Global Method for a Class of Operation Optimization Problem in Steel Rolling Systems

Lianjie Tang,[†,‖] Lixin Tang,[∗,‡] Jiyin Liu,[¶] and Cong Cheng[§]

Key Laboratory of Data Analytics and Optimization for Smart Industry (Northeastern University), Ministry of Education, China, Institute of Industrial & Systems Engineering, Northeastern University, Shenyang, 110819, China, School of Business and Economics, Loughborough University, Leicestershire LE11 3TU, United Kingdom, and Liaoning Engineering Laboratory of Operations Analytics and Optimization for Smart Industry, Northeastern University, Shenyang 110819, China

E-mail: lixintang@mail.neu.edu.cn.

Abstract

Many steel products are produced in hot or cold rolling lines with multiple stands. The steel material becomes thinner after being rolled at each stand. Steady-state parameters for controlling the rolling line need to be set so as to satisfy the final product specifications and minimize the total energy consumption. This paper develops a generalized geometric programming model for this setting problem and proposes a global method for solving it. The

---

[∗]To whom correspondence should be addressed
[†]Key Laboratory of Data Analytics and Optimization for Smart Industry (Northeastern University), Ministry of Education, China
[‡]Institute of Industrial & Systems Engineering, Northeastern University, Shenyang, 110819, China
[¶]School of Business and Economics, Loughborough University, Leicestershire LE11 3TU, United Kingdom
[§]Liaoning Engineering Laboratory of Operations Analytics and Optimization for Smart Industry, Northeastern University, Shenyang 110819, China
[‖]Liaoning Key Laboratory of Manufacturing System and Logistics, Northeastern University, Shenyang 110819, China

model can be expressed with a linear objective function and a set of constraints including non-convex ones. Through constructing lower bounds of some components, the constraints can be converted to convex ones approximately. A sequential approximation method is proposed in a gradually reduced interval to improve accuracy and efficiency. However, the resulting convex programming model in each iteration is still complicated. To reduce the power, it is transformed into a second-order cone programming (SOCP) model and solved using alternating direction method of multipliers(ADMM). The effectiveness of the global method is tested using real data from a hot-rolling line with seven stands. The results demonstrate that the proposed global method solves the problem effectively and reduces the energy consumption.

1. INTRODUCTION

As with the chemical industry, iron and steel production is a process industry,[1–5] but it has its own characteristics, such as a production process with high temperature and high energy consumption,[6,7] production scheduling with complicated planning,[8,9] and different products with different phase changes and technologies.[10–12] Customer demands for small batches and varieties of products have been increasing recently. The product quality requirements in terms of strip thickness, strip crown, and strip flatness are also higher than ever. Those problems bring some challenges for traditional production mode in iron and steel enterprises. Steel rolling process as shown in Figure 1 is a key stage of steel production in determining product quality and affecting energy consumption.

A hot or cold rolling line consists of multiple rolling stands. The steel material gets thinner after being extruded at each stand. Given the equipment of the production line, its efficiency and energy consumption as well as the product quality depend largely on how the line is controlled. The control of a rolling line is realized by load distribution which sets steady-state control parameters and an automatic gauge control(AGC) system[13] which dynamically controls the equipment based on the setting parameters. Figure 2 shows the relationship between them. Load distribution is an operation optimization problem (OOP). Taking product (steel strip) and plant data as input,
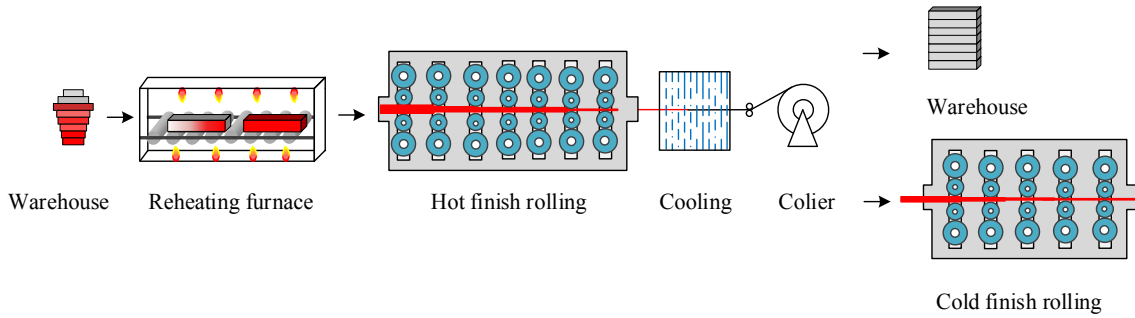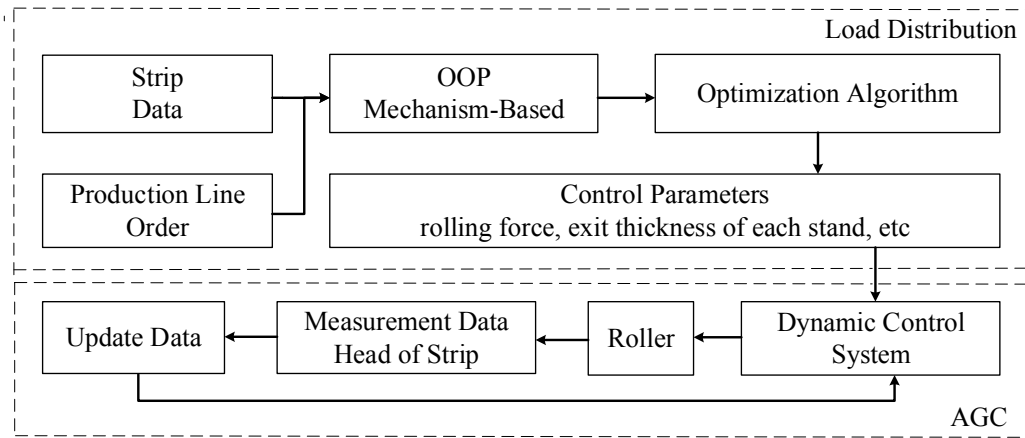
Figure 1: Steel rolling process.

Figure 2: Relationship graph between load distribution and AGC.

the OOP determines the exit thickness of each stand which can be translated to control parameter settings accordingly. As each customer demand is different, the OOP needs to be solved for each production batch. Moreover, before the actual production of each batch, the line needs some trial runs on sample materials until achieving the required rolling effect. OOP is resolved on the basis of the result after each trial run. Effective and efficient solution of the OOP can reduce the number of trials needed and save the total time and materials of the trial process.

There have been some approaches reported in the literature for solving the load distribution problem. Some researchers designed control parameters without considering optimal performance criteria. Lu[14] modeled a set of nonlinear equations assuming relative motor power at different stands and proposed a balance iteration method to solve them without calculating inverse of Ja-

cobian matrix in each iteration. Li[15] presented an improved Newton–Raphson algorithm to solve similar nonlinear equations, which could deal with singular coefficient matrix. The method is not sensitive to the initial value. Many others considered certain objective and modeled the optimization problem based on the relationships among the variables. These models are nonlinear nonconvex optimization problems and corresponding methods had been adopted to solve them. Zhang[16] introduced a dynamic programming algorithm to solve the load distribution problem, where stands were seen as stages. The objective was to minimize the deviation of the rolling crown from a set target. Ozsoy[17] introduced a sequence quadratic programming method to solve the load distribution problem in a hot-rolling line so as to minimize cost of time and energy. Yao et al.[18] proposed an improved differential evolution algorithm for a three-stage multiobjective OOP in hot strip rolling. Chen et al.[7] presented a hybrid self-adaptive strategy to improve genetic algorithm for solving the OOP in hot rolling production process. The aforementioned methods ignore some inherent structural properties of OOP such as the convexity so cannot get the global optimal solution. These structural properties could be used for developing methods to improve the solution speed and obtain the convex optimization problem.[19] Recently, geometric programming (GP) modeling for expressing a practical problem has been proposed.[20] This method provides some tricks to help transform a problem into GP or generalized geometric programming(GGP) for the problems with complex posynomial. Because GP has a special form, GP modeling is a powerful optimization technique and has been widely used for solving a variety of nonlinear optimization problems in chemical engineering,[21–25] condenser design,[26] circuit design,[27] and a number of other areas.[28] Hershenson et al.[29] described a new method for integrated circuit design via geometric programming. Singh et al.[30] presented a convex optimization method for the gate sizing problem under the worst case, which could be transformed into conventional geometric programming problems. Several other practical problems[31–33] were formulated as GGP, in which posynomial functions in the objective and constraints were replaced with signomial.

However, GGP is significantly harder to solve than GP since it may have nonpositive terms. Fortunately, GGP has inherent convexity without the exponential transformation or convex relax-

ation. Among the current methods, Tsai et al.[34] presented two convex conditions and several conversion rules to identify trivariate monomials. Gounaris and Floudas[35] have established a way to infer the convexity of multivariate functions that can be expressed as products of univariate functions satisfying a set of sufficient conditions. Floudas and Lu[36] presented a convex relaxation to deal with the signomial constraints. Lu[37] proposed a beta method for solving GGP, which maximally improved the efficiency of convexification. Nonconvex terms can be approximated as piece-wise linear functions. According to the definition of special ordered set (SOS), the piece-wise function can be transformed into linear functions by introducing integer variables. Thus, the original problem can be converted to a mixed integer convex programming problem.[19,37–39] However, the accuracy of the linear approximation depends on the density of break points. The more break points there are, the more integer variables will be needed. This leads to an increase in the number of additional variables and constraints. The computational difficulty increases with the scale of the approximate problem. Hence, it is necessary to develop methods to avoid using integer variables.

Motivated by convex optimization techniques in the aforementioned papers, we present a novel convex approximate method to get optimal control variables of the OOP for rolling load distribution. The main differences between this paper and previous papers are as follows.

(a)Different from the traditional nonlinear modeling method, the convex modeling of OOP is proposed with structural characteristics.

(b)Unlike piece-wise linear method, the underestimate of nonconvex term can be approximated as a convex function by using the lower and upper limits of some variables. The approximate convex model can be also transformed into a SOCP, which can be solved in polynomial time.

(c) To improve the accuracy of approximate model, we use monotonicity of the constraints to develop the sequential approximation method, in which the variable limits can be tightened in iterations and more precise approximation functions can be obtained on the narrower interval.

The remaining parts of this paper are organized as follows. In the next section, we present the OOP model for rolling load distribution to minimize energy consumption, analyze its struc-

tural properties, and transform it to SOCP. Section 3 proposes a global method based on ADMM framework to solve the OOP. Section 4 presents experiments testing the methods on 10 set of practical data each representing a production batch in a hot rolling line. The final section gives some conclusions.

2. MODELING AND TRANSFORMATION

2.0 Model of OOP. Operation optimization problems are a class of parameter setting problems commonly seen in process industry, especially steel rolling systems. To meet production and quality requirements, the problems are modeled considering some operation conditions in the objective function and constraints. These conditions are such complex that may be reducible to a mathematical model by mechanism process.[43]

A. Deformation Model. The material thickness depends on the gap between the rollers at each stand. For any stand i, the gap is controlled by the rolling force, torque and power models.

$$P_i = BQ_iK_il_i \tag{1a}$$

$$M_i = 2P_i\sqrt{R_i(h_{i-1} - h_i)}\phi \tag{1b}$$

$$N_i = \frac{2\pi \times 10^3}{60 \times 102}n_iM_i \tag{1c}$$

where B is the slab width under the assumption that it remains a constant in a finishing mill, Q_i is the stress state coefficient, K_i is the deformation-resistance coefficient, l_i is deformation length, R_i is the work-roller radius, h_i is the exit thickness, ϕ is the arm coefficient, and n_i is the work-roller circumferential velocity. The definitions of Q_i and K_i are given as follows:

$$Q_i = a_0 + a_1\frac{\sqrt{R_i(h_{i-1} - h_i)}}{h_{i-1} + h_i} \tag{2a}$$

$$K_i = 1.15a_3\exp(a_4T_i + a_5)(\frac{\mu_i}{10})^{a_6T_i+a_7}[a_8(\frac{\gamma_i}{0.4})^{a_2} - (a_8 - 1)(\frac{\gamma_i}{0.4})] \tag{2b}$$

where T_i is the rolling temperature, μ_i is the deformation velocity, γ_i is the deformation degree which is calculated by $\ln \gamma_i = a_9 + a_{10} \ln \varepsilon_i$, ε_i is the relative reduction, and a_0–a_{10} are the regression coefficients.

B. Temperature Model. The temperature of steel strip is an important parameter in rolling processes, which has a directly influence on the deformation and surface quality of the slab. The slab temperature drop model at each stand is given as follows:

$$T_i = T_w + (T_f - T_w) \exp(-K_T \frac{\sum_{j=1}^{i} L_j}{h_{\mathcal{N}} v_{\mathcal{N}}}) \tag{3}$$

where T_w is the cooling water temperature, L_j is the distance of two stands, $v_{\mathcal{N}}$ is the slab exit velocity, K_T is the cooling coefficient, and T_f is the radiation temperature.[7]

The temperature measurement instrument is installed at both ends of a finishing mill. On the basis of data obtained by the thermometer, K_T is evaluated as follows:

$$K_T = \frac{T_{\text{enter}} - T_{\text{out}}}{T_{\text{enter}}} \times \frac{h_{\mathcal{N}} v_{\mathcal{N}}}{L} \tag{4}$$

where T_{enter} is the temperature before entering the finishing mill, T_{out} is the temperature after leaving the finishing mill, and L is the total length of the finishing mill.

C. Two Velocity Models. On the basis of mass conservation law, the slab velocity can be written as follows:

$$v_i = \frac{v_{\mathcal{N}} h_{\mathcal{N}}}{h_i} \tag{5}$$

The slab deformation velocity is related to the deformation degree in a unit time, which can be expressed as

$$\mu_i = \frac{v_i}{\sqrt{R_i(h_{i-1} - h_i)}} \gamma_i \tag{6}$$

D. Shape Model. The crown is a basic index for maintaining the flatness of the strip shape. Assuming that some factors (e.g., bending roll force, hot roll type) can be ignored, the crown

model can be simplified as the following expression.

$$CR_i = \frac{P_i}{K_P} \tag{7}$$

where K_P is the lateral stiffness.

It can be seen that the models are functions of the exit thickness at each stand. So, the exit thicknesses are defined as the decision variables in the rolling process. Because energy consumption is an important performance index in steel production,[7,40] OOP determines an optimal set of the rolling thickness of each stand that minimizes the total rolling energy consumption, subject to meeting the following constraints.

- The slab can successfully enter the first stand to ensure the normal production.

- The equipment's capacity has been brought into full play in middle stands to reach a larger reduction of the thickness.

- The quality requirement of the strip shape is such that the crown is controlled in the last few stands.

- The values of operation parameters are not allowed to be greater than the maximum value of the equipment capacity.

Integrating the mechanism models, the operation optimization problem formulation of $\mathcal{N}$ s-

tands is written as

$$\min_{h_i} \quad \sum_{i=1}^{\mathcal{N}} N_i(h_{i-1}, h_i) \tag{8}$$

$$\text{s.t.} \quad \bar{\delta}_{1,i} P_{i+1}(h_i, h_{i+1}) \leq \delta_{1,i} P_i(h_{i-1}, h_i), i = 1, \cdots, \mathcal{N} - 1 \tag{9}$$

$$\bar{\delta}_{2,i} M_{i+1}(h_i, h_{i+1}) \leq \delta_{2,i} M_i(h_{i-1}, h_i), i = 1, \cdots, \mathcal{N} - 1 \tag{10}$$

$$\bar{\delta}_{3,i} N_{i+1}(h_i, h_{i+1}) \leq \delta_{3,i} N_i(h_{i-1}, h_i), i = 1, \cdots, \mathcal{N} - 1 \tag{11}$$

$$\delta_{5,i} \leq \frac{CR_{i+1}(h_i, h_{i+1})}{h_{i+1}} - \frac{CR_i(h_{i-1}, h_i)}{h_i} \leq \delta_{4,i}, i = \hat{\mathcal{N}}, \cdots, \mathcal{N} - 1 \tag{12}$$

$$0 \leq P_i(h_{i-1}, h_i) \leq P_{i\max}, i = 1, \cdots, \mathcal{N} \tag{13}$$

$$0 \leq M_i(h_{i-1}, h_i) \leq M_{i\max}, i = 1, \cdots, \mathcal{N} \tag{14}$$

$$0 \leq N_i(h_{i-1}, h_i) \leq N_{i\max}, i = 1, \cdots, \mathcal{N} \tag{15}$$

$$h_{i-1} \geq h_i > 0, i = 1, \cdots, \mathcal{N} \tag{16}$$

where P_i, M_i, N_i, and CR_i are the rolling force, torque, power, and crown mechanism models of h_i and h_{i-1}, respectively. The thickness h_i of the ith stand is a decision variable. $P_{i\max}, M_{i\max}$, and $N_{i\max}$ are the given maximum rolling force, maximum rolling torque, and maximum power of the ith stand, respectively. h_0 is the original slab thickness before being rolled. $h_{\mathcal{N}}$ and $CR_{\mathcal{N}}$ are the required thickness and target crown of the final product. δ are model parameters. Constraints 9-12 are called technological conditions. Constraints 9-11 not only guarantee that the strip can be easily to bitten into the first stand but also require the stability of the rolling process. Constraint 12 ensures that the shape variation is within the range $[\delta_{4,i}, \delta_{5,i}]$. Constraints 13-15 are capacity constraints of the equipment require that the rolling force, rolling torque, and rolling power of each stand should not exceed their maximum limits. The decision variables are required to be positive in constraint 16.

In a practical rolling line, the fact that the value of the crown varies over a small interval is acceptable for manufacturers. So, we focus on energy consumption in the production process, while the shape requirements are limited by the constraints. According to slab information, we adjust the parameters δ to meet product requirement. These in OOP modeling are different from the

literature.[7,40] Moreover, mechanism models are nonlinear mathematical models. The traditional method can be employed to deal with it, which may lose the structural features of the optimization problem. So, we hope to find a method to analyze these structure and improve the solution speed.

Fortunately, using well-posed algebraic transformation methods reported by Boyd,[20] the OOP can be transformed into the following GGP form.

$$
(P): \begin{cases}
\min & y_0 \\
\text{s.t.} & G_0(h,y) \leq y_0 \\
& G_m(h,y) \leq d_m,\ m=1,\cdots,M \\
& g_m(h,y) \leq \bar{d}_m,\ m=1,\cdots,K \\
& h_{\mathcal{N}} \geq h \geq h_0, y^u \geq y \geq y^l
\end{cases}
$$

where $G_m(h,y) = \sum\limits_t c_{mt} \prod\limits_{i=1}^{n} h_i^{\gamma_{mti}} \prod\limits_{i=1}^{\bar{n}} y_i^{\bar{\eta}_{mti}}$ $(m=0,1\cdots,M)$ is a nonlinear polynomial function, $\gamma_{mti} \in \mathbb{R}, \bar{\eta}_{mti} \in \mathbb{R}, c_{mt} \in \mathbb{R}, d_m \in \mathbb{R}^+$ are constants, h_0 and $h_{\mathcal{N}}$ are given thicknesses, $g_m(h,y) = \sum\limits_j a_{mj} h_j + \sum\limits_k b_{mk} y_k$ is a linear function, $a_{mj}, b_{mk}, \bar{d}_m \in \mathbb{R}$ are constants, $y = \{y_0, \cdots, y_{\mathcal{N}}\}$ is an intermediate vector, and y^l and y^u are the lower and the upper bounds, respectively.

Proposition 0.1: $h^* \in \mathbb{R}^{\mathcal{N}-1}$ is an optimal solution of the (OOP) if and only if there are positive values for $(y_0^*, y_1^*, \cdots, y_{\mathcal{N}}^*)$ such that vector (h^*, y^*) is an optimal solution of problem P.

Proof. The objective function value of OOP is nonnegative, thus we first should minimize the positive independent variable y_0 subject to the original constraints and the additional constraint $\sum N_i \leq y_0$. This is an equivalent transformation.

Let the complex constraints of the (OOP) be denote as $F_{i+1}(h) \leq F_i(h)$ for convenience. According to the previous study,[42] h is a feasible solution to these constraints of the (OOP) if and only if there are positive values for y such that the augmented vector (h,y) is a feasible solution to the constraint $F_{i+1}(h) \leq y_i \leq F_i(h)$.

Because $F(h)$ is still not a posynomial, we use some tricks[20] to deal with the constraint $F_{\bar{i}}(h) = f_{\bar{i},1}(h)^{\bar{\alpha}_{\bar{i},1}} + f_{\bar{i},2}(h)^{\bar{\alpha}_{\bar{i},2}}$ in different cases of $f_{\bar{i},j}(h)$ and $\bar{\alpha}_{\bar{i},j}, j=1,2, \bar{i}=i,i+1$. For the

left-side inequality, if $f_{i+1,j}(h)$ is a posynomial and $\bar{\alpha}_{i+1,j} \geq 0$, then the inequality $f_{i+1,1}(h)^{\bar{\alpha}_{i+1,1}} + f_{i+1,2}(h)^{\bar{\alpha}_{i+1,2}} \leq y_{i+1,1}$ can be written as

$$f_{i+1,1}(h) \leq y_{i+1,2} \tag{17a}$$

$$f_{i+1,2}(h) \leq y_{i+1,3} \tag{17b}$$

$$y_{i+1,2}^{\bar{\alpha}_{i+1,1}} + y_{i+1,3}^{\bar{\alpha}_{i+1,2}} \leq y_{i+1,1} \tag{17c}$$

When a feasible solution $(h, y_{i+1,1})$ satisfies $f_{i+1,1}(h)^{\bar{\alpha}_{i+1,1}} + f_{i+1,2}(h)^{\bar{\alpha}_{i+1,2}} \leq y_{i+1,1}, y_{i+1,2} = f_{i+1,1}(h)$ and $y_{i+1,3} = f_{i+1,2}(h)$ satisfy the inequalities 17a–17c. Conversely, if $h, y_{i+1,1}, y_{i+1,2}$ and $y_{i+1,3}$ satisfy the inequalities 17a–17c, then we get $f_{i+1,1}(h)^{\bar{\alpha}_{i+1,1}} + f_{i+1,2}(h)^{\bar{\alpha}_{i+1,2}} \leq y_{i+1,2}^{\bar{\alpha}_{i+1,1}} + y_{i+1,3}^{\bar{\alpha}_{i+1,2}} \leq y_{i+1,1}$, in which $f_{i+1,j}^{\bar{\alpha}_{i+1,j}}$ is an increasing function. If $f_{i+1,j}(h)$ is not posynomial, then the equivalence form of the left-side inequality includes $f_{i+1,1}(h) \leq y_{i+1,2}, f_{i+1,2}(h) \leq y_{i+1,3}, f_{i+1,1}(h) \geq y_{i+1,2}, f_{i+1,2}(h) \geq y_{i+1,3}$ and $y_{i+1,2}^{\bar{\alpha}_{i+1,1}} + y_{i+1,3}^{\bar{\alpha}_{i+1,2}} \leq y_{i+1,1}$.

The right-side one is $y_{i+1,1} \leq f_{i,1}(h)^{\bar{\alpha}_{i,1}} + f_{i,2}(h)^{\bar{\alpha}_{i,2}}$, where $f_{i,j}(h)$ is a posynomial and $\bar{\alpha}_{i,j} \geq 0$. Similarly, this constraint can be equivalent to $y_{i,2} \leq f_{i,1}(h), y_{i,3} \leq f_{i,2}(h), y_{i+1,1} \leq y_{i,2}^{\bar{\alpha}_{i,1}} + y_{i,3}^{\bar{\alpha}_{i,2}}$. If $f_{i,j}(h)$ is not posynomial, then we add two inequations $y_{i,2} \geq f_{i,1}(h)$ and $y_{i,3} \geq f_{i,2}(h)$ to constraints.

Thus, the problem OOP can be equivalent to the problem P and the proof is completed.

Although the nonlinear parts of problem P seem not to have any structure characteristics at first sight, we may exploit the inherent convexity of Hessian matrix and monotonicity of the block constraints.

2.1 Structure Characterization. If every monomial is convex, the sum of them is also a convex function. Hence, our basic approach is to analyze every monomial term. For simplicity, we use $g(x) = \hat{c} \prod_{i=1}^{n} x_i^{\alpha_i}$ to represent any term in the constraints $G_m(h, y)$, where $\hat{c} \in \mathbb{R}$ is an arbitrary constant, $\alpha_i \in \mathbb{R}$ is an arbitrary exponent and $x \in \mathbb{R}^n$ is a non-negative variable vector.

Given a twice continuously differentiable function $g(x) = \hat{c} \prod_{i=1}^{n} x_i^{\alpha_i}$, we have the following results:[37]

- Convex property a: when $\hat{c} \geq 0$ and $n = 1$, $g(x)$ is convex on S, if $\alpha \geq 1$;

11

- Convex property b: when $\hat{c} \geq 0$ and $n \geq 1$, $g(x)$ is convex on S, if $\alpha_i \leq 0$ for $\forall i \in I$;

- Convex property c: when $\hat{c} \geq 0$ and $n \geq 2$, $g(x)$ is convex on S, if $\alpha_j \geq 1, \alpha_i \leq 0, \forall i \in I \backslash \{j\}, 1 \leq \alpha_1 + \cdots + \alpha_n$;

- Convex property d: when $\hat{c} \leq 0$ and $n \geq 1$, $g(x)$ is convex on S, if $\alpha_1 + \cdots + \alpha_n \leq 1$ for any weight $\alpha_i \geq 0, \forall i \in I$,

where $S = \{x_i | x_i > 0, i \in I\}, I = \{1, 2, \cdots, n\}$.

For our subsequent developments, we will discuss the monotonicity of constraints in P. We first let

$$\gamma_{mi} := \min\{0, \gamma_{mti}, \bar{\eta}_{mti} | t = 1, \cdots, T_m\}, m = 0, 1, \cdots, M$$

Next, function $G_m(h, y)$ in the constraint can be reformulated as

$$G_m(h, y) = \frac{\sum\limits_{t=1}^{T_m} c_{mt} \prod\limits_{i=1}^{n} h_i^{\gamma_{mti} - \gamma_{mi}} \prod\limits_{i=1}^{\bar{n}} y_i^{\bar{\eta}_{mti} - \gamma_{mi}}}{\prod\limits_{i=1}^{n} h_i^{-\gamma_{mi}} \prod\limits_{i=1}^{\bar{n}} y_i^{-\gamma_{mi}}}, m = 0, 1, \cdots, M$$

where $\gamma_{mti} - \gamma_{mi} \geq 0, \bar{\eta}_{mti} - \gamma_{mi} \geq 0, -\gamma_{mi} \geq 0$. Thus, monotonous increasing functions are defined as $G_m^+(h, y) = \sum\limits_{t \in T_m^+} c_{mt} \prod\limits_{i=1}^{n} h_i^{\gamma_{mti} - \gamma_{mi}} \prod\limits_{i=1}^{\bar{n}} y_i^{\bar{\eta}_{mti} - \gamma_{mi}} (m = 0, 1, \cdots, M)$,

$G_0^-(h, y) = \sum\limits_{t \in T_0^-} (-c_{0t}) \prod\limits_{i=1}^{n} h_i^{\gamma_{0ti} - \gamma_{0i}} \prod\limits_{i=1}^{\bar{n}} y_i^{\bar{\eta}_{0ti} - \gamma_{0i}} - y_0 \prod\limits_{i=1}^{n} h_i^{-\gamma_{0i}} \prod\limits_{i=1}^{\bar{n}} y_i^{-\gamma_{0i}}$, and

$G_m^-(h, y) = \sum\limits_{t \in T_m^-} (-c_{mt}) \prod\limits_{i=1}^{n} h_i^{\gamma_{mti} - \gamma_{mi}} \prod\limits_{i=1}^{\bar{n}} y_i^{\bar{\eta}_{mti} - \gamma_{mi}} - d_m \prod\limits_{i=1}^{n} h_i^{-\gamma_{mi}} \prod\limits_{i=1}^{\bar{n}} y_i^{-\gamma_{mi}} (m = 1, \cdots, M)$, where $T_m^+ := \{i | c_{im} > 0, i = 1, \cdots, T_m\}, T_m^- := \{i | c_{im} < 0, i = 1, \cdots, T_m\}$.

The convex conditions and monotonicity of the functions in the constraints are summarized so that these characteristics can be utilized for efficient problem solution.

2.2 Convexification Strategies for Nonconvex Cases. In the best case, we can get an exact convex operation optimization problem. When this is not possible, the nonlinear constraints can be divided into a convex part and a nonconvex part. By constructing the lower bound on some components of the nonconvex part, the operation optimization problem can be approximated to a

convex formulation based on the convex conditions.

For the sake of briefness, the nonconvex term can be formulated as $f^{\text{nonc}}(x) = \hat{c}x_1^{\alpha_1}x_2^{\alpha_2}\cdots x_n^{\alpha_n}$. Using variable replacement in $f^{\text{nonc}}(x)$, we can have

$$f^{\text{conv}}(x) = \hat{c}x_1^{\alpha_1}x_2^{\alpha_2}\cdots z_k^{\gamma_k}\cdots x_n^{\alpha_n}$$

$$x_k^{\alpha_k} = z_k^{\gamma_k}$$

where f^{conv} is a convex function, $x_k^{\alpha_k} - z_k^{\gamma_k} = 0$ is nonconvex. Because $x_k = z_k^{\frac{\gamma_k}{\alpha_k}}(-1 < \frac{\gamma_k}{\alpha_k} \le 0$ or $1 \le \frac{\gamma_k}{\alpha_k})$ and $0 < l_k \le x_k \le u_k$, we have

$$l_k \le z_k^{\frac{\gamma_k}{\alpha_k}} \le u_k, \frac{1}{u_k} \le x_k^{-1} \le \frac{1}{l_k} \tag{18}$$

If $x_k^{-1} \ge \frac{1}{u_k}, z_k^{\frac{\gamma_k}{\alpha_k}} \ge l_k$, then

$$(z_k^{\frac{\gamma_k}{\alpha_k}} - l_k)x_k^{-1} \ge \frac{(z_k^{\frac{\gamma_k}{\alpha_k}} - l_k)}{u_k} \tag{19}$$

i.e.

$$1 = x_k^{-1}z_k^{\frac{\gamma_k}{\alpha_k}} \ge \frac{1}{u_k}z_k^{\frac{\gamma_k}{\alpha_k}} + l_kx_k^{-1} - \frac{l_k}{u_k} \tag{20}$$

Similarly, if $x_k^{-1} \le \frac{1}{l_k}, z_k^{\frac{\gamma_k}{\alpha_k}} - u_k \le 0$, the following inequality holds

$$1 - u_kx_k^{-1} \ge \frac{1}{l_k}z_k^{\frac{\gamma_k}{\alpha_k}} - \frac{u_k}{l_k} \tag{21}$$

Multiplying both sides by $\frac{l_k}{u_k}$, we obtain the same inequality as 20. The equality can be relaxed as an inequality 20, defined as the lower bound of $x_k^{-1}z_k^{\frac{\gamma_k}{\alpha_k}}$. And inequality 20 is a convex constraint. When $0 < \frac{\gamma_k}{\alpha_k} < 1$ or $\frac{\gamma_k}{\alpha_k} \le -1$, in the same way we can obtain the convex function as follows:

$$1 \ge l_kz_k^{-\frac{\gamma_k}{\alpha_k}} + \frac{1}{u_k}x_k - \frac{l_k}{u_k} \tag{22}$$

Strategy1 : For all $x \in [l, u]$, let

$$Ieq_1 = \{k | \alpha_k(n - m - 1) \leq \sum_{i=m+1}^{n-1} \alpha_i, k = m+1, \cdots, n-1\}$$

$$Ieq_2 = \{k | \alpha_k(n - m - 1) > \sum_{i=m+1}^{n-1} \alpha_i, k = m+1, \cdots, n-1\}$$

$|Ieq_i|$ be the number of elements in set $Ieq_i, i = 1, 2$, if $\hat{c} > 0$, then a term $f^{\text{nonc}}(x) = \hat{c}x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ where $\alpha_1 \leq \cdots \leq \alpha_m < 0 < \alpha_{m+1} \leq \cdots \leq \alpha_n, m < n$ and $1 - \sum_{i=1}^{m} \alpha_i < \alpha_n$ can be convexified as follows:

$$\begin{cases} f^{\text{conv}}(x, z) = \hat{c} \prod_{i=1}^{m} x_i^{\alpha_i} \prod_{k=m+1}^{n-1} z_k^{\gamma_k} x_n^{\alpha_n} \\[2mm] \frac{1}{u_k} z_k^{\frac{\gamma_k}{\alpha_k}} + l_k x_k^{-1} - \frac{l_k}{u_k} \leq 1, if \ -1 < \frac{\gamma_k}{\alpha_k} \leq 0, k = m+1, \cdots, n-1 \\[2mm] l_k z_k^{-\frac{\gamma_k}{\alpha_k}} + \frac{1}{u_k} x_k - \frac{l_k}{u_k} \leq 1, if \ \frac{\gamma_k}{\alpha_k} \leq -1, k = m+1, \cdots, n-1 \end{cases}$$

where $\gamma_k = \begin{cases} (1 - \sum_{i=1}^{m} \alpha_i - \alpha_n)\frac{1}{n-m-1}, & if \ |Ieq_1| \geq |Ieq_2| \\[3mm] (1 - \sum_{i=1}^{m} \alpha_i - \alpha_n)\frac{\alpha_k}{\sum_{i=m+1}^{n-1} \alpha_i}, & otherwise \end{cases}$, $k = m+1, \cdots, n-1$.

Strategy2 : For all $x \in [l, u]$, if $\hat{c} > 0$, then a term $f^{\text{nonc}}(x) = \hat{c}x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ where $\alpha_1 \leq \cdots \leq \alpha_m < 0 < \alpha_{m+1} \leq \cdots \leq \alpha_n$ and $1 - \sum_{i=1}^{m} \alpha_i \geq \alpha_n$ can be convexified as follows:

$$\begin{cases} f^{\text{conv}}(x, z) = \hat{c} \prod_{i=1}^{m} x_i^{\alpha_i} \prod_{k=m+1}^{n-1} z_k^{\gamma_k} z_n^{\gamma_n} \\[2mm] l_k z_k^{\frac{1}{\beta}} + \frac{1}{u_k} x_k - \frac{l_k}{u_k} \leq 1, k = m+1, m+2, \cdots, n-1, (-\frac{1}{\beta} \leq -1) \\[2mm] \frac{1}{u_n} z_n^{\frac{\gamma_n}{\alpha_n}} + l_n x_n^{-1} - \frac{l_n}{u_n} \leq 1, (\frac{\gamma_n}{\alpha_n} \geq 1) \end{cases}$$

where $\gamma_k = -\frac{\alpha_k}{\beta}, k = m+1, \cdots, n-1, \gamma_n = 1 - \sum_{i=1}^{m} \alpha_i - \sum_{k=m+1}^{n-1} \gamma_k, 0 < \beta \leq 1$.

Strategy3 : For all $x \in [l, u]$, let

$$J = \{j| \sum_{j=m+1}^{s} \alpha_j < 1, m+1 \le s \le n\}$$

$$Ieq_3 = \{k| |\alpha_k|(n-s+m) \le \sum_{i \in I \setminus J} |\alpha_i|, k = 1, \cdots, m, s+1, \cdots, n\}$$

$$Ieq_4 = \{k| |\alpha_k|(n-s+m) > \sum_{i \in I \setminus J} \alpha_i, k = 1, \cdots, m, s+1, \cdots, n\}$$

and $|Ieq_i|$ be the number of elements in set $Ieq_i, i = 3, 4, I = \{1, 2, \cdots, n\}$ and if $\hat{c} < 0$, then a term $f^{\text{nonc}}(x) = \hat{c} x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$, where $\alpha_1 \le \cdots \le \alpha_m < 0 < \alpha_{m+1} \le \cdots \le \alpha_s \le \cdots \le \alpha_n, m < n$ and $\sum_{i=m+1}^{s} \alpha_i < 1$, can be convexified as follows:

$$\begin{cases} f^{\text{conv}}(x, z) = \hat{c} \prod_{i \in J} x_i^{\alpha_i} \prod_{k \in I \setminus J} z_k^{\gamma_k} \\ \frac{1}{u_k} z_k^{\frac{\gamma_k}{\alpha_k}} + l_k x_k^{-1} - \frac{l_k}{u_k} \le 1, if \ -1 < \frac{\gamma_k}{\alpha_k} \le 0, k = 1, \cdots, m, or \ \frac{\gamma_k}{\alpha_k} \ge 1, k = s+1, \cdots, n \\ l_k z_k^{-\frac{\gamma_k}{\alpha_k}} + \frac{1}{u_k} x_k - \frac{l_k}{u_k} \le 1, if \ \frac{\gamma_k}{\alpha_k} \le -1, k = 1, \cdots, m, or \ 0 < \frac{\gamma_k}{\alpha_k} < 1, k = s+1, \cdots, n \end{cases}$$

where $\gamma_k = \begin{cases} (1 - \sum_{i \in J} \alpha_i) \frac{1}{n-s+m}, \ if \ |Ieq_3| \ge |Ieq_4| \ and \ n \ne s \\ (1 - \sum_{i \in J} \alpha_i) \frac{|\alpha_k|}{\sum_{i \in I \setminus J} |\alpha_i|}, \ others \end{cases}$, $k = 1, \cdots, m, s+1, \cdots, n.$

Strategy4 : For all $x \in [l, u]$, let

$$Ieq_5 = \{k|n|\alpha_k| \le \sum_{i=1}^{n} |\alpha_i|, k = 1, \cdots, n\}$$

$$Ieq_6 = \{k|n|\alpha_k| > \sum_{i=1}^{n} |\alpha_i|, k = 1, \cdots, n\}$$

and $|Ieq_i|$ is the number of elements in set $Ieq_i, i = 5, 6$, if $\hat{c} < 0$, then a term $f^{\text{nonc}}(x) = \hat{c} x_1^{\alpha_1} x_2^{\alpha_2} \cdots x_n^{\alpha_n}$ where $\alpha_1 \le \cdots \le \alpha_m < 0 < \alpha_{m+1} \le \cdots \le \alpha_n$ and $\alpha_{m+1} \ge 1$ can be convexified as follows:

$$\begin{cases} f^{\text{conv}}(x, z) = \hat{c} \prod_{k=1}^{n} z_k^{\gamma_k} \\ \frac{1}{u_k} z_k^{\frac{\gamma_k}{\alpha_k}} + l_k x_k^{-1} - \frac{l_k}{u_k} \le 1, if \ -1 < \frac{\gamma_k}{\alpha_k} \le 0, k = 1, \cdots, m, or \ \frac{\gamma_k}{\alpha_k} \ge 1, k = s+1, \cdots, n \\ l_k z_k^{-\frac{\gamma_k}{\alpha_k}} + \frac{1}{u_k} x_k - \frac{l_k}{u_k} \le 1, if \ \frac{\gamma_k}{\alpha_k} \le -1, k = 1, \cdots, m, or \ 0 < \frac{\gamma_k}{\alpha_k} < 1, k = s+1, \cdots, n \end{cases}$$

where $\gamma_k = \begin{cases} \frac{1}{n}, & if \ |Ieq_5| \geq |Ieq_6| \\ \frac{|\alpha_k|}{\sum\limits_{i=1}^{n} |\alpha_i|}, & others \end{cases}$ $,k=1,\cdots,n.$

2.3 Second-Order Cone Programming Formulation. By introducing additional intermediate variables, the OOP can be transformed into a convex or approximate convex optimization problem. In general, convex optimization problem can be solved directly using convex optimization software packages. However, the solution process will be slow with the number of new variables. The rolling line is a continuous and high speed process. The time allowed to set optimal parameters is short for the small batches and varieties of products. A second order cone programming formulation has a simple structure and can be solved quickly. As an example, the simple constraint can be cast as a conic quadratic inequality using the following relation.

$$\hat{x}^2 \leq \hat{y} \Leftrightarrow \| (2\hat{x}, \hat{y}-1) \|_2 \leq \hat{y}+1 \tag{23}$$

Considering the power α as any rational number, we will give the second-order cone expression of complex convex function in OOP. Inspired by Alizadeh,[41] we get the following theorem.

Theorem 0.1 : (a) If function $g(x) = \hat{c} \prod\limits_{i=1}^{n} x_i^{\alpha_i}$ satisfies convex property a, where $\alpha = \frac{a}{b}, a, b \in \mathbb{Z}^+$, and there exists no positive integer d such that $a = da'$ and $b = db'$, then $g(x) \leq \bar{y}$, for $\bar{y} \geq 0$, can be transformed to a system of SOC inequalities as

$$\| (2w_j, u_j - v_j) \|_2 \leq u_j + v_j, j = 1, \cdots, J$$

(b) If function $g(x) = \hat{c} \prod\limits_{i=1}^{n} x_i^{\alpha_i}$ satisfies convex property b, where $\alpha_i = \frac{a_i}{b}, a_i \in \mathbb{Z}^-, b \in \mathbb{Z}^+, i = 1, 2, \cdots, n$, then $g(x) \leq \bar{y}$, for $\bar{y} \geq 0$, can be transformed to a system of SOC inequalities as

$$\| (2w_s, w_{2s+1} - w_{2s+2}) \|_2 \leq w_{2s+1} + w_{2s+2}, s \in \{J_1, J_2 \cdots, J_{k+1}\}$$

where $J_1 = 0, J_j = 2^{j-1} - 1, \cdots, 2^j - 2, j = 2, \cdots, k+1.$

(c) If function $g(x) = \hat{c} \prod\limits_{i=1}^{n} x^{\alpha_i}$ satisfies convex property c, where $\alpha_i = \frac{a_i}{b}, a_i \in \mathbb{Z}^-, i = 1, 2, \cdots, n-$

$1, a_n \in \mathbb{Z}^+, b \in \mathbb{Z}^+$, then $g(x) \leq \bar{y}$, for $\bar{y} \geq 0$, can be transformed to a group of SOC inequalities as b.

(d) If function $g(x) = \hat{c} \prod\limits_{i=1}^{n} x^{\alpha_i}$ satisfies convex property d, where $\alpha_i = \frac{a_i}{b}, a_i, b \in \mathbb{Z}^+$, $i = 1, 2, \cdots, n$, then $\bar{y} \leq -g(x)$, for $\bar{y} \geq 0$, can be transformed to a group of SOC inequalities as b.

Proof. (a) When $g(x) = \hat{c} \prod\limits_{i=1}^{n} x_i^{\alpha_i}$ satisfies convex property a, we have $g(x) = \hat{c}x^\alpha$ for $\hat{c} \geq 0$ and $\alpha \geq 1$. If $\alpha = 1$, then $g(x)$ is a linear function. This case directly adds to the linear constraints without any transform. Thus, we only discuss the case of $\alpha > 1$, i.e., $a > b \geq 1$.

Let $2^k < a \leq 2^{k+1}$ and $\sum\limits_{i=1}^{k+1} 2^{i-1}\eta_i = 2^{k+1} - a$, we have

$$
\begin{aligned}
\hat{c}x^\alpha \leq \bar{y} &\Leftrightarrow \hat{c}^b x^a \leq \bar{y}^b \\
&\Leftrightarrow \hat{c}^b x^{2^{k+1}} \leq \bar{y}^b x^{(2^{k+1}-a)} \\
&\Leftrightarrow \hat{c}^b x^{2^{k+1}} \leq \bar{y}^{2^k \beta_{k+1}} x^{2^k \eta_{k+1}} \bar{y}^{2^{k-1}\beta_k} x^{2^{k-1}\eta_k} \cdots \bar{y}^{2^1 \beta_2} x^{2^1 \eta_2} \bar{y}^{2^0 \beta_1} x^{2^0 \eta_1}
\end{aligned}
$$

(24)

where $\eta_i \in \{(0,1) | \sum\limits_{i=1}^{k+1} 2^{i-1}\eta_i = 2^{k+1} - a\}, \beta_i \in \{(0,1) | \sum\limits_{i=1}^{k+1} 2^{i-1}\beta_i = b\}$ and $\{k := k | 2^k < a \leq 2^{k+1}, k \in \mathbb{N}\}$.

Because

$$2^{k+1} \geq 2^{k+1} - \eta_{k+1}2^k - \eta_k 2^{k-1} - \cdots - \eta_1 2^0 > 2^k$$

we can get $\eta_{k+1} = 0$.

Introducing the new variables s, the above inequality 24 can be rewritten as

$$
\begin{cases}
s_{1,1}^2 \leq x^{\eta_1} \\
s_{1,2}^2 \leq \bar{y}^{\beta_1} \\
s_{2,1}^2 \leq s_{1,1}x^{\eta_2} \\
s_{2,2}^2 \leq s_{1,2}\bar{y}^{\beta_2} \\
\vdots \\
s_{k,1}^2 \leq s_{k-1,1}x^{\eta_k} \\
s_{k,2}^2 \leq s_{k-1,2}\bar{y}^{\beta_k} \\
\hat{c}^{\frac{1}{2^k}} x^2 \leq s_{k,1}s_{k,2}\bar{y}^{\beta_{k+1}}
\end{cases}
\Leftrightarrow
\begin{cases}
s_{j,1}^2 \leq s_{j-1,1}x^{\eta_j}, j = 1, \cdots, k \\
s_{j,2}^2 \leq s_{j-1,2}\bar{y}^{\beta_j}, j = 1, \cdots, k \\
s_{k+1}^2 \leq s_{k,1}s_{k,2}\bar{y}^{\beta_{k+1}}
\end{cases}
$$

(25)

where $s_{0,1} = s_{0,2} = 1, s_{k+1} = \hat{c}^{\frac{b}{2^{k+1}}} x$. Thus, the inequalities 25 except the last one can be transformed to SOCs like 24.

The following cancellation rules can remove the corresponding inequalities to void redundant ones when η_j or β_j is equal to 0.

(1)If the right-hand sides of the $\bar{j}$th pair inequalities are both univariates in 25, we can combine them to obtain an inequality of $s_{\bar{j},1}^2$ and cancel the inequality of $s_{\bar{j},2}$ such that $s_{\bar{j},2} = 1$, in which $s_{\bar{j},1}^2$ is less than or equal to multiplication of the univariates.

(2)If $s_{j,\hat{j}}^2 \leq 1, \hat{j} = 1, 2$, then let $s_{j,\hat{j}} = 1$.

According to the cancellation rules, there must be $s_{k,1} = 1$ or $s_{k,2} = 1$. Changing the notation, the above inequalities can be cast as the following system of SOCs:

$$\| (2w_j, u_j - v_j) \|_2 \leq u_j + v_j, j = 1, \cdots, J$$

(b)Because $g(x)$ satisfies convex property b, we have $a_i \leq 0, i = 1, \cdots, n$ and

$$\hat{c} \prod_{i=1}^{n} x_i^{\alpha_i} \leq \bar{y} \Leftrightarrow \hat{c}^b \leq \bar{y}^b \prod_{i=1}^{n} x_i^{|a_i|}$$

$$\Leftrightarrow (\hat{c}^{\frac{b}{2^{k+1}}})^{2^{k+1}} \leq \bar{y}^b \prod_{i=1}^{n} x_i^{|a_i|} 1^{2^{k+1}-b-\sum_{i=1}^{n}|a_i|}$$

$$\Leftrightarrow (\hat{c}^{\frac{b}{2^{k+1}}})^{2^{k+1}} \leq \bar{y}^{\sum_{j=1}^{k+1} \beta_j 2^{j-1}} \prod_{i=1}^{n} x_i^{\sum_{j=1}^{k+1} \eta_j^{(i)} 2^{j-1}} 1^{\sum_{j=1}^{k+1} \bar{\eta}_j 2^{j-1}}$$

where $\bar{\eta}_j \in \{(0,1) | \sum_{j=1}^{k+1} \bar{\eta}_j 2^{j-1} = 2^{k+1} - b - \sum_{i=1}^{n} |a_i|\}, \eta_j^{(i)} \in \{(0,1) | \sum_{j=1}^{k+1} \eta_j^{(i)} 2^{j-1} = |a_i|\}, \beta_j \in \{(0,1) | \sum_{j=1}^{k+1} \beta_j 2^{j-1} = b\}$ and $k := \{k | 2^k < b + \sum_{i=1}^{n} |a_i| \leq 2^{k+1}, k \in \mathbb{N}\}$. The above inequality can be expressed as $k+1$ levels of conic quadratic inequalities, which look like

$$w_s^2 \leq w_{2s+1} w_{2s+2}, s = J_1, \cdots, J_k, J_{k+1} \tag{26}$$

where $J_1 = 0, J_{k+1} = 2^k - 1, \cdots, 2^{k+1} - 2$. Adding new variables depend on the values of $\beta_j, \eta_j^{(i)}, \bar{\eta}_j$ in every level.

The formulation for the first level is

$$(\hat{c}^{\frac{2b}{2^{k+1}}}) = w_0^2 \leq w_1 w_2$$

where the existence of w_1 or w_2 is related to $\beta_{k+1}, \eta_{k+1}^{(i)}$. Interestingly, at most one w is replaced by x_i or y in the first level. If $\beta_{k+1} = 1$, then $b \geq 2^k$. We could get $\sum_{i=1}^{n} |a_i| < 2^k$ on account of $2^k < b + \sum_{i=1}^{n} |a_i| \leq 2^{k+1}$ and $n > 1$. Thus, $|a_i| \neq 2^k$ for all i, i.e., $\eta_{k+1}^{(i)} = 0$ holds. Conversely, because of $2^k < b + \sum_{i=1}^{n} |a_i| \leq 2^{k+1}$ and $b \neq 0$, suppose $\exists |a_l| = 2^k$, there must be $a_i \neq 2^k (i = 1, \cdots, l-1, l+1, \cdots, n)$ and $b < 2^k$, which means $\eta_{k+1}^{l} = 1, \eta_{k+1}^{(i)} = 0, i = 1, \cdots, l-1, l+1, \cdots, n$ and $\beta_{k+1} = 0$. Once there exists no new variable $w_s, s = 1, 2$, the branch is cut off.

The second level is the parent of the first level, for which the resemblant form can be written as

$$w_1^2 \leq w_3 w_4$$
$$w_2^2 \leq w_5 w_6$$

where the existence of $w_s, s = 3, 4, 5, 6$ is related to $\beta_k, \eta_k^{(i)}, \bar{\eta}_k$. If $\beta_k = 1$ or $\eta_k^{(l)} = 1$, then the new variable w_s is equal to y or x_l and there exists no parent branching from the corresponding variable.

Until up to the $k+1$ level, we have

$$w_{2^k-1}^2 \leq w_{2^{k+1}-1} w_{2^{k+1}}$$
$$\vdots$$
$$w_{2^{k+1}-2}^2 \leq w_{2^{k+2}-3} w_{2^{k+2}-2}$$

where the value of $w_s, s = 2^{k+1} - 1, \cdots, 2^{k+2} - 2$ is replaced by $x_1^{\eta_1^1}, x_2^{\eta_1^2}, \cdots, x_n^{\eta_1^n}, \bar{y}^{\beta_1}, 1$, respectively.

Changing the notation for the inequalities 26, we get the result b.

(c) We have $\alpha_n \geq 1, \alpha_i \leq 0, i = 1, 2, \cdots, n-1, 1 \leq \sum_{i=1}^{n} \alpha_i$ because function $g(x) = \hat{c} \prod_{i=1}^{n} x^{\alpha_i}$ satisfies convex property c. If $\alpha_n = 1, \alpha_i = 0, i = 1, 2, \cdots, n-1$, then the function $g(x) = x$ is the linear

one. Others, the inequality $\hat{c} \prod_{i=1}^{n} x^{\alpha_i} \leq \bar{y}$ can be written as follows:

$$\hat{c} \prod_{i=1}^{n} x_i^{\alpha_i} \leq \bar{y} \Leftrightarrow \hat{c}^b x_n^{a_n} \leq \bar{y}^b \prod_{i=1}^{n-1} x_i^{|a_i|}$$

$$\Leftrightarrow \hat{c}^b x_n^{2^{k+1}} \leq \bar{y}^b \prod_{i=1}^{n-1} x_i^{|a_i|} x_n^{2^{k+1}-a_n}$$

where $k := \{k | 2^k < a_n \leq 2^{k+1}, k \in \mathbb{N}\}$. The inequality can be written as a system of SOCs by using b, where $w_0 = \hat{c}^{\frac{b}{2^{k+1}}} x_n$.

(d) We have $\hat{c} < 0$, $\sum_{i=1}^{n} \alpha_i \leq 1$ and $\alpha_i \geq 0$ on account of that $g(x)$ satisfies convex property d. Similarly, we have

$$\bar{y} \leq \prod_{i=1}^{n} (-\hat{c}) x_i^{\alpha_i} \quad \Leftrightarrow \bar{y}^b \leq \prod_{i=1}^{n} (-\hat{c})^b x_i^{a_i}$$

$$\Leftrightarrow \frac{1}{(-\hat{c})^b} \bar{y}^{2^{k+1}} \leq \prod_{i=1}^{n} x_i^{a_i} \bar{y}^{2^{k+1}-b}$$

where $k := \{k | 2^k < b \leq 2^{k+1}, k \in \mathbb{N}\}$. We only need define $\omega_0 = \left(\frac{1}{-\hat{c}}\right)^{\frac{b}{2^{k+1}}} \bar{y}$. It is clear that d is the same as result b.

This completes our proof.

Here we give two examples to explain Theorem 0.1 in detail. For condition a of Theorem 0.1, we have

$$x^{\frac{5}{4}} \leq \bar{y} \Leftrightarrow x^8 \leq \bar{y}^4 x^3$$

$$\Leftrightarrow \begin{cases} s_{1,1}^2 \leq x^{\eta_1} = x \\ s_{1,2}^2 \leq \bar{y}^{\beta_1} = 1 \\ s_{2,1}^2 \leq s_{1,1} x^{\eta_2} = s_{1,1} x \\ s_{2,2}^2 \leq s_{1,2} \bar{y}^{\beta_2} = s_{1,2} \\ x^2 = s_3^2 \leq s_{2,1} s_{2,2} \bar{y}^{\beta_3} = s_{2,1} s_{2,2} \bar{y} \end{cases} \tag{27}$$

Using the above rules, we get the following simplified equations.

$$x^8 \le \bar{y}^4 x^3 \Leftrightarrow \begin{cases} s_{1,1}^2 \le x \\ s_{2,1}^2 \le s_{1,1}x \\ x^2 = s_3^2 \le s_{2,1}\bar{y} \end{cases} \tag{28}$$

Figure 3 is used to show condition b in Theorem 0.1. We give an example $g^+(x) = x_1^{-\frac{1}{4}} x_2^{-\frac{1}{4}} x_3^{-\frac{1}{4}} x_4^{-\frac{1}{4}} \le$ y in the left part of Figure 3. The function can be written as $1^{2^{2+1}} \le y^4 x_1 x_2 x_3 x_4$. Thus, $\beta_{k+1} = \beta_3 = 1, \beta_j = 0(j = 1, 2)$ and $\eta_1^i = 1, \eta_j^i = 0(i = 1, \cdots, 4, j = 2, 3)$. In this case, w_2 does not have parents in the first level and $w_2 = y$. The function $g^+(x) = x_1^{-1} x_2^{-1} x_3^{-1} x_4^{-1} x_5^{-1} x_6^{-1} x_7^{-1} \le \bar{y}$ can be written as $1^{2^{2+1}} \le \bar{y} x_1 x_2 x_3 x_4 x_5 x_6 x_7$ and there will be three levels, where $\beta_1 = 1, \beta_j = 0(j = 2, 3)$ and $\eta_1^i = 1, \eta_j^i = 0(i = 1, \cdots, n = 7, j = 2, 3)$. Since $\beta_3 = \eta_3 = 0$, the first level is added to two new variables w_1 and w_2. Similarly, the next levels have some additional variables until up to the last one. It is shown in the right part of Figure 3.
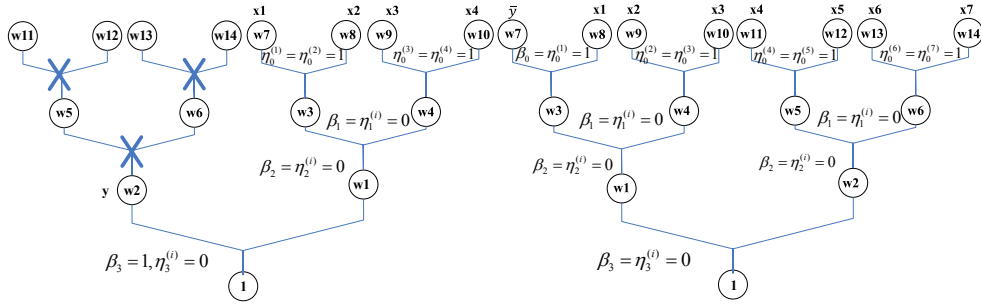

Figure 3: Examples for Theorem 0.1.

On the basis of the above method, the convex formulation of the problem P can be transformed

into the second order cone optimization problem with both inequality and box constraints.

$$
(\text{P1}): \begin{cases}
\min & c^T x \\
\text{s.t.} & A_E x + s = b_E \\
& A_I x \leq b_I \\
& s \in \mathscr{C} \\
& x \in [l, u]
\end{cases}
$$

where $c \in \mathbb{R}^n, b_E \in \mathbb{R}^m, b_I \in \mathbb{R}^{m_I}$ are given vectors, $A_E \in \mathbb{R}^{m \times n}, A_I \in \mathbb{R}^{m_I \times n}$ are given matrices, the second-order cone is $\mathscr{C} = \mathscr{C}_1^{p+1} \times \cdots \times \mathscr{C}_k^{p+1}, \mathscr{C}_i^{p+1} = \{(s_{i0}, s_{i1}) | \|s_{i1}\|_2 \leq s_{i0}, s_{i1} \in \mathbb{R}^p, s_{i0} \in \mathbb{R}\}$.

3. ALGORITHM

Under the above convexification strategies, we propose a global algorithm to avoid using integer variables and improve the accuracy of approximate convex optimization problem, which is called the sequential approximation method based on convex modeling. In the base layer, the approximate model is solved by ADMM. In the main loop, we use the monotonicity of the function to get smaller internal, and then input the reduced interval into the approximate model. The more accurate model is solved as a new iteration. On the basis of the solution of the approximate model, the objective function of the problem OOP is updated until approximate deviation converges within a given error.

According to monotonicity of the function, we give the following rule for deleting the interval such that the ineffective space is cut in each iteration.

If $\max\limits_{m=1,\cdots,M} \{G_m^+(l) - G_m^-(u)\} \leq 0$, then

$$
red_{s+1} = [l_{s+1}, u_{s+1}]
$$

satisfying

$$l_{s+1} = u_s - \sum_{i=1}^{n} \min_{m=1,\cdots,M} \{\varpi_m^i\}(u_s^i - l_s^i)e^i$$

$$u_{s+1} = l_{s+1} + \sum_{i=1}^{n} \min_{m=1,\cdots,M} \{\xi_m^i\}(u_s^i - l_{s+1}^i)e^i$$

where

$$\varpi_m^i = \begin{cases} \bar{\varpi}_m^i, \ if \ G_m^+(l_s) - G_m^-(u_s - \bar{\varpi}_m^i(u_s^i - l_s^i)e^i) = 0, \bar{\varpi}_m^i \in (0,1) \\ 1, \ otherwise \end{cases}$$

$$\xi_m^i = \begin{cases} \bar{\xi}_m^i, \ if \ G_m^+(l_{s+1} + \bar{\xi}_m^i(u_s^i - l_{s+1}^i)e^i) - G_m^-(u_s) = 0, \bar{\xi}_m^i \in (0,1) \\ 1, \ otherwise \end{cases}$$

After having sketched the cut strategy, we can proceed to outline inner-layer ADMM algorithm details, which is inspired by SPADMM.[44] SPADMM is an advanced ADMM algorithm for solving cone optimization problems with three blocks. The algorithm is famous for rapid convergence and solving the large-scale cone optimization problem. Because the slack variables and cone variables are uncoupled in our problem, we can put them together as a new vector. When the problem can be divided into two blocks, the algorithm has the same structure as with the classical ADMM, in which the semiproximal term is equal to 0. Different from solving semidefinite programming problem, we use ADMM to solve problem P1, which is a second-order cone optimization problem. By introducing a non-negative slack variable $\bar{z} \in \mathbb{R}^{m_I+2n}$ and the indicator function δ_U, P1 can be converted to the following standard formulation.

$$(\text{P2}) : \begin{cases} \min \quad c^T x + \delta_{w \in \mathcal{Q}}(w) \\ \text{s.t.} \quad Ax + w = b \end{cases}$$

where $A = (A_s^T, A_{\bar{z}}^T)^T, A_s = A_E, A_{\bar{z}} = (A_I^T, I_{n \times n}, -I_{n \times n})^T$ are coefficient matrices; $b = (b_s^T, b_{\bar{z}}^T)^T, b_{\bar{y}} = b_E, b_{\bar{z}} = (b_I^T, u^T, -l^T)^T$ are coefficient vectors; I is the identity matrix; $w = (s^T, \bar{z}^T)^T$ are the vari-

ables; $\mathscr{Q} = \{(s,\bar{z}) | s \in \mathscr{C}, \bar{z} \geq 0\}$ and the indicator function is defined by

$$
\delta_U(\mu) := \begin{cases} 0, & \text{if } \mu \in U \\ +\infty, & \text{if } \mu \notin U \end{cases}
$$

For a given $\sigma > 0$, the augmented Lagrangian function for P2 is given by

$$
L_\sigma(x, s, \bar{z}; \lambda) = c^T x + \delta_{w \in \mathscr{Q}}(w) + \lambda^T (Ax + w - b) + \frac{\sigma}{2} \|Ax + w - b\|_2^2 \tag{29}
$$

where $\lambda \in R^{m_E + m_I + 2n}$ is lagrangian multiplier.

Algorithm 1: Solve the reformulated SOCP problem P2.

Initialization: $\rho = 1.618, \sigma > 0, s^0 \in \mathscr{C}, \bar{z}^0 \geq 0, \lambda^0 \geq 0, x^0$;

Step 1: Compute $x^{k+1} = (A^T A)^{-1} [A^T (b - w^k - \frac{1}{\sigma} \lambda^k) - \frac{1}{\sigma} c]$.

Step 2: Compute $w^{k+1} = \begin{cases} s^{k+1} := \arg\min L_\sigma(x^{k+1}, s; \lambda_s^k). \\ \bar{z}^{k+1} := \arg\min L_\sigma(x^{k+1}, \bar{z}; \lambda_{\bar{z}}^k). \end{cases}$

Step 3: Compute $\lambda^{k+1} := \lambda^k + \rho\sigma(Ax^{k+1} + w^{k+1} - b)$.

It is important to be clear about how to slove s-minimization problem and $\bar{z}$-minimization problem in each iteration. We next give the analytical expression for every subproblem.

The $\bar{z}$-minimization problem is equivalent to

$$
\begin{aligned}
\min \quad & \lambda_{\bar{z}}^T \bar{z} + \frac{\sigma}{2} \|A_{\bar{z}} x + \bar{z} - b_{\bar{z}}\|_2^2 \\
\text{s.t.} \quad & \bar{z} \geq 0
\end{aligned}
$$

The Lagrangian function for this subproblem can be written as

$$
L(\bar{z}, \nu) = \lambda_{\bar{z}}^T \bar{z} + \frac{\sigma}{2} \|A_{\bar{z}} x + \bar{z} - b_{\bar{z}}\|_2^2 - \nu^T \bar{z}
$$

For which the KKT conditions are

$$
\begin{cases}
\sigma(A_{\bar{z}}x + \bar{z} - b_{\bar{z}} + \frac{\lambda_{\bar{z}}}{\sigma}) - v = 0 \\
\bar{z} \geq 0 \\
v \geq 0 \\
v^T \bar{z} = 0
\end{cases}
$$

If $\bar{z}^* = 0$, then $v^* \geq 0$. We have

$$
v^* = \sigma(A_{\bar{z}}x - b_{\bar{z}} + \frac{\lambda_{\bar{z}}}{\sigma}) \geq 0
$$

If $\bar{z}^* > 0$, then $v^* = 0$. We have

$$
\bar{z}^* = b_{\bar{z}} - (A_{\bar{z}}x + \frac{\lambda_{\bar{z}}}{\sigma}) > 0
$$

Thus, the minimizer of $\bar{z}$ is obtained by the following formulation

$$
\bar{z}^{k+1} = \max\{0, b_{\bar{z}} - (A_{\bar{z}}x^{k+1} + \frac{\lambda_{\bar{z}}^k}{\sigma})\} \tag{30}
$$

By scaling the dual variable λ_s and normalizing the coefficient of variable s, the s-minimization problem can be reformulated as

$$
s^{k+1} = \arg\min\{\delta_{s \in \mathscr{C}}(s) + \lambda_s^T(A_sx + s - b_s) + \frac{\sigma}{2}\|A_sx + s - b_s\|_2^2\}
$$

$$
= \arg\min
\begin{cases}
\frac{\sigma}{2}\|s - (b_s - A_sx - \frac{\lambda_s}{\sigma})\|_2^2 \\
\text{s.t.} s \in \mathscr{C}
\end{cases}
$$

$$
= \prod_{\mathscr{C}}(b_s - A_sx^{k+1} - \frac{\lambda_s^k}{\sigma})
$$

where $\prod_{\mathscr{C}}$ denotes projection onto $\mathscr{C}$. To simplify notation, we define the vector $b_s - A_sx^{k+1} - \frac{\lambda_s^k}{\sigma} =$

g. The projection onto the second-order cone has the following decomposition

$$\prod_{\mathscr{C}} g = \kappa_1 C_1 + \kappa_2 C_2$$

where for $i = 1, 2$,

$$\kappa_i = \max(0, g_n + (-1)^i \parallel [g_1, \cdots, g_{n-1}] \parallel_2)$$

and

$$C_i = \begin{cases} \dfrac{1}{2} \begin{pmatrix} \dfrac{(-1)^i g_1}{\parallel[g_1,\cdots,g_{n-1}]\parallel_2} \\ \vdots \\ \dfrac{(-1)^i g_{n-1}}{\parallel[g_1,\cdots,g_{n-1}]\parallel_2} \\ 1 \end{pmatrix} & \text{if} \quad \parallel [g_1,\cdots,g_{n-1}] \parallel_2 \neq 0 \\[2em] \dfrac{1}{2} \begin{pmatrix} (-1)^i \omega_1 \\ \vdots \\ (-1)^i \omega_{n-1} \\ 1 \end{pmatrix} & \text{if} \quad \parallel [g_1,\cdots,g_{n-1}] \parallel_2 = 0 \end{cases}$$

where $\omega \in \mathbb{R}^{n-1}$ is any vector satisfying $\parallel \omega \parallel_2 = 1$.

Thereby, a sequential approximation method is generated as follows.

Algorithm 2: sequential approximation method based on second order cone optimization

   if (P) *is convex* **then**

       go to Algorithm 1, stop;

   else

       set $\varepsilon, red_0 = [l, u]$;

   end

   while $|G_0(h^{k+1}, y^{k+1}) - L(G_0)| < \varepsilon$, **do**

       1.Go to Algorithm 1 to solve the SOCP problem of convexification $(P2)$. Denote the

       resulting objective value as $L(G_0)$ and the solution as (h^{k+1}, y^{k+1}).

       2. computer red_k and set $k = k + 1$.

   end

Theorem 0.2 : The sequential approximate method is a global optimization method for solving the OOP, if the optimal solution to P2 is a feasible solution to OOP.

Proof. According to Proposition 1, we have

$$\sum_{i=1}^{\mathcal{N}} N_i(h^*) = G_0(h^*, y^*).$$

Thus, we first divide $G_0(h, y)$ into the convex part $G_{conv}(h, y)$ and nonconvex part $G_{nonc}(h, y)$, i.e.,

$$G_0(h, y) = G_{conv}(h, y) + G_{nonc}(h, y).$$

Using the above convexification strategies, we can get the convex approximate formulation of $G_0(h, y)$ as follows.

$$G_0^{conv}(h, y, z) = G_{conv}(h, y) + G_{conv}(h, y, z),$$

where z is the convexification variable.

Next, let (x^*, w^*) be the optimal solution and $L(G_0)$ be objective function of P2. Through a series of transformations, the following equation holds.

$$L(G_0)(x^*, w^*) = G_0^{conv}(\hat{h}^*, \hat{y}^*, \hat{z}^*).$$

Because (h^*, y^*) is the global solution to P, it follows as

$$G_0(h^*, y^*) \leq G_0(\hat{h}^*, \hat{y}^*), \tag{31}$$

if (x^*, w^*) is a feasible solution to P.

From the inequality 31, it is obvious that the upper bound of the global solution to P can be obtained by calculating the optimal objective value of P at the optimal solution of P2.

From the property of the underestimation, we have

$$
\begin{aligned}
G_0^{\text{conv}}(h^*, y^*, z^*) &= G_{\text{conv}}(h^*, y^*) + G_{\text{conv}}(h^*, y^*, z^*) \\
&\leq G_{\text{conv}}(h^*, y^*) + G_{\text{nonc}}(h^*, y^*) \\
&= G_0(h^*, y^*)
\end{aligned}
$$

Besides, (x^*, w^*) is an optimal solution of P2. Then,

$$
L(G_0)(x^*, w^*) = G_0^{\text{conv}}(\hat{h}^*, \hat{y}^*, \hat{z}^*) \leq G_0^{\text{conv}}(h^*, y^*, z^*).
$$

So, the lower bound of the global solution to P is the optimal objective value to $(P2)$.

In summary, if (x^*, w^*) is a feasible solution to P2, then the global solution to OOP can be found by

$$
L(G_0)(x^*, w^*) \leq \sum_{i=1}^{\mathcal{N}} N_i(h^*) \leq G_0(\hat{h}^*, \hat{y}^*).
$$

And the proof is completed.

We give a simple example to show the approximation effect in Figure 4, where $L(G(x))$ is the linear lower bound of $G(x)$. The approximate solution can be as close as possible to the global optimal solution.

4. Numerical Experiments

To evaluate the performance of the proposed method, we have conducted numerical experiments on a set of ten problem instances. Each instance is a practical problem of rolling a different slab. The problem data were collected from a hot-rolling plant in China and used in ref.[7] The experiments were run on a PC with a 3.40 GHz Intel(R) Core(TM) i7–6700 CPU and 16GB RAM.

We apply our proposed sequential approximation method based on second-order cone optimization (GM) to solve the test instances and compare the results with those generated by the empirical method (EM) used in practice.
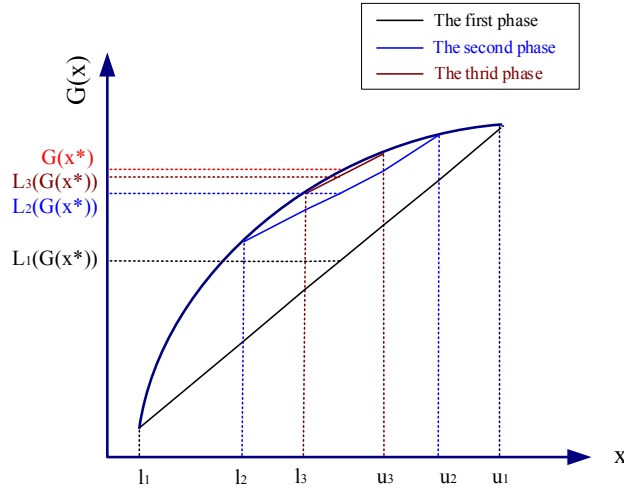
Figure 4: Sequential approximation of a concave function.

The computational results are presented in Table 1. In the table, column Si, i=1–7, shows the exit thickness of the ith stand, time is computation time, energy is the objective function value, and R is the percentage energy savings by the global method as compared to the other method. From the results shown in Table 1, it can be clearly seen that our global method can reduce the energy consumption. The average energy savings as compared with empirical method are 1.15% per ton. The parameters are quickly optimized, compared with the empirical method.

The force and shape quality measures are obtained by using the thickness of every stand in Table 1. And these values are given in the Table 2, where *RRR* is the relative reduction ratio, *RCD* is the relative crown deviation, and o is the change interval of relative crown in the final stand. From the view at microcosmic level, the smaller the grain size is, the better the shape quality will be. In order to prevent excessive growth of the grain, the relative reduction ratio of the final stand should be in [10%,15%]. Table 2 shows that the GM result is within the required range as with EM.

Relative crown is an important index for evaluating intermediate shape and side shape in strip production. According to the Shohet model, the reasonable setting for the relative crown of the last stand should be kept in the range shown in the last column o of Table 2. Our setting also conforms

Table 1: Comparison Results I

no.	method	S1 (mm)	S2 (mm)	S3 (mm)	S4 (mm)	S5 (mm)	S6 (mm)	S7 (mm)	time (s)	energy (KW)	R (%)
1	GM	27.53	19.32	13.37	10.04	7.94	6.43	5.70	58.09	31541	
	EM	25.51	18.55	12.68	9.55	7.84	6.52	5.70	>200	31587	0.15
2	GM	24.74	16.43	10.87	7.98	5.94	4.53	3.92	76.01	28093	
	EM	19.98	13.60	8.91	6.63	5.41	4.49	3.92	>200	28431	1.19
3	GM	22.21	13.85	8.60	5.96	4.42	3.35	2.90	66.38	24366	
	EM	18.35	11.91	7.52	5.34	4.22	3.39	2.90	>200	24816	1.81
4	GM	24.72	16.07	10.44	7.53	5.61	4.27	3.70	73.35	24959	
	EM	19.61	13.32	8.80	6.44	5.20	4.27	3.70	>200	25535	2.26
5	GM	25.40	16.51	10.59	7.45	5.50	4.16	3.60	66.37	25895	
	EM	19.82	13.33	8.72	6.34	5.10	4.17	3.60	>200	26379	1.83
6	GM	24.21	16.12	10.60	7.65	5.74	4.41	3.83	67.32	26520	
	EM	19.52	13.38	8.93	6.58	5.34	4.41	3.83	>200	27061	2.00
7	GM	25.90	18.03	12.38	9.23	7.23	5.80	5.10	66.68	29036	
	EM	20.02	14.56	10.23	7.96	6.70	5.72	5.10	>200	29121	0.29
8	GM	24.50	15.94	10.23	7.19	5.30	4.00	3.46	56.79	19674	
	EM	19.09	12.83	8.56	6.21	4.90	4.00	3.46	>200	19775	0.51
9	GM	23.89	15.47	9.87	6.91	5.09	4.05	3.50	75.62	19797	
	EM	18.85	12.75	8.55	6.23	4.94	4.04	3.50	>200	19910	0.57
10	GM	23.05	14.91	9.50	6.64	4.87	3.88	3.35	61.81	27313	
	EM	18.43	12.4	8.48	6.15	4.79	3.87	3.35	>200	27555	0.88

Table 2: Compared Results II

| no. | method | force (MN) | | | | | | | final stand | | |
		S1	S2	S3	S4	S5	S6	S7	RRR(%)	RCD(10^{-3})	o(10^{-3})
1	GM	16.38	19.65	21.67	16.29	13.65	12.84	7.39	11.30	0.71	
	EM	20.03	18.11	22.44	16.30	11.54	11.19	8.33	12.58	0.02	[-1.68,3.36]
2	GM	16.34	19.50	21.34	15.83	15.42	15.27	8.29	13.38	1.35	
	EM	24.00	19.16	22.57	15.71	11.02	10.53	7.81	12.69	0.90	[-0.84,1.68]
3	GM	19.55	19.80	20.34	18.80	13.37	11.29	7.56	13.43	0.44	
	EM	24.95	19.05	20.38	18.09	10.74	9.07	8.06	14.45	0.11	[-0.38,0.76]
4	GM	16.12	18.14	18.88	16.25	13.58	12.68	6.87	13.37	1.19	
	EM	24.29	17.21	18.91	16.10	10.14	9.23	6.85	13.45	0.80	[-0.68,1.36]
5	GM	16.27	18.48	19.15	18.23	14.18	13.21	7.47	13.47	1.11	
	EM	24.92	17.97	19.19	17.25	10.51	9.58	7.60	13.67	0.59	[-0.57,1.2]
6	GM	17.00	18.93	19.28	17.87	13.73	13.52	7.31	13.22	1.24	
	EM	25.18	18.41	19.35	17.15	10.37	9.92	7.26	13.15	0.89	[-0.72,1.40]
7	GM	14.82	17.72	18.40	17.71	13.87	12.76	8.10	12.06	0.50	
	EM	23.32	16.45	18.05	15.75	10.00	9.15	7.17	10.84	0.53	[-0.94,1.9]
8	GM	12.02	14.00	14.74	13.55	11.16	10.09	5.88	13.51	1.17	
	EM	18.50	13.67	14.19	12.84	8.95	8.05	5.87	13.50	0.80	[-0.59,1.20]
9	GM	12.57	14.85	15.72	12.72	11.96	9.53	5.33	13.58	0.76	
	EM	19.05	14.13	14.76	11.71	9.24	8.40	5.26	13.36	0.28	[-0.62,1.20]
10	GM	18.33	21.04	22.15	20.11	15.51	12.31	7.57	13.44	0.78	
	EM	27.36	20.19	19.81	18.55	12.76	11.48	7.48	13.44	0.50	[-0.51,1.02]
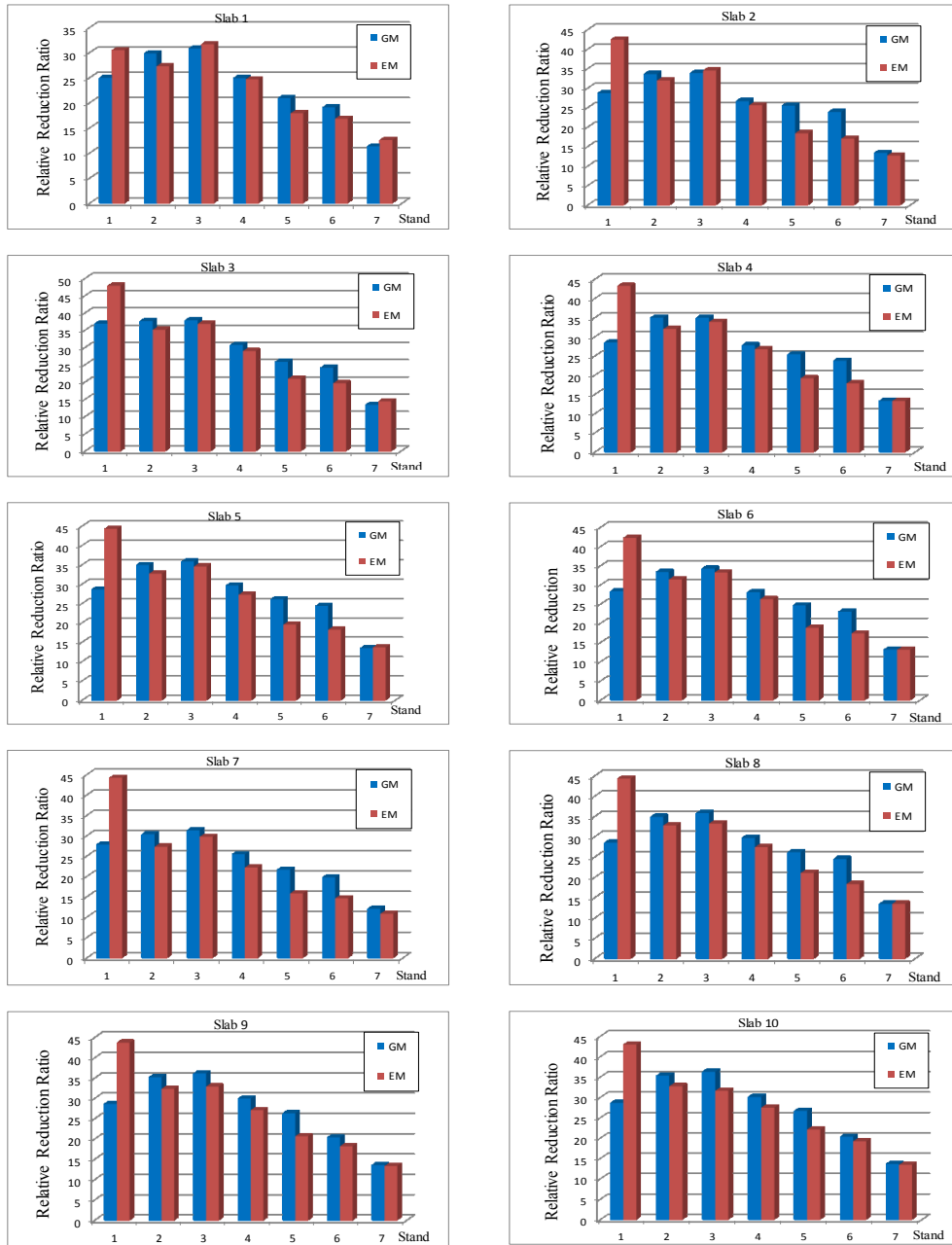
Figure 5: Comparison results for relative reduction ratio of the thickness.

to the specification and can obviously prevent the strip shape problems. So, the comparison results shown in Table 2 indicate that the parameter setting obtained by GM can achieve more reasonable distribution than those of the EM. The parameters of the crown and the reduction ratio guarantee the quality of the product.
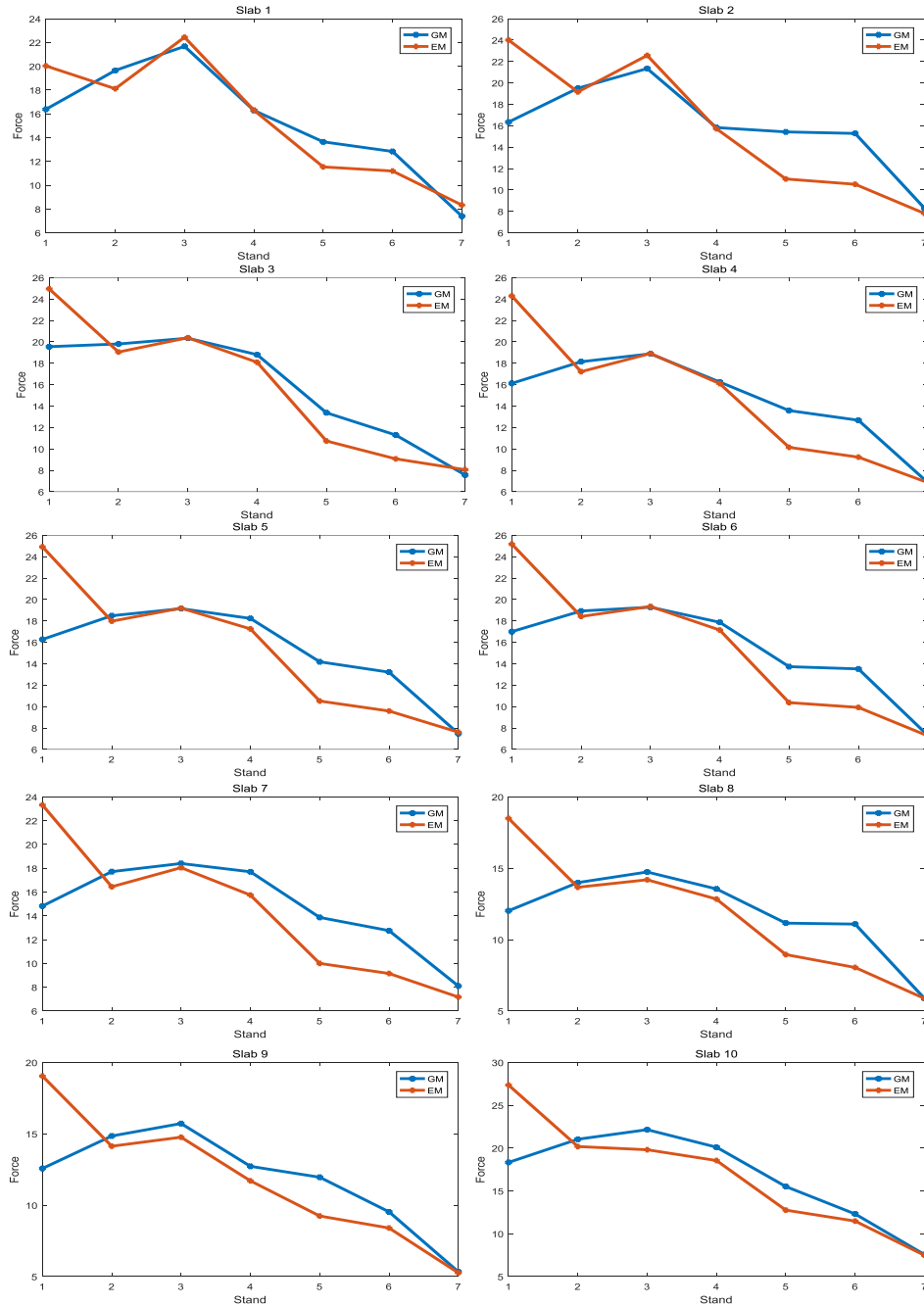
Figure 6: Comparison results for the force.

Figure 5 shows a relative reduction ratio comparison of the thickness by using two methods, where the relative reduction ratio is equal to $\frac{h_{i-1}-h_i}{h_{i-1}}$. Figure 6 presents the graphical illustrations of rolling force at each stand. All parameters of the first stand are lower than the second one in the GM methods, which is different from the EM setting. This guarantees that the slab can be

bitten successfully into the first stand. The slab temperature gradually decreases during the rolling process. Thus, the setting style from the second to fourth stands makes full use of the capacity of the equipment in order to increase the thickness reduction as far as possible. The reduction rate is reduced gradually at the final stages of rolling so to keep the quality and mechanical properties of final strip. As a whole, optimized settings are better than the actual experienced settings.

Conclusion

In this paper, we have discussed a class of operation optimization problem in steel rolling systems and presented a global method to solve it which is based on ADMM. Using the structural feature of the problem, we built the link between the original problem and convex problem. A cut interval method was used to reduce the current feasible region to improve the accuracy of subproblem. This algorithm has been applied to solve 10 different hot rolling operation optimization problems in numerical experiments. The results showed that the global method can generate effective control parameter settings that reduce energy consumption of the rolling process. This can guarantee that the initial best control parameters can successfully transmit to AGC before entering the finishing mill.

Acknowledgements

References

(1) Lally, B; Biegler, L. T., Henein H. Optimization and continuous casting: Part II.Application to industrial casters. *Metall. Trans. B* 1991, 22(5):649-659.

(2) Tang, L.; Liu, J.; Rong, A.; et al. A review of planning and scheduling systems and methods for integrated steel production. *European Journal of Operational Research*. 2001, 133(1):1-20.

(3) Aylen, J. Megabytes for metals: development of computer applications in the iron and steel industry. *Ironmaking & Steelmaking*. 2004, 31(6):465-478.

(4) Kano, M.; Nakagawa, Y. Data-based process monitoring, process control, and quality improvement: Recent developments and applications in steel industry. *Comput. & Chem. Eng.* 2008, 32(1):12-24.

(5) Brunaud, B.; Grossmann, I. E. Perspectives in multilevel decision-making in the process industry. *Frontiers of Engineering Management*, 2017, 4(3), 256-270.

(6) Depree, N.; Taylor, M.; Chen, J.; et al. Development of a Three-Dimensional Heat Transfer Model for Continuous Annealing of Steel Strip. *Ind. Eng. Chem. Res.* 2012, 51(Nigam):1790-1795.

(7) Chen, L.; Wang, X. P.; Tang, L. X. Operation optimization in the hot-rolling production process. *Ind. Eng. Chem. Res.* 2014, 53(28):11393-11410.

(8) Karelahti, J.; Vainiomäki, P.; Westerlund, T. Large scale production planning in the stainless steel industry. *Ind. Eng. Chem. Res.* 2011, 50(9):4893-4906.

(9) Li, J.; Xiao, X.; Tang, Q.; et al. Production scheduling of a large-scale steelmaking continuous casting process via unit-specific event-based continuous-time models: short-term and medium-term scheduling. *Ind. Eng. Chem. Res.* 2012, 51(21):7300-7319.

(10) Santander, J. A.; López, E.; Tonetto, G. M.; et al. Preparation of NiNbO/AISI 430 ferritic stainless steel monoliths for catalytic applications. *Ind. Eng. Chem. Res*. 2014, 53(28):11312-11319.

(11) Zhang, Y.; Zhou, J.; Li, C.; et al. Reaction Kinetics and Mechanism of Iron(II)-Induced Catalytic Oxidation of Sulfur(IV) during Wet Desulfurization. *Ind. Eng. Chem. Res*. 2012, 51(3):1158-1165.

(12) Yin, R. Metallurgical process engineering. *Metallurgical Industry Press,* 2011.

(13) Zhang, J. Z.; Xu, T. Z.; Li, M. The development course of thickness automatic control technology for the strip rolling process. *Baosteel BAC*. 2013, L-29.

(14) Lu, C.; Zhao, Q. L.; Liu, L. Z.; Liu, X. H.; Wang, G. D. Optimizing algorithm of pass schedule for finisher. *J. Iron and Steel Res*. 2001, 13(1):26-29.

(15) Li, H. J.; Xu, J. Z.; Wang, G. D.; Liu, X. H. Improvement on conventional load distribution algorithm in hot tandem mills. *J. Iron and Steel Res*. 2007, 14(2):36-41.

(16) Zhang, H. D.; Zhang, X. P. Rolling schedule optimization based on ϕ function load assignment by dynamic programming method. *Mech. Eng. & Automat*. 2012.

(17) Ozsoy, I. C.; Ruddle, G. E.; Crawley, A. F. Optimum scheduling of a hot rolling process by nonlinear programming. *Can. Metall. Q.*. 1992, 31(3):217-224.

(18) Yao, F.; Yang, W. D. Improved differential evolution algorithm used for load distribution of hot strip rolling. *IEEE Bio-Inspired Computing: Theories and Appl*. 2010, BIC-TA: 526-530, 2010.

(19) Misener, R.; Gounaris, C. E.; Floudas, C. A. Global Optimization of Gas Lifting Operations: A Comparative Study of Piecewise Linear Formulations. *Ind. Eng. Chem. Res*. 2009, 48(13):6098-6104.

(20) Boyd, S.; Kim, S. J.; Vandenberghe, L.; et al. A tutorial on geometric programming. *Optimization & Engineering*. 2007, 8(1):67.

(21) Passy, U.; Wilde, D. J. A Geometric programming algorithm for solving chemical equilibrium problems. *Siam J. Appl. Math.* 1968, 16(2):363-373.

(22) Clasen, R. J. The Solution of the Chemical Equilibrium Programming Problem with Generalized Benders Decomposition. *Oper. Res.* 1984.

(23) Wall, T. W.; Woolsey, D. G. E. D. Solving Complex Chemical Equilibria Using a Geometric-Programming Based Technique. *Oper. Res.* 1986, 34(3):345-355.

(24) Salomone, H. E.; Iribarren, O. A. Posynomial Modeling of Batch Plants: A Procedure to Include Process Decision Variables. *Comput. Chem. Eng.* 1992, 16(3):173-184.

(25) Salomone, H. E.; Montagna, J. M.; Iribarren, O. A. Dynamic simulations in the design of batch processes. *Comput. Chem. Eng.* 1994, 18(3):191-204.

(26) Avriel, M.; Wilde, D. J. Optimal Condenser Design by Geometric Programming. *Ind. Eng. Chem. Process Des. Dev.* 1967, 6(2):256-263.

(27) Kasamsetty, K.; Ketkar, M.; Sapatnekar, S. S. A new class of convex functions for delay modeling and its application to the transistor sizing problem [CMOS gates]. *IEEE Transactions on Computer-Aided Design of Integrated Circuits, and Systems*. 2000, 19(7):779-788.

(28) Ecker, J. G. Geometric programming: methods, computations and applications. *Siam Rev.* 1980, 22(3):338-362.

(29) Hershenson, M. D.; Boyd, S. P.; Lee, T. H. Optimal design of a CMOS op-amp via geometric programming. *IEEE Trans Cmput. -Aided Des. Integr. Circuits Syst.*. 2001, 22(1):1-21.

(30) Singh, J.; Luo, Z. Q.; Sapatnekar, S. S. A Geometric programming-based worst case gate sizing method incorporating spatial correlation. *IEEE transactions on computer-aided design of integrated circuits and systems*. 2008, 27(2):295-308.

(31) Jefferson, T. R.; Scott, C. H. Generalized geometric programming applied to problems of optimal control: Part I, theory. *Journal of Optimization Theory & Applications*. 1980, 30(1):149-149.

(32) Barmi, H. E.; Dykstra, R. L. Restricted multinomial maximum likelihood estimation based upon Fenchel duality. *Statistics & Probability Letters*. 1994, 21(2):121-130.

(33) Scott, C. H.; Jefferson, T. R. Allocation of resources in project management. *International Journal of Systems Science*. 1995, 26(2):413-420.

(34) Tsaia, J. F.; Hu, Y. C. On generalized geometric programming problems with non-positive variables. *European Journal of Operational Research*. 2007, 178(1):10-19.

(35) Gounaris, C. E.; Floudas, C. A. Convexity of products of univariate functions and convexification transformations for geometric programming. *Journal of Optimization Theory & Applications*, 2008, 138(3):407-427.

(36) Lu, H. C.; Li, H. L.; Gounaris, C. E.; et al. Convex relaxation for solving posynomial programs. *Journal of Global Optimization*. 2010, 46(1):147-154.

(37) Lu, H. C. An efficient convexification method for solving generalized geometric problems. *Journal of Industrial & Management Optimization*. 2012, 8(2):429-455.

(38) Pörn, R.; Björk, K. M.; Westerlund, T. Global solution of optimization problems with signomial parts. *Discrete Optimization*. 2008, 5(1):108-120.

(39) Lin, M. H.; Tsai, J. F. Range reduction techniques for improving computational efficiency in global optimization of signomial geometric programming problems. *European Journal of Operational Research*. 2012, 216(1):17-25.

(40) Jun, L. H.; Dong, W. G.; Zhong, X. J.; et al. Optimization of Draft Schedules With Variable Metric Hybrid Genetic Algorithm During Hot Strip Rolling. *J. Iron Steel Res.* 2007, 19(8):33-36.

(41) Alizadeh, F.; Goldfarb, D. Second-order cone programming. *Mathematical Programming*. 2003, 95(1):3-51.

(42) Duffin, R. J.; Peterson, E. L. Geometric programming with signomials. *Journal of Optimization Theory & Applications*. 1973, 11(1):3-35.

(43) Sun, Y. K. Model and Control of Hot Strip Continuous Rolling. *Metallurgical Industry Press*. 2007.

(44) Sun, D.; Toh, K. C.; Yang, L. A Convergent 3-Block SemiProximal Alternating Direction Method of Multipliers for Conic Programming with 4-Type Constraints. *SIAM Journal on Optimization*. 2015, 25(2):882-915.