Prediction-Based Simulation for Molten Metal Logistics

Toshiharu IWATANI,
Electronics Research Laboratory, Kobe Steel, Ltd.
Takatsukadai 5-5 Nishi-ku Kobe 651-2271, Japan
t-iwatani@rd.kcr.kobelco.co.jp

Kouichi MATSUDA and Koichiro SEMURA
Kakogawa Works, Kobe Steel, Ltd.
Kanazawa-cho 1 Kakogawa 657-0137, Japan

Abstract:

In this paper, a prediction-based simulation method is introduced and its application result for molten metal logistics is described. In order to keep some operational constraints, operators control the molten metal logistics by modifying a production process of an iron charging ladle known. Since the modification causes a cost increase, it is desired to minimize its frequency. The simulator for this process must be able to reproduce such logistic control skill. We developed a simulator that can predict the logistic state and change the production process. The advantage of the developed simulator is discussed comparing with conventional simulation result. Further, we consider the feature of the implemented algorithm in comparison with simulation optimization method.

Keyword: molten metal logistics, simulation optimization, prediction-based simulation

1. Introduction

In a steel making plant, molten metal, which is tapped from blast furnaces, is transported to converters using a charging ladle known as a torpedo car. We call this physical distribution process a molten metal logistics (Paul and Balmer, 1993). Since there are several chemical and physical plants between the blast furnaces and converters and quite a few torpedo cars are in use, the logistics is complicated. Further, there exist characteristic operational constraints, which are so unique that a usual simple simulation algorithm cannot keep them. So, it was necessary and technically challenging problem to construct a simulator that can reproduce the logistics and estimate molten metal productivity quantitatively. This paper describes as follows: essential elements, specific constraints, and necessity of the simulator of the molten metal logistics are shown in Section 2. The point under discussion in the simulation algorithm is clarified in Section 3 before comparing the usual simulation result and that derived from a prediction based simulation algorithm in Section 4. Some consideration is given in Section 5 while Section 6 is a conclusion.

2. Outline of the molten metal logistics

2.1 Elements of logistics

Torpedo cars filled with molten metal are transported to relading pits by a locomotive. After the molten metal is transferred from a torpedo car into a converter at the pits, the empty torpedo car is carried back to a blast furnace to receive pig iron again. Torpedo cars cover this route twice or three times a day. There are treatment plants for the molten metal and torpedo cars stop at several plants and undergo treatment. Usually, about 25 torpedo cars and 5 or 6 locomotives are at work in the logistics. Figure 1 shows important plants in the logistics.

![Fig. 1 Plants in molten metal logistics](image-url)

Before arriving at the relading pits, most of the torpedo cars undergo a deslagging process, a dephosphorization pretreatment process and a desulphurization pretreatment process. Table 1 shows the types of production processes of molten metal between the blast furnaces and the converters and its percentage. The ratio shown in Table 1 is an actual result in a month and it...
changes depending on the product mix. After the molten metal discharging, every empty torpedo car undergoes a slag-dumping process. Some torpedo cars return to the blast furnaces directly after dumping and the remaining ones return by way of a removal-of-skull process. Once more, the torpedo cars are charged and carried to the reladling pit again.

Table 1 Types of production processes

<table>
<thead>
<tr>
<th>Type</th>
<th>Ratio</th>
<th>Tapping</th>
<th>De-Slagging</th>
<th>De-St</th>
<th>De-P</th>
<th>De-S</th>
</tr>
</thead>
<tbody>
<tr>
<td>Type 1</td>
<td>60%</td>
<td>☐</td>
<td>☒</td>
<td>☒</td>
<td>☐</td>
<td>☐</td>
</tr>
<tr>
<td>Type 2</td>
<td>10%</td>
<td>☐</td>
<td>☒</td>
<td>☒</td>
<td>☐</td>
<td>☒</td>
</tr>
<tr>
<td>Type 3</td>
<td>14%</td>
<td>☒</td>
<td>☒</td>
<td>☒</td>
<td>☐</td>
<td>☒</td>
</tr>
<tr>
<td>Type 4</td>
<td>2%</td>
<td>☒</td>
<td>☒</td>
<td>☒</td>
<td>☒</td>
<td>☒</td>
</tr>
</tbody>
</table>

2.2 Characteristics of molten metal logistics

In the molten metal logistics, there are two essential constraints, which are described in the Formulae (1.a) and (1.b).

\[
\begin{align*}
N_1(t) &> a_1 \\
N_2(t) &> a_2
\end{align*}
\]

(1.a) (1.b)

where

\[
\begin{align*}
N_1(t) & \text{ The number of torpedo cars before the blast furnaces at time } t \\
N_2(t) & \text{ The number of torpedo cars at the reladling pits at time } t,
\end{align*}
\]

\[
\begin{align*}
a_1, a_2 \text{ constants}
\end{align*}
\]

Formula (1.a) means that there must be a sufficient number of empty torpedo cars before the blast furnaces in order to prepare for a sudden tapping from blast furnaces because it is quite difficult to be sure of the exact time when pig iron will spout. The value of constant \( a_1 \) is usually equal to 9 or 10 and it depends on the condition of the blast furnaces. Formula (1.b) shows that some filled torpedo cars should stand by at the reladling pits in order to provide an unbroken supply of molten metal to the converters. The value of constant \( a_2 \) is decided in accordance with the condition of the converters and it is usually equal to 3 or 4.

When an equipment trouble occurs, it becomes a serious and difficult problem to keep the constraints shown above. For instance, when a plant between the blast furnaces and the converters is out of condition, a disposal time of the plant becomes longer than usual and arrival times at the reladling pits of torpedo cars would be delayed. Consequently, the constraint shown in formula (1.b) would be broken. In such a case, a logistic operator in the real steel plant can predict the decrease in the number of torpedo cars at the reladling pits. He changes the production processes of some filled torpedo cars in order to send them to the reladling pits as early as possible.

Specifically speaking, the operator omits the dephosphorization process from the filled torpedo cars whose manufacturing process are Type 1 or 2 in Table 1 since dephosphorization is executable at the converters. If the operator does nothing even if a plant is in a bad condition, which brings about a lack of molten metal at the reladling pit and a decline in the converters' productivity. Logistic control performed by operators is the most distinctive feature of the logistics. When a plant between the converters and the blast furnace break down, a logistic operator often omits the removal-of-skull process from torpedo cars to control the logistics because of the similar reason.

Dephosphorization at the converter is more expensive than that at the pretreatment equipment and the removal-of-skull process is necessary to maintain torpedo cars in a good condition. Therefore, skilled operators try to control the logistics and keep the ratios of pretreatment dephosphorization and removal-of-skull high. Accordingly, the simulator for molten metal logistics is also expected to maintain these features, that is, to optimize criteria shown in formulae (2.a) and (2.b).

\[
\begin{align*}
\text{Minimize } L_1 & \quad (2.a) \\
\text{Minimize } L_2 & \quad (2.b)
\end{align*}
\]

(2.a) (2.b)

where

\[
\begin{align*}
L_1 & \text{ The number of torpedo cars whose dephosphorization process is omitted} \\
L_2 & \text{ The number of torpedo cars whose removal of skull process is omitted}
\end{align*}
\]

It is desired not to make many torpedo cars wait before some plants except blast furnaces and the converters. However, such dispensable conditions are automatically satisfied when the conditions shown in formulae (1.a) and (1.b). Therefore, there is no other special constraints or criteria in the logistics.

2.3 Necessity of the logistic simulator

While it is usual that a logistic simulator is developed to show some operational guidance or to make a better schedule for the target process, the simulator for the molten metal is developed to attain another purpose. Several plans on plant and equipment investment are proposed to improve logistic indices, namely, a productivity of the converters and a ratio of pretreatment dephosphorization. As the target logistics consists of many plants and transportation facilities, it is difficult to quantitatively discuss the relative merits of the plans. A development of the simulator, which can reproduce the logistics properly for long period and calculate the indices, has been desired so as to evaluate the proposed plans.

23
3. Structure of simulation algorithm

3.1 Input data of the simulator

Table 2 shows necessary input data of the simulator. We derived these data from the actual operation result. Every processing time through the simulation is calculated stochastically using the average value and variation.

Table 2 Necessary input data

<table>
<thead>
<tr>
<th>Item</th>
<th>Average</th>
<th>Deviation</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hot metal tapping interval from BF</td>
<td>T_h</td>
<td>d_h</td>
</tr>
<tr>
<td>Hot metal consumption rate of converter</td>
<td>M_c</td>
<td>d_c</td>
</tr>
<tr>
<td>Processing time of De-slagging</td>
<td>T_d</td>
<td>d_d</td>
</tr>
<tr>
<td>Processing time of De-S preconditioning</td>
<td>T_p</td>
<td>d_p</td>
</tr>
<tr>
<td>Processing time of De-P preconditioning</td>
<td>T_p</td>
<td>d_p</td>
</tr>
<tr>
<td>Torpedo car transition time from plant p to plant q</td>
<td>T_t</td>
<td>d_t</td>
</tr>
</tbody>
</table>

3.2 Outline of simulation algorithm

In this sub-section, a common structure of three types of simulation algorithm, which we developed and applied to the molten metal logistics, is briefly described to elucidate the point under discussion in this paper. Figure 2 shows an outline of the simulation algorithms.

![Fig. 2 Outline of simulation algorithm](image)

An initial setting function makes every torpedo cars empty and set them in a queue before the blast furnaces in (a). After $\Delta t$ minutes passed by in (b), a torpedo car No. $i$ is selected in (c). A state transformation function decides the state of the selected torpedo car at time $t+\Delta t$ in (d). After every torpedo car state is renewed, $\Delta t$ minutes elapses again. When time $t$ passes the given simulation period $T$, the simulator stops after calculating the criterion indices in (e).

The definition of a state is described in the next sub-section and the state transformation function, in which the difference among three applied algorithms, are discussed in Section 4.

3.3 Concept of simulation algorithm and definition of torpedo car state

The simulation algorithm is designed focusing on the torpedo cars. For instance, we considered tapping, relading and all treatments in the plants as processes for torpedo cars. Moreover, the torpedo car movement is also defined as a process performed by 'a locomotives plant'. After a process for a torpedo car is finished, the next process for the torpedo car starts in accordance with a manufacturing process. In the actual operation, the manufacturing process of a torpedo car is decided when charged with molten metal from the blast furnace, the simulator is designed imitating this function. The developed simulator assigns a manufacturing process to each torpedo car stochastically according to given Table 1 after the charging in the simulation and we call an assigned manufacturing process 'an initial schedule' from now on.

It is impossible to start the next process if a necessary plant (or locomotives) is occupied by other torpedo cars even though a process is finished. In such a case, the simulation algorithm sets the torpedo car in a queue for the plant. When a plant finishes processing all the precedent torpedo cars in the queue, the simulation algorithm moves the torpedo car into the plant and lets the plant start the relevant process.

Formula (3.1) shows definitions of $s_i(t)$, the state of a torpedo car $i$ while Formula (3.2) shows that of $S(t)$, the state of the whole logistics.

$s_i(t) = (p_i(t), f_i(t))$  \hspace{1cm} (3.1)

where $p_i(t)$: a place where torpedo car No. $i$ stays, namely, a plant or a queue for a plant at $t$,

$f_i(t)$: remaining time when torpedo car $i$ leaves $p_i$

$S(t) = (s_1(t), s_2(t), ..., s_K(t))$  \hspace{1cm} (3.2)

where $K$: a number of torpedo cars

If $p_i$ is a plant, $f_i$ means remaining processing time and if $p_i$ is a queue $f_i$ is remaining waiting time. These values can be calculated using the processing times determined in accordance with Table 2.

3.4 State transformation function

While the simulation algorithm includes a lot of functions, the most important part is a state transformation function. $Trans()$ In our algorithm, $Trans()$ decides $s_i(t+\Delta t)$ from $s_i(t)$ and $S(t)$ as shown in Formula (4),

24
\[ s_i(t+\Delta t) = \text{Trans}(s_i(t), S(t)) \]  

In order to express the function concisely, we introduce here one technical assumption that torpedo cars always stand by in the queue after the previous process is finished even if the next plant is ready to process the torpedo car. This assumption simplifies the transfer patterns of torpedo cars: every torpedo car transfers from a plant to a queue and from a queue to a plant and repeats this pattern. Moreover, the state transformation function \( \text{Trans}() \) becomes to consists of only three functions as follows:

**Function 1.** Determine when a plant starts to process a torpedo car in the queue.

**Function 2.** Determine when a plant completes processing the torpedo car.

**Function 3.** Determine to which queue the torpedo car should go next.

For convenience, we name Function 3 'TC (Torpedo Car) control function'. There was no special technical problem in developing Function 1 or Function 2. In the next section, we describe the large influence that the torpedo car control function has on the ability of the simulator.

4. **Comparison between three types of simulation algorithm**

In this Section, three types of simulation algorithm are explained. The difference among these algorithms exists in TC control functions.

4.1 **Type 1: Usual and simple simulation algorithm**

The first developed simulator adopts a usual and simple algorithm. So, the torpedo car control function simply follows the initial schedule of each torpedo car (Figure 3). Namely, a torpedo car stops at all plants specified in its initial schedule in order.

![Fig. 3 State transformation in Type 1](image)

An output of the first simulator is shown in Figure 4, which describes the transition of the number of torpedo cars at the reladling pits.

**Fig. 4 Problem of Type 1 simulator**

This figure indicates that the simulation result does not keep the constraint shown in Formula (1.1b) (To simplify the discussion, we focus on Formula (1.1b)). The distribution of initial schedules used in this simulation is derived from one real operation period, in which no constraint violation took place.

**Table 3 Simulation result of Type 1 simulator**

<table>
<thead>
<tr>
<th>case</th>
<th>Percentage of de-P process omission</th>
<th>Percentage of constraint violation</th>
</tr>
</thead>
<tbody>
<tr>
<td>real operation</td>
<td>73.5%</td>
<td>0%</td>
</tr>
<tr>
<td>result of Type 1</td>
<td>0%</td>
<td>14.5%</td>
</tr>
</tbody>
</table>

Table 3 shows percentages of constraint violation and dephosphorization process omission in the real operation result and a simulation result. Naturally, simulation algorithm did not omit any dephosphorization pretreatment process and it failed to reproduce the actual operation result mainly because the constraints are not satisfies.

4.2 **Type 2: Torpedo car schedule modification by present state**

As shown in Section 2.2, logistic operators omit a dephosphorization pretreatment process from the initial schedule and dispatch the filled torpedo car to the reladling pits so as to avoid a lack of molten metal. We tried to realize such torpedo car control in the simulation algorithm. Figure 5 expresses the TC control function in Type 2 simulator. A schedule modification function, which is newly attached, can change the initial schedules of some torpedo cars in accordance with \( S(t) \) and given modifying rules. To put it more concretely, if the present number of torpedo cars at reladling pits is less than a constant \( m \), the TC control function selects a proper torpedo car and omits a dephosphorization pretreatment process from its initial schedule.
While the target number of the torpedo car is always more than 3 in this section, we carry out the simulation several times changing the value of \( m \). The simulation results are summarized in Table 4. The constraint is satisfied but the percentage of dephosphorization process omission is higher than the actual operation when \( m \) is equal to 4 or 5, though the constraint is not yet satisfied with \( m \) being equal to 2 or 3. In other words, Type 2 simulator is able to keep the constraints by omitting dephosphorization pretreatment process too often. Consequently, Type 2 is improved compared with Type 1, but it cannot reproduce the logistics satisfactorily.

### Table 4 Simulation result of Type 2 simulator

<table>
<thead>
<tr>
<th>CASE</th>
<th>( m )</th>
<th>Percentage of de-P process omission</th>
<th>Percentage of constraint violation</th>
</tr>
</thead>
<tbody>
<tr>
<td>real operation</td>
<td></td>
<td>7.3%</td>
<td>0.0%</td>
</tr>
<tr>
<td>result 1 of Type 2</td>
<td>1</td>
<td>3.0%</td>
<td>11.0%</td>
</tr>
<tr>
<td>result 2 of Type 2</td>
<td>2</td>
<td>6.0%</td>
<td>5.0%</td>
</tr>
<tr>
<td>result 3 of Type 2</td>
<td>4</td>
<td>9.8%</td>
<td>0.5%</td>
</tr>
<tr>
<td>result 4 of Type 2</td>
<td>5</td>
<td>15.2%</td>
<td>0.0%</td>
</tr>
</tbody>
</table>

### 4.3 Type 3: Schedule modification by prediction

It can be said that the insufficient result of Type 2 simulator is quite natural. Because it takes about 20 or 30 minutes to move a torpedo car that has not finished dephosphorization pretreatment process to the relading pits, omitting the pretreatment process of a torpedo car considering the present state is too late to recover the torpedo number at the relading pits. Figure 6 describes how the TP control function in Type 3 simulator works.

Before modifying the initial schedule, Type 3 simulator predicts \( S(t + n \Delta t) \), which is a future state \( n \) minutes later by utilizing a Type 1 simulator. The modifying rule in Type 3 is activated if the number of torpedo cars at the relading pit \( n \) minutes later is less than constant \( m \). The simulation results are described in Table 5. When \( n \) is equal to 30 and \( m \) is equal to 3, the constraint is always satisfied and the ratio of dephosphorization process omission is a little better than the result of real operation. Therefore, Type 3, which uses prediction to modify the torpedo car scheduling, is able to simulate the molten metal logistics.

### Table 5 Simulation result of Type 3 simulator

<table>
<thead>
<tr>
<th>CASE</th>
<th>( m )</th>
<th>( n )</th>
<th>Percentage of de-P process omission</th>
<th>Percentage of constraint violation</th>
</tr>
</thead>
<tbody>
<tr>
<td>real operation</td>
<td></td>
<td></td>
<td>7.3%</td>
<td>0.0%</td>
</tr>
<tr>
<td>result 1 of Type 3</td>
<td>2</td>
<td>10</td>
<td>4.0%</td>
<td>10.8%</td>
</tr>
<tr>
<td>result 2 of Type 3</td>
<td>3</td>
<td>10</td>
<td>7.2%</td>
<td>3.0%</td>
</tr>
<tr>
<td>result 3 of Type 3</td>
<td>4</td>
<td>10</td>
<td>9.5%</td>
<td>0.5%</td>
</tr>
<tr>
<td>result 4 of Type 3</td>
<td>5</td>
<td>10</td>
<td>14.8%</td>
<td>0.0%</td>
</tr>
<tr>
<td>result 5 of Type 3</td>
<td>2</td>
<td>30</td>
<td>4.0%</td>
<td>7.0%</td>
</tr>
<tr>
<td>result 6 of Type 3</td>
<td>3</td>
<td>30</td>
<td>7.0%</td>
<td>0.0%</td>
</tr>
<tr>
<td>result 7 of Type 3</td>
<td>4</td>
<td>30</td>
<td>9.5%</td>
<td>0.0%</td>
</tr>
<tr>
<td>result 8 of Type 3</td>
<td>5</td>
<td>30</td>
<td>15.2%</td>
<td>0.0%</td>
</tr>
<tr>
<td>result 9 of Type 3</td>
<td>2</td>
<td>60</td>
<td>7.0%</td>
<td>5.0%</td>
</tr>
<tr>
<td>result 10 of Type 3</td>
<td>3</td>
<td>60</td>
<td>12.0%</td>
<td>2.0%</td>
</tr>
<tr>
<td>result 11 of Type 3</td>
<td>4</td>
<td>60</td>
<td>14.2%</td>
<td>0.0%</td>
</tr>
<tr>
<td>result 12 of Type 3</td>
<td>5</td>
<td>60</td>
<td>20.5%</td>
<td>0.0%</td>
</tr>
</tbody>
</table>
5. Consideration
Features of developed algorithm in comparison with Simulation Optimization

It had been an established theory that simulation method can evaluate a solution but can not generate a solution (Harrell and Tumay 1994). However, because of recent rapid progress of optimization method and computer power, the theory has changed. Simulation method is an indispensable factor when solving a complex optimization problem using the structure shown in Figure 7 (Barretto, et al. 1999). Genetic Algorithm (Goldberg 1989) and Simulated Annealing Algorithm (Kirkpatrick, et al., 1983) are typically used in optimization procedure. Now, it can be said that simulator can answer not only "what if" question but also "How to" question (Azadivar 1992).

Fig. 7 Simulation optimization method

The conditions that the molten metal logistic simulator should satisfy are shown in formulae (1) and (2). These conditions can be converted into an optimization problem shown in formula (5).

Minimize \( f(x) \)
subject to \( N_2(x, t) > 3 \) for any \( t \) (5)

where
\( x \): a set of torpedo cars each of which initial schedule contains de-P pretreatment process
\( f(x) \): a number of elements of set \( x \)
\( N_2(x, t) \): a number of torpedo cars at reloading pits at time \( t \) when dephosphorization pretreatment process of any torpedo car that belongs to \( x \) is omitted

The result in subsection 4.3 shows that Type 3 simulator can decide a torpedo set \( x \). The structure of Type 3 simulator developed is shown in Figure 8. At time \( t \), this system predicts the logistic state at time \( t + n^* \Delta t \) and modifies the given schedule after \( t \). As a result, the simulator derives feasible solution that is as good as the real operator's decision. Implemented algorithm in the simulator is quite simple in comparison with SA and GA and the modification of given scheduling is more restrictive. So, it takes only 5 minutes to simulate the molten metal logistics for one month using PC with 1.5 GHz CPU. The reason why such a simple algorithm is effective for the molten metal logistics problem is that the searching space is not so vast since the scheduling problem was alternative one and the omission ratio is low. It is important to select a properly simple / complex algorithm in respect of the complexity of the given problem.

Fig. 8 Function of Type 3 simulator as a scheduler

6. Conclusion

We developed a simulator for a molten metal logistics that can closely imitate an actual operation. The simulator has a function to modify given torpedo schedules utilizing state prediction. It was difficult for us to find a suitable logistic simulator in a market that can express the molten metal logistics using only its basic functions. The most important reason is the lack of a prediction-based logistic control function shown in this paper. There remain numerous simulation problems that needs such a logistic control function in real-world industries. So, we would like to refine and apply a prediction-based simulation in future.

REFERENCES


