Process Systems Engineering

Genetic Algorithm Based on Duality Principle for Bilevel Programming Problem in Steel-making Production

Shuo Lin 1,⁎, Fangjun Luan 1, Zhonghua Han 1, Xisheng Li 2, Xiaofeng Zhou 2, Wei Liu 3

1 Information & Control Engineering Faculty, Shenyang Jianzhu University, Shenyang 110168, China
2 Shenyang Institute of Automation, Chinese Academy of Sciences, Shenyang 110016, China
3 School of Information Science & Engineering, Northeastern University, Shenyang 110004, China

A R T I C L E   I N F O

Keywords:
Steel-making
Genetic algorithm
Bilevel problem
Scheduling

A B S T R A C T

Steel-making and continuous/ingot casting are the key processes of modern iron and steel enterprises. Bilevel programming problems (BLPPs) are the optimization problems with hierarchical structure. In steel-making production, the plan is not only decided by the steel-making scheduling, but also by the transportation equipment. This paper proposes a genetic algorithm to solve continuous and ingot casting scheduling problems. Based on the characteristics of the problems involved, a genetic algorithm is proposed for solving the bilevel programming problem in steel-making production. Furthermore, based on the simplex method, a new crossover operator is designed to improve the efficiency of the genetic algorithm. Finally, the convergence is analyzed. Using actual data the validity of the proposed algorithm is proved and the application results in the steel plant are analyzed.
© 2014 Chemical Industry and Engineering Society of China, and Chemical Industry Press. All rights reserved.

1. Introduction

During recent years, the continuous casting technology developed in the 1950s underwent a rapid growth. Because the continuous casting technology can save more energy consumption in the production process from hot steel to slab, the main steels are produced by continuous casting. During the process of steel-making and continuous casting, most of the material transportations within the whole process are finished by cranes. The crane scheduling is the scheduling important branch to assist the scheduling of the main equipments (steel-making, refining, continuous casting machines). A coordinating steel-making and continuous casting system consists of the scheduling of main equipments and assist transportation equipments. Because a good iron and steel making scheduling system could not only realize the reduction of the energy but also improve the production, and the steel-making and continuous casting is the bottleneck of the whole iron and steel production, the crane scheduling plays an important role in the whole process. The major problem of this paper is how to make a good crane schedule in a computationally efficient manner and effectively assist the scheduling of the main equipments in order to ensure a well-organized rhythm in the whole production and to improve the transportation efficiency of the cranes.

Harjunkoski and Grossmann [1] developed a decomposition strategy for solving large scheduling problems using mathematical programming methods. Lee et al. [2] solved a scheduling problem for operating the continuous caster by using the concept of a special class of graphs known as interval graphs. Ouelhadj et al. [3] described a new model for robust predictive/reactive scheduling of SCC based on the use of multi-agents, tabu search and heuristic approaches. At Stahl Linz GmbH, Neuwirth [4] reported a linear programming model with machine convicts and provided key modeling factors of SCC scheduling and charge allocation scheme in the furnace, but the mathematical representation of the model was not given. An expert system was used by Jimichi et al. [5] to determine parameters and operational conditions to match slab production with customer orders.

Stahl Linz GmbH, Stohl and Spokek [6] established a hybrid co-operative expert system model for the SCC scheduling problems, but they were unable to construct an optimized mathematical model. Ferretti et al. [7] presented the algorithmic solution based on an ant system metaheuristic for an industrial production inventory problem in a steel continuous casting plant. The model took into account the relevant parameters of the finite-capacity production system. The study focused on the optimization of the production sequence of the billets. Blackburn and Millen [8] examined the impact of a rolling-schedule implementation on the performance of three of the better known lot-sizing methods for single-level assembly systems. Atighehchian et al. [9] developed a novel iterative algorithm for scheduling steel-making continuous casting production, named HANO. This algorithm was based on a combination of ant colony optimization (ACO) and non-linear optimization methods. Zhu et al. [10] set a novel optimization model to improve the efficiency and performance for production planning in steelmaking and continuous casting (SCC) process.

http://dx.doi.org/10.1016/j.cjche.2014.05.006
1004-9541/© 2014 Chemical Industry and Engineering Society of China, and Chemical Industry Press. All rights reserved.
2. Genetic Algorithm for Bilevel Problem

Bilevel-programming techniques were initiated by Von Stackelberg [11] for bilevel programming problems (BLPPs) based on the dual programming of the follower’s problem and the dual theorem. The region of the leader’s variables was divided into several sub-regions, such that for all values of the leader’s variables in each sub-region, the follower’s problem had the same solution expression. Furthermore, a genetic algorithm was used to solve the leader’s problems, in which a new crossover operator was designed based on the simplex method. An advantage of the proposed genetic algorithm was that the optimal solutions to the follower’s problem can be obtained very easily without massive calculation of the follower’s problem. Also, a method of dynamic penalty function was presented.

2.1. Bilevel problem

The bilevel problem is as follows:

\[
\begin{align*}
\min_{x} & \quad F(x, y) \\
\text{s.t.} & \quad G(x, y) \leq 0 \\
& \quad d^T y \in \mathbb{R}^n, A \in \mathbb{R}^{m \times n}, B \in \mathbb{R}^{p \times n}, C \in \mathbb{R}^q. \\
& \quad s.t. \quad Ax + By \leq C, y \geq 0.
\end{align*}
\] (1)

The dual problem is as follows:

\[
\begin{align*}
\min_{u} & \quad (C - Ax)^T u \\
\text{s.t.} & \quad B^T u \geq -d^T u > 0.
\end{align*}
\] (2)

\{u_1, u_2, \ldots, u_k\} is the basic feasible solution of problem (2). \{B_1, B_2, \ldots, B_k\} is the feasible basis. \{(C - Ax)^T_u B_i^{-1} B^T_i\} is the corresponding objective function of \{B_i\}. The necessary and sufficient condition that \{B_i\} is the optimal feasible basis is \((C - Ax)^T_u (C - Ax)^T_u B_i^{-1} B^T_i \geq 0, i = 1, 2, \ldots, k\). For each inequality \(i\), the inequality \((C - Ax)^T_u (C - Ax)^T_u B_i^{-1} B^T_i \geq 0\). So there are \(k\) inequality groups. Domain \(i\) is the divided domain of inequality. If leader variable \(x\) satisfies inequality \(i\), it belongs to domain \(i\). According to the duality principle, the optimal solution of the follower problem is \(\{(C - Ax)^T_u B_i^{-1}\}^1\). There are two steps in the calculation.

1. Solve the domain \(i\) of leader variable \(x\), calculate \(y(x) = \{(C - Ax)^T_u B_i^{-1}\}^1\).

2. Calculate the problem:

\[
\begin{align*}
\min_{x \in X} & \quad F(x, y(x)) \\
\text{s.t.} & \quad G(x, y(x)) < 0.
\end{align*}
\] (3)

2.2. Fitness function

Set \(R(x) = F(x, y(x)) + M_g\) and \(\max(G(x, y(x))), 0, i = 1, 2, \ldots, p\) is individual fitness. Where \(g\) is the iteration of genetic; \(G(x, y(x))\) is the component of \(G(x, y(x))\), \(M_g\) is the dynamic penalty factor and \(M_g\) is a pre-given positive integer decided by the point in population. Set \(x_0, x_1, x_2, \ldots, x_n\) is \(n + 1\) optimal and no identical points which are collected by fitness. Calculate

\[
d = \frac{1}{n+1} \sum_{i=1}^{n} \|x_i - x_G\| \quad \text{where} \quad x_G = \frac{1}{n+1} \sum_{i=0}^{n} x_i
\] (4)

\[
M_{g+1} = \max\left\{ M_g, \frac{1}{g} \right\}
\] (5)

If \(n + 1\) points are affine independence, \(\{x_0, x_1, x_2, \ldots, x_n\}\) is a simplex. With an increasing iteration number, points in population tend to accordance, when \(M_g\) tends to infinity.

2.3. Crossover operator

\(p_c\) is population size. Sort the fitness of pop(\(k\)) points ascending as \(x_1, x_2, \ldots, x_{p_c}\). If \(x_1, x_2, \ldots, x_{\lceil \frac{p_c}{2} \rceil} \) \((\frac{p_c}{2} < n)\) is affine independence, set \(S = \{x_1, x_2, \ldots, x_{\lceil \frac{p_c}{2} \rceil}\}\). If point \(x_{\lceil \frac{p_c}{2} \rceil} + 1\) points in \(S\) is affine independence, \(S = S_{\lceil \frac{p_c}{2} \rceil + 1}\); else study to the next point \(x_{\lceil \frac{p_c}{2} \rceil + 1}\) until that \(S\) includes \(n\) points. If there are not \(n\) affine independence points, \(x_1\) is the vertex and construct other \(n - 1\) points in its neighborhood. Make sure \(n\) points are affine independence; The construction method is from Takahama T. and Sakai S. [11]. Re-sort the population, delete the worst \(n - 1\) points, make \(p_c\) unchanged. Calculate the center of gravity: \(x_c = \frac{1}{n} \sum_{i=1}^{n} x_i\), \(x_1\) is the last point in \(S\), and \(\hat{S} = \{x(x) \leq R(x_G), x \in \text{pop}(k)\}\), so \(S \subseteq \hat{S}\). Offspring of crossover individual \(x_i\) is

\[
o_c = \left\{ \begin{array}{ll}
x + r(\hat{x}_c - x), & x \notin S \\
x + r(\Delta(x_1 - x)), & x \in S
\end{array} \right.
\] (6)

where, \(r \in [0, 1]\) is random number, \(\hat{x}_c\) and \(\Delta\) is positive constant. When \(r = 1\), \(\hat{x}_c\) reaches boundary of search field. \(x_1\) is the best point in population. When crossover individual \(x\) and points in \(S\) is affine independence, there is a simplex which can use the simple algorithm.

2.4. Mutation operator

\(\hat{x}\) is the father individual:

If \(|x_1 - \hat{x}| \leq \varepsilon\), mutationoffspring \(a_m = \hat{x} + \Delta_1; \text{else } a_m = \hat{x} + \text{diag}(\Delta_2)(x_1 - \hat{x})\);

where \(\Delta_1 = (\Delta_1, \Delta_2, \ldots, \Delta_m)^T, i = 1, 2; \Delta_{ij} - N(0, \sigma^2), j = 1, 2, \ldots, n; \Delta_{ij} - N(1, \sigma^2), j = 1, 2, \ldots, n; N(u, \sigma^2)\) is the Gaussian distribution, \(u\) is the mean and \(\sigma^2\) is the variance.

2.5. Genetic algorithm

The genetic algorithm is as follows:

Step 0: Calculate the feasible basis of follower dual problem \(B_1, \ldots, B_k\);

Step 1: Generate \(p_c\) points \(x_i, i = 1, 2, \ldots, p_c\) in boundary constraint set \(X\) uniformly. Substitute each \(x_i\) into the explicit expression in the follower level. To get \(y(x_i)\), let all points \(x_i\) be the population pop(0) whose size is \(p_c\), Calculate the fitness of each individual and sort the fitness in ascending order. For conscience, denote pop(0) = \(\{x_1, x_2, \ldots, x_{p_c}\}\), let \(k = 0\);

Step 2: \(p_c\) is the crossover probability. Get crossover individual \(x\) from pop(k), \(\hat{x}\) is the crossover offspring of \(\hat{O}_1\) is set of all crossover offspring;

Step 3: For each individual \(x\) in pop(k), mutate with probability \(p_m\), get mutation offspring \(x_{O2}\) is set of mutation offspring;

Step 4: Calculate fitness of all offspring individual, get \(N_c(N_1 < p_c)\) best individuals from \(\text{pop}(k) \cup \hat{O}_1 \cup \hat{O}_2\). Get the next generation population \(\hat{O}(k + 1)\) from other \(p_c - N_c\) individuals;

Step 5: If match terminating condition, algorithm end, else let \(k = k + 1\), go to Step 2.

2.6. Convergence analysis

**Definition 1.** \((\xi_k)\) is a vector sequence of probability space. If \(\xi\) exists, \(\lim_{m \to \infty} \xi_m = 1\), or \(\forall \varepsilon > 0, \prob \left( \bigcap_{k=m}^{\infty} \{||\xi_k - \xi|| \geq \varepsilon\} \right) = 0\). \((\xi_k)\) converge to \(\xi\) on probability 1.
Lemma 1. (Borel–Cantelli) If \(A_1, A_2, \ldots, A_m, \ldots\) is a vector sequence of probability space, set \(p_k = \text{prob}(A_k)\). If \(\sum_{k=1}^{\infty} p_k < \infty\), \(\text{prob}\left(\bigcap_{k=1}^{\infty} A_k\right) = 0\). If \(\sum_{k=1}^{\infty} p_k = \infty\), \(A_1, A_2, \ldots, A_m, \ldots\) is mutual independence, so \(\text{prob}\left(\bigcap_{k=1}^{\infty} A_k\right) = 1\).

Upper space projection of IR is \(S'(X) = \{x \in X \mid \exists y, (x, y) \in IR\}\). There are five assumptions:

Assumption 1. Feasible region of follower problem and dual problem is bounded.

Assumption 2. There is only optimal solution for each \(x\) in upper level.

Assumption 3. \(S'(X)\) is a bounded closed set, and \(F(x, y)\) is continuous on \(\Omega\).

Assumption 4. There is a global minimizer \(x^*\) at least; \(\forall \delta > 0\), the Lebesgue measure of set \(S'(X) \cap \{x||x - x^*|| < \delta\}\) is more than 0.

Assumption 5. \(\forall x \in X\), the follower problem is nondegenerate.

For \(\forall \varepsilon > 0\), \(Q_1 = \{x \in S'(X)||F(x, y(x)) - F^*|| \leq \varepsilon\}\), \(Q_2 = S'(X)|Q_1\) \(F^* = \min F(x, y) : (x, y) \in IR\) \(F^* = \{F(x^*, y(x^*)) : x^* \in S'(X)\}\). So there are two states \(S_1\) and \(S_2\):

- \(S_1\): If there is a point that belongs to \(Q_1\) at least, \(p(k)\) is in \(S_1\)
- \(S_2\): If there is on point that belongs to \(Q_1\), \(p(k)\) is in \(S_2\).

Theorem 1. \(p_{ij}(i, j = 1, 2)\) means \(p(k)\) in state \(S_i\). The probability that \(p(k+1)\) is in state \(S_j\) is as follows:

(a) For each \(p(k)\) in state \(S_1\), \(p_{11} = 1\).
(b) For each \(p(k)\) in state \(S_2\), \(\exists c \in (0, 1)\), \(p_{22} < c\).

Proof. If \(p(k) \in S_1, p(k) \in S_1, p(k+1) \in S_1, \) so \(a\) is tenable. \(S_0\) is the optimal solution set. \(\forall x^* \in S_0\) which satisfies Assumption (4). When \(x \rightarrow x^*, y(x) \rightarrow y(x^*)\), where \(y(x)\) is the follower optimal solution corresponding to \(x\). From linear programming theory and Assumption (5), basis variable component \(B^{-1}(C-Ax^*) < 0\) is the follower optimal solution for \(x = x^* + \Delta x\), \(B^{-1}(C-Ax^* + \Delta x) = B^{-1}(C-Ax^*) - B^{-1}\Delta x\). Obviously, \(\text{|\Delta x|} \rightarrow 0\), \(B^{-1}(C-Ax^* + \Delta x) > 0\); and \(y(x) \rightarrow y(x^*)\), so \(y(x)\) is continuous at \(x^*\).

Because \(F(x, y)\) is continuous, \(\forall \varepsilon > 0, \exists \gamma > 0\), when \(x \in S'(X)\) and \(||x - x^*|| \leq \gamma\),

\[
|F(x, y(x)) - F^*| < \frac{\varepsilon}{2}.
\]

When \(x \in S'(X) \cap \{x||x - x^*|| \leq \gamma\}\), \(F(x, y(x)) - F^* < \varepsilon\). Let \(N_{\gamma}(x^*) = \{x \in R^n ||x - x^*|| < \gamma\}\). When \(p(k)\) is in state \(S_2\), \(o_m\) is the mutation offspring of \(x^*\) \(p(k)\), so \(o_m = x + \Delta o\) or \(o_m = \delta + \text{diag}(\Delta)(x_1 - x)\).

Because each component of \(\Delta o\) is independent, and Gaussian distribution \(i \in (1, 2)\). So each component of \(o_m\) is independent and Gaussian distribution. Denote \(o_m = N(\mu, \sigma^2)\), \(\mu\) and \(\sigma^2\) is continuous about \(x\). The probability of \(o_m \in S'(X) \cap N_{\gamma}(x^*)\) is as follows:

\[
\text{prob}\left(\bigcap_{k=1}^{\infty} A_k\right) = \int_{S'(X) \cap N_{\gamma}(x^*)} \left(\frac{1}{\sqrt{2\pi\sigma^2}}\right)^n \exp\left(-\frac{1}{2\sigma^2}(x - \mu)^2\right) dx_1, \ldots, dx_n
\]

Theorem 2. \(p(k)\) is a population sequence created by algorithm. There is one point in \(p(0)\) that belongs to \(S'(X)\). \(x^*(k)\) is the optimal point in \(p(k)\). When Assumptions (1)–(5) are satisfied:

\[
\text{prob}\left(\left| F(x^*(k), y(x^*(k))) - F^* \right| \geq \varepsilon \right) = 1.
\]

Proof. \(\forall \varepsilon > 0\), denote \(p_k = \text{prob}\left(|F(x^*(k), y(x^*(k))) - F^*| \geq \varepsilon\right)\), so

\[
p_k = \int_{\mathbb{R}^n} \chi_{P_{\gamma}}(t)\chi_{Q_1}(t) dt, \quad t = 1, 2, \ldots, k
\]

From Theorem 1, \(P_{\gamma} = p_{22} \leq c^k\). So \(\sum_{k=1}^{\infty} p_k \leq \sum_{k=1}^{\infty} c^k = \frac{c}{1-c} \leq \text{c}\). From Lemma 1, Borel–Cantelli,

\[
\text{prob}\left(\bigcup_{m=1}^{\infty} \{ |F(x^*(k), y(x^*(k))) - F^*| \geq \varepsilon \} \right) = 0. \text{Complete.}
\]

3. Mathematical Model

During the process of steel-making and continuous casting, based on the scheduling of the main equipments and the corresponding steel ladle destination, the routes of the ladle could be divided into the following modes:

1) LD \(\rightarrow\) no finery \(\rightarrow\) casting mode: after receiving the molten steel when the process of LD process is finished, the crane transports the motel steadily by steel ladle to the casting, takes it for preprocessing and takes the steel ladle to the continuous casting by crown crane.

2) LD \(\rightarrow\) no finery \(\rightarrow\) IC mode: after receiving the molten steel when the LD process is finished, the crane transports the motel steely by steel ladle to the casting, takes it for preprocessing and takes the steel ladle by crane to another casting line, and the crown crane will take the steel ladle to the ingot casting finally.

3) LD \(\rightarrow\) finery \(\rightarrow\) casting mode: after receiving the molten steel when the LD process is finished, the charge is sent to the casting line. After the pre-process, the charge is sent to the refining stage, the crown cranes take the action of loading and unloading, and the crown crane takes the charge into the casting line.

4) LD \(\rightarrow\) finery \(\rightarrow\) IC mode: after receiving the molten steel when the LD process is finished, the charge is sent to the casting line. After the pre-process, the charge is sent to the refining stage, the crown
cranes take the action of loading and unloading, and the crown crane takes the charge into the ingot casting line.

To describe the continuous/ingot casting production process, we introduce the following special terms:

| Charge: | The process of a ladle of molten iron from converter to refinery and continuous/ingot casting. It contains several units, like job. |
| Work station: | In one operating procedure, the work station means the operation time of a charge plan in a certain kind of steel-making equipment. A charge plan includes several work stations. |

### 3.1. The scheduling model of continuous and ingot casting

Half-charge plan adjustment keeps processing equipment \( y_{jk} \) invariant, which is located in the nonproducing work station. Making plan of equipment’s processing starting time \( s_{jk} \).

The mathematical model for the mixed half-charge plan and scheduling problem is formulated as follows:

\[
 f(t) = \min_{i=1,j=1,k=1} \left( s_{jk}Y_{jk} \right). 
\]  

Subject to:

\[
st_{im,k} + pt_{im,k} + pt_{im,k} + pt_{im,k} - s_{im,k}Y_{im,k}Y_{im,k} - \Delta t_{ing} = st_{im,k}Y_{im,k} \quad \Omega \tag{11}
\]

\[
est_{jk}y_{jk} = st_{jk}y_{jk} + pt_{jk}y_{jk} \quad \Omega \tag{13}
\]

\[
st_{jk}y_{jk} + pt_{jk}y_{jk} + ut_{j-1,j} \leq st_{j-1,j}kY_{j-1,j-1,k} \tag{15}
\]

\[
st_{jk}, et_{jk} > T_{now}. \tag{16}
\]

Constraint (11) represents that the charges in a same continuous casting must have close cohesion, pre and post. This station is the \( m \) process which is also the last process. Constraint (12) represents that there is a setup time interval between adjacent ingot castings. Constraint (13) represents that each charge cannot be interrupted when it is processing on a station. Constraint (14) represents the constraints of the equipment. The same equipment can only process the next charge when the last one is over. Constraint (15) represents the constraints of the job order. The same charge can only continue the next station when the last one is over. Constraint (16) represents the starting time of the station.

### 3.2. Crane schedule

In order to facilitate the mathematical analysis of the crane scheduling problem, several conditions are assumed.

- The finishing operation order of every plan is decided.
- Every plan must be only the overall production and cannot be divided into multiple parts.
- More of the same quality product cannot be produced as a whole.
- Each processing time on the device is known.
- The crane position in the same rail remains the same sequence to maintain the shortest distance \( \delta \) between two adjacent cranes.

Introduce the following variables:

\[
x_{jk}^m: \text{When} \ O_{jk} \text{works at device} \ m, x_{jk}^m = 1; \text{otherwise} \ x_{jk}^m = 0
\]

\[
x_{hj,k_2}^m: \text{When} \ O_{hj} \text{is operated after} \ O_{j,k_2} \text{at device} \ m, x_{hj,k_2}^m = 1; \text{otherwise} \ x_{hj,k_2}^m = 0.
\]

\[
y_{jk}^k: \text{When crane} \ k \text{is assigned to a transportation job of completed} \ O_{jk}, y_{jk}^k = 1; \text{otherwise} \ y_{jk}^k = 0.
\]

\[
y_{jk}^k: \text{When crane} \ k \text{is occupied to transportation job of completed} \ O_{jk} \text{at time} \ t, y_{jk}^k = 1; \text{otherwise} \ y_{jk}^k = 0.
\]

\[
G: \text{Sufficiently large constant value.}
\]

The objective is to minimize the sum of project completion time. The objective function is as follows:

\[
\min \sum_{i \in D} \left( ET_{i,n_i} - ST_{i,1} \right). \tag{17}
\]

Subject to:

\[
\sum_{m=1}^{M_0} x_{m}^m = 1 \tag{18}
\]

\[
\sum_{O_{jk}} x_{jk}^m = 1 \tag{19}
\]

\[
\sum_{O_{jk}} x_{jk}^m = 1 \tag{20}
\]

\[
ST_{j,k} \geq ET_{j,k} - G \left( 1 - x_{k,j} \right) \left( 1 - x_{j,k} \right) \tag{21}
\]

\[
ST_{j,k} \geq ET_{j,k} - G \left( 1 - x_{k,j} \right) \left( 1 - x_{j,k} \right) \tag{22}
\]

\[
ET_{j,k} \geq ST_{j,k} + OT_{q} + 2d \tag{23}
\]

\[
ST_{j,k} \geq ET_{j,k} + CT \left( a_{jk}, a_{j+k} \right) \tag{24}
\]

\[
\sum_{k} y_{jk} = 1 \tag{25}
\]

\[
\sum_{O_{jk}} y_{jk} = 1 \tag{26}
\]

\[
a_{jk} = \sum_{m=1}^{M_0} p_{m}^m x_{jk}^m \tag{27}
\]

\[
p_{k}^t + \nu \geq p_{k}^{t-1} \geq p_{k}^t - \nu \tag{28}
\]

\[
p_{k}^t_{k+1} \geq p_{k}^t + \delta \tag{29}
\]
be completed before the follow-up job starting in the same device. Constraints (23) and (24) are the requirement of crane transportation. Constraint (25) ensures that every $O_{ij}$ has a crane. Constraint (26) means one crane must have a job. Constraint (27) shows that the processing position and the position assigned to $O_{ij}$ is the same. Constraint (28) means the crane has its maximum speed. Constraint (29) shows the shortest distance $\delta$ between two adjacent cranes.

4. Numerical Results

In the plant, the scheduling plan is a manual plan. The steelmaking plant has so many equipments: 3 converters, 7 refineries, 3 continuous casters, 3 ingot casting lines and 4 cranes.

Set the parameter of the algorithm with the steelmaking plant plan and steel species. Save the parameter to rule base. Get the parameter from rule base based on schedule plan and steel species each time. A comparison is made between the scheduling method and model system that is used in steel plant. The finish time of each heat is shown in Fig. 1.

Using the scheduling method, the average heat finish time is reduced by 13.5 min. The average rate of this scheduling method in the steel-making plant is shown in Fig. 2. The method has a better performance. After being applied in the operation in the steel plant at Shanghai, the performance index is as follows: The average planning time is 8.4 s and the dynamic adjusting time is about 4.5 s in the production mode such that the daily average production is 66 heats. Comparing with the manual and general model systems, the average daily load of the converter increases from 42.21% to 45.00%, the average rate of equipment load increases from 50.44% to 55.16% and the average heat completion time goes from 195.88 min down to 176.38 min.

5. Conclusions

Bilevel programming problems (BLPPs) are nonconvex optimization problems with hierarchical structure. The vast majority of the existing research works on BLPPs is concentrated on linear BLPPs and some special nonlinear BLPPs in which all of the functions involved are convex and twice differentiable. In this paper, a genetic algorithm used in steel plant is given to solve the bilevel problem. The convergence of the algorithm is verified. The algorithm result is verified that the scheduling method and policy of the steelmaking continuous and ingot casting are efficient.

**Nomenclature**

- $a_{ij}$: position of device which processes $O_{ij}$
- $adj_{\text{ingot}}$: the standard interval time between ingot castings
- $CT(a,b)$: crane running time from position $a$ to position $b$
- $d$: lifts and drops time when crane transports the ladle
- $ET_{ij}$: processing end time of $O_{ij}$
- $et_{ijk}$: the work station $j$’s processing end time of charge $i$ processed on equipment $k$
- $j$: serial number of work station, $1 \leq j \leq m_i$
- $K$: the total amount of equipments;
- $K_j$: the parallel machine number of work station $j$
- $k,k'$: serial number of equipment. $1 \leq k, k' \leq K$
- $M$: number of available crane
- $m_i$: the total work station amount of charge $i$
- $N_i$: manufacturing procedure set of charging plan $i$, $i \in U$
- $n$: serial numbers of cast, $N$ is the total amount of continuous casting batch casts, $n = 1,2,...,N$
- $n_i$: the number of plan $i$’s processes, $n_i = |N_i|$
- $O_{ij}$: the process $j$ of plan $i$, $j = 1, ..., n_i$
- $OM_{ijk}$: device collection can be used for $O_{ij}$
- $OT_{ij}$: process time of $O_{ij}$
- $P_k$: position of crane $k$ at time $t$
- $P_{m}$: position of device $m$
- $p_{ijk}$: the work station $j$’s standard processing time of charge $i$ processed on equipment $k$
- $SI(i,j,k)$: the charge after process $j$ of charge $i$ processed on equipment $k$
\[ ST_{ij} \] processing start time of \( O_{ij} \)

\[ ST_{jk} \] the work station \( j \)'s processing start time of charge \( i \) processed on equipment \( k \)

\( T_{now} \) current time;

\[ U = \{ j_1, j_2, \ldots, j_i, \ldots \} \] charging plan set, \( j_i \) is charging plan \( i \)

\[ ut_{j_j+1} \] the standard transportation time between work station \( j \) and work station \( j + 1 \) of charge \( i \)

\( v \) crane speed, it is a constant

\( y_{ijk} \) if the work station \( j \) of charge \( i \) processed on equipment \( k \), \( y_{ijk} = 1 \), else \( y_{ijk} = 0 \)

\( \delta \) the shortest distance between two adjacent cranes

\( \Omega \) the set of waiting continuous casting charge \( i \in \Omega \), \(|\Omega|\) is the total amount of continuous casting charge

\( \Omega^* \) the set of no-producing ingot casting charge

\(|\Omega^*|\) the total amount of ingot casting charge

References


