RESCHEDULING OPTIMIZATION OF STEELMAKING-CONTINUOUS CASTING PROCESS BASED ON THE LAGRANGIAN HEURISTIC ALGORITHM

LIANGLIANG SUN*, FANGJUN LUAN AND YU YING

Department of Information and Control Engineering, Shenyang Jianzhu University
No. 9, Hunnan East Road, Hunnan New District
Shenyang City, Liaoning 110168, China

KUN MAO

Department of Information Science and Engineering, Northeastern University
NO. 3-11, Wenhua Road, Heping District
Shenyang City, Liaoning 110004, China

(Communicated by Adil Bagirov)

ABSTRACT. This study investigates a challenging problem of rescheduling a hybrid flow shop in the steelmaking-continuous casting (SCC) process, which is a major bottleneck in the production of iron and steel. In consideration of uncertain disturbance during SCC process, we develop a time-indexed formulation to model the SCC rescheduling problem. The performances of the rescheduling problem consider not only the efficiency measure, which includes the total weighted completion time and the total waiting time, but also the stability measure, which refers to the difference in the number of operations processed on different machines for the different stage in the original schedule and revised schedule. With these objectives, this study develops a Lagrangian heuristic algorithm to solve the SCC rescheduling problem. The algorithm could provide a realizable termination criterion without having information about the problem, such as the distance between the initial iterative point and the optimal point. This study relaxes machine capacity constraints to decompose the relaxed problem into charge-level subproblems that can be solved using a polynomial dynamic programming algorithm. A heuristic based on the solution of the relaxed problem is presented for obtaining a feasible reschedule. An improved efficient subgradient algorithm is introduced for solving Lagrangian dual problems. Numerical results for different events and problem scales show that the proposed approach can generate high-quality reschedules within acceptable computational times.

1. Introduction. The scheduling of steelmaking-continuous casting (SCC) process is a hybrid flow job shop problem. There are three major processing stages which includes steelmaking, refining, and casting. Since the SCC process is a complicated technological process that also involves a high-temperature and high-weight material

2010 Mathematics Subject Classification. Primary: 58F15, 58F17; Secondary: 53C35.

Key words and phrases. Rescheduling, steelmaking-continuous casting, subgradient algorithm, Lagrangian heuristic algorithm, dynamic programming.

The research is financially sponsored by the National Natural Science Foundation Committee of China (Subject Numbers: 61503259), Hanyu Plan of Shenyang Jianzhu University and Research Funding from the Networked Control System Key Laboratory of the Chinese Academy of Sciences.

* Corresponding author: Liangliang Sun.

1431
flow, near optimal scheduling plays an important role in improving the production efficiency of the entire system [20, 33]. However, the SCC process operates in dynamic environments subject to different unforeseen or random disturbances such as machine failures (MFs) and processing time variations (PTVs). These disturbances not only interrupt the SCC operation but also render the original schedule obsolete. In light of these unexpected changes, the importance of rescheduling is comparable to that of scheduling.

Extensive studies have been conducted on SCC static scheduling [1, 3, 12, 14, 27, 28, 31, 32, 39, 40], only a few studies have been conducted on SCC rescheduling. Some researchers [2, 24, 25, 35] proposed the multi-agents method to solve the optimal scheduling problems in steel plant. Tang et al. [34] proposed an improved differential evolution algorithm to solve scheduling for the steelmaking and continuous casting process based on the real production data. Literature [2, 25, 35] proposed some surveys and reviews on rescheduling in steelmaking systems. Yu and Pan [38] proposed a heuristic-based algorithm for solving the optimal rescheduling for the delay disturbance of the operation time.

The present study adopts a problem-oriented approach for solving the SCC optimal rescheduling problem. Because the special structure of the SCC scheduling problem, the Lagrangian relaxation (LR) algorithm could get a near optimal solution in an acceptable time for the SCC rescheduling by decoupling some complicated constraints such as machine capacity constraints or cast-break constraints by using Lagrangian multipliers [18]. The relaxed problem is decomposed into charge-level or cast level subproblems which are solved by the forward dynamic problems (FDP) or backward dynamic problems (BDP) [30]. Based on the optimal subgradient problems, the corresponding Lagrange dual (LD) problem is updated by subgradient optimization methods (SG) [15] or surrogate subgradient optimization methods (SSG) [30]. The multipliers are updated iteratively based on the SG or SSG. Finally, in order to avoid the confliction of the decoupled constraints, a feasible solution is obtained from the relaxed solution by the use of the different heuristic-based algorithm. The combination of SG or SSG with FDP or BDP under the LR frame has been used in different SCC scheduling problem, such as static scheduling problems [7, 15, 16, 22, 23] or dynamic scheduling problems with different objectives and under different conditions [19, 31, 37]. As for the rescheduling problems with LR algorithm, Xiong et al. [36] proposed a dynamic scheduling algorithm with constant processing times in the rescheduling problems without considering the processing time and the other disturbance factors. As for the real production, SCC rescheduling considers not only the efficiency measure but also the stability measure. The processing routes for each charge may in the different operations by the different machines. Providing that some disturbances happened, changing the routes of certain charges will be a rather cumbersome task and could results in more energy consumption.

As mentioned above, a time-indexed SCC rescheduling formulation and an improved LR algorithm is proposed for solving the SCC rescheduling problem. We establish a rescheduling model with consideration of MFs and PTVs. On the basis of the established model, we relax the machine capacity constraints and decompose the relaxed problem into charge-level subproblems with variable processing times. The charge-level subproblems are solved using a polynomial dynamic programming (DP) algorithm. The corresponding LD problem is solved using an efficient subgradient algorithm with global convergence. It is remarkable that all charges in
the SCC process have the same processing flow. This implies that SCC scheduling can be viewed as a kind of hybrid flow shop scheduling. Therefore, the proposed approach can also be applied to similar rescheduling problems. The rest of the paper is organized as follows. Section 2 describes the SCC rescheduling problem and the SCC rescheduling formulation. Section 3 presents the proposed LR approach for the scheduling problem, solution methods for the subproblems, the construction method of constraint conflict solution, and the subgradient algorithm for the LD problem. Computational experiments are described in Section 4. Finally, Section 5 concludes our study and states our future work.

2. Problem description.

2.1. Description of SCC rescheduling. Typical steelmaking process includes three main stages, which are iron making stage, steelmaking-continuous casting (SCC) stage and rolling stage. SCC is the bottleneck of the entire steel production. In the process of SCC, there are three stages steelmaking, refining and continuous casting, as shown in Figure 1. Refining stage consists of RH and LF refining position. Under the cooperation with accessorial equipment, the main equipment composes the process of SCC by means of various refining methods. Blast furnace produces liquid iron, then converters smelt liquid iron into molten steel, afterwards molten steel will be poured into the corresponding ladles according to its steel grade (The molten steel smelted in a converter is called a charge). Crane moves laterally while trolley moves vertically. By the reasonable distribution and collaborative transportation of crane and trolley, limited quantity of steel ladles are carried to refining positions to conduct single or multiple refining according to the different demands for molten steel (a charge processed at single or multiple refining positions respectively are called single or multiple refining). Refined molten steel are loaded by steel ladles, then the crane and trolley convey the steel ladles to casters and the molten steel will finally be casted into slabs (the set of charges that casted continuously in the same caster is called a cast).

Since the practical production environment is complicated, it is difficult to prevent disruptions from affecting the production operation. These disruptions break the original production rhythm, requiring rescheduling of operations. All disruptions are eventually reflected on two events: MFs and PTVs. There are three statuses for an operation while a PTV event has occurred: finished (F), in process (P), or unstarted (U). Except the above three statuses, the possible status of an operation in the MF event is the canceled status (C), which means that the operation cannot be processed because of the MF. If the status of an operation of a charge is C, the statuses of the remaining unstarted operations in this charge are also C. Only those operations with status U can be rescheduled, and the corresponding

---

**Figure 1. Steelmaking-continuous casting process.**
charges are termed unfinished charges in the rescheduling problem. Additionally, the available time of a machine is one of three time instants: the time instant at which an event occurred, the completion time of its operation with status P, and the completion time of the maintenance caused by its failures.

Unlike the constant processing times in static scheduling, the processing times of charges in rescheduling can be adjustable within a certain scope in response to unforeseen disruptions. Nevertheless, owing to such disruptions, two adjacent operations of the same cast on the caster cannot be processed consecutively, which implies that the cast is broken. In real production, the schedulers must keep the cast-break time as small as possible in order to ensure the cast machine worked well. In addition, to reduce the temperature lost and the work-in-process inventory level, the due date for the last charge in each cast and the sojourn time of each charge should be as small as possible. This is also the objective of static scheduling. Therefore, the objective of the revised schedule includes the efficiency measure, which includes the total weighted completion time, sojourn time, and cast-break loss. On the other hand, the changing of a charge from one machine to another is rather cumbersome albeit with the near distance due to the heavy material and high temperature process. This requires the amount of change between the original schedule and the revised schedule to be as small as possible. Hence the stability measure is set as one of the performance index for the revised schedule of SCC.

2.2. Mathematical model for the reschedule of SCC. Based on the previous introduction and analysis, a time-indexed mathematical formulation is proposed for the rescheduling of SCC while considering the SCC production disturbances. In this mathematical formulation, \( M_j \) is the number of machines available in the \( j \)-th stage in a certain time. \( \Omega \) is the set of all unfinished charges in the cast on the \( k \)-th machine in the continuous casting stage and \( k = 1, 2, ..., M_s \), and \( s = 1, 2, ..., S \) is the number of stage. Here, \( \Omega_1 \cup \Omega_2 \cup ... \cup \Omega_{M_s} = \Omega \), and \( \Omega_{k_1} \cap \Omega_{k_2} = \emptyset \) for any \( k_1, k_2 \in 1, 2, ..., M_s \) and \( k_1 \neq k_2 \). In each cast, there is an ideal due time for the last charge. Here, \( b(k) \) is set as the index of the last charge in the \( k \)-th cast and \( b(k) = \sum_{k=0}^{k} |\Omega_k| \). \( T \) is the time horizon and \( t \in 1, 2, ..., T \). \( O_i \) is the set of index of the started operations of charge \( i \), where started operations refer to those that are finished and in process when an event occurs. The connection between the “started operations” and the statuses of an operation is shown in Figure 2.

In the real steelmaking plant, the processing time for each machine is in the interval \((P_{L}^{U}j, P_{L}^{L}j, P_{L}^{L}j)\) is the upper bound of processing time for the charge \( i \) in the stage \( j \) and \( P_{L}^{L}j \) is the lower bound of processing time for the charge \( i \) in the stage \( j \). The transportation time is the same as the processing time for each charge, it is in the interval \((T_{L}^{U}j, T_{L}^{L}j, T_{L}^{L}j)\) is the upper bound of transportation time for the charge \( i \) and \( T_{L}^{L}j \) is the lower bound of transportation time for the charge \( i \). \( W_{1} \) is the due date penalty coefficient for each charge, \( W_{2} \) is the sojourn time penalty coefficient for each charge, \( W_{3} \) is the cast-break penalty coefficient for each charge in the same cast in the process of casting. \( W_{4} \) is the penalty coefficient for the number of processed charges on different machines in the initial and revised schedules. During the process of reschedule, the available time of the \( k \)-th machine in the stage \( j \) is \( A_{j,k} \). In the initial schedule, The processing time including the PTV of charge \( i \) in the stage \( j \) is \( p_{i,j} \); the completion time including the PTV of the \( j \)-th operation of charge \( i \) is \( c_{i,j} \); if the beginning time of charge \( i \) on the \( k \)-th machine in the \( j \)-th operation is \( t \), \( g_{i,j,k,t} \) is equal to 1, otherwise \( g_{i,j,k,t} \) is equal to
0. For this problem, there are five kinds of decision variables. $c_{i,j}$ is the completion time of charge $i$ in stage $j$. $p_{i,j}$ is the processing time of charge $i$ in stage $j$. $t_{i,j}$ is the transportation time of charge $i$ in stage $j$. $y_{i,j,k,t}$ is equal to 1 if charge $i$ is processing on the $k$-th machine in the $j$-th operation at time $t$, otherwise $y_{i,j,k,t}$ is equal to 0. $x_{i,j,t}$ is equal to 1 if charge $i$ is processing in the $j$-th operation at time $t$, otherwise $x_{i,j,t}$ is equal to 0.

2.2.1. Objective function. In the real steelmaking plant, there are four performance indexes to evaluate the quality of the revised schedule which includes the due date for each cast; the sojourn time for the two adjacent charges; the cast-break for the charges in the same cast and the stability measure, as shown in Figure 3. Based on the four penalty coefficients for the four performance indexes, the objective function of the revised schedule is,

$$\begin{align*}
\min G &= F_1 + F_2 + F_3 + F_4. \\
(1)
\end{align*}$$

For each cast, there is an ideal due date. The first performance index $F_1$ is to keep the completion time of the last charge in each cast to be closer to its due date.

$$F_1 = W_1 \sum_{k=1}^{M_s} \sum_{i=b(k-1)+1}^{b(k)} |c_{i,S} - d_i|, \quad (2)$$

The second performance index $F_2$ is to reduce the waiting time among all the adjacent charges.

$$F_2 = W_2 \sum_{i=1}^{|\Omega|} (c_{i,S} - c_{i,O_i+1} + p_{i,O_i+1}), \quad (3)$$

The third performance index $F_3$ is to reduce the time gap between the adjacent charges in the same cast.

$$F_3 = W_3 \sum_{k=1}^{M_s} \sum_{i=b(k-1)+1}^{b(k)} (c_{i+1,S} - c_{i,S} - p_{i+1,S}), \quad (4)$$
The forth performance $F_4$ is the function of the stability measure for the revised schedule. In order to linear the function, two new sets of variables are introduced,

$$F_4 = W_4 \sum_{i \in \Omega}^T \sum_{t = 1}^{S-1} \sum_{j = 1}^{M_j} \sum_{k = 1}^1 \left[ \max(y_{i,j,k,t} - \bar{y}_{i,j,k,t}, 0) + \max(\bar{y}_{i,j,k,t} - y_{i,j,k,t}, 0) \right], \quad (5)$$

$$y_{i,j,k,t}^1 = \max(\bar{y}_{i,j,k,t} - y_{i,j,k,t}, 0), \quad (6)$$

$$y_{i,j,k,t}^2 = \max(y_{i,j,k,t} - \bar{y}_{i,j,k,t}, 0). \quad (7)$$

There are three cases as follows:

**Case 1.** if $y_{i,j,k,t} > \bar{y}_{i,j,k,t}$, then $y_{i,j,k,t}^1 - y_{i,j,k,t}^2 = y_{i,j,k,t} - \bar{y}_{i,j,k,t} - 0 = y_{i,j,k,t} - \bar{y}_{i,j,k,t}$;

**Case 2.** if $y_{i,j,k,t} < \bar{y}_{i,j,k,t}$, then $y_{i,j,k,t}^1 - y_{i,j,k,t}^2 = 0 - \bar{y}_{i,j,k,t} + y_{i,j,k,t} - y_{i,j,k,t} - \bar{y}_{i,j,k,t}$;

**Case 3.** if $y_{i,j,k,t} = \bar{y}_{i,j,k,t}$, then $y_{i,j,k,t}^1 - y_{i,j,k,t}^2 = 0$.

Therefore, we can get,

$$y_{i,j,k,t}^1 + y_{i,j,k,t}^2 = y_{i,j,k,t} - \bar{y}_{i,j,k,t}, i \in \Omega, 1 \leq j < S, 1 \leq k \leq M_j, 1 \leq t \leq T. \quad (8)$$

So, the forth performance index mathematical formulation could be changed into,

$$F_4 = W_4 \sum_{i \in \Omega}^T \sum_{t = 1}^{S-1} \sum_{j = 1}^{M_j} \sum_{k = 1}^1 (y_{i,j,k,t}^1 + y_{i,j,k,t}^2). \quad (9)$$

**Figure 3.** Illustration of the four performance indexes for the revised scheduling of SCC.
2.2.2. **Constraints.** Except the four performances indexes, the revised schedule of SCC should also meet the following constraints.

**Completion time constraints.** These constraints indicate that each charge in the $g$-th operation should be processed continuously, the relationship between the completion time and the beginning time is,

$$c_{i,j} = \sum_{t=1}^{T-p_{ij}+1} (t-1)x_{i,j,t} + p_{ij}, 1 \leq j < S, 1 \leq j < S. \quad (10)$$

**Processing time constraints.** These constraints indicate that each charge could be processed only in an operation at any time,

$$\sum_{t=1}^{T-p_{ij}+1} x_{i,j,t} = 1, i \in \Omega, 1 \leq j < S, 1 \leq j < S. \quad (11)$$

**Uniqueness of processing equipment constraints.** These constraints indicate that each charge can be processed on one and only one machine in any operation,

$$x_{i,j,t} = M_j \sum_{k=1}^{M_i} y_{i,j,k,t}, i \in \Omega, 1 \leq j < S, 1 \leq j < S, 1 \leq t \leq T. \quad (12)$$

**Processing precedence constraints.** These constraints indicate that for the two consecutive processed charges, the beginning time of the later charge cannot be earlier than the completion time of the previous processed charge,

$$c_{i,j} - c_{i,j+1} + T_{j,j+1} + p_{i,j+1} \leq 0, 1 \leq j < S, 1 \leq j < S. \quad (13)$$

**Machine capacity constraints.** These constraints indicate that the number of charges simultaneously processed at time period $t$ in the operation $g$ cannot be greater than the total number of units available.

$$\sum_{\tau=max(t-P_{ij}+1,1)}^{T} x_{i,j,\tau} \leq M_j \sum_{k=1}^{M_i} \delta(A_{j,k}-t), 1 \leq j < S, 1 \leq t \leq T. \quad (14)$$

**Cast constraints.** These constraints indicate that the relationship of all charges in the same cast, each charge in the same cast should also meet the processing precedence constraints. For the charges in the same cast, the beginning time of the later charge cannot be earlier than the completion time of the previous processed charge.

$$c_{i,S} + p_{i+1,S} - c_{i+1,S} \leq 0, i, i+1 \in \Omega_k, 1 \leq k \leq M_S. \quad (15)$$

**Beginning time constraints.** These constraints indicate that the beginning time of the first charge in each cast cannot be earlier than the available time of the machine in the casting stage.

$$c_{i,S} - p_{i,S} + 1 \geq A_{S,k}, i = b(k-1) + 1, 1 \leq k \leq M_S. \quad (16)$$

**Other constraints.** The related value range of the variables are as follows,

$$c_{i,j} \geq 0, i \in \Omega, 1 \leq j \leq S. \quad (17)$$

$$x_{i,j,t} \in \{0, 1\}, \delta_{i,j,t} \in \{0, 1\}, i \in \Omega, 1 \leq j < S, 1 \leq t \leq T. \quad (18)$$

$$c_{i,j} = \bar{c}_{i,j}, p_{ij} = \bar{p}_{ij}, j \in O_i, i \in \Omega. \quad (19)$$

$$P_{i,j} \leq p_{ij} \leq P_{i,j}, i \in \Omega, 1 \leq j \leq S. \quad (20)$$

$$T_{i,j} \leq t_{ij} \leq T_{i,j}, i \in \Omega, 1 \leq j \leq S. \quad (21)$$
with nonnegative multipliers

Solution methodology. Based on the introduction of the couple constraints mathematical formulation in Section 2, the charge-level, cast-level and the machine-level splitting policy could be got to optimize the revised schedule. However, if the algorithm adopts machine-level splitting policy, the subproblems of the relaxed problem are parallel machine scheduling problems, which are NP-hard \[7\]. This implies that the relaxed problem cannot be solved exactly within a reasonable computational time; if the algorithm adopts cast-level splitting policy, the subproblems of the relaxed problem are also time consuming compared with charge-level splitting policy. Hence, the paper adopts the charge-level splitting policy based on the constraints (14) machine capacity constraints and constraints (15) cast constraints.

3.1. Lagrangian relaxation. By relaxing the couple constraints (14) and (15) with nonnegative multipliers $\mu_{1,j,t}$ and $\mu_{2,i}$, we can get the following LR problem:

\[(LR)L(\mu) = \min(G + G_1 + G_2),\]

where $\mu$ is a vector of Lagrangian multipliers $\{\mu_{1,j,t}, \mu_{2,i}\}$, $x$ is a vector of variables $\{x_{i,j,t}, y_{i,j,k,t}, p_i;j, y_{i,j,k,t}^1, y_{i,j,k,t}^2\}$, $X$ is the feasible region corresponding to the other constraints, $G$ has been defined previously, we can get,

\[G_1 = \sum_{j=1}^{S-1} \sum_{t=\max\{t-P_{i,j}+1\}}^{t} \sum_{k=1}^{M_i} \mu_{1,j,t} (x_{i,j,t} - \delta(A_{j,k} - t)).\]

\[G_2 = \sum_{k=1}^{M_S} \sum_{i=b(k-1)+1}^{b(k)-1} \mu_{2,i} (c_iS + p_{i+1,S} - c_{i+1,S}).\]

Similarly, the corresponding LD problem is,

\[(LD)L(\mu) = \min_{x \in X} (G + G_1 + G_2), \mu_{1,j,t} \geq 0, \mu_{2,i} \geq 0, 1 \leq j < S, 1 \leq t \leq T, i \in \Omega.\]

For a given multiplier $\mu$, the LR problem can be decomposed into $|\Omega|$ charge-level subproblems,

\[L(\mu) = \sum_{i \in \Omega} L_i(\mu_{1,j,t}, \mu_{2,i}).\]

The subproblem $L_i(\mu)$ is presented as follows,

\[L_i(\mu) = \min_{j=|O_i|+1}^{S} G_{i,j}(\mu_{1,j,t}, \mu_{2,i}, x).\]

Considering the other constraints, where,

\[w_{ij} = \begin{cases} W_1 + W_2 - W_3 + \mu_{2,i}, & i = b(k-1) + 1, 1 \leq k \leq M_S, j = S; \\
W_1 + W_2 + W_3, & i = b(k), 1 \leq k \leq M_S, j = S; \\
W_1 + W_2 + \mu_{2,i} - \mu_{2,i-1}, & b(k-1) + 1 < i < b(k), 1 \leq k \leq M_S, j = S; \\
-W_2 & i \in \Omega, j = |O_i| + 1. \end{cases}\]
\[ G_{t,j}^1 = \sum_{i=1}^{T-p_{i,j}+1} \sum_{\tau=t}^{T} x_{i,j,\tau} \mu_{1,j,\tau}, 1 \leq j < S, \]

\[ G_{t,j}^2 = \sum_{i=1}^{T} M_j \sum_{t=1}^{T} \left( y_{i,j,k,t}^1 + y_{i,j,k,t}^2 \right), 1 \leq j < S, \]

\[ G_{t,j}^3 = \begin{cases} 
(\mu_{2,i-1} - W_j) p_{i,j}, & b(k - 1) + 1 \leq i \leq b(k), 1 \leq k \leq M_S, j = S; \\
W_2 p_{i,j}, & i \in \Omega, j = |O_i| + 1; \\
- \sum_{i=1}^{T} M_i \mu_{j,t} \delta(A_{j,k} - t), & i \in \Omega, 1 \leq j < S.
\end{cases} \]

\[ (31) \]

\[ (32) \]

\[ (33) \]

3.2. Solution of LR problem. After the nonnegative multipliers \( \mu_{1,j,t} \) and \( \mu_{2,i} \) are introduced, the LR problem is split into \(|\Omega| \) charge-level subproblems. In order to solve each charge-level subproblem, the dynamic programming (DP) algorithm is introduced while considering the processing precedence constraints. Before presenting the DP algorithm, we introduce some necessary notations as follows. For a given \( \{\mu_{1,j,t}, \mu_{2,i}\} \), let \( f_{i,j}(\mu_{1,j,t}, \mu_{2,i}, t, P_{i,j}) \) be the cost for the beginning the \( j \)-th operation of charge \( i \) at time period \( t - p_{i,j} + 1 \) and completing this operation at time period \( t \) such that,

\[ f_{i,j}(\mu_{1,j,t}, \mu_{2,i}, t, P_{i,j}) = \begin{cases} 
G_{t,j}(\mu_{1,j,t}, \mu_{2,i}, x)|_{c_{i,j}=t, p_{i,j}=p_{i,j}}, & c_{i,j}^L \leq t \leq c_{i,j}^U; \\
+\infty, & otherwise.
\end{cases} \]

\[ (34) \]

Where \( c_{i,j}^L = c_{i,j}, c_{i,j}^L = c_{i,j}^L - T_{j-1,j}, (|O_i| < j \leq S), c_{i,S}^L = T, \) and \( c_{i,j}^U = c_{i,j+1} - T_{j-1,j} - T_{j,j+1} \). Let \( F_{i,j}^{\mu_{1,j,t},\mu_{2,i}}(|O_i| < j \leq S) \) be the optimal criterion value for the \( j \)-th operation of charge \( i \) completed no later than time \( t \).

Initial condition,

\[ F_{i,j}(\mu_{1,j,t}, \mu_{2,i}, t) = +\infty, \]

for \( t < c_{i,j}^L \) or \( c > c_{i,j}^U, i \in \Omega, \) and \(|O_i| < j \leq S\).

Recursive equation,

\[ F_{i,j}(\mu_{1,j,t}, \mu_{2,i}, t) = \min_{p_{i,j} \leq P_{i,j} \leq P_{i,j}^U} \min_{T_{i,j}^L \leq T_{i,j} \leq T_{i,j}^U} \{ f_{i,j}(\mu_{1,j,t}, \mu_{2,i}, t, P_{i,j}, F_{i,j}(\mu_{1,j,t}, \mu_{2,i}, t - 1)) \}, \]

\[ (35) \]

\[ (36) \]

for \( c_{i,j}^L \leq t \leq c_{i,j}^U, i \in \Omega. \)

Therefore, \( L(\mu) \) is given as \( \sum_{i \in \Omega} F_{i,S}(\mu_{1,j,t}, \mu_{2,i}, t - 1, c_{i,j}^U) \). The optimal charge-level subproblem penalty cost is obtained as the minimal cumulative cost from the steelmaking stage to the casting stage. Finally, the optimal solution for \( LR_i \) can be got by the previous stage. The overall computational complexity of the DP algorithm is \( O(TP|\Omega|) \), where \( T = \max_{i \in \Omega, 1 \leq j \leq S} (T_{i,j}^U - T_{i,j}^L) \) and \( P = \max_{i \in \Omega, 1 \leq j \leq S} (P_{i,j}^U - P_{i,j}^L). \)
3.3. Obtain a feasible revised schedule to the original problem. The optimal solution to the relaxed revised schedule is generally infeasible for the original problem because the couple constraints processing precedence and the machine capacity have been relaxed. In this paper, a two-stage heuristic algorithm is proposed to construct a feasible solution based on the optimal solution to the relaxed problem. The first stage is adjusted to ensure that the processing precedence constraints are satisfied without considering machine capacity constraints. The second stage is adjusted to ensure that the machine confliction are resolved to get a feasible optimal revised schedule.

**Step 1. Initialization.** Based on the initial scheduling for SCC, the value of the available time of each machine \(\overline{A}_{j,k}(1 \leq j < S, 1 \leq k \leq M_j)\) and the machine selection for the charge \(i\) in the stage \(j\), \(m_{i,j}(1 \leq j < S, 1 \leq m_{i,j} \leq M_j, i \in \Omega_j)\) are given. The multipliers \(\mu_{1,j,t}, \mu_{2,i}\), processing time \(p_{i,j}\) and transportation time \(t_{i,j}\) are substitute. The \(LR\) problem becomes a linear programming problem.

**Step 2. Conflict resolution.** While considering the charge sequence in the same stage, set \(a_{j,k}\) and \(k^* = \arg \min_{1 \leq k \leq M_j} a_{j,k}\). Charge \(i\) is arranged into the \(k^*\)-th machine in stage \(j\) and \(\overline{A}_{j,k^*} = \overline{A}_{j,k} + p_{i,j}\). Repeat the process until all the charges is arranged.

**Step 3. Machine adjustment.** Let \(T_j(1 \leq j < S)\) be an ascending-order list of all elements in \(\overline{t}_{i,j}\). Let \(i\) be the first unscheduled charge in the list \(T_j(1 \leq j < S)\). Set \(a_{j,k}\) and \(k^* = \arg \min_{1 \leq k \leq M_j} a_{j,k}\). Charge \(i\) is arranged into the \(k^*\)-th machine in stage \(j\) and \(\overline{A}_{j,k^*} = \overline{A}_{j,k} + p_{i,j}\). Repeat the process until all the charges is arranged.

**Step 4. Optimal feasible solution.** By using the linear programming, the feasible completion time \(c_{i,j}\), processing time \(p_{i,j}\) and transportation time \(t_{i,j}\) of each operation are got and defined as a feasible solution for the revised schedule for SCC.

3.4. Traditional subgradient methods. For the optimization of revised schedule for SCC, the traditional subgradient algorithm for the dual problem is given by,

\[
\{\mu_{1,j,t}, \mu_{2,i}\}_{m+1} = P_{\phi}(\{\mu_{1,j,t}, \mu_{2,i}\}_m - \alpha_m g(\{\mu_{1,j,t}, \mu_{2,i}\}_m)), m = 0, 1, 2, \ldots, \tag{38}
\]

here, \(g(\{\mu_{1,j,t}, \mu_{2,i}\}_m)\) is the subgradient of the original problem \(F(\{\mu_{1,j,t}, \mu_{2,i}\}_m)\), \(F(\{\mu_{1,j,t}, \mu_{2,i}\}_m) = -L(\{\mu_{1,j,t}, \mu_{2,i}\}_m), \Phi = \{\mu_{1,j,t}, \mu_{2,i}\}_{\mu_{1,j,t} \geq 0, \mu_{2,i} \geq 0}\), \(\mu_{1,j,t}\) and \(\mu_{2,i}\) are the related multipliers with coupling constraints (14) machine capacity constraints and (15) cast constraints.

Although the subgradient algorithm has advantages of simplicity and low computational complexity, it has two drawbacks. One is the slow convergence caused by two kinds of zigzagging phenomena \([11]\), and the other is the strict convergence condition that the optimum dual value should be known a priori \([4]\). Some studies \([6, 11, 13]\) have proposed various search-direction methods to improve the efficiency of the subgradient algorithm, whereas some others \([5, 10, 17, 29]\) have proposed various step-size methods to replace the strict convergence condition. Mao et al. \([19]\) integrated these two types of methods into a new subgradient algorithm, called the deflected-conditional subgradient level algorithm (DCSLA), which not only improves the efficiency of the subgradient algorithm but also guarantees global
Before presenting the DCSLA, we need to provide the definition of a conditional subgradient, which generalized the standard subgradient. The conditional subdifferential is given as,

\[ \partial^c F(\mu) = \{ g \in \mathbb{R}^n | F(\mu_1) \geq F(\mu) + g^T(\mu_1 - \mu), \forall \mu_1 \in \Phi, \mu \in \Phi \}, \]

whose elements are referred to as conditional subgradients. \( \partial^c F(\mu_m) \geq \partial F(\mu_m) \) for all \( \mu \in \Phi \).

The DCSA algorithm [6] is introduced as follows,

\[ d_m = \hat{g}_m + \beta_m d_{m-1}, \]

\[ \mu_{m+1} = P_{\Phi}(\mu_m - \alpha_m d_m), m = 0, 1, 2, ..., \]

Here, \( d_{-1} = 0 \), \( \hat{g}_m \) is the conditional subgradient of \( F(\mu_m) \) at \( \mu \) and \( \beta \geq 0 \) is termed a deflection parameter. Based on the characterization of the conditional subdifferential [9] and the projection method [13], the conditional subgradient \( \hat{g} \) can be determined as follows,

\[ \hat{g} = \begin{cases} 0, & \text{if } g^i \geq 0, \text{ and } \mu_i = 0, \\ g^i, & \text{others.} \end{cases} \]

Where \( g_i \) is the i-th element of \( g \), \( \tau_m = 1 \). and \( g \) is the subgradient of \( F(\mu_m) \) at \( \mu \). The deflection parameter \( \beta_m \) can be determined as follows [4].

\[ \beta_m = \begin{cases} -\tau_m g^T_m d_{m-1}/\|d_{m-1}\|^2, & g^T_m d_{m-1} < 0, \\ 0, & g^T_m d_{m-1} \geq 0, \end{cases} \]

3.5. DCSLA for the revised schedule of SCC. To overcome the difficulty of an unrealizable convergence condition of the conventional subgradient algorithm, some works [5, 10] have proposed an adaptive estimation strategy for the optimum, termed Brannlunds level control strategy, which can guarantee global convergence and overcome the abovementioned difficulty. Another study [17] has used this method to solve a static hybrid flow shop scheduling problem. The present study incorporates Brannlunds level control strategy into the DCSA, resulting in the DCSLA, which is applied to the SCC rescheduling.

**Step 1. Definition.** Let \( P(\{\mu_{1,j,t}, \mu_{2,i}\}) \) be a primal function value of \( \{\mu_{1,j,t}, \mu_{2,i}\} \), which is derived using the heuristic presented in Subsection 3.3. Let \( F(\{\mu_{1,j,t}, \mu_{2,i}\}) = -L(\{\mu_{1,j,t}, \mu_{2,i}\}), g(\{\mu_{1,j,t}, \mu_{2,i}\}) \) be the subgradient of \( F(\{\mu_{1,j,t}, \mu_{2,i}\}) \), \( \hat{g}(\{\mu_{1,j,t}, \mu_{2,i}\}) \) be the conditional subgradient of \( F(\{\mu_{1,j,t}, \mu_{2,i}\}) \). We denote \( g_m = g(\{\mu_{1,j,t}, \mu_{2,i}\}), \hat{g}_m = \hat{g}(\{\mu_{1,j,t}, \mu_{2,i}\}), \) and \( d_m = d(\{\mu_{1,j,t}, \mu_{2,i}\}) \).

**Step 2. Parameters setting.** Based on the real production of revised schedule of SCC, the initialization of the related parameter are as follows.

1) The parameter \( \epsilon_1 = 1e - 3 \) denotes the proportion of the decline of the dual function. Because the scheduling of SCC is the NP hard problem and a duality gap exists. Generally, the duality gap is more than 1%, observed on the basis of experimental regularity. Reducing the duality gap to less than 0.1% would be favorable for the optimization of SCC scheduling. Therefore, we set this parameter as \( \epsilon_1 = 1e - 3 \).
2) The parameter $\epsilon_2 = 1e - 3$ denotes the 2-norm value of the subgradient. Because the dual problem is a nonsmooth continuous function, providing that the 2-norm value of the subgradient is small enough, we could get the conclusion that the current iterative value is close enough to the optimal value.

3) The parameter $\delta_1 = (P(\{\mu_{1,j,t}, \mu_{2,i}\})_0 + P(\{\mu_{1,j,t}, \mu_{2,i}\})_0)/5$ is an estimated value of $F(\{\mu_{1,j,t}, \mu_{2,i}\}) - F^*$. Because $P(\{\mu_{1,j,t}, \mu_{2,i}\})_0$ is the feasible solution of the original problem, its value is always greater than the optimal value of the dual problem. $F(\{\mu_{1,j,t}, \mu_{2,i}\})_0 = -L(\{\mu_{1,j,t}, \mu_{2,i}\})_0$ is one of the values of the dual problem; thus, $\delta_1 = (P(\{\mu_{1,j,t}, \mu_{2,i}\})_0 + P(\{\mu_{1,j,t}, \mu_{2,i}\})_0)/5$ is an estimated value of $F(\{\mu_{1,j,t}, \mu_{2,i}\}) - F^*$.

4) The parameter $\sigma_{\max} = (P(\{\mu_{1,j,t}, \mu_{2,i}\})_0 + F(\{\mu_{1,j,t}, \mu_{2,i}\})_0)/g(\{\mu_{1,j,t}, \mu_{2,i}\}_0)$ is an estimated value of $\|\{\mu_{1,j,t}, \mu_{2,i}\}_0 - \mu^*\|_2$. The value of $P(\{\mu_{1,j,t}, \mu_{2,i}\})_0 + F(\{\mu_{1,j,t}, \mu_{2,i}\})_0$ can be easily obtained by the LR algorithm. We calculate the lower bound of $F^*$ according to the objective function of the original problem. Thus, we can obtain an estimated value of $F(\{\mu_{1,j,t}, \mu_{2,i}\}) - F^*$. Finally, we can obtain a rough estimated value of $\|\{\mu_{1,j,t}, \mu_{2,i}\}_0 - \mu^*\|_2$ on the basis of the parameter $\sigma_{\max} = (P(\{\mu_{1,j,t}, \mu_{2,i}\})_0 + F(\{\mu_{1,j,t}, \mu_{2,i}\})_0)/g(\{\mu_{1,j,t}, \mu_{2,i}\}_0)$.

5) The parameter $\{\mu_{1,j,t}, \mu_{2,i}\}_0 = 0$ is the default value for the initial Lagrange multipliers and $\{\mu_{1,j,t}, \mu_{2,i}\}_0 = \mu_{\text{best}}$. $F_{\text{best}} = P(\{\mu_{1,j,t}, \mu_{2,i}\}_0), h(s) = 1/(s + 1)$ and $F_{\text{best}}^{-1} = \infty$. The other parameters of $t, N, \beta, r, m, l, s, d_{-1}, n, M[l]$ are set as the value of 0.8, 4, 0.8, 0, 0, 0, 0, 0, 0, 0, 1 separately, where $M[l]$ is the iteration number when the l-th update of $F_{\text{lev}}$ occurs.

**Step 3. Function selection.** During the iteration, the algorithm need select the proper value for $F_{\text{best}}^m$ and $P_{\text{best}}$. If $F(\{\mu_{1,j,t}, \mu_{2,i}\}_0) < F_{\text{best}}^{m-1}$, then $F_{\text{best}}^m = F(\{\mu_{1,j,t}, \mu_{2,i}\}_0)$ and $\{\mu_{1,j,t}, \mu_{2,i}\}_{\text{best}} = \{\mu_{1,j,t}, \mu_{2,i}\}_0$; else $F_{\text{best}}^m = F_{\text{best}}^m$. At the same time, if $P_{\text{best}} < P(\{\mu_{1,j,t}, \mu_{2,i}\}_0)$, then $P_{\text{best}} = P(\{\mu_{1,j,t}, \mu_{2,i}\}_0)$.

**Step 4. Descent condition.** If $F(\{\mu_{1,j,t}, \mu_{2,i}\}_k) \leq F_{\text{best}}^m - 0.5\delta_l$, set $M[l+1] = m, l = l + 1$, and $\delta_{l+1} = \delta_l$.

**Step 5. Overestimation adjustment.** In case of the small over estimation. Set $r = r + 1$, D[r] = $F(\{\mu_{1,j,t}, \mu_{2,i}\}_m)$. If $r > N$ and $\sum_{n=1}^{W-2} |2(D[n + 2] - D[n])| < \epsilon_2$, then $m_1 = m + [h(s)\sigma_{\max} - \sigma_m + 1]/\|d_m\| + 1, \sigma_{m_2} = h(s)\sigma_{\max} + 1, F_{\text{best}}^m = F_{\text{best}}^m, d_{m_1} = d_m, r = 0, \mu_{1,j,t}, \mu_{2,i}_m = \mu_{1,j,t}, \mu_{2,i}_m$, and $m = m_1$. In case of the large estimation. Set $l = l + 1$ and $s = s + 1$. If $\sigma_m > h(s)\sigma_{\max}$, then $M[l + 1] = m, \mu_{1,j,t}, \mu_{2,i}_m = \mu_{1,j,t}, \mu_{2,i}_m$, $\sigma_{l+1} = 0, d_{m-1} = 0, \delta_{l+1} = \beta \delta_l$.

**Step 6. Lagrange multipliers update.** Let $F_{\text{lev}} = F_{\text{best}}^m$, by using $d_m = \hat{g}_m + \beta_m d_{m-1}$ and $\{\mu_{1,j,t}, \mu_{2,i}\}_{m+1} = P_{\mu}(\{\mu_{1,j,t}, \mu_{2,i}\}_m - \sigma_m d_m)$ to update the Lagrange multipliers $\{\mu_{1,j,t}, \mu_{2,i}\}$.

**Step 7. Accumulation of iteration.** Set $\sigma_m + 1 = \sigma_m + \|\sigma_m d_m\|$ and $m = m + 1$. If $|\delta_l|/F_{\text{best}}^m[L] < \epsilon_1$ or $|d_m| < \epsilon_1$, Stop; Otherwise go to step 3.

From the description of the algorithm, we can make the following conclusions.

1) The DCSLA algorithm provides a practical termination criterion without having any information about the problem, such as the distance between the initial iterative point and the optimal point. In particular, for a large calculation scale of the SCC problem, the algorithm can yield a high-quality solution with acceptable computational times.

2) The conventional optimal algorithm for SCC scheduling cannot provide an accurate evaluation of the solution. The enhanced strategies improve not only the
practicality of the LR approach but also the exactness of evaluation of the quality of the solution, and these strategies can instruct the production of SCC schedule effectively. The global convergence of the proposed algorithm can be proved by means of the previously reported methods [10, 19, 21] and the characterization of the deflected-conditional subgradient.

4. Numerical testing based on steelmaking plant data.

4.1. Test problems used for experimentation. We evaluate the Lagrangian Heuristic approach for SCC rescheduling based on real steelmaking plant data. For the actual steelmaking plant, there are four stages, the first stage is steelmaking, the last is continuous casting and the other stages are the refining stages. In each stage, there are three machines. The coefficient of completion time of each charge is $W_1 = 10 + 40(S - 1)$, the coefficient of waiting time of each charge is $W_2 = 10$, the coefficient of cast-break loss in the last stage is $W_3 = 30$ and the coefficient of the number of charges processed on different machines in the initial and revised schedules is $W_4 = 20$. The processing time in the different machines is ranged from $[36, 50]$. The transportation time between the two adjacent stages is ranged from $[3, 10]$. The maximum processing time $P(i, j)^U$ is 1.1 times $P(i, j)$, and the minimum processing time $P(i, j)^L$ is 0.9 times $P(i, j)$.

4.2. Initial scheduling of SCC. We acquired the initial schedule of SCC by means of the surrogate subgradient Lagrangian relaxation algorithm (SSLRA) [23]. In the SSLRA, a two-stage strategy for the SCC schedule is introduced after construction of the global optimal mathematical model without considering the disturbances factors. The first stage transforms the model, which is based on the relaxation of the constraints no break and no confliction, by using the LR method. The second stage proposes the LR framework to optimize the schedule of steelmaking-refining-continuous casting by the SSLRA, which can give an appropriate direction without solving of all the separated charges schedule to optimize the dual functions for separable steelmaking and continuous casting schedule problems in order to update the LR multipliers by using the appropriate step size and subgradient. Then, a near-optimal schedule solution is obtained by adjusting the duality gap between the lower bound and the upper bound by using the heuristic algorithm. Based on the initial scheduling of SCC, the related factors and the levels for the rescheduling of SCC are described as follows. First, the number of charges in each cast is the same, they are 30, 60, 90 and 120 separately. Secondly, two levels of time points are given for the SCC rescheduling, they are at 30% of the make-span (early) and at 70% of the make-span (late) separately. Third, based on the standard processing time, the length of the processing time delay is set at 0% of the standard processing time, 10% of the standard processing time, and 20% of the standard processing time separately. At last, the length of the machine breakdown time is set as 0% of the make-span, 3% of the make-span, and 6% of the make-span separately. According to the set of rescheduling point, processing time delay and the machine breakdown time, we can see that there are 16 types of events for the SCC rescheduling. We generate 10 cases randomly. Thus, there are 640 test instances.

4.3. Computational results for the optimal rescheduling. In order to evaluate the optimal solution of LR, we proposed the duality gap $g = (UB - LB)/UB \times 100\%$ as a criterion to measure the solution optimality of SCC rescheduling, where $UB$ is the upper bound derived from the modified feasible solution and $LB$ is the
lower bound obtained from the solution of the LD problem. All the test problem are implemented in C# computer programming language under the Windows 7 operating system (64-bit) environment with Intel Core i7-3667U 3.4 GHz CPU PC.

4.3.1. Case 1: Comparison between SSLRA and DCSLA for SCC scheduling. Based on the algorithm parameters and the problem instances mentioned above, without considering the disturbances of the SCC process, we use the SSLRA \[18\] and DCSLA to optimize the SCC scheduling problem under different problem sizes (where the number of casts is 2, 3, 4, and 5 and the number of charges in each cast is 5, 6, 7, and 8, respectively). Table 1 and Table 2 presents a comparison of results obtained using these two algorithms.

Table 1. The results obtained by SSLRA

| Cast vs. charge(SLLRA) | LB       | UB       | Gap (%) | Time (s) |
|------------------------|----------|----------|---------|----------|
| 2 vs.5                 | 937456   | 983672   | 4.70    | 299.5    |
| 2 vs.6                 | 1182728  | 1398892  | 15.45   | 313.3    |
| 2 vs.7                 | 1497377  | 1732913  | 13.59   | 327.8    |
| 2 vs.8                 | 1837244  | 2252386  | 18.43   | 323.7    |
| 3 vs.5                 | 1923113  | 2183725  | 11.93   | 309.4    |
| 3 vs.6                 | 2487753  | 2711245  | 8.24    | 355.2    |
| 3 vs.7                 | 3294573  | 3690245  | 10.72   | 377.2    |
| 3 vs.8                 | 3999272  | 5294742  | 24.47   | 466.1    |
| 4 vs.5                 | 3100023  | 3274848  | 5.34    | 378.5    |
| 4 vs.6                 | 4134749  | 4591234  | 9.94    | 449.2    |
| 4 vs.7                 | 5368271  | 7545422  | 28.85   | 598.4    |
| 4 vs.8                 | 6650012  | 9082765  | 26.78   | 739.6    |
| 5 vs.5                 | 4648823  | 5119374  | 9.19    | 504.7    |
| 5 vs.6                 | 6168391  | 9998116  | 38.30   | 663.2    |
| 5 vs.7                 | 8102927  | 11924753 | 32.05   | 2199.5   |
| 5 vs.8                 | 10373752 | 13583721 | 23.63   | 7824     |
| Average                | 4106654  | 7532570  | 17.60   | 1008.08  |

From these results, we find that the average values of the duality gap are 17.60% and 3.92% for the SSLRA and DCSLA, respectively, and the average calculation times for these algorithms are 1008.08 s and 3.1 s, respectively. The optimized results of the SCC schedule obtained by the DCSLA are superior to those obtained by the SSLRA.

4.3.2. Case 2: Computational results of DCSLA for SCC rescheduling. Table 3 presents the computational results of the DCSLA for different events in consideration of the test factors and levels explained in the previous section. The result in each column for each event is an average value obtained from four problem-size levels (30, 60, 90, and 120). The average duality gap of each event as obtained from these four problem-size levels is related to the following three performance indexes of SCC rescheduling.

1) EV-1: the consecutive charges in each cast should be processed continuously: no break cast.
2) EV-2: the consecutive processed charges in the same machine cannot conflict: no confliction.
Table 2. The results obtained by DCSLA

| Cast vs. charge(DCSLA) | LB      | UB      | Gap (%) | Time (s) |
|------------------------|---------|---------|---------|----------|
| 2 vs. 5                | 937456  | 954151  | 1.75    | 1.6      |
| 2 vs. 6                | 1182728 | 1294621 | 8.64    | 2.2      |
| 2 vs. 7                | 1497377 | 1519847 | 1.48    | 3.2      |
| 2 vs. 8                | 1837244 | 1997636 | 8.03    | 1.9      |
| 3 vs. 5                | 1923113 | 2003743 | 4.02    | 2        |
| 3 vs. 6                | 2487753 | 2505632 | 0.71    | 3.2      |
| 3 vs. 7                | 3294573 | 3349425 | 1.64    | 1.5      |
| 3 vs. 8                | 3999272 | 4186443 | 4.47    | 1.5      |
| 4 vs. 5                | 3100023 | 3153846 | 1.71    | 2.1      |
| 4 vs. 6                | 4134749 | 4200474 | 1.56    | 2.7      |
| 4 vs. 7                | 5368271 | 5438362 | 1.29    | 3.8      |
| 4 vs. 8                | 6650012 | 6739436 | 1.33    | 3.1      |
| 5 vs. 5                | 4648823 | 4753628 | 2.20    | 2.4      |
| 5 vs. 6                | 6168391 | 6374522 | 3.23    | 3.9      |
| 5 vs. 7                | 8102927 | 9193736 | 11.86   | 5.9      |
| 5 vs. 8                | 10373752| 11376463| 8.81    | 8.8      |
| Average                | 4106654 | 4315122 | 3.92    | 3.11     |

3) EV-3: the summation of the difference value between the actual completion time and the due date of the casts in the casters.

The smaller the duality gap, the better will be the three performance indexes. From the computational results, we can get the following conclusions.

Table 3. Computational results of DCSLA for SCC rescheduling

| ET | Events | EV-1 (s) | EV-2 (s) | EV-3 (min) | DG (%) | Time (s) | IN |
|----|--------|----------|----------|------------|--------|----------|----|
| 1  | R1-T2-M1| 0        | 0        | 22         | 12.94  | 76.63    | 162|
| 2  | R1-T3-M1| 0        | 0        | 15         | 11.85  | 66.31    | 115|
| 3  | R1-T1-M2| 0        | 0        | 16         | 13.64  | 75.92    | 141|
| 4  | R1-T2-M2| 0        | 0        | 19         | 13.09  | 58.7     | 128|
| 5  | R1-T3-M2| 0        | 0        | 21         | 11.55  | 44.76    | 103|
| 6  | R1-T1-M3| 0        | 0        | 18         | 8.68   | 121.91   | 196|
| 7  | R1-T2-M3| 0        | 0        | 11         | 12.97  | 134.73   | 187|
| 8  | R1-T3-M3| 0        | 0        | 10         | 11.32  | 83.66    | 165|
| 9  | R2-T2-M1| 0        | 0        | 19         | 7.74   | 9.32     | 63 |
| 10 | R2-T3-M1| 0        | 0        | 12         | 8.31   | 8.69     | 60 |
| 11 | R2-T1-M2| 0        | 0        | 16         | 9.22   | 12.81    | 52 |
| 12 | R2-T2-M2| 0        | 0        | 19         | 7.93   | 9.33     | 61 |
| 13 | R2-T3-M2| 0        | 0        | 22         | 8.88   | 8.89     | 68 |
| 14 | R2-T1-M3| 0        | 0        | 14         | 9.12   | 15.9     | 59 |
| 15 | R2-T2-M3| 0        | 0        | 16         | 8.45   | 7.63     | 42 |
| 16 | R2-T3-M3| 0        | 0        | 11         | 8.67   | 8.69     | 74 |
| Average | 0    | 0        | 16.31    | 10.27     | 46.49  | 104     |

(ET: Event Type, IN: Number of Iterations, DG: Duality Gap, EV: Evaluation Values)
The largest gap among all the test instances is 13.64% with a computational time of 75.92s. The distance between the optimized value and the optimal value, which lies between the lower bound and the upper bound, is small enough to satisfy the production requirements based on the evaluation of the related performance indexes under the considered different events. The computational times for SCC rescheduling are much faster than the calculation times in the case of manual operation. The results in Table 1 show that the proposed approach can get a near optimal solution within acceptable computational times.

2) The later the time of occurrence of rescheduling, the smaller are the computational times of the two methods. For example, the computational times for the events containing R1 are much larger than those for the events containing R2. If rescheduling occurs later, the number of charges requiring rescheduling is smaller and the problem size is smaller. This implies smaller computational times. Interestingly, the gaps for events containing R1 are bigger than those for events containing R2.

3) Without consideration of the rescheduling time point, the events have a significant influence on the computational times of the two methods, although they have a relatively weak influence on the gaps. For example, the computational times of $R1 - T1 - M2$, $R1 - T2 - M2$, and $R1 - T3 - M3$ are 75.92 s, 58.7 s, and 83.66 s, respectively, whereas the gaps are 13.64%, 13.09%, and 11.32%, respectively. The differences among the gaps are much smaller than those among the computational times. The reason for this is that the subproblems are solved using the DP algorithm. The processing times of each charge and the number of charges in the DP algorithm are the major factors for the differences among the gaps. On the other hand, events excluding the event of rescheduling time-point do not change the problem size and the processing sequence of the unstarted charges. This implies that these events do not change the structure of the SCC rescheduling problem and the differences in the final optimal results qualities of the proposed approach for these events would be not large.

5. Conclusion. In steelmaking-continuous casting (SCC) production, processing time variations and machine failures are the main unforeseen disturbance factors, how to make a good reschedule in an efficient computational manner is the key factor for the steelmaking production. In this paper, the rescheduling model was built based on the time-indexed mathematical formulation. An efficient Lagrangian heuristic algorithm was proposed to optimize the rescheduling problem. The industrial application is verified based on the background of the actual steelmaking-continuous casting schedule in domestic steelmaking plant. The computational results showed that the proposed approach could yield a near-optimal solution within acceptable computational times. The research will improve the reschedule method for the actual steelmaking-continuous casting process, promote the development of the reschedule theory with the separable structure characteristics and lay a foundation in the industry application for the reschedule theory.

Acknowledgments. The authors would like to thank the editor and the referees for their constructive comments and suggestions. The author also gratefully acknowledge the contributions of Prof. Tianyou Chai of Northeastern University and Prof. Peter B. Luh of University of Connecticut, Professor Hong Wang of United Kingdom of Manchester, Qianchuan Zhao of Tsinghua University and Xiaodong Xia.
from Shenyang Jianzhu University for their valuable suggestions, which have been used to improve the readability of the paper.

REFERENCES

[1] A. Atighehchian, M. Bijari and H. Tarkesh, A novel hybrid algorithm for scheduling steelmaking continuous casting production, Computers and Operations Research, 36 (2009), 2450–2461.

[2] H. Aytug, M. Lawley, K. McKay, S. Mohan and R. Uzsoy, Executing production schedules in the face of uncertainties: A review and some future directions, European Journal of Operational Research, 161 (2005), 86–110.

[3] A. Bellabdaoui and J. Teghem, A mixed-integer linear programming model for the continuous casting planning, International Journal of Production Economics, 104 (2005), 2450–2461.

[4] D. Bertsekas, Nonlinear Programming 2nd edition, Athena Scientific, Massachusetts, 1999.

[5] U. Brannlund, On Relaxation Methods for Nonsmooth Convex Optimization, Ph.D thesis, Royal Institute of Technology in Stockholm, 1993.

[6] P. Camerini, L. Fratta and F. Maffioli, On improving relaxation methods by modified gradient techniques, Mathematical Programming Study, 3 (1975), 26–34.

[7] H. Chen and P. Luh, An alternative framework to Lagrangian relaxation approach for job shop scheduling, European Journal of Operational Research, 149 (2003), 499–512.

[8] P. Cowling, D. Ouelhaad and S. Petrovic, Dynamic scheduling of steel casting and milling using multi-agents, Production Planning and Control, 15 (2004), 178–188.

[9] V. Demjanov and V. Somesova, Conditional subdifferentials of convex functions, Soviet Mathematics Doklady, 19 (1978), 1181–1185.

[10] J. Goffin and K. Kiwiel, Convergence of a simple subgradient level method, Mathematical Programming, 85 (1999), 207–211.

[11] B. Guta, Subgradient Optimization Methods in Integer Programming with an Application to a Radiation Therapy Problem, Ph.D thesis, Teknishe Universitat Kaiserlautern in Kaiserlauter, 2003.

[12] I. Harjunkoski and I. Grossmann, A decomposition approach for the scheduling of a steel plant production, Computers and Chemical Engineering, 25 (2001), 1647–1660.

[13] T. Larsson, M. Patriksson and A. Stromberg, Conditional subgradient optimization – theory and applications, European Journal of Operational Research, 88 (1996), 382–403.

[14] J. Li, X. Xiao, Q. Tang and C. Floudas, Production scheduling of a Large-scale steelmaking continuous casting process via unit-specific event-based continuous-time models: Short-term and medium-term scheduling, Industrial and Engineering Chemistry Research, 51 (2012), 7300–7319.

[15] P. Luh and D. Hoitomt, Scheduling of manufacturing systems using the Lagrangian relaxation technique, IEEE Transactions on Automatic Control, 38 (1993), 1066–1079.

[16] P. Luh, D. Hoitomt, E. Max and K. Pattipati, Scheduling generation and reconfiguration for parallel machines, IEEE Transactions on Robotics and Automation, 6 (1990), 687–696.

[17] K. Mao, Q. Pan, X. Pang and T. Chai, A novel Lagrangian relaxation approach for the hybrid flowshop scheduling problem in a steelmaking-continuous casting process, European Journal of Operational Research, 236 (2014), 51–60.

[18] K. Mao, Q. Pan, X. Pang and T. Chai, An effective Lagrangian relaxation approach for rescheduling a steelmaking-continuous casting process, Control Engineering Practice, 30 (2014), 67–77.

[19] K. Mao, Q. Pan, X. Pang, T. Chai and P. Luh, An Effective Subgradient Method for Scheduling a Steelmaking-Continuous Casting Process, IEEE Transactions on Automation Science and Engineering, 12 (2014), 1–13.

[20] H. Missbauer, W. Hauber and W. Werner Stadler, A scheduling system for the steelmaking-continuous casting process: A case study from the steelmaking industry, International Journal of Production Research, 47 (2009), 4147–4172.

[21] A. Nedic and D. Bertsekas, Incremental Subgradient Methods for Nondifferentiable Optimization, SIAM Journal on Optimization, 12 (2001), 109–138.

[22] T. Nishi, Y. Hiranaka and M. Imiguchi, Lagrangian relaxation with cut generation for hybrid flowshop scheduling problems to minimize the total weighted tardiness, Computers and Operations Research, 37 (2010), 189–198.
[23] T. Nishi, Y. Isoya Y and M. Inuiguchi, An integrated column generation and lagrangian relaxation for flowshop scheduling problems. *Proceedings of the 2009 IEEE International Conference on Systems, Man and Cybernetics*, (2009), 209–304.

[24] D. Ouelhadj, P. Cowling and S. Petrovic, Utility and stability measures for agent-based dynamic scheduling of steel continuous casting. *IEEE International Conference on Robotics and Automation*, (2003), 175–180.

[25] D. Ouelhadj and S. Petrovic, A survey of dynamic scheduling in manufacturing systems. *Journal of Scheduling*, 12 (2009), 417–431.

[26] D. Ouelhadj, S. Petrovic, P. Cowling and A. Meisels, Inter-agent cooperation and communication for agent-based robust dynamic scheduling in steel production. *Advanced Engineering Informatics*, 18 (2004), 161–172.

[27] D. Pacciarelli and M. Pranzo, Production scheduling in a steelmaking-continuous casting plant. *Computers and Chemical Engineering*, 28 (2004), 2823–2835.

[28] Q. Pan, L. Wang, K. Mao, J. Zhao and M. Zhang, An Effective Artificial Bee Colony Algorithm for a Real-World Hybrid Flowshop Problem in Steelmaking Process. *IEEE Transactions on Automation Science and Engineering*, 10 (2013), 307–322.

[29] H. Sherali, G. Choi and C. Tuncbilek, A Variable Target Value Method for Nondifferentiable Optimization. *Operation Research Letters*, 26(2000), 1–8.

[30] L. Sun, Research on the Optimal Scheduling Method for the productive Process of Steelmaking-Refining-Continuous Casting, Ph.D thesis, Northeastern University in Shenyang, 2015.

[31] L. Tang, J. Liu, A. Rong and Z. Yang, A review of planning and scheduling systems and methods for integrated steel production. *European Journal of Operational Research*, 133 (2001), 1–20.

[32] L. Tang, P. Luh, J. Liu and L. Fang, Steelmaking process scheduling using Lagrangian relaxation. *International Journal of Production Research*, 40 (2002), 55–70.

[33] L. Tang, G. Wang and Z. Chen, Integrated charge batching and casting width selection at Baosteel. *Operations Research*, 62 (2014), 772–787.

[34] L. Tang, Y. Zhao and J. Liu, An Improved Differential Evolution Algorithm for Practical Dynamic Scheduling in Steelmaking-continuous Casting Production. *IEEE Transactions on Evolutionary Computation*, 18 (2014), 209–213.

[35] G. Vieira, J. Hermann and E. Lin, Rescheduling manufacturing systems: a framework of strategies, policies and methods. *Journal of Scheduling*, 6 (2003), 36–92.

[36] R. Xiong, Y. Fan and C. Wu, A dynamic job shop scheduling method based on Lagrangian relaxation. *Tsinghua Science and Technology*, 4 (1999), 1297–1302.

[37] H. Xuan and L. Tang, Scheduling a hybrid flowshop with batch production at the last stage. *Computers and Operations Research*, 34 (2007), 2718–2733.

[38] S. Yu and Q. Pan, A Rescheduling Method for Operation Time Delay Disturbance in Steelmaking and Continuous Casting Production Process. *International Journal of Iron and Steel Research*, 19 (2012), 33–41.

[39] H. Zhong, X Dong and H. Shi, Research on the load balancing scheduling problem of reentrant hybrid flowshops. *Chinese High Technology Letters*, 25 (2015), 70–81.

[40] H. Zhong, Y Zhu and S. Lin, A dynamic co-evolution compact genetic algorithm for E/T problem. *The 17th IFAC Symposium on System Identification*, (2015), 1433–1437.

Received January 2015; 1st revision August 2015; final revision April 2016.

E-mail address: swinburnsun@163.com
E-mail address: luanfangjun@sjzu.edu.cn
E-mail address: yingyu1984@163.com
E-mail address: maokunneustu@163.com