent interruption in advance, instead of passively reacting when uncertainty reveals.

6. Conclusions

In this paper, we study solving two-stage robust optimization model in gunbarrel structured natural gas network. Uncertainties with budget parameter are considered in this paper since it is not likely that all the demands in intermediate nodes reach the maximal deviation simultaneously. The robust model is solved by C&CG algorithm which consists of MP and SP. To solve the MP, we take advantages of the separability of the structure of problem and decompose it into two hierarchies. The inner hierarchy is solved by dynamic programming and the outer hierarchy is solved by piecewise linear approximation based on lambda method. The SP is a bilevel problem and quite challenging.

To solve the bilevel SP, we first formulate it as a dynamic programming problem, and then two approximate methods are proposed. The first method is multi-dimensional neighborhood search. We provide the theoretical justification of the bounds and the algorithm in multi-dimensional cases. The second method is aggregation, in which we aggregate similar states.

The numerical results reveal that the two approximate methods together show significant speed gain of an average of 7 times. The numerical tests also verify the benefits of robust model. It reduces the possibility of infeasibility when facing demand uncertainties.

From a management perspective, we provide a proactive method to cope with uncertainties in operation management of natural gas network, instead of passively reacting when change already happens. This method ensures that no matter what the uncertainty realization is, the performance of the operation decisions won't be too bad, thus by this way we add robustness to the system.

Possible future research direction of this work is to consider tree structured or even cyclic structured natural gas network, which will be much more complicated. In tree structured network, dynamic programming methods are also applicable. However, in cyclic network the optimal nominal flow rate in each pipeline requires complicated calculating and cannot be determined directly as in tree structured and gunbarrel structured network. What's more, in this paper we only consider steady-state model, robust optimization under transient state may be another future direction. Also, distributional robust optimization can be studied to fully use some of the information of uncertainty sets.

Acknowledgments

This research was partially supported by the National Science Foundation of China (Grants 71822105 and 91746210).

Appendix A. Description of Basic Model

If a compressor k at station n is switched-on, the power consumed is:

$$\frac{H_{n,k}\cdot f_{n,k}}{\eta_{n,k}}, \tag{A1}$$

where $H_{n,k}$ and $\eta_{n,k}$ are the isentropic head and isentropic efficiency of the compressor unit respectively, and $f_{n,k}$ is the mass flow rate passing through the compressor. All compressors in the same station are installed in parallel, thus their isentropic head values are identical when in use.

The measurement of the isentropic head is the amount of work needed to compress the gas from suction pressure p_n^{s} to the discharge pressure value p_n^{d} isentropically. In practice, this work amount is related to the actual volumetric flow rate $Q_{n,k}$ and compressor rotational speed $s_{n,k}$ as follow:

$$H_{n,k} = \widehat{b}_{1,n,k} s_{n,k}^2 + \widehat{b}_{2,n,k} Q_{n,k} s_{n,k} + \widehat{b}_{3,n,k} Q_{n,k}^2. \tag{A2}$$

Similarly, the isentropic efficiency has a quadratic form representation of $Q_{n,k}$ and $s_{n,k}$:

$$\eta_{n,k} = \widehat{b}_{4,n,k} + \widehat{b}_{5,n,k}\left(\frac{Q_{n,k}}{s_{n,k}}\right) + \widehat{b}_{6,n,k}\left(\frac{Q_{n,k}}{s_{n,k}}\right)^2, \tag{A3}$$

where constants $\{\widehat{b}_{j,n,k} \big| j = 1, 2, \ldots, 6\}$ are characteristic parameters depending on the compressor unit and can be estimated via experiments which collect data of the quantities $Q_{n,k}$, $s_{n,k}$, $H_{n,k}$, and $\eta_{n,k}$. We can infer from (A2) that:

$$s_{n,k} = \frac{-\widehat{b}_{2,n,k} Q_{n,k} + \sqrt{\widehat{b}_{2,n,k}^2 Q_{n,k}^2 - 4\widehat{b}_{1,n,k}\left(\widehat{b}_{3,n,k} Q_{n,k}^2 - H_{n,k}\right)}}{2\widehat{b}_{1,n,k}}. \tag{A4}$$

The following two equations give the relationships between $(H_{n,k}, Q_{n,k})$ and $(f_{n,k}, p_n^{\mathrm{s}}, p_n^{\mathrm{d}})$:

$$H_{n,k} = \frac{Z_n R T_n^{\mathrm{s}}}{m}\left[\left(\frac{p_n^{\mathrm{d}}}{p_n^{\mathrm{s}}}\right)^m - 1\right], \tag{A5}$$

$$Q_{n,k} = Z_n R T_n^{\mathrm{s}} \frac{f_{n,k}}{p_n^{\mathrm{s}}}, \tag{A6}$$

where $m = (\sigma - 1)/\sigma$, with p_n^{s} and p_n^{d} denoting the suction and discharge pressure of the compressor station respectively. All compressors in a station share the same discharge pressure and suction pressure. The isentropic exponent σ, the gas compressibility factor Z_n and the gas constant R are positive parameters with given suction temperature T_n^{s} in our work. We denote $\mathscr{S}_n$ as the set of switched-on compressors in station n. To summarize, the power consumed by a single compressor k can be expressed by a function $P_{n,k}(p_n^{\mathrm{s}}, T_n^{\mathrm{s}}, p_n^{\mathrm{d}}, Q_{n,k}, \mathscr{S}_n)$.

The working domain of a switched-on compressor $k \in \mathscr{S}_n$ in station n is given by the following constraints:

$$s_{n,k} \le s_{n,k}^{\max}, \tag{A7}$$

$$s_{n,k}^{\min} \le s_{n,k}, \tag{A8}$$

$$Q_{n,k} \ge \widehat{a}_{1,n,k} + \widehat{a}_{2,n,k}\cdot s_{n,k} + \widehat{a}_{3,n,k} s_{n,k}^2, \tag{A9}$$

$$Q_{n,k} \le \widehat{a}_{4,n,k} + \widehat{a}_{5,n,k}\cdot s_{n,k} + \widehat{a}_{6,n,k} s_{n,k}^2, \tag{A10}$$

where $\{\widehat{a}_{j,n,k} \big| j = 1, 2, \ldots, 6\}$ are constants depending on the compressor k and can be estimated by experiments collecting data of the quantities and applying the least squares method. Equation (A7) and (A8) give the upper and lower rotational speed bounds. Equation (A9) is the surge line constraint limiting the lower bound of $Q_{n,k}$ as a quadratic function of $s_{n,k}$. Equation (A10) is the stonewall line constraint limiting the upper bound of $Q_{n,k}$ as a quadratic function of $s_{n,k}$.

Appendix B. Piece-wise Linear Approximation of MP

To solve the non-convex and nonlinear problem MP, we use linear-approximation method combined with single station optimization to significantly reduce the total number of variables in the approximation. We first denote

$$\sum_{n=1}^{N} P_n\left(p_n^{\mathrm{s},j}, T_n^{\mathrm{s},j}, p_n^{\mathrm{d}}, \widehat{Q}_n^j, \mathscr{S}_n\right) = \min_{Q_{n,k}^j} \sum_{n=1}^{N} \sum_{k \in \mathscr{S}_n} P_{n,k}\left(p_n^{\mathrm{s},j}, T_n^{\mathrm{s}}, p_n^{\mathrm{d}}, Q_{n,k}^j, \mathscr{S}_n\right), \tag{B1}$$

where $P_n(p_n^{\mathrm{s}}, T_n^{\mathrm{s}}, p_n^{\mathrm{d}}, Q_n, \mathscr{S}_n)$ is the optimization results in a station with given suction, discharge pressure, temperature and working status. It can be solved by dynamic programming or piece-wise linear approximation (Deng, 2015; Deng et al., 2019). By this way we decompose the problem into single-station optimization and network optimization.

In the network optimization, the core of the approximation is to transform expression (B1) and

$$\left(p_n^{\mathrm{s},j}, T_n^{\mathrm{s}}\right) = g_n\left(p_{n-1}^{\mathrm{d}}, T_{n-1}^{\mathrm{d}*}\left(p_{n-1}^{\mathrm{d}}\right), \widehat{\boldsymbol{Q}}^{\boldsymbol{j}}_{(\boldsymbol{n-1},\boldsymbol{n})}\right), \quad \forall j \in J, n \in [N]\backslash\{1\}$$

into a piecewise-linear formulation of $p_{n-1}^{\mathrm{d}}, p_n^{\mathrm{d}}$ using lambda method, thus we only need to consider linearization of all pairs of $(p_{n-1}^{\mathrm{d}}, p_n^{\mathrm{d}})$ in network optimization. We rewrite the term $P_n(p_n^{\mathrm{s},j}, T_n^{\mathrm{s},j}, p_n^{\mathrm{d}}, \widehat{Q}_n^j, \mathscr{S}_n)$ by function of pair $(p_{n-1}^{\mathrm{d}}, p_n^{\mathrm{d}})$ as $\mathscr{Z}_n(p_{n-1}^{\mathrm{d}}, p_n^{\mathrm{d}}, \widehat{\boldsymbol{Q}}^{\boldsymbol{j}}_{(\boldsymbol{n-1},\boldsymbol{n})}, \mathscr{S}_n)$, which denotes the optimal power cost of station n under discharge pressure p_n^{d}, previous station discharge pressure p_{n-1}^{d}, volumetric flow rate realization $\widehat{\boldsymbol{Q}}^{\boldsymbol{j}}_{(\boldsymbol{n-1},\boldsymbol{n})}$ and working compressor combination of $\mathscr{S}_n$:

following feasibility/optimality cuts for iteration index $j+1$:

$$\mathscr{Z}_n\left(p_{n-1}^{\mathrm{d}},p_n^{\mathrm{d}},\widehat{\boldsymbol{Q}}_{(\boldsymbol{n}-1,\boldsymbol{n})}^{\boldsymbol{j}},\mathscr{S}_n\right)=P_n\left(p_n^{\mathrm{s},j},T_n^{\mathrm{s},j},p_n^{\mathrm{d}},\widehat{Q}_n^{j},\mathscr{S}_n\right),=\min_{Q_{n,k}^{j}}\sum_{k\in\mathscr{S}_n}P_{n,k}\left(p_n^{\mathrm{s},j},T_n^{\mathrm{s}},p_n^{\mathrm{d}},Q_{n,k}^{j},\mathscr{S}_n\right)=\min_{Q_{n,k}^{j}}\sum_{k\in\mathscr{S}_n}P_{n,k}\left(g_n\left(p_{n-1}^{\mathrm{d}},T_{n-1}^{\mathrm{d}*}\left(p_{n-1}^{\mathrm{d}}\right),\widehat{\boldsymbol{Q}}_{(\boldsymbol{n}-1,\boldsymbol{n})}^{\boldsymbol{j}}\right),p_n^{\mathrm{d}},Q_{n,k}^{j},\mathscr{S}_n\right),\ \forall j\in J,k\in\mathscr{S}_n,n\in[N]\backslash\{1\}.$$

Suppose the domain of the pair $(p_{n-1}^{\mathrm{d}},p_n^{\mathrm{d}})$ is $D\subseteq\mathscr{R}^2$, p_{n-1}^{d} and p_n^{d} are divided into $N_{n-1,1}-1$ and $N_{n,1}-1$ parts respectively. Then the set of grid points is given by $\Lambda=1\ldots N_{n-1,1}N_{n,1}$ and the total number of triangles is $2(N_{n-1,1}-1)(N_{n,1}-1)$. We introduce $\lambda_{n,i},\ i\in I$ for each grid point and binary variables $z_{n,l},\ l\in L$ for each triangle. We denote whether to choose $\mathscr{S}_n$ by binary variable $Y_{\mathscr{S}_n}$, if $Y_{\mathscr{S}_n}=1$ then a particular combination of working status of compressors in station n is selected. The original formulation of the piece-wise linear approximation of MP is:

$$\min_{p_n^{\mathrm{d}},\eta,Y_{\mathscr{S}_n}}\ \eta$$

$$\text{s.t. }\eta\geq\sum_{n=1}^{N}\sum_{i=1}^{N_n}\sum_{\mathscr{S}_n\in\mathscr{S}}\lambda_{n,i}\cdot Y_{\mathscr{S}_n}\cdot\mathscr{Z}_n\left(\widehat{p}_{n-1,i}^{\mathrm{d}},\widehat{p}_{n,i}^{\mathrm{d}},\widehat{\boldsymbol{Q}}_{(\boldsymbol{n}-1,\boldsymbol{n})}^{\boldsymbol{j}},\mathscr{S}_n\right)+\sum_{i=1}^{N_1}\sum_{\mathscr{S}_1\in\mathscr{S}}\lambda_{1,i}\cdot Y_{\mathscr{S}_1}\cdot\mathscr{Z}_1\left(\mathscr{P}_0^{\mathrm{s}},\widehat{p}_{1,i}^{\mathrm{d}},\widehat{\boldsymbol{Q}}_{(\boldsymbol{0},1)}^{\boldsymbol{j}},\mathscr{S}_1\right),\ \forall j\in J,$$

$$(p_1^{\mathrm{s}},T_1^{\mathrm{s}})=g_1\left(\mathscr{P}_0^{\mathrm{s}},\mathscr{T}_0^{\mathrm{s}},\widehat{\boldsymbol{Q}}_{(\boldsymbol{0},1)}^{\boldsymbol{j}}\right),\ \forall j\in J,\qquad p_{n-1}^{\mathrm{d}}=\sum_{i=1}^{N_n}\lambda_{n,i}\widehat{p}_{n-1,i}^{\mathrm{d}},\ \forall n\in[N]\backslash\{1\},\qquad p_{n-1}^{\mathrm{d}}=\sum_{i=1}^{N_{n-1}}\lambda_{n-1,i}\widehat{p}_{n,i}^{\mathrm{d}},\ \forall n\in[N]\backslash\{1\},\lambda_{n,i}$$

$$\leq\sum_{l\in L(\lambda_{n,i})}z_{n,l},\ \forall n\in[N],i\in I,\sum_{i=1}^{N_n}\lambda_{n,i}=1,\ \forall n\in[N],\sum_{n\in\mathscr{S}_n}Y_{\mathscr{S}_n}=1,\ \forall n\in[N],\sum_{l=1}^{N_n'}z_{n,l}=1,\ \forall n\in[N],z_{n,l}\in\{0,1\},$$

$$\forall n\in[N],l\in L,\qquad Y_{\mathscr{S}_n}\in\{0,1\},\ \forall n\in[N],\qquad\lambda_{n,i}\geq0,\ \forall n\in[N],i\in I,\tag{B2}$$

where $N_n=N_{n,1}N_{n,1}$, $N_n'=2(N_{n,1}-1)(N_{n,1}-1)$, and $L(\lambda_{n,i})$ is the index set of all triangles adjacent to grid point $\lambda_{n,i}$. To reduce the number of variables, when we discretize the pairs of $(p_{n-1}^{\mathrm{d}},p_n^{\mathrm{d}})$, we only consider those within the working domain. It is worth pointing out constraint (B2) is nonlinear since it contains product of $\lambda_{n,i}$ and $Y_{\mathscr{S}_n}$. To linearize this term, we introduce continuous variable $x_{n,i,\mathscr{S}_n}$ and force it to equal to $\lambda_{n,i}\cdot Y_{\mathscr{S}_n}$ by following constraints:

$$x_{n,i,\mathscr{S}_n}\leq M\cdot Y_{\mathscr{S}_n},\quad\forall n\in[N],i\in I,\mathscr{S}_n\in\mathscr{S},\tag{B3}$$

$$0\leq x_{n,i,\mathscr{S}_n}\leq\lambda_{n,i},\quad\forall n\in[N],i\in I,\mathscr{S}_n\in\mathscr{S},\tag{B4}$$

$$x_{n,i,\mathscr{S}_n}\geq\lambda_{n,i}-\left(1-Y_{\mathscr{S}_n}\right)\cdot M,\quad\forall n\in[N],i\in I,\mathscr{S}_n\in\mathscr{S}.\tag{B5}$$

Replacing constraint (B2) with constraints (B3)-(B5), we obtain an MILP approximation of MP. In C&CG algorithm, we add the

$$\eta\geq\sum_{n=1}^{N}\sum_{i=1}^{N_n}\sum_{\mathscr{S}_n\in\mathscr{S}}x_{n,i,\mathscr{S}_n}\cdot\mathscr{Z}_n\left(\widehat{p}_{n-1,i}^{\mathrm{d}},\widehat{p}_{n,i}^{\mathrm{d}},\widehat{\boldsymbol{Q}}_{(\boldsymbol{n}-1,\boldsymbol{n})}^{\boldsymbol{j}+1},\mathscr{S}_n\right),\tag{B6}$$

$$(p_1^{\mathrm{s}},T_1^{\mathrm{s}})=g_1\left(\mathscr{P}_0^{\mathrm{s}},\mathscr{T}_0^{\mathrm{s}},Q_1^{j+1}\right).\tag{B7}$$

Appendix C. the Dynamic Programming Formulation of SP

This appendix provides the original dynamic programming formulation of SP.

Stage index: each stage of the DP corresponds to one compressor station.

State variables: let γ_n be the total deviation of volumetric flow rates for stage $1,2,\ldots,n$. For notation convenience, we denote $\boldsymbol{Q}_{(\boldsymbol{n}-1,\boldsymbol{n})}$ as $\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1}$, then the state of stage n is $(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1},\gamma_n)$.

State-transition equation: recall that g_n calculates $(p_n^{\mathrm{s}},T_n^{\mathrm{s}})$ from $(\widehat{p}_{n-1}^{\mathrm{d}},\widehat{T}_{n-1}^{\mathrm{d}},\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1})$:

$$(p_n^{\mathrm{s}},T_n^{\mathrm{s}})=g_n\left(\widehat{p}_{n-1}^{\mathrm{d}},\widehat{T}_{n-1}^{\mathrm{d}},\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1}\right),\quad n\in[N]\backslash\{1\},$$

where $\widehat{p}_{n-1}^{\mathrm{d}}$, $\widehat{T}_{n-1}^{\mathrm{d}}$ denote that the discharge pressure and

temperature are given in this problem. Recall that φ_n is the mapping from each pair of (p_n^s, T_n^s) to the set of feasible pairs $(\widehat{p}_n^d, \widehat{T}_n^d, Q_{n,k})$. Then, the feasibility constraints on (p_n^s, T_n^s) can be written as:

$$(p_n^s, T_n^s) = g_n\left(\widehat{p}_{n-1}^d, \widehat{T}_{n-1}^d, \tilde{\mathbf{Q}}_{\boldsymbol{n}-1}\right) \in \varphi_{n,k}\left(\widehat{p}_n^d, \widehat{T}_n^d, Q_{n,k}\right),$$

Value function: we denote by $V_n(\tilde{\mathbf{Q}}_{\boldsymbol{n}}, \gamma_n)$ the value function with respect to the system state $(\tilde{\mathbf{Q}}_{\boldsymbol{n}}, \gamma_n)$. Then $V_n(\tilde{\mathbf{Q}}_{\boldsymbol{n}}, \gamma_n)$ calculates the maximum power required to transport natural gas from station 1 to n under total deviation γ_n as follows:

$$\begin{aligned}
V_n(\tilde{\mathbf{Q}}_{\boldsymbol{n}}, \gamma_n) &= \max_{\tilde{\mathbf{Q}}_{\boldsymbol{i}}, \gamma_i} \quad \sum_{i=1}^{n} C_i((\tilde{\mathbf{Q}}_{\boldsymbol{i}-1}, \gamma_{i-1}), (\tilde{\mathbf{Q}}_{\boldsymbol{i}}, \gamma_i)) \\
&= \max_{\tilde{\mathbf{Q}}_{\boldsymbol{i}}, \gamma_i} \quad \sum_{i=1}^{n} P_i\left(p_i^s, T_i^s, \widehat{p}_i^d, Q_i, \widehat{\mathscr{S}}_n\right) \\
&\text{s.t. } (p_i^s, T_i^s) \in \varphi_{i,k}\left(\widehat{p}_i^d, \widehat{T}_i^d, \tilde{\mathbf{Q}}_{\boldsymbol{i}-1}\right), \quad \forall k \in \widehat{\mathscr{S}}_n, 1 \le i \le n, \\
&\quad (p_i^s, T_i^s) = g_i\left(\widehat{p}_{i-1}^d, \widehat{T}_{i-1}^d, \tilde{\mathbf{Q}}_{\boldsymbol{i}-1}\right), \quad \forall 2 \le i \le n, \\
&\quad Q_{l(m^+)} - dQ_m = Q_{l(m^-)}, \quad \forall m \in [M], \\
&\quad Q_i = Q_{l(i^-)} = Q_{l(i^+)}, \quad \forall 1 \le i \le n, \\
&\quad \gamma_i = \sum_{m \in M(i-1,i)} \frac{|\overline{dQ}_m - dQ_m|}{\delta \overline{dQ}_m} + \gamma_{i-1}, \quad \forall 2 \le i \le n, (p_1^s, T_1^s) = g_1\left(\mathscr{P}_0^s, \mathscr{T}_0^s, \mathbf{Q}_{(\mathbf{0},1)}\right), \gamma_n \le \Gamma.
\end{aligned}$$

$k \in \widehat{\mathscr{S}}_n, n \in [N]\backslash\{1\}$.

Also we have:

$$Q_{l(m^+)} - dQ_m = Q_{l(m^-)}, \quad m \in [M],$$

$$Q_n = Q_{l(n^-)} = Q_{l(n^+)}, \quad n \in [N].$$

What's more, since we consider the budget uncertainty set, the total deviation should be within the budget:

$$\gamma_n = \sum_{m \in M(n-1,n)} \frac{|\overline{dQ}_m - dQ_m|}{\delta \overline{dQ}_m} + \gamma_{n-1} \le \Gamma, \quad n \in [N]\backslash\{1\}.$$

Immediate reward function: Recall that $P_n(p_n^s, T_n^s, p_n^d, Q_n, \mathscr{S}_n)$ is compressor station n's optimal total power consumption. Given former state $(\tilde{\mathbf{Q}}_{\boldsymbol{n}-1}, \gamma_{n-1})$, we are now ready to start stage n, and select $(\tilde{\mathbf{Q}}_{\boldsymbol{n}}, \gamma_n)$ as the immediate destination. Then the immediate reward function is:

$$\begin{aligned}
&C_n((\tilde{\mathbf{Q}}_{\boldsymbol{n}-1}, \gamma_{n-1}), (\tilde{\mathbf{Q}}_{\boldsymbol{n}}, \gamma_n)) = P_n\left(p_n^s, T_n^s, \widehat{p}_n^d, Q_n, \widehat{\mathscr{S}}_n\right) \\
&= P_n\left(g_n\left(\widehat{p}_{n-1}^d, \widehat{T}_{n-1}^d, \tilde{\mathbf{Q}}_{\boldsymbol{n}-1}\right), \widehat{p}_n^d, Q_n, \widehat{\mathscr{S}}_n\right), \quad n \in [N]\backslash\{1\}.
\end{aligned}$$

Then, the Bellman's equation can be written as follows:

$$\begin{aligned}
V_n(\tilde{\mathbf{Q}}_{\boldsymbol{n}}, \gamma_n) &= \max_{\tilde{\mathbf{Q}}_{\boldsymbol{n}-1}, \gamma_{n-1}} \{C_n((\tilde{\mathbf{Q}}_{\boldsymbol{n}-1}, \gamma_{n-1}), (\tilde{\mathbf{Q}}_{\boldsymbol{n}}, \gamma_n)) + V_{n-1}(\tilde{\mathbf{Q}}_{\boldsymbol{n}-1}, \gamma_{n-1})\} \\
&= \max_{\tilde{\mathbf{Q}}_{\boldsymbol{n}-1}, \gamma_{n-1}} \left\{P_n\left(p_n^s, T_n^s, \widehat{p}_n^d, Q_{l(n^-)}\right) + V_{n-1}(\tilde{\mathbf{Q}}_{\boldsymbol{n}-1}, \gamma_{n-1})\right\} \\
&\text{s.t.} \quad (p_n^s, T_n^s) \in \varphi_{n,k}\left(\widehat{p}_n^d, \widehat{T}_n^d, \tilde{\mathbf{Q}}_{\boldsymbol{n}-1}\right), \\
&\quad (p_n^s, T_n^s) = g_n\left(\widehat{p}_{n-1}^d, \widehat{T}_{n-1}^d, \tilde{\mathbf{Q}}_{\boldsymbol{n}-1}\right), \\
&\quad Q_{l(m^+)} - dQ_m = Q_{l(m^-)}, \forall m \in M(n-1,n), \\
&\quad Q_{l(n^-)} = Q_{l(n^+)}, \\
&\quad \gamma_n = \sum_{m \in M(n-1,n)} \frac{|\overline{dQ}_m - dQ_m|}{\delta \overline{dQ}_m} + \gamma_{n-1}.
\end{aligned} \tag{C1}$$

Appendix D. Neighborhood Search Related Proposition for SP

In (C1), we denote the optimal decision for given volumetric flow rate $\tilde{\mathbf{Q}}_{\boldsymbol{n}}$ and total deviation γ_n as $\mathscr{X}^*(\tilde{\mathbf{Q}}_{\boldsymbol{n}}, \gamma_n)$. We also define

$$G_n((\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1},\gamma_{n-1}),(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)\times)=P_n\left(g_n\left(\hat{p}^{\mathrm{d}}_{n-1},\hat{T}^{\mathrm{d}}_{n-1},\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1}\right),\hat{p}^{\mathrm{d}}_{n},Q_{l(n^-)},\widehat{\mathscr{T}}_n\right)+V_{n-1}(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1},\gamma_{n-1}),$$

which describes the total power cost at stage n as a function of $(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1},\gamma_{n-1})$ and $(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)$. For given direction $\boldsymbol{e}_1$, $\boldsymbol{e}_2$ and given discretization interval size Δ, we define

$$\overline{\mathscr{X}}(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n,\Delta,\boldsymbol{e}_1,\boldsymbol{e}_2)=\inf\{x\geq 0)|G(\boldsymbol{X}^*(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+x*\boldsymbol{e}_1,(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+\Delta\cdot\boldsymbol{e}_2)<G(\boldsymbol{X}^*(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+(x-\Delta)*\boldsymbol{e}_1,(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+\Delta\cdot\boldsymbol{e}_2)\}$$

and

$$\underline{\mathscr{X}}\left(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n,\Delta,\boldsymbol{e}_1,\boldsymbol{e}_2\right)=\sup\{x\leq 0)|G(\boldsymbol{X}^*(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+x*\boldsymbol{e}_1,(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+\Delta\cdot\boldsymbol{e}_2)<G(\boldsymbol{X}^*(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+(x+\Delta)*\boldsymbol{e}_1,(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+\Delta\cdot\boldsymbol{e}_2)\}$$

Essentially, $\overline{\mathscr{X}}(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n,\Delta,\boldsymbol{e}_1,\boldsymbol{e}_2)$ is the minimum x in direction $\boldsymbol{e}_1$ such that function $G(\boldsymbol{X}^*(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+x*\boldsymbol{e}_1,(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+\Delta\cdot\boldsymbol{e}_2)$ starts decreasing with x. And $\underline{\mathscr{X}}(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n,\Delta,\boldsymbol{e}_1,\boldsymbol{e}_2)$ is the maximum x in direction $\boldsymbol{e}_1$ such that function $G(\boldsymbol{X}^*(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+x*\boldsymbol{e}_1,(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+\Delta\cdot\boldsymbol{e}_2)$ starts increasing with x. The fundamental theorem of the algorithm is shown as follows:

LEMMA 1. *For finite closed sets $\boldsymbol{X},\boldsymbol{Y}\subset\mathbb{R}^n$, $\mathbb{R}^m$ and vector $\boldsymbol{x}\in\mathbb{R}^n,\boldsymbol{y}\in\mathbb{R}^m$, we denote $G(\boldsymbol{x},\boldsymbol{y}):\boldsymbol{X}\times\boldsymbol{Y}\to\mathbb{R}$ as an arbitrary continuous function. We choose $\Delta\boldsymbol{y}$ as an arbitrary vector where $\Delta\boldsymbol{y}>0$. Suppose $G(\boldsymbol{x},\boldsymbol{y})$ is quasi-concave in $\boldsymbol{x}$. For any given $\boldsymbol{y}\in Y$ such that $\boldsymbol{y}+\Delta\boldsymbol{y}\in Y$, and a given maximum $\boldsymbol{x}^*(\boldsymbol{y})\in\arg\max_{\boldsymbol{x}\in\boldsymbol{X}}G(\boldsymbol{x},\boldsymbol{y})$ and an arbitrary positive direction $\boldsymbol{e}$. We denote*

$$\begin{aligned}\bar{x}(\boldsymbol{y})&=\inf\{x\geq 0|\exists\Delta x>0,G(\boldsymbol{x}^*(\boldsymbol{y})+x*\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y})+(x-\Delta x)*\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})\},\\ \underline{x}(\boldsymbol{y})&=\sup\{x\leq 0|\exists\Delta x>0,G(\boldsymbol{x}^*(\boldsymbol{y})+x*\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y})+(x+\Delta x)*\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})\},\\ \bar{x}(\boldsymbol{y},\Delta x)&=\inf\{x\geq 0)|G(\boldsymbol{x}^*(\boldsymbol{y})+x*\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y})+(x-\Delta x)*\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})\},\\ \underline{x}(\boldsymbol{y},\Delta x)&=\sup\{x\leq 0|G(\boldsymbol{x}^*(\boldsymbol{y})+x*\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y})+(x+\Delta x)*\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})\}.\end{aligned}$$

Then, for an arbitrary $\Delta x>0$,

$$\underline{x}(\boldsymbol{y})-\Delta x\leq\underline{x}(\boldsymbol{y},\Delta x)\leq(\boldsymbol{x}^*(\boldsymbol{y}+\Delta\boldsymbol{y},\boldsymbol{e})-\boldsymbol{x}^*(\boldsymbol{y}))^{\mathrm{T}}*\boldsymbol{e}\leq\bar{x}(\boldsymbol{y},\Delta x)\leq\bar{x}(\boldsymbol{y})+\Delta x,$$

where

$$\boldsymbol{x}^*(\boldsymbol{y}+\Delta\boldsymbol{y},\boldsymbol{e})\in\left\{\arg\max_{\boldsymbol{x}\in\boldsymbol{X}}G(\boldsymbol{x},\boldsymbol{y}+\Delta\boldsymbol{y})\middle|\frac{(\boldsymbol{x}-\boldsymbol{x}^*(\boldsymbol{y}))^{\mathrm{T}}}{\|(\boldsymbol{x}-\boldsymbol{x}^*(\boldsymbol{y}))\|_2}\cdot\boldsymbol{e}=\pm 1\right\}$$

The full version is provided in Appendix F. with further improved boundaries if function $G(\boldsymbol{x},\boldsymbol{y})$ has additional characteristics.

PROPOSITION 1. *For any $n\in[N]$, given direction $\boldsymbol{e}_1$, $\boldsymbol{e}_2$ and given discretization interval size $\Delta>0$, if $G_n((\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1},\gamma_{n-1}),(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n))$ is continuous and quasi-concave in $(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1},\gamma_{n-1})$,*

$$\underline{\mathscr{X}}(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n,\Delta,\boldsymbol{e}_1,\boldsymbol{e}_2)\leq(\mathscr{X}^*((\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n)+\Delta*\boldsymbol{e}_2,\boldsymbol{e}_1)-\mathscr{X}^*(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n))^{\mathrm{T}}*\boldsymbol{e}_1\leq\overline{\mathscr{X}}(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n,\Delta,\boldsymbol{e}_1,\boldsymbol{e}_2).$$

PROPOSITION 1 is a direct result of LEMMA 1. It gives search boundaries $\underline{\mathscr{X}}(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n,\Delta,\boldsymbol{e}_1,\boldsymbol{e}_2)$ and $\overline{\mathscr{X}}(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n,\Delta,\boldsymbol{e}_1,\boldsymbol{e}_2)$ of the optimal solution in direction $\boldsymbol{e}_1$, when function $G_n((\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1},\gamma_{n-1}),(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}},\gamma_n))$ is quasi-concave in $(\tilde{\boldsymbol{Q}}_{\boldsymbol{n}-1},\gamma_{n-1})$. However, we should still consider the case when the conditions in PROPOSITION 1 do not hold. Motivated by Deng et al. (2019), we develop a hybrid search algorithm in which after every certain steps the approximation result is compared with the one without approximation, then the errors can be corrected if there is any.

Appendix E. the Proposed Approximate DP Algorithm Pseudocode for SP

The proposed hybrid search algorithm in shown in Algorithm 1. The global search algorithm, which is the original formulation of multi-dimensional dynamic programming without approximation is shown in Algorithm 2. It is a benchmark of the proposed hybrid search algorithm 3. It is worth noting that there is a chance that conditions in PROPOSITION 1 do not hold. To handle these cases, in hybrid search algorithm 3 we add a checking mechanism similar to Deng et al. (2019) to correct the search error if there exists any. The checking mechanism is called every certain number of steps. At each checking point, the approximate neighborhood search is compared with global search. If there is a gap between the two results, we backtrack previous steps to recalculate. In this way gaps and errors are corrected.

Algorithm 1. Multi-dimensional Neighborhood Search Function

The aggregation technique can be applied to both Algorithm 2 and algorithm 3. In aggregation we skip certain search points as interpreted in section 4.2. For example, a full search space for stage $n-1$ contains all combinations of $(\tilde{\mathbf{Q}}_{\boldsymbol{n-1}}, \gamma_{n-1})$. If the minimal discretization of γ_{n-1} is 0.01, we can only search in the points with a minimal γ_{n-1} discretization of 0.5. With well-designed parameters, we may gain a significant calculation time improvement.

Algorithm 1 Multi-dimensional Neighborhood Search Function

[Function] ***Neighborhood Search*** $(n,\ \hat{Q}^1_n, \hat{Q}^2_n \ldots \hat{\gamma}_n, tQ^1_{n-1}, tQ^2_{n-1} \ldots t\gamma_{n-1})$

Find $i_1^*, i_2^* \ldots \in [J_{n-1,1}], [J_{n-1,2}] \ldots$ where $\hat{Q}^1_{n-1,i_1^*} = tQ^1_{n-1}$, $\hat{Q}^2_{n-1,i_2^*} = tQ^2_{n-1} \ldots$

$V^{\mathrm{b}}, V^{\mathrm{f}} \leftarrow -\infty$

while $k <= \dim^*(\boldsymbol{Q}_n)$ **do**

  $V_{\mathrm{p}} \leftarrow \max(V^{\mathrm{b}}, V^{\mathrm{f}})$

  (***Forward Neighborhood Search***)

  $i_1, i_2 \ldots \leftarrow i_1^*, i_2^* \ldots$

  while $i_k \le J_{n-1,k}$ **do**

    if $\hat{Q}^1_{n-1,i_{|\boldsymbol{Q}_n|}} = \hat{Q}^1_n$ and $\hat{\gamma}_n \le \Gamma$ and $G_n((\hat{Q}^1_{n-1,i_1}, \hat{Q}^2_{n-1,i_2} \ldots \hat{\gamma}_{n-1,i_0}), (\hat{Q}^1_n, \hat{Q}^2_n \ldots \hat{\gamma}_n)) < V^{\mathrm{f}}$ **then**

      break

    else if $\hat{Q}^1_{n-1,i_{|\boldsymbol{Q}_n|}} = \hat{Q}^1_n$ and $\hat{\gamma}_n \le \Gamma$ and $G_n((\hat{Q}^1_{n-1,i_1}, \hat{Q}^2_{n-1,i_2} \ldots \hat{\gamma}_{n-1,i_0}), (\hat{Q}^1_n, \hat{Q}^2_n \ldots \hat{\gamma}_n)) \ge V^{\mathrm{f}}$ **then**

      $V^{\mathrm{f}} \leftarrow G_n((\hat{Q}^1_{n-1,i_1}, \hat{Q}^2_{n-1,i_2} \ldots \hat{\gamma}_{n-1,i_0}), (\hat{Q}^1_n, \hat{Q}^2_n \ldots \hat{\gamma}_n))$

      $i_k \leftarrow i_k + 1$

      $i_k^* \leftarrow i_k$

  end while

  (***Backward Neighborhood Search***)

  $i_1, i_2 \ldots \leftarrow i_1^*, i_2^* \ldots$

  while $i_k \ge 0$ **do**

    if $\hat{Q}^1_{n-1,i_{|\boldsymbol{Q}_n|}} = \hat{Q}^1_n$ and $\hat{\gamma}_n \le \Gamma$ and $G_n((\hat{Q}^1_{n-1,i_1}, \hat{Q}^2_{n-1,i_2} \ldots \hat{\gamma}_{n-1,i_0}), (\hat{Q}^1_n, \hat{Q}^2_n \ldots \hat{\gamma}_n)) < V^{\mathrm{b}}$ **then**

      break

    else if $\hat{Q}^1_{n-1,i_{|\boldsymbol{Q}_n|}} = \hat{Q}^1_n$ and $\hat{\gamma}_n \le \Gamma$ and $G_n((\hat{Q}^1_{n-1,i_1}, \hat{Q}^2_{n-1,i_2} \ldots \hat{\gamma}_{n-1,i_0}), (\hat{Q}^1_n, \hat{Q}^2_n \ldots \hat{\gamma}_n)) \ge V^{\mathrm{b}}$ **then**

      $V^{\mathrm{b}} \leftarrow G_n((\hat{Q}^1_{n-1,i_1}, \hat{Q}^1_{n-1,i_2} \ldots \hat{\gamma}_{n-1,i_0}), (\hat{Q}^1_n, \hat{Q}^2_n \ldots \hat{\gamma}_n))$

      $i_k \leftarrow i_k - 1$

      $i_k^* \leftarrow i_k$

  end while

  if $k = \dim(\boldsymbol{Q}_n)$ **then**

    $k = 0$

  if $|V_{\mathrm{p}} - \max(V^{\mathrm{b}}, V^{\mathrm{f}})| < \dot{o}$ **then**

    break

end while

Find $Q^1_{n-1}, Q^2_{n-1} \ldots \gamma_{n-1}$ corresponding to V_{p}.

return $(V_{\mathrm{p}}, Q^1_{n-1}, Q^2_{n-1} \ldots \gamma_{n-1})$

* The term dim here represents the total number of dimensions of a vector.

Algorithm 2. Multi-dimensional Global Search Function

Algorithm 2 Multi-dimensional Global Search Function

[Function] ***Global Search*** $(n,\ \hat{Q}_n^1, \hat{Q}_n^2 ... \hat{\gamma}_n)$

$V \leftarrow 0$

for $k = 1, 2...$ **do**

  for $i_k = 1, 2... J_{n-1,k}$ **do**

    if $\hat{Q}_{n-1,i_{|\boldsymbol{Q}_n|}}^1 = \hat{Q}_n^1$ and $\hat{\gamma}_n \le \Gamma$ and $G_n((\hat{Q}_{n-1,i_1}^1, \hat{Q}_{n-1,i_2}^2 ... \hat{\gamma}_{n-1,i_0}), (\hat{Q}_n^1, \hat{Q}_n^2 ... \hat{\gamma}_n)) \ge V^{\mathrm{b}}$ **then**

      $V \leftarrow G_n((\hat{Q}_{n-1,i_1}^1, \hat{Q}_{n-1,i_2}^1 ... \hat{\gamma}_{n-1,i_0}), (\hat{Q}_n^1, \hat{Q}_n^2 ... \hat{\gamma}_n))$

Find $Q_{n-1}^1, Q_{n-1}^2 ... \gamma_{n-1}$ corresponding to V.

return $(V, Q_{n-1}^1, Q_{n-1}^2 ... \gamma_{n-1})$

Algorithm 3. Multi-dimensional Hybrid Search Algorithm

Algorithm 3 Multi-dimensional Hybrid Search Algorithm

Initialize: Solve for $V_1(\boldsymbol{Q}_{1,\boldsymbol{j}}, \gamma_{1,j})\ \ (\forall j, k \in [J_{1,k}])$

for $n = 2, 3, 4... N$ **do**

  Set $tQ_{n-1}^1, tQ_{n-1}^2 ...$ as the minimum value and $t\gamma_{n-1}$ as the maximum value.

  for $k = 0, 1, 2...$ **do**

    for $j_k = 1, 2... J_{n,k}$ **do**

      $j \leftarrow \sum_k j_k$

      $(V_n(\boldsymbol{Q}_{\boldsymbol{n,j}}, \gamma_{n,j}), \mathrm{X}^*(\boldsymbol{Q}_{\boldsymbol{n,j}}, \gamma_{n,j})) \leftarrow$ ***Neighborhood Search*** $(n,\ \hat{Q}_{n,j}^1, \hat{Q}_{n,j}^2 ... \hat{\gamma}_{n,j}, tQ_{n-1}^1, tQ_{n-1}^2 ... t\gamma_{n-1})$

      if $j = k\hat{j}(k \in \mathrm{Z})$ **then**

        $(V_n^{\mathrm{g}}(\boldsymbol{Q}_{\boldsymbol{n,j}}, \gamma_{n,j}), \mathrm{X}^{\mathrm{g}^*}(\boldsymbol{Q}_{\boldsymbol{n,j}}, \gamma_{n,j})) \leftarrow$ ***Global Search*** $(n,\ \hat{Q}_{n,j}^1, \hat{Q}_{n,j}^2 ... \hat{\gamma}_{n,j})$

        if $V_n^{\mathrm{g}}(\boldsymbol{Q}_{\boldsymbol{n,j}}, \gamma_{n,j}) > V_n(\boldsymbol{Q}_{\boldsymbol{n,j}}, \gamma_{n,j})$ **then**

          $V_n(\boldsymbol{Q}_{\boldsymbol{n,j}}, \gamma_{n,j}) \leftarrow V_n^{\mathrm{g}}(\boldsymbol{Q}_{\boldsymbol{n,j}}, \gamma_{n,j})$

          $(tQ_{n-1}^1, tQ_{n-1}^2 ... t\gamma_{n-1}) \leftarrow \mathrm{X}^{\mathrm{g}^*}(\boldsymbol{Q}_{\boldsymbol{n,j}}, \gamma_{n,j})$

          for $k' = k, k-1 ... 0$ **do**

            for $j_{k'} = j_k, j_k - 1 ... 0$ **do**

              $(V_n^{\mathrm{b}}(\boldsymbol{Q}_{\boldsymbol{n,j_{k'}}}, \gamma_{n,j_{k'}}), \mathrm{X}^{\mathrm{b}^*}(\boldsymbol{Q}_{\boldsymbol{n,j_{k'}}}, \gamma_{n,j_{k'}})) \leftarrow$ ***Global Search*** $(n,\ \hat{Q}_{n,j_{k'}}^1, \hat{Q}_{n,j_{k'}}^2 ... \hat{\gamma}_{n,j_{k'}})$

              $(tQ_{n-1}^1, tQ_{n-1}^2 ... t\gamma_{n-1}) \leftarrow \mathrm{X}^{\mathrm{b}^*}(\boldsymbol{Q}_{\boldsymbol{n,j_{k'}}}, \gamma_{n,j_{k'}})$

              if $V_n^{\mathrm{b}}(\boldsymbol{Q}_{\boldsymbol{n,j_{k'}}}, \gamma_{n,j_{k'}}) > V_n(\boldsymbol{Q}_{\boldsymbol{n,j_{k'}}}, \gamma_{n,j_{k'}})$ **then**

                $V_n(\boldsymbol{Q}_{\boldsymbol{n,j_{k'}}}, \gamma_{n,j_{k'}}) \leftarrow V_n^{\mathrm{b}}(\boldsymbol{Q}_{\boldsymbol{n,j_{k'}}}, \gamma_{n,j_{k'}})$

              end if

            end for

          end for

        end if

      end if

    end for

  end for

end for

Appendix FFull Version of all the Lemmas

LEMMA 2. 1. *For finite closed sets* $\boldsymbol{X}, \boldsymbol{Y} \subset \mathbb{R}^n$, $\mathbb{R}^m$ *and vector* $\boldsymbol{x} \in \mathbb{R}^n$, $\boldsymbol{y} \in \mathbb{R}^m$, *we denote* $G(\boldsymbol{x}, \boldsymbol{y}) : \boldsymbol{X} \times \boldsymbol{Y} \to \mathbb{R}$ *as an arbitrary continuous function. We choose* $\Delta \boldsymbol{y}$ *as an arbitrary vector where* $\Delta \boldsymbol{y} > 0$. *Suppose* $G(\boldsymbol{x}, \boldsymbol{y})$ *is quasi-concave in* $\boldsymbol{x}$. *For any given* $\boldsymbol{y} \in Y$ *such that* $\boldsymbol{y} + \Delta \boldsymbol{y} \in Y$, *and a given maximum* $\boldsymbol{x}^*(\boldsymbol{y}) \in \arg\max_{\boldsymbol{x} \in \boldsymbol{X}} G(\boldsymbol{x}, \boldsymbol{y})$ *and an arbitrary positive direction* $\boldsymbol{e}$. *We denote*

$$\overline{x}(\boldsymbol{y}) = \inf\{x \geq 0 | \exists \Delta x > 0, G(\boldsymbol{x}^*(\boldsymbol{y}) + x^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}) + (x - \Delta x)^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y})\},$$
$$\underline{x}(\boldsymbol{y}) = \sup\{x \leq 0 | \exists \Delta x > 0, G(\boldsymbol{x}^*(\boldsymbol{y}) + x^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}) + (x + \Delta x)^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y})\},$$
$$\overline{x}(\boldsymbol{y}, \Delta x) = \inf\{x \geq 0) | G(\boldsymbol{x}^*(\boldsymbol{y}) + x^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}) + (x - \Delta x)^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y})\},$$
$$\underline{x}(\boldsymbol{y}, \Delta x) = \sup\{x \leq 0 | G(\boldsymbol{x}^*(\boldsymbol{y}) + x^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}) + (x + \Delta x)^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y})\}.$$

Then, for an arbitrary $\Delta x > 0$,

$$\underline{x}(\boldsymbol{y}) - \Delta x \leq \underline{x}(\boldsymbol{y}, \Delta x) \leq (\boldsymbol{x}^*(\boldsymbol{y} + \Delta \boldsymbol{y}, \boldsymbol{e}) - \boldsymbol{x}^*(\boldsymbol{y}))^{\mathrm{T}} * \boldsymbol{e} \leq \overline{x}(\boldsymbol{y}, \Delta x) \leq \overline{x}(\boldsymbol{y}) + \Delta x,$$

where $\boldsymbol{x}^*(\boldsymbol{y} + \Delta \boldsymbol{y}, \boldsymbol{e}) \in \left\{ \arg\max_{\boldsymbol{x} \in \boldsymbol{X}} G(\boldsymbol{x}, \boldsymbol{y} + \Delta \boldsymbol{y}) \middle| \frac{(\boldsymbol{x} - \boldsymbol{x}^*(\boldsymbol{y}))^{\mathrm{T}}}{\|(\boldsymbol{x} - \boldsymbol{x}^*(\boldsymbol{y}))\|_2} \cdot \boldsymbol{e} = \pm 1 \right\}$

2. In addition to 1, if $G(\boldsymbol{x}, \boldsymbol{y})$ is L^∞-Lipschitz in $\boldsymbol{y}$, i.e. $\exists L \geq 0$ such that

$$\left|G(\boldsymbol{x}, \boldsymbol{y}_1) - G(\boldsymbol{x}, \boldsymbol{y}_2)\right| \leq L \cdot \|\boldsymbol{y}_1 - \boldsymbol{y}_2\|_2, \quad \forall \boldsymbol{y}_1, \boldsymbol{y}_2 \in Y, \boldsymbol{x} \in X,$$

then

$$\overline{x}(\boldsymbol{y}) \leq \inf\{x \geq 0) | G(\boldsymbol{x}^*(\boldsymbol{y}) + x^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}), \boldsymbol{y}) - L\|\Delta \boldsymbol{y}\|_2\},$$
$$\underline{x}\left(\boldsymbol{y}\right) \leq \sup\{x \leq 0) | G(\boldsymbol{x}^*(\boldsymbol{y}) + x^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}), \boldsymbol{y}) - L\|\Delta \boldsymbol{y}\|_2\}.$$

3. In addition to 2, if $G(\boldsymbol{x}, \boldsymbol{y})$ is continuously differentiable in **x.** Denote $g(x, \boldsymbol{y}) = G(\boldsymbol{x}^*(\boldsymbol{y}) + x^*\boldsymbol{e}, \boldsymbol{y})$ and $g(x, \boldsymbol{y} + \Delta \boldsymbol{y}) = G(\boldsymbol{x}^*(\boldsymbol{y}) + x^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y})$, if there exists $\varepsilon > 0$ such that $\partial_x g(x_1, \boldsymbol{y}) - \partial_x g(x_2, \boldsymbol{y}) \leq \varepsilon^*(x_2 - x_1)$, $\forall x_1 > x_2$ and $\boldsymbol{x}^*(\boldsymbol{y}) + x_1^*\boldsymbol{e}, \boldsymbol{x}^*(\boldsymbol{y}) + x_2^*\boldsymbol{e} \in X$, then it holds that

$$\overline{x}(\boldsymbol{y}) - \underline{x}(\boldsymbol{y}) \leq 4\sqrt{L\|\Delta \boldsymbol{y}\|_2/\varepsilon}.$$

Proof of LEMMA 2 1. We prove the first inequality by showing that

$$\underline{x}(\boldsymbol{y}) - \Delta x - \varepsilon \leq \underline{x}(\boldsymbol{y}, \Delta x), \quad \forall \varepsilon > 0.$$

For an arbitrary $\varepsilon > 0$, the inequality holds if $\underline{x}(\boldsymbol{y}, \Delta x) \geq \underline{x}(\boldsymbol{y}) - \varepsilon$ because $\Delta x > 0$. Therefore, we only need to discuss the case when $\underline{x}(\boldsymbol{y}, \Delta x) < \underline{x}(\boldsymbol{y}) - \varepsilon$.

By the definition of $\underline{x}(\boldsymbol{y})$, there exist $0 > x_0 > \underline{x}(\boldsymbol{y}) - \varepsilon$ and $\Delta x_0 > 0$, such that

$$G(\boldsymbol{x}^*(\boldsymbol{y}) + x_0 * \boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}) + (x_0 + \Delta x_0) * \boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}).$$

Then we argue that there exists $\Delta x_1 > 0, \Delta x_1 \leq \Delta x$ such that

$$G(\boldsymbol{x}^*(\boldsymbol{y}) + x_0 * \boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}) + (x_0 + \Delta x_1) * \boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}).$$

If $\Delta x_0 \leq \Delta x$, then $\Delta x_1 = \Delta x_0$. If $\Delta x_0 > \Delta x$, we have

$$G(\boldsymbol{x}^*(\boldsymbol{y}) + x_0 * \boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}) + (x_0 + \Delta x_0 - \Delta x) * \boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}).$$

Otherwise, we should have

$$\begin{aligned} G(\boldsymbol{x}^*(\boldsymbol{y}) + (x_0 + \Delta x_0 - \Delta x)^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) &\leq G(\boldsymbol{x}^*(\boldsymbol{y}) + x_0^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) \\ &< G(\boldsymbol{x}^*(\boldsymbol{y}) + (x_0 + \Delta x_0)^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) \end{aligned}$$

and $x_0 + \Delta x_0 > \underline{x}(\boldsymbol{y}) - \varepsilon > \underline{x}(\boldsymbol{y}, \Delta x)$, which contradicts the definition of $\underline{x}(\boldsymbol{y}, \Delta x)$. If necessary, repeat the above subtraction, and obtain $\Delta x_1 = \Delta x_0 - n \cdot \Delta x > 0$, $\Delta x_1 \leq \Delta x$, $n \in \mathbb{Z}^+$. Let $x_1 = x_0 + \Delta x_1 - \Delta x \leq x_0$, then $\|\boldsymbol{x}_3 - \boldsymbol{x}^*(\boldsymbol{y})\|_2 > \|\boldsymbol{x}_0 - \boldsymbol{x}^*(\boldsymbol{y})\|_2$. By the quasi-concavity of function $G(\boldsymbol{x}, \boldsymbol{y} + \Delta \boldsymbol{y})$ in $\boldsymbol{x}$ and in direction $\boldsymbol{e}$, we have

$$\begin{aligned} G(\boldsymbol{x}^*(\boldsymbol{y}) + x_1^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) &\leq G(\boldsymbol{x}^*(\boldsymbol{y}) + x_0^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) \\ &< G(\boldsymbol{x}^*(\boldsymbol{y}) + (x_0 + \Delta x_1)^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}). \end{aligned}$$

Therefore, $x_1 \in \{x \leq 0 | G(\boldsymbol{x}^*(\boldsymbol{y}) + x^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}) + (x + \Delta x)^*\boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y})\}$. As a result,

$$\underline{x}(\boldsymbol{y}, \Delta x) \geq x_1 > x_0 - \Delta x > \underline{x}(\boldsymbol{y}) - \Delta x - \varepsilon.$$

For any arbitrary $\varepsilon > 0$, the above inequality holds, thus we can derive that

$$\underline{x}(\boldsymbol{y}, \Delta x) \geq \underline{x}(\boldsymbol{y}) - \Delta x.$$

We next prove the second inequality by contradiction. Assuming that there exists $\boldsymbol{x}^*(\boldsymbol{y} + \Delta \boldsymbol{y}, \boldsymbol{e}) : (\boldsymbol{x}^*(\boldsymbol{y} + \Delta \boldsymbol{y}, \boldsymbol{e}) - \boldsymbol{x}^*(\boldsymbol{y}))^{\mathrm{T}} * \boldsymbol{e} < \underline{x}(\boldsymbol{y}, \Delta x)$. By the definition of $\underline{x}(\boldsymbol{y}, \Delta x)$, there exists x_2:

$$(\boldsymbol{x}^*(\boldsymbol{y} + \Delta \boldsymbol{y}, e) - \boldsymbol{x}^*(\boldsymbol{y}))^{\mathrm{T}} * \boldsymbol{e} < x_2 < \underline{x}(\boldsymbol{y}, \Delta x) \tag{F1}$$

such that

$$G(\boldsymbol{x}^*(\boldsymbol{y}) + x_2 * \boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}) + (x_2 + \Delta x) * \boldsymbol{e}, \boldsymbol{y} + \Delta \boldsymbol{y}).$$

Because $\boldsymbol{x}^*(\boldsymbol{y} + \Delta \boldsymbol{y}, \boldsymbol{e}) \in \left\{ \arg\max_{x \in X} G(\boldsymbol{x}, \boldsymbol{y} + \Delta \boldsymbol{y}) \middle| \frac{(\boldsymbol{x} - \boldsymbol{x}^*(\boldsymbol{y}))}{\|(\boldsymbol{x} - \boldsymbol{x}^*(\boldsymbol{y}))\|_2} = \pm \boldsymbol{e} \right\}$, we have

$$G(\boldsymbol{x}^*(\boldsymbol{y})+x_2{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y}) \quad < G(\boldsymbol{x}^*(\boldsymbol{y})+(x_2+\Delta x){*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y}) \le G(\boldsymbol{x}^*(\boldsymbol{y}+\Delta\boldsymbol{y},\boldsymbol{e}),\boldsymbol{y}+\Delta\boldsymbol{y}). \tag{F2}$$

Combining Eq. (F1) and Eq. (F2), we observe that $G(\boldsymbol{x},\boldsymbol{y}+\Delta\boldsymbol{y})$ first decreases strictly from $\boldsymbol{x}^*(\boldsymbol{y}+\Delta\boldsymbol{y},\boldsymbol{e})$ to $\boldsymbol{x}^*(\boldsymbol{y})+x_2{*}\boldsymbol{e}$ and then increases strictly from $\boldsymbol{x}^*(\boldsymbol{y})+x_2{*}\boldsymbol{e}$ to $\boldsymbol{x}^*(\boldsymbol{y})+(x_2+\Delta x){*}\boldsymbol{e}$, which contradicts that $G(\boldsymbol{x},\boldsymbol{y}+\Delta\boldsymbol{y})$ is quasi-concave in $\boldsymbol{x}$. Therefore, we have

$$\underline{x}\left(\boldsymbol{y},\Delta x\right) \le \left(\boldsymbol{x}^*\left(\boldsymbol{y}+\Delta\boldsymbol{y},\boldsymbol{e}\right)-\boldsymbol{x}^*(\boldsymbol{y})\right)^{\mathrm{T}}{*}\boldsymbol{e},$$

$$\forall \boldsymbol{x}^*\left(\boldsymbol{y}+\Delta\boldsymbol{y},\boldsymbol{e}\right) \in \left\{\arg\max_{\boldsymbol{x}\in X} G\left(\boldsymbol{x},\boldsymbol{y}+\Delta\boldsymbol{y}\right) \middle| \frac{(\boldsymbol{x}-\boldsymbol{x}^*(\boldsymbol{y}))^{\mathrm{T}}}{\|(\boldsymbol{x}-\boldsymbol{x}^*(\boldsymbol{y}))\|_2}\boldsymbol{e}=\pm 1\right\}.$$

The last two inequalities can be proved following the same procedures.

2. We prove the first inequality by showing that

$$\{x\ge 0 | G(\boldsymbol{x}^*(\boldsymbol{y})+x{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y})-L^*\|\Delta\boldsymbol{y}\|_2\} \subset \{x\ge 0 | \exists \Delta x>0, G(\boldsymbol{x}^*(\boldsymbol{y})+x{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y})+(x-\Delta x){*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})\}.$$

For an arbitrary $x_0\in\{x\ge 0 | G(\boldsymbol{x}^*(\boldsymbol{y})+x{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y})-L^*\|\Delta\boldsymbol{y}\|_2\}$, we argue that $x_0\neq 0$, because according to the Lipschitz condition, we have

$$G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y}+\Delta\boldsymbol{y}) \ge G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y}) - L^* \| \Delta\boldsymbol{y}\|_2.$$

Let $\Delta x_0 = x_0 > 0$. We observe that

$$G(\boldsymbol{x}^*(\boldsymbol{y})+x_0{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y}) - L^* \| \Delta\boldsymbol{y}\|_2 \le G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y}+\Delta\boldsymbol{y}) = G(\boldsymbol{x}^*(\boldsymbol{y})+(x_0-\Delta x_0){*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y}).$$

Therefore,

$$x_0\in\{x\ge 0 | \exists \Delta x>0, G(\boldsymbol{x}^*(\boldsymbol{y})+x{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y}) < G(\boldsymbol{x}^*(\boldsymbol{y})+(x-\Delta x){*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})\}.$$

Similarly, we have

$$\{x\le 0 | G(\boldsymbol{x}^*(\boldsymbol{y})+x{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y})-L^*\|\Delta\boldsymbol{y}\|_2\} \subset \{x\le 0 | \exists \Delta x>0, G(\boldsymbol{x}^*(\boldsymbol{y})+x{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y})+(x+\Delta x){*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})\}.$$

Therefore, we have proved that

$$\overline{x}(\boldsymbol{y}) \le \inf\{x\ge 0) | G(\boldsymbol{x}^*(\boldsymbol{y})+x{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y})-L\|\Delta\boldsymbol{y}\|_2\},$$
$$\underline{x}\left(\boldsymbol{y}\right) \le \sup\{x\le 0) | G(\boldsymbol{x}^*(\boldsymbol{y})+x{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y})-L\|\Delta\boldsymbol{y}\|_2\}.$$

3. We prove the inequality by contradiction. We assume that $\overline{x}(\boldsymbol{y}) - \underline{x}(\boldsymbol{y}) > 4\sqrt{L \| \Delta\boldsymbol{y}\|_2/\varepsilon}$. If we denote

$$x^{\mathrm{l}}(\boldsymbol{y}) = \sup\{x\le 0 | G(\boldsymbol{x}^*(\boldsymbol{y})+x{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y})-L\|\Delta\boldsymbol{y}\|_2\},$$
$$x^{\mathrm{r}}(\boldsymbol{y}) = \inf\{x\ge 0 | G(\boldsymbol{x}^*(\boldsymbol{y})+x{*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y})<G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y})-L\|\Delta\boldsymbol{y}\|_2\},$$

where $\boldsymbol{x}^*(\boldsymbol{y})+x^{\mathrm{l}}(\boldsymbol{y}){*}\boldsymbol{e},\boldsymbol{x}^*(\boldsymbol{y})+x^{\mathrm{r}}(\boldsymbol{y}){*}\boldsymbol{e}\in X$, we have

$$x^{\mathrm{r}}(\boldsymbol{y}) - x^{\mathrm{l}}(\boldsymbol{y}) > 4\sqrt{L \| \Delta\boldsymbol{y}\|_2/\varepsilon}. \tag{F3}$$

For the arbitrary minimum point $x^*(\boldsymbol{y})\in\arg\max_{\boldsymbol{x}\in X}G(\boldsymbol{x},\boldsymbol{y}+\Delta\boldsymbol{y})$. Eq. (F3) gives either $-x^{\mathrm{l}}(\boldsymbol{y}) > 2\sqrt{L \| \Delta\boldsymbol{y}\|_2/\varepsilon}$ or $x^{\mathrm{r}}(\boldsymbol{y}) > 2\sqrt{L \| \Delta\boldsymbol{y}\|_2/\varepsilon}$. Without loss of generality, we assume that $-x^{\mathrm{l}}(\boldsymbol{y}) > 2\sqrt{L \| \Delta\boldsymbol{y}\|_2/\varepsilon}$ to derive a contradiction in the following proof.

First, since $G(\boldsymbol{x},\boldsymbol{y})$ is continuous in $\boldsymbol{x}$, we have

$$G\left(\boldsymbol{x}^*(\boldsymbol{y})+x^{\mathrm{l}}(\boldsymbol{y}){*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y}\right) = G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y}) - L \| \Delta\boldsymbol{y}\|_2.$$

Consequently, by the Lipschitz condition, we have

$$G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y}+\Delta\boldsymbol{y}) \le G(\boldsymbol{x}^*(\boldsymbol{y}),\boldsymbol{y}) + L \| \Delta\boldsymbol{y}\|_2 = G\left(\boldsymbol{x}^*(\boldsymbol{y})+x^{\mathrm{l}}(\boldsymbol{y}){*}\boldsymbol{e},\boldsymbol{y}+\Delta\boldsymbol{y}\right) + 2L \| \Delta\boldsymbol{y}\|_2. \tag{F4}$$

Then, from the assumption that $\partial_x g(x_1,\boldsymbol{y}) - \partial_x g(x_2,\boldsymbol{y}) \le \varepsilon^*(x_2-x_1)$, $\forall x_1 > x_2$ and $\boldsymbol{x}^*(\boldsymbol{y})+x_1{*}\boldsymbol{e}$, $\boldsymbol{x}^*(\boldsymbol{y})+x_2{*}\boldsymbol{e}\in X$, we have

$$\partial_x g(\xi,\boldsymbol{y}+\Delta\boldsymbol{y}) \ge -\varepsilon\cdot\xi + \partial_x g(0,\boldsymbol{y}+\Delta\boldsymbol{y}) = -\varepsilon\cdot\xi, \quad \forall \xi<0, \boldsymbol{x}^*(\boldsymbol{y})+\xi{*}\boldsymbol{e}\in X.$$

As a result,

$$\begin{aligned} g(0,\boldsymbol{y}+\Delta\boldsymbol{y}) &= g\left(x^{\mathrm{l}}(\boldsymbol{y}),\boldsymbol{y}+\Delta\boldsymbol{y}\right) + \int_{x^{\mathrm{l}}(\boldsymbol{y})}^{0} \partial_x g(\xi,\boldsymbol{y}+\Delta\boldsymbol{y})\mathrm{d}\xi \\ &\ge g\left(x^{\mathrm{l}}(\boldsymbol{y}),\boldsymbol{y}+\Delta\boldsymbol{y}\right) - \int_{x^{\mathrm{l}}(\boldsymbol{y})}^{0} \varepsilon\cdot\xi\mathrm{d}\xi \\ &= g\left(x^{\mathrm{l}}(\boldsymbol{y}),\boldsymbol{y}+\Delta\boldsymbol{y}\right) - \frac{\varepsilon}{2}\cdot\xi^2\Big|_{\xi=x^{\mathrm{l}}(\boldsymbol{y})}^{\xi=0} \\ &= g\left(x^{\mathrm{l}}(\boldsymbol{y}),\boldsymbol{y}+\Delta\boldsymbol{y}\right) + \frac{\varepsilon}{2}\cdot\left(x^{\mathrm{l}}(\boldsymbol{y})\right)^2. \end{aligned}$$

Because $-x^{\mathrm{l}}(\boldsymbol{y}) > 2\sqrt{L \| \Delta\boldsymbol{y}\|_2/\varepsilon}$, the above inequality gives

$$g(0,\boldsymbol{y}+\Delta\boldsymbol{y}) > g\left(x^{\mathrm{l}}(\boldsymbol{y}),\boldsymbol{y}+\Delta\boldsymbol{y}\right) + 2L \| \Delta\boldsymbol{y}\|_2. \tag{F5}$$

We observe Eq. (F4) and Eq. (F5) together lead to a contradiction. Therefore, we have

$$\overline{x}(\boldsymbol{y}) - \underline{x}(\boldsymbol{y}) \le 4\sqrt{L \| \Delta\boldsymbol{y}\|_2/\varepsilon}.$$

Appendix G. Main parameter settings of the numerical experiment

In this section, we present the numerical settings for the tested networks in Fig. 6. We list the pipeline parameters, the descriptions of components of compressor stations and the physical parameters of the fitting compressor model formulation. The same structure is tested for two compressor station component settings. The first setting is just the same as Deng et al. (2019), in which the sub-network contains only one compressor in each station. In the second setting, each station contains 1, 2 or 3 nonidentical compressors, details are in Table 8. The pipeline parameters and compressor type parameters in Tables 7 and 9 all come from Deng et al. (2019). The size of numbers of compressors in each station is

consistent with network settings in previous related literature (; Deng et al., 2019, Behrooz (2016).

Table 7
Pipeline Parameters for the test network

Pipeline ID	Starting Node	End Node	Length, km	Elevation Change, m	Heat Transfer Factor, W/(m^2·K)
1	0	1	0	0	0
2	1	2	139.652	−6	4
3	2	3	13.34	0	4
4	3	4	101.86	−777.8	4
5	4	5	62.3	37.2	4
6	5	6	42.41	0	4
7	6	7	134.18	−113	4
8	7	8	152.37	−19.8	4
9	8	9	95.66	18	5
10	9	10	54.671	14.1	4

Table 8
Compressor station components for the test network

Station ID	Setting1		Setting2	
	Compressor number	Compressor Type/ID	Compressor number	Compressor Type/ID
1	1	1	1	1
2	1	2	2	2,6
3	1	3	2	3
4	1	4	1	4,7,8
5	1	5	3	5,9

Table 9
Compressor Model Parameters for the test network

ID	H_1	H_2	H_3	ETA_1	ETA_2	$SURGE_1$	$SURGE_2$	$STONE_1$	$STONE_2$	$Speed_{max}$	$Speed_{min}$
1	−5.35E-04	−3.71E-04	1.32E-04	−1.31E-03	2.14E-04	3420	2.77E-01	5900	8.61E-01	5000	3120
2	−9.81E-04	2.33E-05	2.08E-05	−1.30E-03	1.62E-04	5620	3.55E-01	8120	1.41E+00	5330	3510
3	−5.37E-04	−3.78E-04	1.31E-04	−1.31E-03	2.24E-04	3360	2.58E-01	5940	8.54E-01	4940	3180
4	−9.70E-04	3.39E-05	2.12E-05	−1.32E-03	1.76E-04	5100	3.57E-01	8140	1.24E+00	5320	3500
5	−8.88E-04	6.20E-05	2.37E-05	−1.17E-03	1.70E-04	5060	3.02E-01	6990	9.15E-01	6560	4050
6	−4.65E-04	−3.44E-04	6.56E-05	−1.48E-03	1.46E-04	5600	3.95E-01	10090	1.29E+00	4970	2960
7	−1.19E-03	1.64E-04	4.35E-05	−1.48E-03	3.21E-04	3520	1.45E-01	4650	5.63E-01	6340	4030
8	−4.38E-04	−3.42E-04	6.85E-05	−1.47E-03	1.57E-04	5580	3.84E-01	10290	1.33E+00	5080	3150
9	−5.82E-04	7.91E-05	7.50E-06	−6.89E-04	1.17E-04	5050	4.62E-01	8560	1.58E+00	6430	3730

References

Aßmann, D., Liers, F., Stingl, M., Vera, J.C., 2018. Deciding robust feasibility and infeasibility using a set containment approach: an application to stationary passive gas network operations. SIAM Journal on Optimization 28 (3), 2489–2517. https://doi.org/10.1137/17m112470x.

Aßmann, D., Liers, F., Stingl, M., 2019. Decomposable robust two-stage optimization: an application to gas network operations under uncertainty. Networks 74 (1), 40–61. https://doi.org/10.1002/net.21871.

Bard, J.F., 1991. Some properties of the bilevel programming problem. Journal of Optimization Theory and Applications 68 (2), 371–378. https://doi.org/10.1007/bf00941574.

Baxter, J., Bartlett, P.L., 2001. Infinite-horizon policy-gradient estimation. Journal of Artificial Intelligence Research 15, 319–350. https://doi.org/10.1613/jair.806.

Behrooz, H.A., 2016. Managing demand uncertainty in natural gas transmission networks. Journal of Natural Gas Science and Engineering 34, 100–111. https://doi.org/10.1016/j.jngse.2016.06.051.

Ben-Tal, A., Goryashko, A., Guslitzer, E., Nemirovski, A., 2004. Adjustable robust solutions of uncertain linear programs. Mathematical Programming 99 (2), 351–376. https://doi.org/10.1007/s10107-003-0454-y.

Bertsimas, D., Sim, M., 2003. Robust discrete optimization and network flows. Mathematical Programming 98 (1), 49–71. https://doi.org/10.1007/s10107-003-0396-4.

Bertsimas, D., Sim, M., 2004. The price of robustness. Operations Research 52, 35–53. https://doi.org/10.1287/opre.1030.0065.

Borraz-Sánchez, C., Haugland, D., 2011. Minimizing fuel cost in gas transmission networks by dynamic programming and adaptive discretization. Computers & Industrial Engineering 61, 364–372. https://doi.org/10.1016/j.cie.2010.07.012.

Chen, X., Zhang, Y., 2009. Uncertain linear programs: extended affinely adjustable robust counterparts. Operations Research 57, 1469–1482. https://doi.org/10.1287/opre.1080.0605.

Cobos-Zaleta, D., Ríos-Mercado, R.Z., 2002. A Minlp Model for Minimizing Fuel Consumption on Natural Gas Pipeline Networks. XI Latin-Ibero-American conference on operations research, pp. 90–94.

Deng, T.H., Liang, Y., Zhang, S., Ren, J.Z., Zheng, S.Y., 2019. A dynamic programming approach to power consumption minimization in gunbarrel natural gas networks with nonidentical compressor units. INFORMS Journal on Computing 31 (3), 593–611. https://doi.org/10.1287/ijoc.2018.0833.

Deng, T.H., 2016. Minimization of power usage in a compressor station with multiple compressors. Journal of Energy Engineering 142 (4), 04015048. https://doi.org/10.1061/(asce)ey.1943-7897.0000325.

De Wolf, D., Smeers, Y., 2000. The gas transmission problem solved by an extension of the simplex algorithm. Management Science 46 (11), 1454–1465. https://doi.org/10.1287/mnsc.46.11.1454.12087.

Gabrel, V., Murat, C., Thiele, A., 2014. Recent advances in robust optimization: an overview. European Journal of Operational Research 235 (3), 471–483. https://doi.org/10.1016/j.ejor.2013.09.036.

Geißler, B., Morsi, A., Schewe, L., Schmidt, M., 2015. Solving power-constrained gas transportation problems using an MIP-based alternating direction method. Computers & Chemical Engineering 82, 303–317. https://doi.org/10.1016/

j.compchemeng.2015.07.005.
Gorissen, B.L., Yanıkoğlu, İ., den Hertog, D., 2015. A practical guide to robust optimization. Omega 53, 124–137. https://doi.org/10.1016/j.omega.2014.12.006.
Gotzes, C., Heitsch, H., Henrion, R., Schultz, R., 2016. On the quantification of nomination feasibility in stationary gas networks with random load. Mathematical Methods of Operations Research 84 (2), 427–457. https://doi.org/10.1007/s00186-016-0564-y.
Hanasusanto, G.A., Kuhn, D., Wiesemann, W., 2015. K-adaptability in two-stage robust binary programming. Operations Research 63 (4), 877–891. https://doi.org/10.1016/j.orl.2015.10.006.
Han, J., Xu, Y., Liu, D., Zhao, Y., Zhao, Z., Zhou, S., Deng, T., Xue, M., Ye, J., Shen, Z.M., 2019. Operations research enables better planning of natural gas pipelines. INFORMS Journal on Applied Analytics 49 (1), 23–39. https://doi.org/10.1287/inte.2018.0974.
Hansen, P., Jaumard, B., Savard, G., 1992. New branch-and-bound rules for linear bilevel programming. SIAM Journal on Scientific and Statistical Computing 13 (5), 1194–1217. https://doi.org/10.1137/0913069.
Huang, W., Chen, J., Fu, C., Huang, Y., 2019. Approach for natural gas to be a primary energy source in China. Frontiers of Engineering Management 6 (4), 467–476. https://doi.org/10.1007/s42524-019-0068-6.
Hu, B., 2014. Oil and gas cooperation between China and Central Asia in an environment of political and resource competition. Petroleum Science 11 (4), 596–605. https://doi.org/10.1007/s12182-014-0377-7.
Konidaris, G., Osentoski, S., Thomas, P., 2011. Value Function Approximation in Reinforcement Learning Using the Fourier Basis. Twenty-fifth AAAI conference on artificial intelligence.
Liang, T., Chai, J., Zhang, Y.J., Zhang, Z.G., 2019. Refined analysis and prediction of natural gas consumption in China. Journal of Management Science and Engineering 4 (2), 91–104. https://doi.org/10.1016/j.jmse.2019.07.001.
Li, S.Q., Zhang, B.S., Tang, X., 2016. Forecasting of China's natural gas production and its policy implications. Petroleum Science 13 (3), 592–603. https://doi.org/10.1007/s12182-016-0101-x.
Martin, A., Möller, M., Moritz, S., 2006. Mixed integer models for the stationary case of gas network optimization. Mathematical Programming 105 (2), 563–582. https://doi.org/10.1007/s10107-005-0665-5.
Ma, Y., Li, Y., 2010. Analysis of the supply-demand status of China's natural gas to 2020. Petroleum Science 7 (1), 132–135. https://doi.org/10.1007/s12182-010-0017-9.
Nguyen, H.H., Uraikul, V., Chan, C.W., Tontiwachwuthikul, P., 2008. A comparison of automation techniques for optimization of compressor scheduling. Advances in Engineering Software 39 (3), 178–188. https://doi.org/10.1016/j.advengsoft.2007.02.003.
Powell, W.B., 2007. Approximate Dynamic Programming: Solving the Curses of Dimensionality, 703. John Wiley & Sons. https://doi.org/10.1002/9780470182963.
Powell, W.B., 2009. What you should know about approximate dynamic programming. Naval Research Logistics (NRL) 56 (3), 239–249. https://doi.org/10.1002/nav.20347.
Ríos-Mercado, R.Z., Borraz-Sánchez, C., 2015. Optimization problems in natural gas transportation systems: a state-of-the-art review. Applied Energy 147, 536–555. https://doi.org/10.1016/j.apenergy.2015.03.017.
Rogers, D.F., Plante, R.D., Wong, R.T., Evans, J.R., 1991. Aggregation and disaggregation techniques and methodology in optimization. Operations Research 39 (4), 553–582. https://doi.org/10.1287/opre.39.4.553.
Sutton, R.S., McAllester, D.A., Singh, S.P., Mansour, Y., 2000. Policy gradient methods for reinforcement learning with function approximation. Advances in Neural Information Processing Systems 1057–1063.
Vahid-Pakdel, M.J., Nojavan, S., Mohammadi-Ivatloo, B., Zare, K., 2017. Stochastic optimization of energy hub operation with consideration of thermal energy market and demand response. Energy Conversion and Management 145, 117–128. https://doi.org/10.1016/j.enconman.2017.04.074.
Wang, M., Yu, H., Jing, R., Liu, H., Chen, P., Li, C., 2020. Combined multi-objective optimization and robustness analysis framework for building integrated energy system under uncertainty. Energy Conversion and Management 208, 112589. https://doi.org/10.1016/j.enconman.2020.112589.
Wintergerst, D., 2017. Application of Chance Constrained Optimization to Gas Networks.
Wong, P.J., Larson, R.E., 1968. Optimization of tree-structured natural-gas transmission networks. Journal of Mathematical Analysis and Applications 24 (3), 613–626. https://doi.org/10.1016/0022-247x(68)90014-0.
Wu, S., Rios-Mercado, R.Z., Boyd, E.A., Scott, R.L., 2000. Model relaxations for the fuel cost minimization of steady-state gas pipeline networks. Mathematical and Computer Modelling 31 (2–3), 197–220. https://doi.org/10.1016/s0895-7177(99)00232-0.
Xue, M.Y., Deng, T.H., Liu, D., 2016. CNPC uses an iterative two-stage convex relaxation approach to operate natural gas pipelines. Interfaces 46 (6), 533–546. https://doi.org/10.1287/inte.2016.0867.
Zeng, B., Zhao, L., 2013. Solving two-stage robust optimization problems using a column-and-constraint generation method. Operations Research Letters 41 (5), 457–461. https://doi.org/10.1016/j.orl.2013.05.003.
Zhang, S., Liu, S., Deng, T., Shen, Z.M., 2020. Transient-state natural gas transmission in gunbarrel pipeline networks. INFORMS Journal on Computing 32 (3), 697–713. https://doi.org/10.1287/ijoc.2019.0904.
Zhao, L., Zeng, B., 2012. An exact algorithm for two-stage robust optimization with mixed integer recourse problems. Optimization-Online. org.
Zhao, W., Jia, A., Zhang, G., 2019. Analysis and countermeasures of natural gas development in China. Frontiers of Engineering Management 6 (4), 77–484. https://doi.org/10.1007/s42524-019-0064-x.