

Contract Net Protocol for Cooperative Optimisation and Dynamic Scheduling of Steel Production

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

^{1,3} Automated Scheduling, Optimisation and Planning Research Group, School of Computer Science and IT, University of Nottingham, Nottingham NG8 1BB, UK,
¹Email: dxs@cs.nott.ac.uk, ³Email: sxp@cs.nott.ac.uk.

²Modelling Optimisation Scheduling and Intelligent Computing Research Group, Department of Computing, University of Bradford, Bradford BD7 1DP, UK,
Email: P.I.Cowling@Bradford.ac.uk.

Abstract. This paper describes a negotiation protocol proposed for inter-agent cooperation in a multi-agent system we developed for optimisation and dynamic integrated scheduling of steel production. The negotiation protocol is a two-level bidding mechanism based on the contract net protocol. The purpose of this protocol is to allow the agents to cooperate and coordinate their actions in order to find globally near-optimal robust schedules, which are able to optimise the original production goals whilst minimising the disruption caused by the occurrence of unexpected real-time events. Experimental results show the performance of this negotiation protocol to coordinate the agents in generating good quality robust schedules. This performance is evaluated in terms of stability and utility measures used to evaluate the robustness of the steel production processes in the presence of real-time events.

Keywords. steel production, multi-agents, cooperation and coordination, contract net protocol, dynamic scheduling

1 Introduction

Steel production involves a variety of processes [1, 2] including continuous casters, hot strip mills and furnaces. Continuous casters cast the molten steel into slabs with different widths and chemical composition. The hot strip mill rolls these slabs in order to produce the output of the primary steelmaking production process, steel coils. The scheduling systems of the caster and the hot strip mill have very different objectives and constraints and operate in an environment where there is a substantial quantity of real-time information concerning production failures and customer requests. Note that there are usually several continuous casters

that can supply a single hot strip mill with slabs. Due to the complexity of the steel production environment involving multiple distributed production processes, in our previous work, we proposed a multi-agent architecture for integrated dynamic scheduling of the hot strip mill and continuous casters [3, 4]. Each steel production process is represented by an agent, including the continuous caster agents (CCA), the hot strip mill agent (HSMA), the slabyard agent (SYA) and the user agent (UA). The HSMA performs the robust scheduling of the hot strip mill. The CCA performs the robust scheduling of the continuous caster. Robust scheduling focuses on building predictive/reactive schedules to minimise the effects of disruption while optimising some measures of performance for the realised schedule. The SYA communicates with the CCA(s) to convey details of the slabs requested by the HSMA, and maintains information on slabs already produced which are currently cooling down. The UA manages and announces the orders to produce, and allows outside input of data for dynamic changes of order conditions. These agents communicate by exchanging asynchronous messages formatted in XML (Extensible Mark-up Language). An experimental prototype was developed as a multi-thread application in Microsoft Visual C++/MFC.

In order for the agents to find global feasible schedules, the agents must cooperate and coordinate their local actions. The most widely used cooperation and coordination method is the Contract-Net Protocol (CNP). The CNP is a high level protocol for achieving efficient cooperation introduced by Smith [13] based on a market-like protocol. The CNP has been extensively used for inter-agent cooperation in dynamic production scheduling. Yams [9] is one of the earliest agent-based manufacturing system where part agents negotiate with resource agents to assign tasks to the resource agents using the CNP. Shaw [11] developed a dynamic scheduling system in a cellular manufacturing system, where the manufacturing cell agent could sub-contract work to other cells through a bidding mechanism to schedule the tasks. Lin and Solberg [5] used the CNP for inter-agent cooperation in a shop floor. In their model part agents enter the system with certain currency and negotiate with resource agents via a bidding mechanism. When a resource agent is in failure, it informs the corresponding part agent, and the latter proceeds to a renegotiation process on the operations in failure with the resource agents. Sousa and Ramos [14] used the CNP for dynamic scheduling in manufacturing systems. The task agents negotiate the operations of the task with the resource agents using the CNP. When a resource agent detects a malfunction, it sends a machine fault message to the task agents that have contracted its operations. The task agents renegotiate the operations in failure with other resource agents capable of performing the operations. Ouelhadj et al. [7] proposed a multi-agent autonomous architecture for dynamic scheduling in flexible manufacturing systems where resources agents negotiate using the CNP for a global schedule. Following the occurrence of real-time events, the resource agent of the resource in failure sends a failure message to the contractor resource agents specifying the operations in failure. On receiving the message, the corresponding resource agents renegotiate the operations in failure with other resource agents. Maturana and Norrie [6] proposed a mediator architecture, Metaphor I, for dynamic scheduling of virtual enterprises combining mediation and sub-tasking using the CNP. Mediator agents

are used to coordinate the resource agents using the CNP. A resource breakdown is simulated by introducing a breakdown period into the resource. Each job allocated within the halt-period is rescheduled to other available time slots found in the malfunctioning resource) or in a different resource. The same idea was adopted by Shen et al. [12] with Metaphor II, for integrating the manufacturing enterprise's activities. The manufacturing resource agents are coordinated by appropriate mediators by combining the mediation mechanism and the CNP. They defined several rescheduling mechanisms for different real-time events such as: rush orders, order cancellation, machine breakdown, etc. Recently leveled commitment contracts were proposed as an extension of the CNP for increasing the economic efficiency of contracts between self-interested agents in the presence of incomplete information about future events. Sandholm [10] described a leveled commitment contracting protocol for automated contracting in distributed manufacturing by giving the possibility for each agent to decommit from the contract by simply paying a decommitment penalty to the other contract party.

The research presented in this paper extends our previous work concerning the integration of the dynamic scheduling of the continuous caster and the hot strip mill in steel production using multi-agent systems. It focuses on inter-agent cooperation for optimal dynamic robust scheduling of steel production. Section 2 describes the negotiation protocol developed for cooperative optimisation and robust scheduling. Section 3 presents the cooperative robust scheduling in the presence of real-time events. The experimental results are presented in Section 4. Conclusions are presented in Section 5.

2 The negotiation protocol for cooperative optimisation and robust scheduling

The HSMA and CCA(s) present different objectives and constraints and no agent possesses a global view of the entire agency, therefore cooperation is essential to allow the agents to adjust their local schedules in order to achieve global objectives and to react to the presence of real-time events. In this paper, we propose a negotiation protocol based on the contract net protocol for inter-agent cooperation. The negotiation protocol is a two-level bidding mechanism involving negotiation at HSMA-SYA level and SYA-CCA(s) level. At the HSMA-SYA negotiation level, the HSMA requests the supply of slabs from the SYA. At the SYA-CCA (s) negotiation level, the SYA requests the production of slabs not available in the slabyard from the CCA(s). In the CNP no agent takes any responsibilities before reaching mutual agreements on the tasks, which can cause the degradation of the global coordination performance. By attaching the commitment duration to the negotiation messages, the performance of the CNP is improved especially for dynamic environments where desired tasks and available resources may be continuously changing. The commitment duration specifies the time windows by which the agents must respond to a given negotiation message. The negotiation protocol is a three-step process, which begins after the session has been initiated by the

UA, which sends a request message to the HSMA to produce a collection of coil orders. The order describes the grade (carbon and aluminium composition) and physical (weight, width, length, and thickness) properties of the coils and their due dates. The steps of the negotiation protocol are described above.

2.1 Task announcement

Upon receiving the request message from the UA, the HSMA generates a schedule of coils which satisfies the hot strip mill's constraints and maximises its objective function. Then, negotiation at the HSMA-SYA level starts. The HSMA issues an announcement message (HSMA-announcement) to the SYA to supply the slabs necessary to produce the coils, which have been scheduled. Due to hot strip mill hard constraints, the HSMA generates a number of sub-schedules (coffins) [4], and announces each sub-schedule separately to the SYA. The HSMA-announcement message describes the following information:

Sender: HSMA,

Receiver: SYA,

Type: task announcement,

Task specification: the list of slabs to produce with the description of their grade, dimensions, and the requested production ready time,

Bid-reception deadline: the time by which the SYA must respond with a bid.

Upon receiving the HSMA-announcement message, the SYA analyses the announcement and negotiation at the SYA-CCA level starts to identify the CCA to produce the non-available slabs in the slabyard. The SYA sends the SYA-announcement message to the CCA(s) to request the casting of slabs, which are not currently available in the slabyard. The SYA-announcement message describes the following information:

Sender: SYA,

Receiver: CCA,

Type: task announcement,

Task specification: the list of slabs requested by the HSMA which are currently not available in the slabyard with the description of their grade, dimensions, and the requested production ready time,

Bid-reception deadline: the time by which the CCA must respond with a bid.

2.2 Bidding

A bid represents an offer to execute the task specified in the SYA-announcement concerning the production of the slabs non-available in the slabyard. Each CCA will inspect the SYA-announcement message and will decide whether or not it should respond with a bid, considering the quality of the caster schedule resulting

from adding the requested slabs to its current engagements. If it chooses to respond, it will develop its locally optimised scheduling and sends a CC-bid message to the SYA. The CC-bid message describes the following information:

Sender: CCA,

Receiver: SYA,

Type: bid,

Bid specification: the slabs to produce with the proposed production dates,

Bid-acceptance deadline: the time by which the SYA must respond with a contract.

After receiving the bids before the bid-reception deadline, the SYA evaluates the CCA-bids and selects the best bid based on the earliest production date. The SYA sends a SYA-bid to the HSMA with the information about the available slabs in the slabyard and production dates of the slabs to be produced by the CCA(s) proposed in the selected bid. This bid is used at the HSMA-SYA negotiation level. The SYA-bid message describes:

Sender: SYA,

Receiver: HSMA,

Type: bid,

Bid specification: slabs available in the slabyard plus the slabs from the accepted CCA bid with the proposed production dates,

Bid-acceptance deadline: the time by which the HSMA must respond with a contract.

2.3 Contracting

After the HSMA has received the bid from the SYA, it can respond before the bid-acceptance deadline with the following alternatives:

- Accept the bid and sends the award message to the SYA. The SYA on its turn sends the award message to the appropriate CCA before the bid-acceptance deadline. The production of slabs then can start.
- Accept a subset of the slabs in the bid and renegotiates the production of slabs with unsatisfied production dates. The HSMA restarts a re-negotiation session with a new announcement message, which specifies the slabs with unsatisfied production dates and an additional allowance tolerance on the width. Width tolerance is a relaxation on the width hard constraint, since a coil can be made from slabs having a variety of different physical dimensions. The renegotiation process iterates cyclically until the HSMA states that the solution matches the requirements of its schedule within acceptable tolerances.

A failure to send a contract message before the bid-accept deadline means the HSMA or the SYA is rejecting the bid.

3 Cooperative robust rescheduling

In the proposed multi-agent architecture, both the hot strip mill and the continuous caster are subject to various real-time events. Among the most frequent real-time events that can occur on the continuous caster are heat of molten steel may arrive with wrong chemical composition. These real-time events can affect the hot strip mill production process by causing delayed delivery of slabs or failure to meet prescribed quality control standards of the slabs.

On the occurrence of real-time events, the CCA and the HSMA react locally to the real-time events and cooperate in order to define a globally feasible robust schedule. In order for the CCA and the HSMA to react locally to real-time events, we defined several domain-specific rescheduling strategies, as well as utility, stability and robustness measures. The CCA and the HSMA reschedule so as to maximise robustness. Robustness combines the maximisation of utility and the minimisation of stability. Utility measures the change in the value of the schedule objective function following the schedule revision. It is expressed by the difference between the value of the objective function of the new schedule after reacting into the real-time events and the objective function of the predictive schedule before taking into account real-time events. Stability measures the deviation from the original predictive schedule caused by schedule revision to quantify the undesirability of making large changes to the initial schedule unless absolutely necessary. For the CCA the stability is expressed by the weighted sum of the absolute difference between the original starting production time of each slab in the original schedule, and the new starting production time after the occurrence of the real-time event. On the HSMA, the stability is expressed by the sum of the absolute difference between the original completion time of each coil of the original schedule and the new completion time after the occurrence of the real-time event. For a detailed description of the utility, stability, robustness measures and rescheduling strategies refer to [8] for the CCA, and to [3, 4] for the HSMA.

The rescheduling strategies of the HSMA are: Do-nothing (NOT), Simple Replacement (SR), Closed Schedule Repair (CSR), Open Schedule Repair (OSR), Hybrid Open Schedule Repair (HOSR), Partial Reschedule (PR), and Complete Rescheduling (CR).

The rescheduling strategies of the CCA are: Insert-at- End Schedule Repair (IESR), Insert-Heat Schedule Repair (IHSR), Shift Schedule Repair (SHSR), Swap Schedule Repair (SWSR), Hybrid-shift-swap Schedule Repair (HBSR), and Complete rescheduling (CR).

When the CCA applies a rescheduling strategy to react to the real-time events heat with wrong chemical composition specification, it does a least commitment scheduling by proposing an alternative schedule which specifies the new production dates for the slabs in the current schedule. These new production dates are sent to the SYA in order to inform it on the new production dates. The SYA on receiving the message analyses the new production dates, and starts renegotiation at the SYA-CCA level with other CCA (s) to find alternative CCA that can produce the delayed slabs. If the SYA cannot manage to find the appropriate CCA to pro-

duce on time the slabs in failure it will send an alert message to the HSMA on the delayed slabs which will not be available on time. The HSMA on receiving the message reacts to the real-time events by applying the rescheduling strategies so as to maximise robustness to find alternative slabs from the slabyard to be inserted into the gap. The HSMA sends a new announcement message to the SYA to request in real-time the remaining slabs of its current schedule found after reacting to the delayed slabs. These slabs should be available in the slabyard or in the process of casting.

4 Experimental results

To show the performance of the negotiation protocol proposed to generate feasible global schedules in the presence of real-time events, we conducted several simulation experiments. In the simulation, the multi-agent architecture involves two CCA(s), a HSMA, a SYA and a UA. We carried out 5 runs for various instances of data obtained from a steel manufacturer. For each run, 5 real-time events were generated on the CCA(s). Each real time-event specifies up to 5 heats of molten steel with the wrong aluminium and carbon composition, which affects up to 4 slabs per heat. Each real-time event on the CCA can affect the production of up to 20 slabs on the HSMA. For each real-time event, both the HSMA and the CCA evaluate the best rescheduling strategy, which maximises the robustness to react to the real-time events. Note that low values of the stability measure yield low schedule disruption, and high values of the utility measure give a good value of the objective function.

Figure 1 presents a Gantt chart of a simulation run to illustrate the negotiation and re-negotiation sessions. The HSMA generates its optimal schedule of 150 orders using tabu search and requests the SYA to supply it with the slabs. On receiving the message, the SYA requests the CCA1 and CCA2 to produce 100 slabs not available in the slabyard. The CCA1 and CCA2 generate their local schedules using tabu search which is a sequence of 24 molten steel heats, and submit bids with different production dates of the slabs to the SYA. The SYA selects the CCA1 bid and submits its bid to the HSMA. We suppose in this example that the HSMA is satisfied with the production dates proposed and then it sends a contract to the SYA. The SYA on its turn sends a contract to CCA1. During production, real-time event1 occurred on the CCA1, which specifies that heat 2 and heat 3 are out of carbon and aluminium ranges which affects 9 slabs in total. The CCA1 reacts to the real-time events by applying HSR strategy and submits an alternative bid to the SYA which specifies the revised schedule. The SYA notices that 9 slabs will be late. Then it requests the CCA2 to produce the late slabs. The CCA2 bid cannot satisfy the required production dates. So the SYA sends an alert message to the HSMA on late slabs. The HSMA reacts to the real-time event by applying the CSR strategy and requests the revised schedule by the SYA. On the occurrence of real-time event 2 which affects 3 slabs of heat 6, the revised schedule of CCA1 does not affect the production dates of the slabs. The SYA on receiving the new

CCA1 bid on the revised schedule sends a contract message to the CCA1 since the slabs will still be produced on time by the CCA1.

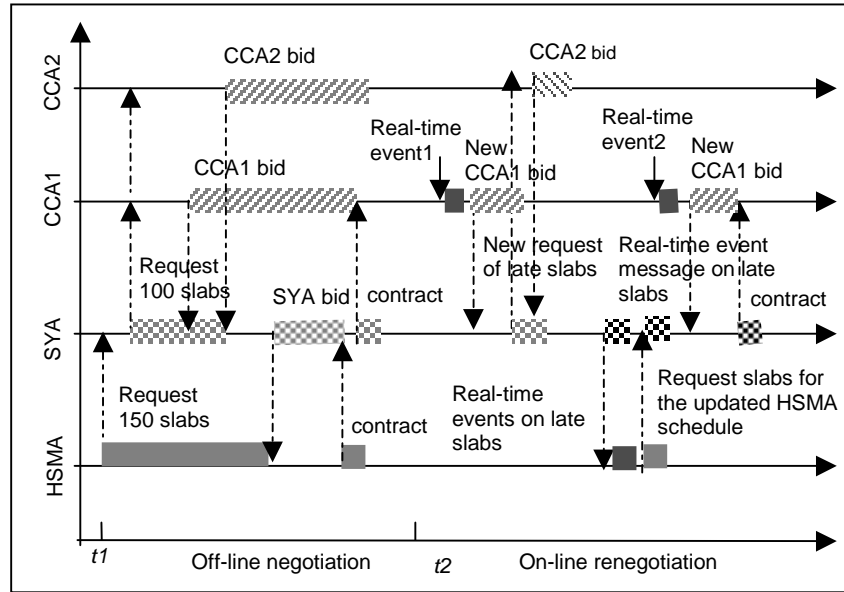


Fig. 1. Gantt Chart of an example of negotiation and renegotiation sessions

Figure 2 and figure 3 show the performance of the utility and stability measures for various problem instances on both the CCA and the HSMA obtained by applying the schedule repair and complete rescheduling strategies. Each point is the new value of utility and stability after reacting to a real-time event. In figure 2, the experiment results show that usually schedule repair strategies (NOT, SR, CSR, OSR, HCSR, HOSR, HCSR, HOSR, PR) maintain a very good stability of the system and at the same time they can achieve a very good utility comparable to the one of complete rescheduling. Consequently, schedule repair strategies define more robust schedules to react to real-time events. Complete rescheduling is competitive with schedule repair strategies in terms of utility measure, but does not dominate the schedule-repair strategies, which attain similar utility, but for better stability. Similar results are observed on the CCA. Figure 3 shows that schedule repair strategies IESR, SWSR, SHSR, HBSR maintain a good stability measure compare to complete reschedule except IHSR because of the large number of heats added to the schedule. HBSR outperforms the other rescheduling strategies in both the utility and stability measures.

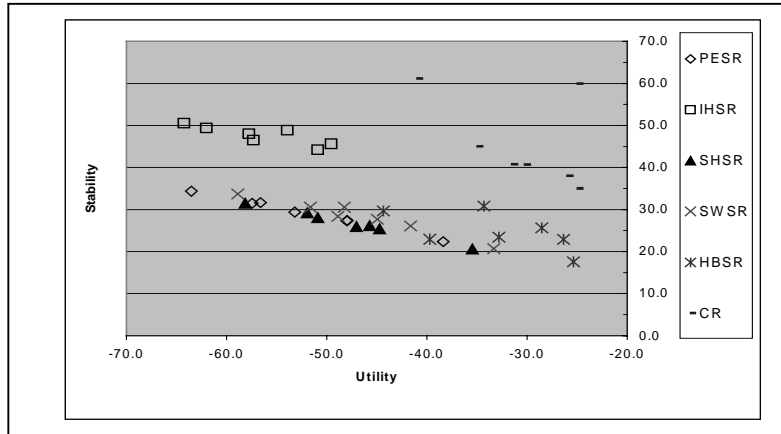


Fig. 2. Performance of the utility and stability measures of the CCA

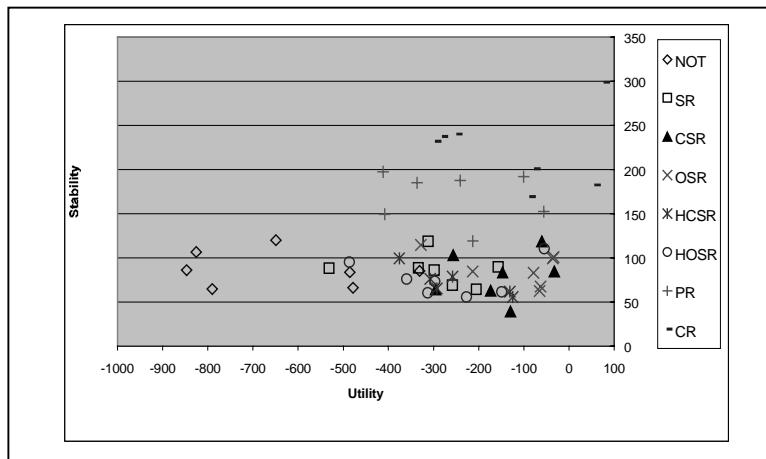


Fig. 3. Performance of the utility and stability measures of the HSMA

5 Conclusion

This paper has presented an inter-agent cooperation protocol for optimisation and dynamic scheduling in steel production. The cooperation protocol is a two level bidding mechanism based on the contract net protocol. The negotiation is a three-step process, which involves announcing, bidding and contracting at the HSMA-

SYA and SYA-CCA(s) levels. In the presence of real-time events, the CCA and the HSMA react locally to real-time events using the utility, stability, and robustness measures and rescheduling strategies. The proposed negotiation protocol is able to deal with the presence of real-time events by renegotiation in order to maintain robust global schedules. The experimental results showed the performance of the negotiation protocol in increased performance of utility and stability measures of the schedule-repair strategies compare to complete rescheduling. We have provided evidence that local autonomy and cooperation capabilities of multi-agent systems yield system robustness against failures. Moreover, using this flexible cooperation architecture allows the agents to be very simply added or removed from the system.

6 References

1. Cowling P I (1995) Optimisation in Steel Rolling. In *Optimisation in Industry*, John Wiley & Sons, Chichester, England.
2. Cowling P I, Rezig W (2000) Integration of Continuous Caster and Hot Strip Mill Planning for Steel Production. *Journal of Scheduling*, 3:185-208.
3. Cowling P I, Ouelhadj D, Petrovic S (2001) A multi-agent architecture for dynamic scheduling of steel hot rolling. In the *Proceedings of the Third International ICSC World Manufacturing Congress*, Rochester, NY, USA.
4. Cowling P I, Ouelhadj D, Petrovic S (2003) A multi-agent architecture for dynamic scheduling of steel hot rolling. To appear in the *Journal of Intelligent Manufacturing*.
5. Lin G Y, Solberg J J (1992) Integrated shop floor control using autonomous agents. *IIE Transactions*, 24:57-71.
6. Maturana F, Shen Weiming, Norrie D (1999) MetaMorph: an adaptive agent-based architecture for intelligent manufacturing. *International Journal of Production Research*, 37: 2159-2173.
7. Ouelhadj D, Hanachi C, Bouzouia B (1999) A Multi-contract net protocol for dynamic scheduling in flexible manufacturing systems. In the *Proceedings of the IEEE International Conference on Robotics and Automation*, Detroit, USA.
8. Ouelhadj D, Cowling P I, Petrovic S (2003) Utility and Stability Measures for Agent-Based Dynamic scheduling of Steel Continuous Casting. Accepted for publication in *Proceedings of IEEE Conference on Robotics and Automation*, Taiwan, USA.
9. Parunak H V (1987) Manufacturing experience with the contract net, In M. N. Huhns, ed., *Distributed Artificial Intelligence*, Pitman, pp 285-310.
10. Sandholm Tuomas W (2000) Automated contracting in distributed manufacturing among independent companies. *Journal of Intelligent Manufacturing*, 11:271-283.
11. Shaw J M (1988) Dynamic scheduling in cellular manufacturing systems: a framework for Network decision making. *Journal of Manufacturing Systems*, 7:83-94.
12. Shen Weiming, Maturana F, Norrie D H (2000) Metaphor II: an agent-based architecture for distributed intelligent design and manufacturing. *Journal of Intelligent Manufacturing*, 11:237-251.
13. Smith R (1980) The contract net protocol: high level communication and control in distributed problem solver. *IEEE Transactions on Computers*, 29:1104-1113.
14. Sousa P, Ramos C (1999) A distributed architecture and negotiation protocol for scheduling in manufacturing systems. *Computers in Industry*, 38:103-113.