MasDISPO_xt: Heat and Sequence Optimisation based on Simulated Trading inside the Supply Chain of Steel Production

Sven Jacobi
German Research Center for Artificial Intelligence
Stuhlsatzenhausweg 3
66123 Saarbrrücken
Sven.Jacobi@dfki.de

David Raber
German Research Center for Artificial Intelligence
Stuhlsatzenhausweg 3
66123 Saarbrrücken
David.Raber@dfki.de

Klaus Fischer
German Research Center for Artificial Intelligence
Stuhlsatzenhausweg 3
66123 Saarbrrücken
Klaus.Fischer@dfki.de

ABSTRACT

The production of steel normally constitutes the inception of most Supply Chains in different areas. Steel manufacturing companies are strongly affected by bull whip effects. Due to nondeterministic incoming orders and changes of customer requirements on accepted orders, making the right decision at a certain stage can be the difference between earning or loosing a great turnover. Improving their operational efficiency is required to keep a competitive position on the market. Therefore, flexible planning and scheduling systems are needed to support these processes which are based on considerable amounts of data which can hardly be processed manually. Existing systems are dominated by centralized decision making processes, mostly data driven and often not modeling the business processes they should. MasDISPO_xt is an agent-based generic online planning and online scheduling system for monitoring of the complete Supply Chain of Saarstahl AG, a globally respected steel manufacturer. This paper concentrates on the creation and optimisation of heats and sequences as a presetting for the production inside the steelwork.

1. PRODUCTION AT SAARSTAHL AG

The production chain of the Saarstahl AG consists of a multitude of specialized and complex metallurgical manufacturing processes with a lot of interdependencies among them. First, a blast furnace factory produces hot metal for the steel production. In certain intervals equally distributed over the day, a certain quantity of hot metal is sent by rail to the steel works for the next production step. Inside the melting shop, steel of different quality grades is produced according to concrete customer orders and requirements. It is cast at continuous casting plants into billets. A single production unit inside the steel works is called heat grouped together in sequences – a totally ordered set of similar heats of equal formats and qualities.

Cite as: MasDISPO_xt: Heat and Sequence Optimisation based on Simulated Trading inside the Supply Chain of Steel Production, S. Jacobi, D. Raber and K. Fischer, *Proc. of 7th Int. Conf. on Autonomous Agents and Multiagent Systems (AAMAS 2008)- Industry and Applications Track*, Berger, Burg, Nishiyama (eds.), May, 12-16., 2008, Estoril, Portugal, pp. 23-26.

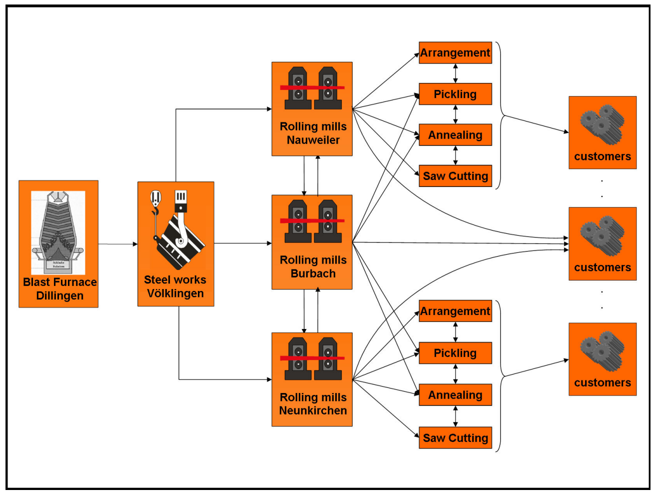

Figure 1: supply chain of Saarstahl AG

Afterwards, these billets are delivered to the rolling mills. Here, steel bars and wire rods of different shapes and formats are produced. In fixed, cyclic rolling campaigns of limited capacities certain formats are produced. These cycles are dependent on the rolling mills, billet supply of the steel works and explicit orders from customers. They vary between one to four weeks. After the rolling, potentially following processes concerning steel bars are arrangement, pickling, annealing and saw cutting; wire rods probably need a annealing, a pickling or both. Finally, the products are delivered to the customers – mostly suppliers of the automotive, shipbuilding or aerospace sectors. Figure 1 depicts the roughly described Supply Chain.

Given a working plan, the system schedules the execution of each concrete order along the production chain. It monitors production on a rough—in weeks—and detailed—in days and hours—level, and executes an online detailed planning and scheduling for the different manufacturing phases. It has to detect problems in the production and handle them in order to return to normal production. The rough working plan for each manufacturing phase is calculated on demand, before final order commitment. Depending on delivery date, order size and vertical integration certain capacities at specified aggregates have to be roughly allocated.

Usual orders to Saarstahl vary between five to several hundreds of tons. Batch sizes on each manufacturing level are fixed or limited, hence, orders have to be grouped together in process units on each stage with local constraints to keep. For instance, inside the steelwork, a production unit is called heat with fixed size of $160t$. The orders covered by a heat have to be of same quality, same casting format and should have the same calculated processing step date. Additional restrictions concerning the production inside the steelwork, and how they are handled in MasDISPO – the Multiagent System [11, 12] for the steelworks' optimisation – are described in [7], [6].

The average order backlog at Saarstahl is about 17500 orders, which makes it already a complex challenge to find an optimal mapping which keeps all constraints and deadlines. However, the system has to, additionally, deal with the online problem of dealing with new incoming orders and changing requirements by customers.

Normally, as a process step gets closer to a certain phase, the more concrete its allocation and the more detailed its planning has to be. Therefore, this system has to deal with smooth transitions between rough and detailed planning – a challenge which is often only non satisfying matched by traditional centralised approaches [10]. Dependencies between rough and detailed planning, as well as, interconnections between different manufacturing phases have to be modeled.

The overall process chain is characterized by changes in customer orders and it is affected by production setbacks or problems. Therefore, steel manufacturing companies must be flexible and dynamic, by adapting production plans fast in order to meet customer requirements while still being cost-efficient. Since these are requirements which need to be covered in almost every industrial sector, there are a lot of commercial systems handling this. But these ERP systems (enterprise resource planning) [5], [9] like APO (Advanced Planner and Optimiser) [3] or APS (Advanced Planning and Scheduling) are suitable for a rough planning only, but often not very suitable for operations planning. Big software companies have adopted the strategy to provide integration mechanisms for MES-level solutions [10] like the presented solution.

MasDISPO_xt, a decentralised agent based approach, is the proposed solution of this paper. In MasDISPO_xt, every order is modelled as an agent. The agent calculates and observes its own schedule from order entry, across rough and detailed planning, and monitors the production up to the point of delivery. It responds to changes during planning, scheduling and production by dynamically adapting the schedules. Also, each aggregate of any factory is modelled as an agent which also calculates its schedule autonomously based on further local knowledge and restrictions.

The complete production chain is very complex and could not be addressed with the appropriate detail in the context of just one paper. Therefore, this paper concentrates on the support of the creation and optimisation of heats and sequences. The detailed description of the problem as well as its solution is presented in Section 2. A general discussion of deployed system is described in 3. Finally , in Section 3 and 4, the conclusions and ongoing work are presented.

2. HEAT AND SEQUENCE CREATION

In higher planning levels (sales), the global production capacities for the different production phases are booked. After that, the planning process continues by planning at lower levels. In the case of the creation and optimisation of heats and sequences, the global planning level provides the lower level with a set of orders. This set consists normally of about 3500 order positions of different sizes, deadlines, qualities and further restrictions related to each order position. These positions have to be mapped into heats of a fixed size of $160t$ and then mapped into sequences the lengths of which have to be maximised – due to certain quality related restrictions.

Heat and sequence creation therefore is divided in two levels. First, an initial heat creation is calculated. In this phase, the order's deadline is the major criterion. The aim is to minimise the number of heats to optmise order throughput and minimise costs. Secondly, sequences are created. Maximising a sequence's length means to minimise down times of the continuous casting aggregate and hence optimsing the aggregate's throughput.

The result of these two phases is a base for the creation of a daily target schedule (DTS) as a presetting for the production inside the melting shop. This DTS consists of a partial ordered set of sequences for the continuous castings inside the steelwork. Each sequence consists of a total ordered set of heats.

2.1 Heat Creation

As mentioned, the first phase is the creation of heats as batch size for the steelwork. Input for this process is the order backlog $\mathcal{R}$ of order positions which still have to be melted, average $|\mathcal{R}| \approx 4000$. Major criterion is the latest possible manufacturing completion date of the steelwork. Also, other restrictions are mandatory, these are:

- Steel Grade: Different order positions may not be inside the same heat.
- Casting Dimension: The formats of all different positions must be equal in order to be inside the same heat.
- Continuous Casting Aggregate: Order positions are mapped to determined aggregates

Subject to these restrictions, $\mathcal{R}$ has to be partitioned into u subsets $\mathcal{R}_j$ with $\bigcup_{j=1}^{u} \mathcal{R}_j = \mathcal{R}$. u depends on the explicit composition of $\mathcal{R}$. Since every single order is modeled as an agent, this partition is straightforward. Each $\mathcal{R}_j$ consists of a set of order positions $o_i (i = 1, ..., n)$ with different delivery dates and different sizes. Aim is to minimise the number of heats in each $\mathcal{R}_j$.

Definition 1. Each order position o_i has capacity c_{o_i}. Let

$$\mathcal{C}_j := \sum_{i=1}^{n} c_{o_i}$$

be the total weight of $\mathcal{R}_j$.

$$j_{min} := \lfloor \tfrac{\mathcal{C}_j}{160} \rfloor$$

is the lower bound of quantity of heats needed to cover all order positions o_i in $\mathcal{R}_j$.

A heuristic is used to create the heats as initial solution for the second phase. At first, each $\mathcal{R}_j$ is sorted lexicographically by delivery date and order size. Some orders have capacity greater than one heat, hence at least one complete heat is allocated by such orders and the orders have to be separated into several parts. Because of different lengths of billets (limited by certain legacy systems) and order sizes further orders need to be separated and distributed on more than one heat. Some customers demand their materials of a single charge. This is also taken into account in this first step. Now, new heats are created anytime a certain order o_i does not fit completely into an instantiated heat. The system should not separate and distribute orders if not necessary. Hence, according to urgency and size heats are created until j_{min} is reached.

Secondly, all orders which have not been assigned yet have to be mapped to existing heats or probably new heats have to be created. In this step, a score function is used by each order to determine how worthwhile it is to get into a certain heat. Hence, the overall costs are minimised. The score must be inside an user defined range. Since it is possible, that certain orders might not be assigned according to this range, in the next step, the "best" score is criterion. The first phase is closed by a plausibility check on the filling degree. As an result, heats with a filling degree greater than 95% have to be received.

2.2 Sequence Optimisation

In the second phase sequences have to be created and its compositions have to be optimised. The planning department chooses a certain number of heats $\mathcal{H} \subseteq \mathcal{R}_j$ as a sequence, $|\mathcal{H}| = m$. This set $\mathcal{H}$ consist of a set $\mathcal{O}$ of order positions o_k, $\bigcup_{k=1}^{m} o_k = \mathcal{O}$. Since a sequence is created for production, the major criterion "latest possible manufacturing completion date" of the first phase has become irrelevant. Other, new restrictions are mandatory, these are:

- The filling degree of a heat must be kept inside a certain tolerance range
- Number of semi finished products lengths $\mathcal{L}$ is limited: $\mathcal{L} \leq 4$
- o_k might not be separated on more than three heats

Most important criterion in this scenario is the degree of degassing. Certain orders o_k need to be degassed for reasons of homogeneity. During the first phase, this criterion has not been taken into account since it is counterproductive to the latest possible manufacturing completion date. Orders which need a degassing are evenly distributed on each $\mathcal{R}_j$ in the initial solution. Purpose of the second phase is to group all orders $o_k \in \mathcal{O}$ which need a degassing into equal heats. So, the number of degassed heats and therefore production costs will be minimised.

Definition 2. For a given subset $\mathcal{H} \subseteq \mathcal{R}_j$ with m heats $|\mathcal{H}| = m$, the composition of each heat $h_i, h_i \in \mathcal{H}$ with order positions o_k with the aim to minimise production costs is defined as *Sequence Optimisation Problem* $\mathcal{SOP}$.

$\mathcal{SOP}$ can be transformed into a decision problem $\mathcal{SODP}$ simply by searching for a solution of $\mathcal{SOP}$ with value *cost* less than any value t, $cost \leq t$. This $\mathcal{SODP}$ is equivalent to a t-SuperCluster-DecisionProblem $tSDP$ as defined by [8]. $tSDP$ is $\mathcal{NP}$-complete.

The former approach at Saarstahl was to solve it manually. An employee of the planning department choosed a certain subset – the length of the correspondent sequence – and tried to exchange order positions between the heats in order to optmise the number of degassed heats. Because of the complexity of the problem and the fact that this has to be done for almost every sequence of DTS, an automated solution was needed.

The presented approach uses *Simulated Trading* [1] to solve it. It is an improvement mechanism starting from any initial solution—in this context $\mathcal{H} \subseteq \mathcal{R}_j$ with heats h_i as generated during the heat creation.

By successively "selling" and "buying" certain order positions each heat tries to optimise its composition of order positions. Objective is to achieve a new assignment of the already accepted order positions to the heats with an optimised cost. The trading goes over several rounds. In each cycle the heat agents submit one offer to sell or buy an order position. At the end of each round a trading agent tries to match the sell and buy offers. This is a special kind of hill-climbing algorithm, which can be interrupted anytime to pick the best solution found. This has to be done with all created subsets $\mathcal{H}$ in parallel – the number of sequences in DTS. The protocol is depicted in figure2.

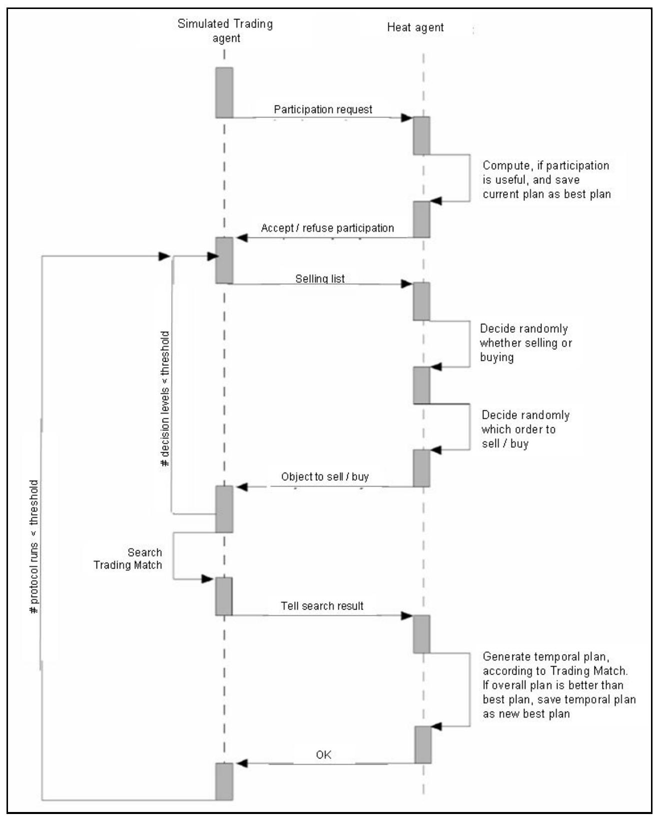

Figure 2: Simulated Trading Protocol

For this purpose, the fact of modeling heats as agents comes in handy, since a solution using *Simulated Trading* is formulated in the following way:

For each heat $h_i \in \mathcal{H}$, there is a set $\mathcal{O}_i$, $\bigcup_{i=1}^{n} \mathcal{O}_i = \mathcal{O}$ of *planned* orders with allocated capacity in heat h_i.

$$O_i = \{o \mid o \in h_i \wedge o \in \mathcal{O}\}$$

Each heat h_i advertises an order position $o_j \in \mathcal{O}_i$ with allocated capacity in h_i by auction and accepts orders o_l of other aggregates if and only if o_i also is also accepted by another suitable heat $h_k \in H$ $(k \neq i)$. The objective of this is to achieve a new allocation of $\mathcal{O}$ of already accepted order positions, which optimises the costs for all heats inside the sequence.

Original *Simulated Trading Protocol* was designed for one broker agent and hence one objective function, only. The described scenario demands different objective functions to model the problem since objectives of different heats might be counterproductive. Depending on the type of a certain heat, different functions are chosen. This is an extension of the protocol which makes it more applicable to other scenarios.

MasDISPO_xt is not a standalone system, it is embedded in the IT-environment of Saarstahl with interfaces to interact with other parts of the complete Supply Chain. Therefore, an external service-oriented architecture [2], [4] has been developed. It provides Web-Service interfaces to other partners involved, with which relevant decisions can be made. By using the described service oriented approach, flexibility is kept and interfaces are modeled for a further extension.

3. APPLICATION USE AND PAYOFF

One benefit of the deployed system is compared to the former work flow the system itself. In the past, the tasks solved by the system were handled manually. Because of the complexity it is clear that the creation of sequences wasn't solved efficiently. Now, by use of the system the planning department is able to simulate and compare several settings with different $|\mathcal{H}| = m$ lenghts of sequences leading to a well-balanced DTS. This simply was too circumstantial by manual approach.

Average runtime for the first phase is only a few seconds for one single $\mathcal{R}_j$, less than 20 seconds for the second phase. Since runtime is inside the range of seconds, extensions of the deployed system are already demanded. The planning department asks for a similar *Simulation Client* as described in [7]. Different subset substitutions of each $\mathcal{R}_j$ could be compared and evaluated. Finally, a complete production plan for the steelwork could be suggested.

A benefit is not really measurable in effective costs since it cannot be evaluated correctly. But it is obvious that the presented substitutions of heats and sequences are better than a manual approach.

The project had a duration of one year and involved one employee of Saarstahl and one of DFKI. So, a transfer of research results was expected rather than complete new innovations. Nevertheless, the extension of *Simulated Trading* proves the solution's novelty. A graphical user interface was not needed, the result are written into the legacy systems.

Since a certain part of development was on Saarstahl's responsibility, a business case was not needed. Also, a feasibility study's result as well as [10] prove that traditional, commercial approaches fail according to the special needs in this case. This is also proves significance of the deployed system. In certain areas of industry like steel production, innovative solutions like complete MasDISPO_xtmodel the business processes and meet the needs of planning departments as well as production planners on MES-levels. Smooth transitions between the different levels of planning are received – real interoperability in between these levels is achieved, another great benefit.

4. CONCLUSIONS

The described examples of this paper state a subset of the different problems which need to be solved along production inside the supply chain. But these cases already prove that the problems are too complex to be handled manually. An automatic and responsive planning system is needed. The decentralised approach with multiagent systems make the system easier to handle, really models the demanded business processes and is able to manage the huge data amount along production – requests which are not always met by the existing centralised approaches.

The authors like to thank Saarstahl AG. Without their innovation related mind the realisation of MasDISPO_xt would not have been possible.

5. REFERENCES

[1] A. Bachem, W. Hochstaettler, and M. Malich. Simulated trading: A new approach for solving vehicle routing problems. Technical Report Tech.Rep. 92.125, Mathematisches Institut der Universitaet zu Koeln, 1992.

[2] D. K. Barry. *Web Services and Service-Oriented Architectures*. Morgan Kaufmann, 2003.

[3] H. Bartsch and P. Bickenbach. *Supply Chain Management mit SAP APO*. SAP Press, 2nd ed., 2002.

[4] D. Booth, H. Haas, F. McCabe, E. Newcomer, M. Champion, C. Ferris, and D. Orchard. Web services architecture, working group note, http://www.w3.org/tr/ws-arch/, 2004.

[5] N. Gronau. *Enterprise Resource Planning und Supply Chain Management: Architektur und Funktionen*. Oldenbourg (Muenchen), 2004.

[6] S. Jacobi, E. Leon-Soto, C. Madrigal-Mora, and K. Fischer. Agentsteel: An agent-based online system for the planning and observation of steel production. *Proc. 4th Inter. Joint Conference on Autonomous Agents and Multiagent Systems (AAMAS 05)*, 2005.

[7] S. Jacobi, E. Leon-Soto, C. Madrigal-Mora, and K. Fischer. Masdispo: A multiagent decision support system for steel production and control. *AAAI Innovative Applications of Artificial Intelligence*, 2007.

[8] M. Malich. Simulated trading: Ein paralleles verfahren zur lösung von kombinatorischen optimierungsproblemen, 1994.

[9] G. W. Plossle and J. A. Orlicky. *Orlicky's Material Requirements Planning*. Prentice Hall, 2nd ed., 1985.

[10] SAP. Integration von mes-systemen in sap for mill products, 2004.

[11] G. Weiss, editor. *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*. KIT Press, 1999.

[12] M. Wooldridge. *An Introduction to Multiagent Systems*. John Whiley & Sons, 2002.