# Utility and Stability Measures for Agent-Based Dynamic Scheduling of Steel Continuous Casting*

D. Ouelhadj[1], P. I. Cowling[2], S. Petrovic[3]

[1,3] Automated Scheduling, Optimisation and Planning Research Group, School of Computer Science and IT, University of Nottingham, Jubilee Campus, Nottingham, NG8 1BB, UK.
[2] Modelling Optimisation Scheduling and Intelligent Computing Research Group, Department of Computing, University of Bradford, Bradford BD7 1DP,UK.

*Abstract* - **This paper describes a new model for robust predictive/reactive scheduling of steel continuous casting based on the use of multi-agents, tabu search and heuristic approaches. A continuous caster agent generates a predictive production schedule taking into account manufacturing requirements and local constraints using tabu search. The predictive schedule is modified so as to minimise deviation between the performance measure values of the realised and predictive schedules in order to react to real-time events. We propose several schedule-repair and complete reschedule strategies to handle the real-time events, evaluate and compare their performance. The decision as to whether to locally repair the schedule or reschedule from scratch (complete reschedule) is based on three measures: robustness, utility and stability. Utility measures the change in schedule objective following schedule revision. Stability measures the deviation from the original schedule caused by schedule revision to quantify the undesirability of making large changes to the initial predictive schedule unless absolutely necessary. Robustness combines the utility and stability measures. In order to investigate the performance of these measures and strategies, simulation experiments were carried out and results are reported**.

**Key words**: steel production, continuous caster, multi-agents, dynamic scheduling, reactive scheduling.

## 1. INTRODUCTION

Continuous casting is the process whereby molten steel is cast into slabs for subsequent rolling into coils in the hot strip mill. Various static scheduling techniques of the continuous caster have been reported in the literature. They may be classified into two categories: mathematical programming methods and artificial intelligence methods. Tang et al. [15] proposed a mathematical programming model considering both punctual delivery and production operation continuity. A knapsack-constrained travelling salesmen scheduling model was implemented for LTV steel's Cleveland Works' Twin stand continuous slab caster [2][12]. A complex mixed integer linear programming model was developed and solved using heuristic techniques for twin strand continuous caster at LTV and Geneva steel works [12]. In the case of artificial intelligence methods, Numao [13], Epp et al. [10] developed expert systems to perform cooperative scheduling of the continuous caster. Dorn et al. [8] compared several iterative methods including tabu search, simulated annealing and genetic algorithms for the continuous caster scheduling. Lee et al. [12] developed a scheduling system for the two twin-strand casters at LTV using fast genetic algorithms to obtain satisfactory solutions.

Most of the literature on continuous caster scheduling considers static problems. However, these scheduling systems may not be effective in a dynamic manufacturing environment where a variety of real-time events can disturb the schedule. Dynamic scheduling is an important issue in steel production. Few research works have addressed the problem of dynamic scheduling of the continuous caster. Tang et al. [16] in its overview on integrated scheduling of the continuous caster and the hot strip mill addressed the importance of dynamic scheduling in steel production. Cowling and Johanson [6] addressed as well the importance of dynamic scheduling in steel continuous casting. Dorn et al. [9] studied the dynamic scheduling of steel production using fuzzy reasoning to represent and propagate schedule uncertainty within the context of scheduling the operations of a continuous caster.

The main alternatives to react to the presence of real-time events are either to generate a new schedule from scratch (complete reschedule) or by making alterations to the previous schedule (schedule repair) [6][9][14]. Complete reschedule is rarely achievable in practice because frequent schedule regeneration can result in instability on the shop floor. Schedule repair approaches are often more appropriate. Robustness is a desirable attribute of a predictive schedule as it focuses on minimising the effects of disruptions on the performance

---

[1] dxs@cs.nott.ac.uk, Tel: +44 (0)115 846 6521. Fax: +44 (0)115 951 4254.
[2] Peter.Cowling@scm.brad.ac.uk, Tel: +44 (0)1274 234005. Fax: +44 (0)1274 234054.
[3] sxp@cs.nott.ac.uk, Tel: +44 (0)115 951 4222. Fax: +44 (0)115 951 4249.

*Research supported by EPSRC grant number GR/N04225/01, DASH associates and a committee of vice chancellors and Principals ORS award.

measures. In the robust scheduling approach, the predictive schedule is built using available information on the disruptions that are likely to occur during execution of the schedule to minimise deviation between the performance measure values of the realised and predictive schedules. Abumazair and Svestka [1] compared three strategies for rescheduling the affected operations in a job shop with respect to measures of efficiency and stability. The scheduling objective is to maximise shop efficiency, and at the same time minimise system impact caused by schedule changes. Cowling and Johansson [6] proposed two measures, utility and stability, to decide whether to repair a schedule or reschedule from scratch, and surveyed rescheduling and schedule-repair techniques.

The research presented in this paper is a further development of previous work concerning the integration of the dynamic scheduling of the continuous caster and the hot strip mill in steel production. It focuses on dynamic robust scheduling of the continuous caster to respond to the presence of real-time information. Section 2 briefly describes the multi-agent architecture proposed in a previous work on the integration of the scheduling systems of the continuous caster and the hot strip mill. Section 3 presents the mathematical model and the heuristic methods developed for continuous caster predictive scheduling. Section 4 elaborates the predictive-reactive scheduling of the continuous caster, more precisely the utility and stability measures, and the rescheduling strategies. The experimental results are presented in Section 5. Conclusions are presented in Section 6.

## 2. AN AGENT_BASED SYSTEM FOR INTEGRATED DYNAMIC SCHEDULING OF STEEL PRODUCTION

Steel production involves a variety of processes [4][12] including continues casters, hot strip mills and furnaces. The continuous caster process casts the steel into slabs with different widths and chemical composition. The hot strip mill processes these slabs in order to produce the final product, steel coils. Due to the complexity of the steel production environment involving multiple distributed production processes, we proposed an agent architecture for integrated dynamic scheduling of the hot strip mill and the continuous caster [5][7]. Each of these processes is represented by an agent, including the continuous caster agent, hot strip mill agent, the slabyard agent and the user agent. The hot strip mill agent performs the robust predictive/reactive scheduling of the hot strip mill. The continuous caster agent generates the robust predictive/reactive schedules of the continuous caster. The slabyard agent is responsible for the management of the slabs produced by the continuous caster agent. The slabyard agent communicates with the continuous caster agent to convey details of the slabs requested by the hot strip mill agent, and maintains information on slabs already produced which are currently cooling down. The user agent provides the user interface to the system. It manages and announces the orders to produce, and deals with dynamic changes of order conditions. The agents cooperate using the contract net protocol with commitment duration. The communication is realised by the exchange of asynchronous messages between the agents defined using XML. In this paper we focus upon the dynamic scheduling of the continuous caster.

## 3. CONTINUOUS CASTER PREDICTIVE SCHEDULING

### *3.1. The mathematical model*

The scheduling problem of the continuous caster is to define a sequence of heats, each of which corresponds to a fixed tonnage of molten steel. Each heat is made for a group of orders, which can be made using compatible chemical composition (grade). Each heat has a fixed capacity (50 tonnes in the case we consider here), and takes about one hour to produce approximately 4 slabs of similar and the same grade. The scheduling system of the continuous caster presents widely different constraints to those of the hot strip mill [2][4][12]:

- Grade compatibility constraints: slabs in the same heat will have similar width and the same chemical grade. Generally each order can be made at one of several chemical grades.
- Width compatibility constraints: there must be only small changes in width between slabs of the same heat, and between two consecutive heats.
- Time constraints: the slabs required by the hot strip mill should be provided on time.

We have modelled the problem of sequencing slabs using a mixed integer-programming model and solved it using constrained bin-packing heuristics and tabu search. These models capture the most important scheduling constraints and objectives, and could be further extended, for example to consider other chemical constituents or inter-heat chemical compatibility. The continuous caster scheduling model can be stated as: given *n* slabs of varying chemistry, width and weight, schedule them in a minimum number *m* of heats, such that the total weight of each heat is no more than the heat capacity. All heats have the same capacity, and the steel from each heat, which does not go towards orders, is made to stock, which is undesirable.

Let $Max\Phi$ be the maximum weight of the molten steel in the heat, and $\Phi_j$ be the weight of slab$_j$. Define: ($MinC_j$, $MaxC_j$) are the minimum and maximum carbon contents of slab$_j$, ($MinAL_j$, $MaxAL_j$) as the minimum and maximum aluminium contents of slab$_j$, and ($MinW_j$, $MaxW_j$) are the maximum and minimum widths of slab$_j$.

Let us consider the following variables:

$$\delta_{ij} = \begin{cases} 1, \ if \ \ slab_j \ \ is \ assigned \ \ to \ heat_i \\ 0, Otherwise \end{cases}$$

Where $i \in \{1, 2, \ldots, m\}$ and $j \in \{1, 2, \ldots, n\}$.
$ETP_{ij}$ is the earliness or tardiness delivery penalty to produce $slab_j$ in $heat_i$, when the hot strip mill's requirements are considered. These $ETP_{ij}$ are important to the integration of the continuous caster and the hot strip mill scheduling process. Lateness and earliness are penalised according to the amount of lateness and earliness, with lateness receiving a higher penalty.

Because of the high fixed cost of producing a heat and the time window constraints, we want to minimise the number of heats and the earliness and tardiness delivery penalties. The economic impact of these two effects are expressed using a weighted sum:

$$\min \sum_{i=1}^{m} \sum_{j=1}^{n} ETP_{ij} \, \delta_{ij} \qquad (1)$$

Subject to the following constraints:

- The total weight packed into a heat must not exceed its capacity (unused capacity is made to stock),

$$\sum_{j=1}^{n} \Phi_j \, \delta_{ij} \le Max\Phi, \quad i \in \{1,2,\cdots,m\} \qquad (2)$$

- Each slab must be assigned to at most one heat,

$$\sum_{i=1}^{m} \delta_{ij} \le 1 \ , \ j \in \{1,2,\ldots, \ n\} \qquad (3)$$

- Width incompatibility,

$$\begin{aligned} &If \quad MaxW_k - MinW_j < 0 \qquad (4) \\ &or \quad MaxW_j - MinW_k < 0 \\ &then \quad \delta_{ik} + \delta_{i'j} \le 1 \end{aligned}$$

- Grade incompatibility

$$\begin{aligned} &If \quad MaxAL_k - MinAL_j < 0 \qquad (5) \\ &or \quad MaxAL_j - MinAL_k < 0 \\ &then \quad \delta_{ik} + \delta_{ij} \le 1 \end{aligned}$$

$$\begin{aligned} &If \quad MaxC_k - MinC_j < 0 \qquad (6) \\ &or \quad MaxC_j - MinC_k < 0 \\ &then \quad \delta_{ik} + \delta_{ij} \le 1 \end{aligned}$$

### *3.2. Tabu search to schedule casters*

To resolve the continuous caster scheduling problem, modelled using the constrained bin-packing model given above, we used a tabu search meta-heuristic [11].

- **Initial solution:** to find the initial solution, we propose two heuristics used by the continuous caster agent, Constrained First Fit (*CFF*) and Constrained Best Fit (*CBF*), based on the FF and the BF heuristics of classical bin-packing problem [3]. These two heuristics first sort the slabs in increasing order of their production date requested by the hot strip mill and then assign them to the heats taking into account compatibility constraints. In each step of *CFF*, the current slab is added to the partially filled heat with the smallest index that has sufficient residual capacity, subject to compatibility constraints. If no such heat is available, the slab is added into a new heat. In each step of *CBF*, the current slab is assigned to the partially filled heat that has the smallest remaining capacity into which it will fit, subject to compatibility constraints. In absence of such a heat, a new one is used.
- **Neighbourhood structure**: the neighbourhood of a current solution is obtained by two moves: shift move and swap move. For each slab, we consider all possible moves. Shift move shifts a slab *k* from heat *i* to heat *j*. Swap move swaps slab *k* in heat *i* with slab *l* in heat *j*.
- **Tabu list**: in order to prevent the procedure cyclically revisiting solutions which have been generated in earlier iterations, certain moves must be forbidden to a fixed number of iterations, chosen randomly in the range [6, 13]. A tabu-list is used to temporarily store those attributes which describes the move. Whenever a shift move is performed, the attribute (*j*, *h*) is set tabu to prevent slab *j* returning to heat *h*. For each applied swap (*j*, *h*)-(*i*, *k*) move, the attributes (*j*, *h*) and (*i*, *k*) are set tabu.
- **Aspiration criteria**: a move can be released from the tabu condition if it has a solution value better than the best solution found by the search until that moment.
- **Stopping criteria**: it is a specified number of iterations.

## 4. ROBUST PREDICTIVE-REACTIVE SCHEDULING

The continuous caster production is often affected by various real-time events, such as: failure to produce the required chemical composition of a heat, heats in the wrong sequence, etc. We proposed three measures utility, stability, and robustness, which measure the effect of real-time information on the schedule. These measures allow the continuous caster agent to choose the best strategy to react to the real-time events. If we have a range of good techniques for repairing and rescheduling in the presence of real-time information, we must still address the importance issue of whether to repair or reschedule and which schedule repair strategy should be used in response to real-time information.

### *4.1. Utility and stability measures*

Utility measures the improvement in the original schedule objectives (1) due to schedule revision, and stability measures the deviation from the original schedule caused by schedule revision. The utility is the difference between the value of the objective function $F'$ (1) of the new schedule $S'$ after taking into account the real-time information $E$, and the objective function $F$ of the schedule $S$ before taking into account real-time information. Where $t$ is the time of arrival of information, which may be highly significant since information known earlier may generally be handled more effectively. The utility is expressed by:

$$Utility(S, S', E, t) = F - F' \quad (7)$$

The stability is expressed by the weighted sum of the absolute difference between the original starting production time $T_i$ of each slab$_i$ in the heats of the original schedule, and the new starting production time $T_i'$ after the occurrence of the real-time event for each slab in the heats, and the stability effect of adding new heats. It is expressed by:

$$Stability(S, S', E, t) = \sum_{i=1}^{n} \left| T_i - T_i' \right| + \Delta H \, Q_s \quad (8)$$

$Q_s$ is the stability effect of adding a single heat = $m$.

$\Delta H$ is the change in the total number of heats in the schedule.

The robustness measure of a schedule combines the maximisation of the *utility* and the minimisation of the *stability*. The robustness is expressed by the maximisation of the weighted objective function expressed by:

$$Robustness(S, S', t) = \alpha \; Utility - (1 - \alpha) \; Stability \quad (9)$$

$\alpha$ is a real valued weight in [0, 1].

### *4.2. Rescheduling strategies*

To deal with real-time events, we proposed several rescheduling strategies:

- **Insert- at- end schedule repair (IESR)**: keeps the existing schedule and creates additional heats at the end of the caster schedule to house the non-compliant slabs (fig. 1).

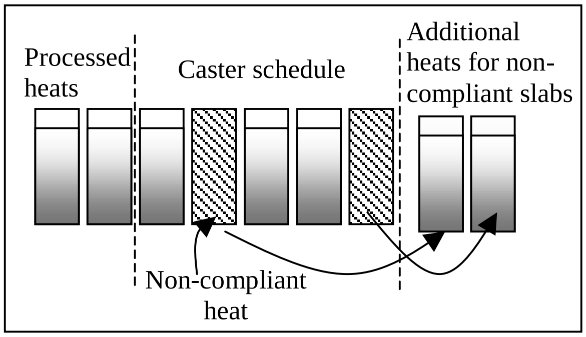

Fig.1 Insert at end schedule repair

- **Insert-heat schedule repair (IHSR)**: inserts a new heat after the one with non-compliant slabs in the schedule, in which the non-compliant slabs will be made (fig. 2).

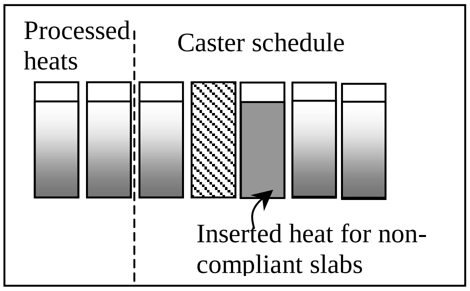

Fig. 2 Insert heat schedule repair

- **Shift schedule repair (SHSR)**: shifts the non-compliant slabs to the best heats of the current schedule so as to maximise robustness (fig. 3). If the non-compliant slabs cannot be scheduled on the current heats, additional heats are created at the end of the schedule.
- **Swap schedule repair (SWSR)**: swaps the non-compliant slabs with the best slabs within the heats of the current schedule so as to maximise robustness. If the non-compliant slabs cannot be scheduled on the current heats, additional heats are created at the end of the schedule (fig. 4).

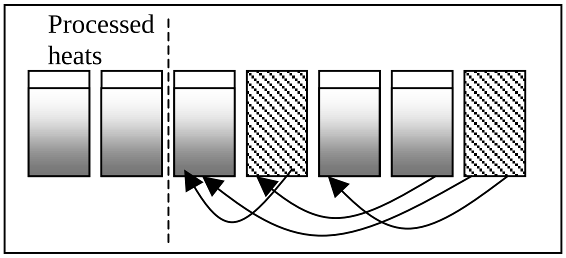

Fig. 3 Shift schedule repair

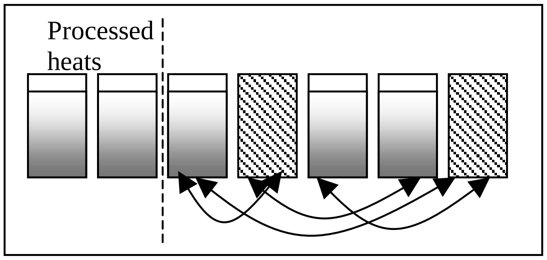

Fig. 4 Swap schedule repair

- **Hybrid-shift-swap schedule repair (HBSR)**: swaps or shifts the non-compliant slabs within the current schedule so as to maximise robustness using tabu search.

- **Complete reschedule (CR)**: reschedules the sequence of the remaining scheduled slabs on the current sequence of heats, and create additional heats if necessary to house required slabs. To reschedule the remaining scheduled slabs so as to maximise robustness, we use tabu search starting from an initial solution defined by *CBF* heuristic.

## 5. EXPERIMENTAL RESULTS

We carried out 5 runs for various instances of data obtained from a steel manufacturer (90 slabs, 120 slabs, and 145 slabs) [4]. For each run, 5 real-time events were generated. Each real-time event specifies up to 5 heats with the wrong aluminium and carbon composition that affects up to 4 slabs per heat. To generate random real-time events, the continuous caster agent uses a normal distribution of carbon and aluminium content. The first event is generated at 10% of the caster schedule, and the next events every 20% thereafter. On the occurrence of each real-time event, the continuous caster agent evaluates the robustness measures of the different schedule-repair and complete reschedule strategies for different values of $\alpha$ = (0, 0.01, 0.25, 0.50, 0.75, 0.95, 1). The strategy, which maximises the robustness measure, is applied to adjust the schedule. Fig. 5 shows the average frequency of each strategy applied, in response to real-time events. The results demonstrate that when we tolerate significant changes in stability and we wish to maintain a good utility *hybrid-swap-shift schedule repair* and *complete reschedule* have a higher frequency compare to the other strategies. Furthermore, *hybrid-swap-shift schedule repair* is the most dominant strategy. When we wish to maintain a good stability *hybrid-swap-shift schedule repair*, *swap schedule repair* and *shift schedule repair* are the most dominant strategies. However, we remark that the *complete reschedule* strategy generally appears after a large number of real-time events have disturbed the schedule.

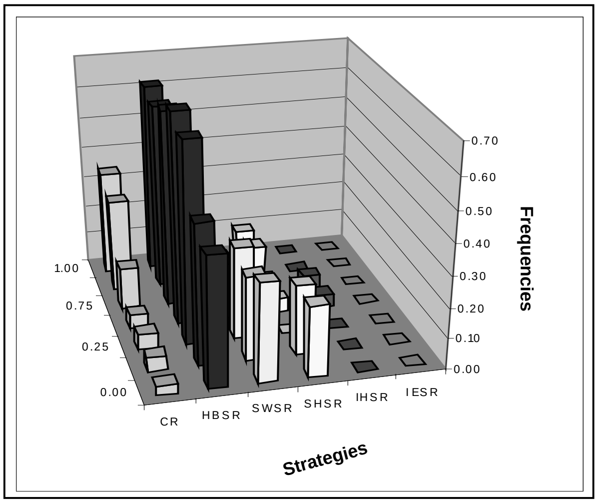

Fig. 5 Average frequency of the different strategies

Fig. 6 summarises our results on the performance of the utility and stability measures of each strategy for different values of $\alpha$. The results demonstrate that in an environment where stability should be maintained (0, 0.01, 0.25, 0.50), the schedule repair strategies *hybrid-swap-shift schedule repair*, *swap schedule repair* and *shift schedule repair* give the best stability measures compared to *complete reschedule*, *insert heat schedule repair* and *insert at end schedule repair* due to the number of heats added. In an environment where importance in given to utility the ($\alpha$=0.75, 0.95 and 1), *hybrid-swap-shift schedule repair* and *complete reschedule* yield better utility values than the other schedule repair strategies, due to the fact that *swap schedule repair*, *shift schedule repair*, *insert heat schedule repair* and *insert at end schedule repair* can cause unnecessary stock steel if they do not manage to just swap or just shift some of the non-compliant slabs on the heats of the caster schedule. *Hybrid-swap-shift schedule repair* outperforms all the rescheduling strategies in terms of both utility and stability measures. Similar results were obtained with the schedule repair and complete reschedule strategies developed for the predictive/reactive scheduling of the hot strip mill.

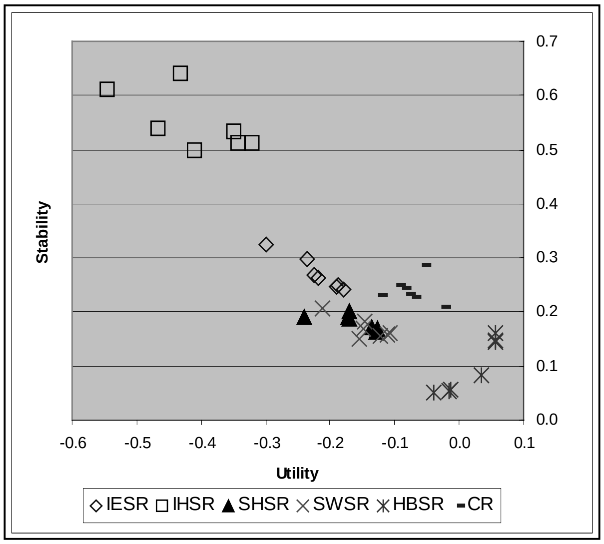

Fig. 6 Performance of the utility and stability measures

## 6. CONCLUSION

This paper has presented a new model for robust predictive/reactive scheduling of the continuous caster, which is part of our multi-agent architecture for integrated dynamic scheduling of the continuous caster and the hot strip mill. In the multi-agent system the continuous caster agent generates predictive schedules using tabu search. For reactive scheduling, we proposed different measures to evaluate the effect of real-time events on the predictive schedule, and use these measures to make the decision whether to proceed with a schedule-repair or complete reschedule strategy to react to the real-time events. In addition, we have investigated various new schedule-repair and complete reschedule strategies and compared their performance with respect to utility, stability and robustness measures. The experimental results obtained in simulation indicate that in an environment where it is desirable to maintain the stability of the system schedule repair strategies are more efficient in generating robust schedules. Even in an environment where we tolerate significant changes in stability and require improvements in utility, schedule repair strategies remain competitive with complete reschedule.

## 7. REFERENCES

[1] R. J. Abumaizar and J. A. Svestka, Rescheduling job shops under random disruptions, *International Journal of Production Research*, vol. 35, no. 7, 1998, pp. 2065-2082.

[2] R. E. Box and D. G. Herbe, A scheduling model for LTV steel's Cleveland work's twin-strand continuous slab caster, *Interface*, vol.18, no. 1, 1988, pp.42-56.

[3] E. G. Coffman, M. R. Garey and D. Johnson, Approximation algorithms for bin-packing: a survey, Approximation Algorithms for NP-hard problems, D. Hochbaum (ed.), PWS Publishing, 1997, pp. 46-93.

[4] P. I. Cowling, W. Rezig, Integration of Continuous Caster and Hot Strip Mill Planning for Steel Production, Journal of Scheduling, vol. 3, no. 4, 2000, pp. 185-208

[5] P. I. Cowling, D. Ouelhadj and S. Petrovic, A multi-agent architecture for dynamic scheduling of steel hot rolling, in Proceeding of WMC '2001, Rochester, NY.

[6] P. I. Cowling and M. Johansson, Using real-time information for effective dynamic scheduling, *European Journal of Operational Research*, vol. 139, no. 2, 2001, pp. 230-244.

[7] P. I. Cowling, D. Ouelhadj and S. Petrovic, A multi-agent architecture for dynamic scheduling of steel hot rolling, to appear in the Journal of Intelligent Manufacturing, 2003.

[8] J. Dorn, M. Girsch, G. Skele, and W. Slany, Comparison of Iterative Improvement Techniques for Schedule Optimization, *European Journal on Operations Research*, vol. 94, no. 2, 1996, pp. 349-361.

[9] J. Dorn, Roger Kerr and Gabi Thalhammer, Reactive scheduling: improving the robustness of schedules and restricting the effects of shop floor disturbances by fuzzy reasoning, International Journal of Human Computer Studies, vol. 42, 1995, pp. 687-704.

[10] H. P. Epp, M. G. Kalin, D. P. Miller, K. R. Karwan and J. R. Sweigart, An interactive, adaptive, real-time scheduler for steel-making, in proceedings of the Third International Conference of Expert Systems and the Leading Edge in Production and Operations management, Hilton Heat Island, SC, USA, 1989, pp. 495-504.

[11] F. Glover, Tabu search, Fred Glover, Manuel Laguna. Boston: Kluwer Academic Publishers, 1997.

[12] H. S. Lee, S. S. Murthy, S. W. Haider and D. V. Morse, Primary production scheduling at steel making industries, *IBM Journal Research Development*, vol. 40, no. 2, 1996, pp. 231-252.

[13] M. Numao, Development of Cooperative Scheduling System for the Steel-Making Process, Eds. *Intelligent Scheduling*, Morgan Kaufman, 1994, pp. 607-628.

[14] J. Sun and D. Xue, A dynamic reactive scheduling mechanism for responding to changes of production orders and manufacturing resources, *Computers in Industry*, vol. 46, 2002, pp.189-207.

[15] J. L. Tang, A. Rong and Z. Yang, A mathematical programming model for scheduling steel-making-continuous casting production, *European Journal of Operational Research*, vol. 120, no. 1, 2000, pp. 423-435.

[16] Tang, J. L., A. Rong and Z. Yang, A review of planning and scheduling systems and methods for integrated steel production, *European Journal of Operational Research*, vol. 133, no.1, 2001, pp.1-20.