INTRODUCTION

With the worsening air pollution problem, controlling carbon emissions has attracted a great attention of all the countries in the world. According to the estimation of international energy agency, about 4% ~ 5% of carbon dioxide emissions in the world is caused by the iron and steel industry. Therefore, the study on the energy conservation and emissions reduction of iron and steel industry is extremely urgent. At present, the main method of energy conservation and emission reduction is to update the machine and improve process. However, the method of energy conservation and emission reduction through the optimization of production schedule was neglected to a great extent[1]. This method has significant advantages compared with the traditional method without the limit of the ability of improvement and the necessity of huge capital investment. Steelmaking and continuous casting is the main process in iron and steel production and also the key link of energy consumption and carbon emission. Therefore, the reasonable arrangement of production scheduling plan can remarkably reduce carbon emission.

Currently, the research on carbon emission from the perspective of production scheduling is just beginning, so few relevant literatures are available. Li and Head [2] improved the original auto industry scheduling model with minimizing single cost as the objective, made research about the problem of reducing carbon emissions, and proposed to realize the optimized comprehensive objective through scheduling instead of energy supplement for fossil fuel. Fang et al.[3] thought the maximum completion time of artifact, peak power consumption load and carbon footprint peak power were the objectives, adopted multi-objective mixed integer programming method, and made instance analysis on processing system composed of two machines. Because of the high complexity of this problem, optimization method is only suitable for small-scale problems. Liu[4-5] constructed the optimization model with minimizing carbon emissions and the total tardiness penalties as double objectives for the batch production scheduling and used adaptive and hierarchical genetic algorithm to solve the problem.

Steelmaking continuous casting production scheduling problem has received extensive attention of many scholars. Tang et al.[6] discussed the steelmaking continuous casting production scheduling problem, adopted nonlinear programming model to solve the machine conflict, and then...
converted it into linear programming model to solve the problem. Bellabdaoui and Teghem[7] constructed a mixed integer programming model and used standard software package to solve the steelmaking continuous casting production scheduling problem. According to Yu Shengping et al.[8], the hybrid intelligent optimization method based on the combination of equipment assignment of expert system and neighborhood search of human-computer interaction was proposed to solve three-stage steelmaking continuous casting scheduling problem. And then the ant colony algorithm and the hybrid algorithm based on nonlinear optimization method were proposed by Atighehchian et al.[9] for solving steelmaking continuous casting production scheduling problem. Tang et al.[10] thought the maximum completion time was the optimization goal, conducted the research on the coordinated scheduling problem of steelmaking continuous casting production and transportation, and made use of tabu search algorithm to solve model. Tan et al.[11-12] constructed the energy cost minimization model for each cast in view of controllable processing times and alterable electricity price, and used the hybrid algorithm of elicitation and constraint propagation to solve it. In the above literatures, process constraints or economic indicators in steelmaking continuous casting production scheduling are regarded as optimization goal. Carbon footprint as an environmental indicators has not been considered yet. Based on this, the optimization model taking the minimized makespan has not been considered yet. Based on this, the optimization model taking the minimized makespan and carbon footprint as double objectives was built in order to solve steelmaking continuous casting strong-constrained production scheduling problem.  

Steelmaking continuous casting production scheduling can be abstracted as the strong-constraint hybrid flow shop scheduling with limited waiting time, continuous batch processing in the stage of continuous casting, and casting on time. For the strong constraint scheduling problem, the encoding gene of its solution is subject to the order constraint. It is difficult to get effective solution through the genetic crossover and mutation operation. In literatures[6-9], the constraint conditions were multiplied by the penalty coefficient of their own, which was taken as objective function to solve the problem (but the solution not satisfying constraint conditions was allowed ). However, the strong-constraint problem was never solved. PBIL algorithm is different from the traditional evolutionary algorithm which is based on population updating. According to PBIL, algorithm probability model is used for learning and sampling and probability selection is conducted for the gene-bits of solution. Therefore, the Inferior solution space can be effectively avoided according to the volume of probability, which can make the population search more targeted. The algorithm can fast and effectively learn the arrangement information of high quality solution and concentration solution, which can provide guide for effectively searching the solution space of the problem. PBIL algorithm has been used for solving many optimization problems. According to Li Zuocheng et al.[13], the minimized and earliest completion time was the optimization goal and the PBIL algorithm integrated with subpopulation probability model was designed to solve the heterogeneous parallel machine scheduling problem. Bo-Yeong Kang et al.[14] proposed a new population-based incremental learning (PBIL) algorithm for robot path planning. Zhang, Qingbin et al.[15] proposed an improved PBIL algorithm to solve the signal timing problem of an isolated intersection and the experimental results show that the algorithm can get rational signal timing effectively with more insensitive to the parameters setting. Komla A. Folly.[16] proposed a method of optimally tuning the parameters of power system stabilizers (PSSs) for a multi-machine power system using Population-Based Incremental Learning (PBIL). According to the related literature research, there are no relevant research results about steelmaking and continuous casting production scheduling problem and the effective algorithm to solve the problem is also not available.

The paper is organized as follows. A double-objective optimization model is proposed to minimize the makespan and carbon footprints. Then, the Improved Population-Based Increased Learning (IPBIL) algorithm is proposed to solve the problem. The IPBIL algorithm uses a two-stage search strategy and a time-window moving scheme to guide the population to the iterative search of the optimal solution region by establishing a new probability model and its updating mechanism. Finally, the effectiveness of the algorithm is validated through simulation experiment and the influencing factors of carbon footprint are analyzed.

2. Steelmaking-Casting production Scheduling Model

2.1 Problem Description

Steelmaking processes include: smelting, refining and continuous casting. Scrap steel and other raw materials are smelted by EAF (Electric arc furnace) to molten steel, which has the required chemical composition and temperature after being processed by LF (Ladle furnace). And then it is continuously
casted in the continuous casting machine into steel slab meeting the scheduled steel grade requirements for being used in hot rolling process. The molten steel smelted by the same EAF at a time is called charge. The set of charge continuously casted in the same continuous casting machine is called cast.

Due to the strict requirement of continuous casting process on the temperature of molten steel, the dwell time of charge between processes cannot exceed the upper limit and be less than the transportation time. In order to ensure the close jointing between continuous casting and hot rolling processes, in continuous casting stage poring must be started on time and production must be carried out in batches according to the cast without breaking. In theory, Steelmaking continuous casting production scheduling can be regarded as strong-constrained hybrid flow shop scheduling with limited charge dwell time, continuous batch processing in continuous casting stage, and poring on time. The production process is shown in Figure 1.

In this paper, the optimization problem with minimizing makespan and the carbon footprint as double objectives in steelmaking continuous casting production scheduling was studied in the case of giving the production plan of upper level (the charges in one cast, cast corresponding machine, cast sequence and casting time were known).

### 2.2 Notation

- $j$: stage index, $j = 1, 2, 3$ refers to smelting, refining and casting respectively;
- $i$: charge index (also called job index), $i = 1, 2, \ldots, I$;
- $n$: cast index, $n = 1, 2, \ldots, N$;
- $\Omega_n$: set of indices of the $n$-th batch, $n \in \{1, 2, \ldots, N\}, \Omega_{n1} \cap \Omega_{n2} = \emptyset$, and for any $n1 \neq n2 \in \{1, 2, \ldots, N\}$, $\bigcup \Omega_1 \cup \Omega_2 \cup \ldots \cup \Omega_N = \Omega$;
- $L(n)$: index of the last job in the $n$-th cast;
- $m_j$: number of machines available in stage $j$, $m_j \in \{1, 2, \ldots, M_j\}$;
- $B_n$: set of indices of all batches on the cast stage;
- $ST$: setup time between two adjacent casts on the same machine in the last stage;
- $D_n$: planning time of batch $n$ starting;
- $t_i$: processing time of job $i$ in stage $j$;
- $s_i$: starting time of job $i$ in the process $j$;
- $c_i$: finishing time of job $i$ in the process $j$;
- $T_{j+1}$: transportation time from stage $j$ to stage $j+1$;
- $\alpha_{ik}$: 0/1, variable that is equal to one if and only if job $i$ is processed on the $k$-th machine in stage $j$;
- $\gamma_{ik}$: 0/1, variable that is equal to one if and only if charge $i$ is processed before charge $i_k$ in stage $j$;
- $P_i$: power of the machine on stage $j$;
- $P_{ij}$: idle power of the machine on stage $j$;
- $T_{\max}$: upper limit of dwell time of the job between two adjacent stages;
- $\beta$: decreased temperature per unit time in the process of a ton molten steel being transported and waiting;
- $c$: specific heat capacity of molten steel;
- $m$: quality of molten steel.

### 2.3 Mathematical model

The environment indicator $CO_2$ is regarded as objective function $f_1$. The carbon emissions in the process of steelmaking continuous casting production scheduling is made up of three main parts:

I. The carbon emissions caused by the machine processing. The carbon footprint shown in formula (1) was obtained by the total energy consumption of EAF, LF and CCM processing heat being multiplied by the coefficient of carbon emissions, where $\varepsilon$ is the coefficient of carbon emissions.

$$EB_{\text{co2}} = \varepsilon \sum_{i=1}^{I} \sum_{j=1}^{3} P_{ij} t_i$$

II. The carbon emission caused by machine running empty between two adjacent charges. After the continuous casting machine finished a heat, it was necessary to turn off the machine for cleaning or replacing the mould. Until the next heat begun, the machine was rebooted. Therefore, in the continuous casting stage, the machine never run empty. Formula (2) refers to the carbon footprint caused by machine running empty in smelting and refining processes.
Formula (5) expresses the temperature decreased follows: two optimization objectives, which is shown as Scheduling can be obtained on the basis of the above continuous casting strong-constrained production energy. Formula (4) expresses the dwell time of all heat (including transportation and waiting time). Formula (5) expresses the heat generated per unit of electrical the dwelling process of all heat. Formula (7) expresses the heat generated per unit of electrical energy.

\[ R_{vq2} = \varepsilon cm\beta T^R / q \]  
(3)

\[ T^R = \sum_{i=1}^{I} \sum_{j=1}^{J} (s_{i,j+1} - c_{ij}) \]  
(4)

\[ \Delta t = \beta T^R \]  
(5)

\[ Q = cm\Delta t \]  
(6)

\[ q = 3600kJ / kwh \]  
(7)

The carbon footprint caused in the steelmaking continuous casting strong-constrained production scheduling is as follows:

\[ f_1 = EB_{vq2} + EI_{vq2} + R_{vq2} \]  
(8)

II. The economic indicator, namely makespan (the maximum completion time) was regarded as objective function \( f_2 \).

The completion time of the last operation should be determined, so the pouring in the continuous casting process can start on time. The factor that influences the maximum completion time is the starting time of the first heat in the first process. The dwell time of each heat from one process to the next should be larger than transportation time but shorter than the maximum time limit. Reverse deducing form the continuous casting process to refining process, the allowed starting time range for each heat can be determined, so the pouring in the continuous casting process can start on time. The factor that influences the maximum completion time is the starting time of the first heat in the first process. The dwell time of each heat from one process to the next should be larger than transportation time but shorter than the maximum time limit. Reverse deducing form

\[ s_{i,j+1} - c_{ij} \geq T_{j,i+1}, \forall i \in I, \forall j \in \{1, 2\} \]  
(14)

\[ s_{i,j+3} = s_{i,j} + t_{ij}, \forall i \in I, \forall j \in \{1, 2, \ldots, N\} \]  
(15)

\[ s_{L(a)+1},3 - c_{L(a),3} \geq ST, \forall m_{ij}, a \in B_{an} \]  
(16)

\[ x_{ij} - c_{ij} + (3 - x_{ij} - k_{ij} - y_{ij})U \geq 0 \]  
(17)

\[ \sum_{k=1}^{M} x_{ij} = 1, \forall i \in I, \forall j \in \{1, 2\} \]  
(18)

\[ y_{ij} = 1, \forall i \neq i1, i2, \forall j \in \{1, 2\} \]  
(19)

\[ T_{j,i+1} \leq \sum_{i=1}^{I} \sum_{j=1}^{J} (s_{i,j+1} - c_{ij}) \leq T_{max}, \forall i \in I, \forall j \in \{1, 2\} \]  
(20)

\[ x_{ij} \in \{0, 1\}, \forall i \in I, \forall j \in \{1, 2\}, \forall k \in \{1, 2, \ldots, M\} \]  
(21)

\[ y_{ij} \in \{0, 1\}, \forall i \neq i1, i2, \forall j \in \{1, 2\} \]  
(22)

This is a double-objective optimization model. Formula (13) defines objective function. The minimum carbon footprint and the maximum completion time are the optimization objectives. Formula (14) indicates that for the same heat, the next process begins only until the previous process has been finished and the artifact has been transported to the machine of the next process. Formula (16) indicates that certain adjustment time is required between the adjacent casts on the continuous casting machine to change the mould and tundish and clean machine. Formula (17) indicates that the machine can only process one heat at the same time, which also means resource constraint. Formula (18) indicates that each heat can only be processed by one machine in all processes. Formula (19) indicates that any two different heats is processed successively in each stage. Formula (20) indicates that the dwell time of each heat should exceed transportation time but be less than maximum dwell time. Formula (21) and (22) define the scope of variables.

3 IPBIL Algorithm

IPBIL is an improved PBIL algorithm. It combines the standard 2-D PBIL probability matrix into a three-dimensional matrix and establishes a new updating mechanism. The double-probability model is used to guide the population two-stage search. The local search algorithm of the transition time window is designed to optimize the makespan and carbon footprint. IPBIL algorithm uses the improved PBIL probability model to generate each generation of population, so as to perform a highly efficient global search on the solution space, and to improve the local search capability of the algorithm by using the two-stage local search of time window. Due to the good balance between global and local search, the
algorithm has the ability to obtain the excellent solution of the problem.

### 3.1 Encoding Mode

In the paper, ten-digit coding mode is used. Let the solution \( \pi = [\pi_1, \pi_2, ..., \pi_s, \pi'_1, \pi'_2, ..., \pi'_m] \) expresses the heat order of all production processes, where \( s \) is the total number of production process, and \( m \) is the number of production machines in each stage. For example, \( \rho^k_q = [4,5,10,13,15] \) expresses the production heat sequence of machine 3 in production process 2, where \( \pi^i_1, ..., \pi^i_m \) are the known conditions.

Let \( \rho^k_q = [\rho^k_q[1], ..., \rho^k_q[i], ..., \rho^k_q[n]] \) expresses the production sequence of machine \( q \) in process \( k \), where \( |\rho^k_q| \) is the number of production heat. In process \( k \), heat \( \rho^k_q[i] \) has actual starting time, actual completion time, allowed earliest starting time, allowed latest starting time, corresponding probability matrix, the production sequence of each machine in smelting process is extended to a 3-dimensional one.

Let \( \rho^k_q = [\rho^k_q[1], ..., \rho^k_q[i], ..., \rho^k_q[n]] \) expresses the actual production time sequence of \( \rho^k_q \).

### 3.2 Local Search Algorithm based on Time Window Backward Moving method

The time window backward moving method to the local search algorithm is to push the start time of the furnace as much as possible to the maximum allowable start time, so that the time from the ending of one process to the next is reduced and the carbon emissions in the dwell process of heat can be reduced.

Since the completion time of the last furnace in the final step is fixed, the makespan can be minimized if the time of all previous processes is postponed. For the production sequence \( \rho_q = [\rho_q[1], ..., \rho_q[i], ..., \rho_q[n]] \), the operation steps of the time window backward moving method are as follows:

1. Calculate the allowed starting time window of \( \rho^k_q[i] \), \( T^k_q[i] \), \( T^k_q[i] \), \( i = 1, 2, ..., n \).
2. Let \( i = n \), postpone the actual starting production time of heat, and then make \( T^k_q[i] = T^k_q[i] \).
3. Let \( i = i-1 \), calculate \( T^k_q[i] \) and \( T^k_q[i] \), \( T^k_q[i] \). If \( T^k_q[i] = T^k_q[i] \), and then \( T^k_q[i] = T^k_q[i] + T^k_q[i] \), or \( T^k_q[i] = T^k_q[i] + T^k_q[i] \).

### 3.3 The Search Strategy of IPBIL

According to the production process requirements of steelmaking and continuous casting, The IPBIL uses double probability model to guide the population two-stage search. In the case of given operation plan of continuous casting, the process using the heat of each machine in the continuous casting process to solve the heat sequence of each machine in refining process is called the first search stage of IPBIL. The process using the heat of each machine in the refining process to solve the heat sequence of each machine in smelting process is called the second search stage of IPBIL.

At first, according to the requirements on the steelmaking and continuous casting processes, the standard 2-dimensionaol probability matrix is extended to a 3-dimensional one. That means that each search stage of PBIL corresponds to a single 2-dimensional probability matrix. According to the corresponding probability matrix, the production heat in each stage is guided and the number of machine used to process the heat is selected.

Next, the time window backward moving method mentioned in section 3.2 is used to solve the corresponding production starting time of the heat in each stage for achieving more optimized objectives.

Finally, the optimal solution of this population is used to update two-stage probability model, which can ensure to globally search near the optimal solution.

### 3.4 Steps of IPBIL Algorithm

Step 1: initialize population and probability matrix mode according to formula(23), where \( k = 1, 2, m \) is the quantity of machines in each process, \( n \) is number of all charge, \( N = [1, 2, ..., n] \) is the set of all charges. Each column of vector in matrix expresses the probability of corresponding charge choosing machine. \( Pro^k \) is the probability of process \( k \), \( Pro^k \) is the row \( i \), column \( j \) of the matrix.

\[
Pro^k = \begin{bmatrix}
1/m & \cdots & 1/m \\
\vdots & \ddots & \vdots \\
1/m & \cdots & 1/m_{n,m}
\end{bmatrix}
\] (23)

Step 2: Let \( k = 2 \), sort for all charge according to the starting time of each charge in process \( k+1 \),
getting the sort $\gamma^k = [\gamma^k[1], \gamma^k[2], \ldots, \gamma^k[l], \ldots, \gamma^k[n]]$.

Step 3: select the machine used to process each charge in $\gamma^k$, getting the sort $\xi^k = [\xi^k[1], \xi^k[2], \ldots, \xi^k[l], \ldots, \xi^k[n]]$, $\xi^k$ is the corresponding production machine sort of $\gamma^k$. In the initializing time, $l = 1$.

Step 3.1: Calculate the allowed earliest and the latest starting time of $\gamma^k[I]$ in the process, which are expressed by $T^k_i[I]$ and $T^k_i[l]$ respectively.

Step 3.2: calculate the time of machine $r$ completing the assigned charge $T^k_i[r]$, where, $r = 1, 2, \ldots, m$ . If no charge was assigned to machine $r$ in the initializing time, $T^k_i[r] = 0$.

Step 3.3: let the set of selectable machine of charge $\gamma^k[I]$ be $\mathcal{M}^k_i$, $\mathcal{M}^k_i = \Phi$. if $T^k_i[I] \geq T^k_i[l]$, $\mathcal{M}^k_i = \mathcal{M}^k_i + \{r\}$, where $r = 1, 2, \ldots, m$.

Step 3.4: according to the probability derived by formula (24), roulette wheel method was used in $\mathcal{M}^k_i$ to select production machine $w$ of charge in $\gamma^k$. $\xi^k[I] = w$, $w \in \mathcal{M}_i$. Where, $p_{i,j}$ expresses the probability of charge $j$ selecting machine $i$. The sequence of machine $w$ producing charge $\xi^k$ was recorded. If $T^k_i[w] \leq T^k_i[I]$, the actual starting time of charge $\gamma^k[I]$ in machine $w$ is $T^k_i[I] = T^k_i[I]$; if $T^k_i[I] \leq T^k_i[w] \leq T^k_i[I]$, $T^k_i[I] = T^k_i[I]$.

$$p_{i,j} = P_{o_{ij}} / (1 + \theta), \quad (k = 1, 2, \quad i = \xi^k[I], \quad j = \gamma^k[I], \quad l = 1, 2, \ldots, n) \quad (26)$$

Step 7: If the termination condition cannot be met, return to step 3; or the algorithm is finished, and then output the results.

4 Numerical experiments

4.1 Problem instances

Take the actual production scheduling problem of a steel plant as example. The main production mode of the plan is two electric arc furnace (EAF1 EAF2), two refining furnace (LF1 LF2), and two continuous casting machine (CC1 CC2).

The production time range of charge and the processing and idle power of machine in each process are shown in Table 1. The upper limit of dwell time for each charge is 25min, and transportation time is 5min. Processing capacity of electric arc furnace is 150t. The casting plan made according to the production plan of the higher level includes 2casts, one cast in a continuous casting machine and 6 charges in each cast.

<table>
<thead>
<tr>
<th>Table 1 Machine production time and power</th>
</tr>
</thead>
<tbody>
<tr>
<td>machine</td>
</tr>
<tr>
<td>Production time/min</td>
</tr>
<tr>
<td>$P_r$(kw/min)</td>
</tr>
<tr>
<td>$P_l$(kw/min)</td>
</tr>
</tbody>
</table>

Note: The first number in [] is the minimum production time, the second is the maximum production time.

4.2 Experimental results and Analysis

The experiment parameters are set as follows: population size=30, learning rate $\theta = 0.01$, maximum evolution generation as 300. The carbon footprint of all machine production charge is constant. Therefore, the carbon print caused by machine running empty and charge dwell do not effect by scheduling plan. $\eta = (E_{v_{oa}} + R_{oa}) / f_i$ defines the proportion of carbon footprint caused by machine running empty, $f_i$ is the carbon footprint caused by charge dwell, and $f_i$ is the total carbon footprint.

I. Single - objective Optimization Simulation Experiment for Minimizing makespan

According to formula (12), the lower limit of the makespan for the scheduling is 365. The IPBIL algorithm with minimizing makespan as objective is run 15 times, with the results shown in Table 2.
Table 2 Date of makespan and carbon footprint by running the algorithm 15 times

<table>
<thead>
<tr>
<th>Number</th>
<th>makespan (min)</th>
<th>Total carbon footprint (kg)</th>
<th>the carbon footprint caused by machine running empty (kg)</th>
<th>The carbon footprint caused by charge dwell (kg)</th>
<th>η (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>365</td>
<td>615955</td>
<td>111250</td>
<td>3705</td>
<td>18.66</td>
</tr>
<tr>
<td>2</td>
<td>365</td>
<td>605335</td>
<td>100250</td>
<td>4085</td>
<td>17.24</td>
</tr>
<tr>
<td>3</td>
<td>365</td>
<td>618350</td>
<td>114500</td>
<td>3111</td>
<td>19.02</td>
</tr>
<tr>
<td>4</td>
<td>365</td>
<td>585330</td>
<td>80150</td>
<td>4180</td>
<td>14.41</td>
</tr>
<tr>
<td>5</td>
<td>365</td>
<td>607200</td>
<td>102400</td>
<td>3800</td>
<td>17.49</td>
</tr>
<tr>
<td>6</td>
<td>365</td>
<td>584295</td>
<td>78450</td>
<td>4845</td>
<td>14.26</td>
</tr>
<tr>
<td>7</td>
<td>365</td>
<td>606455</td>
<td>100255</td>
<td>4845</td>
<td>17.33</td>
</tr>
<tr>
<td>8</td>
<td>365</td>
<td>619110</td>
<td>114500</td>
<td>3610</td>
<td>19.08</td>
</tr>
<tr>
<td>9</td>
<td>365</td>
<td>614480</td>
<td>109300</td>
<td>4180</td>
<td>18.47</td>
</tr>
<tr>
<td>10</td>
<td>365</td>
<td>585330</td>
<td>80150</td>
<td>4180</td>
<td>14.41</td>
</tr>
<tr>
<td>11</td>
<td>365</td>
<td>611680</td>
<td>106500</td>
<td>4180</td>
<td>18.09</td>
</tr>
<tr>
<td>12</td>
<td>365</td>
<td>581515</td>
<td>76050</td>
<td>4465</td>
<td>13.85</td>
</tr>
<tr>
<td>13</td>
<td>365</td>
<td>600950</td>
<td>96150</td>
<td>3800</td>
<td>16.63</td>
</tr>
<tr>
<td>14</td>
<td>365</td>
<td>595020</td>
<td>89650</td>
<td>4370</td>
<td>15.80</td>
</tr>
<tr>
<td>15</td>
<td>365</td>
<td>595550</td>
<td>90750</td>
<td>3800</td>
<td>15.88</td>
</tr>
<tr>
<td>Average</td>
<td>365</td>
<td>601770.3</td>
<td>96687</td>
<td>4077.07</td>
<td>16.71</td>
</tr>
</tbody>
</table>

In the results, the value of makespan optimization objective is 365, reaching the allowed minimum value. The average value of carbon footprint caused by scheduling plan is 601770.3. According to the Figure 2, the carbon footprint caused by machine running empty has positive relationship with the total carbon footprint and accounts for a higher proportion in the total carbon footprint, which means that the carbon footprint caused by machine running empty is the major factor influencing total carbon footprint. Due to the upper and lower limit of dwell time, the variation range of carbon footprint caused by charge dwell is small, which is from 3000kg to 5000kg.
The double-objective optimization procedure is run 15 times, getting the Table 3. According to Table 3, the average value of makespan is 365, which is the allowed minimum value. The average value of carbon footprint is decreased to 562410 kg. The carbon footprint caused by machine running empty and charge dwell are decreased by 40.5% and 4.7% respectively. $\eta$ is reduced to 10.92%. The proportion of carbon footprint caused by scheduling plan is significantly reduced. Thus, the IPBIL algorithm can effectively optimize the dual goals of makespan and carbon footprint.

Figure 3 shows a gantt chart of scheduling example about the above problems gained by the algorithm, where the black rectangle expresses the charge dwell time, and dashed rectangle expresses the time of machine running empty. Each charge in the scheduling plan meets the constraint of upper and lower limit, which can ensure to pour on time in the continuous casting stage and produce each charge in the planned sequence. The average dwell time of all charge is 9 min, much less than the upper limit of time. The process constraints are met. At the same time, the carbon emissions are effectively reduced.

The related data of smelting and refining process in this scheduling plan are shown in Table 4. The
average value of the Machine energy utilization rate is 84.19%. And the energy utilization rate of EAF2 is 100%. Although the utilization rate of the machine EAF1 is 30% higher than that of LF1, the carbon emissions caused by EAF1 are 17000kg more than LF1 because the idle capacity of EAF1 is much larger than that of LF1. On the conditions of fixed sequence and limited dwell time between processes of each charge in the continuous stage, enhancing the machine utilization rate of smelting process can more effectively reduce energy consumption and carbon footprint compared with refining process.

<table>
<thead>
<tr>
<th>Machine No</th>
<th>Idle time (min)</th>
<th>Idle capacity (kw/min)</th>
<th>Production time (min)</th>
<th>Production capacity (kw/min)</th>
<th>Mechanical energy utilization rate (%)</th>
<th>Idle carbon footprint (kg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EAF1</td>
<td>35</td>
<td>950</td>
<td>245</td>
<td>1000</td>
<td>88.05</td>
<td>33250</td>
</tr>
<tr>
<td>EAF2</td>
<td>0</td>
<td>950</td>
<td>175</td>
<td>1000</td>
<td><strong>100.00</strong></td>
<td>0</td>
</tr>
<tr>
<td>LF1</td>
<td>125</td>
<td>130</td>
<td>150</td>
<td>150</td>
<td>58.06</td>
<td>16250</td>
</tr>
<tr>
<td>LF2</td>
<td>25</td>
<td>130</td>
<td>210</td>
<td>150</td>
<td>90.65</td>
<td>3250</td>
</tr>
<tr>
<td>Average</td>
<td>46</td>
<td>—</td>
<td>195</td>
<td>—</td>
<td><strong>84.19</strong></td>
<td>—</td>
</tr>
</tbody>
</table>

### Table 4 The related energy consumption date of production scheduling example

III. The extended experiment for the algorithm of several groups of typical data. The experiment is conducted for the commonly used several groups of production data in the actual production. There are 7-10 charges in each cast. Within the range from 24 hours to $3 \times 24$ hours of planned quantity, usually adapted charge number is 6, 7, 10, 12, 14, 16, 18. The experiment results are shown in Table 5. And the lower limit of makespan is calculated out by formula(12). Many groups of experimental data show that IPBIL algorithm can minimize the makespan and effectively reduce carbon footprint on the basis of the local search strategy integrated with time window backward moving method. For the production data of 18 charges and 151 casts (about $3 \times 24$ hours of planned quantity), the average running time of the algorithm is only 11.72s, which can completely satisfy the requirements of actual production.

<table>
<thead>
<tr>
<th>Cast number</th>
<th>Charge number in each cast</th>
<th>Makespan Lower limit(min)</th>
<th>Without local search</th>
<th>With local search</th>
<th>The reduced carbon footprint (kg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>6</td>
<td>[9,10,10,10,8,9]</td>
<td><strong>1245</strong></td>
<td>1295</td>
<td>Carbon footprint (kg)</td>
<td>2696100</td>
</tr>
<tr>
<td>7</td>
<td>[10,9,9,9,8,10,10]</td>
<td><strong>1555</strong></td>
<td>1595</td>
<td>Carbon footprint (kg)</td>
<td>3204660</td>
</tr>
<tr>
<td>10</td>
<td>[10,10,8,8,8,8,7,8,10,9,10]</td>
<td><strong>1826</strong></td>
<td>1849</td>
<td>Carbon footprint (kg)</td>
<td>4176856</td>
</tr>
<tr>
<td>12</td>
<td>[9,9,10,10,8,8,7,8,10,9,10]</td>
<td><strong>2155</strong></td>
<td>2195</td>
<td>Carbon footprint (kg)</td>
<td>4921315</td>
</tr>
<tr>
<td>14</td>
<td>[7,10,7,10,8,8,7,8,8,7,9,10]</td>
<td><strong>2458</strong></td>
<td>2486</td>
<td>Carbon footprint (kg)</td>
<td>5787934</td>
</tr>
<tr>
<td>16</td>
<td>[8,10,9,10,8,8,8,7,8,8,8,7,9,10,9,7,8,8,9,9]</td>
<td><strong>2765</strong></td>
<td>2798</td>
<td>Carbon footprint (kg)</td>
<td>6122845</td>
</tr>
<tr>
<td>18</td>
<td>[9,9,8,8,8,8,7,9,9,9,10,10,7,7,8,8,9,9]</td>
<td><strong>2955</strong></td>
<td>2995</td>
<td>Carbon footprint (kg)</td>
<td>6590145</td>
</tr>
</tbody>
</table>

5 Conclusions

In the paper, the objective of carbon emissions is considered in the steelmaking and continuous casting production scheduling optimization problem. Furthermore, the optimization model with minimizing makespan and carbon footprint is constructed for the strong-constrained production scheduling problem with the characteristics of limited operation waiting time, continuous batch production in the last stage, and casting on time. Because the traditional gene crossover and mutation operators would ruin the strong constraint conditions, it is difficult to gain the efficient solution. However, the problem is solved by PBIL algorithm, in which the gene-bits of solution are selected based on probability. Double probability model is used for guiding population to conduct two-stage search. Finally, the time-window moving scheme is designed for guiding population to iteratively search in high quality solution region.
The experimental results shows that IPBIL algorithm can effectively solve the steelmaking and continuous casting strong-constraint production scheduling problem with double optimization objectives. According to experiment analysis, we find that: (1) Idling carbon emissions are the main contributors to total carbon emissions. (2) As the EAF machine power is much larger than LF, in the production schedule under strong constraints, compared with the refining process, improving the smelting process machine utilization can effectively reduce energy consumption and carbon emissions. (3) The total carbon footprint of the 8-hour operation plan for 12 charges and 2 casts is reduced by 39,600kg. Therefore, optimizing the steelmaking and continuous casting production schedule can effectively reduce carbon footprint, which is significant for the iron and steel enterprises to adopt new ways to achieve energy saving and emissions reduction.

6. ACKNOWLEDGEMENTS

This work was financially supported by the National Natural Science Foundation of China (71271054, 70971022).

7. REFERENCES