Knowledge-based Production Planning, Scheduling and Control

IJCAI-93 Workshop

Chambery, France

August 29, 1993

A Distributed Heuristic Expert System for Simulation and Production Planning in Petrochemical Industries

FRÄMLING Kary*, HAMMARSTRÖM Leif**

(*) Ecole des Mines de Saint-Etienne

158 cours Fauriel, 42023 Saint-Etienne Cedex 2, FRANCE

Tel. 33-77.42.02.39, Fax. 33-77.42.00.00

E-mail: framling@emse.fr

(**) Neste Engineering, Technology Centre

P.O.B. 310, SF-06101 Porvoo, FINLAND

Tel. 358-15-1873981, Fax. 358-15-1877221

E-mail: lgh@jukka.neste.fi

Abstract

This paper describes a production scheduling application for a petrochemical plant. The main techniques used are heuristic reasoning, simulation and object oriented (OO) models. Each product manufactured by the plant has its' own scheduling object, which contains a set of rules for producing a heuristic estimate of the need to manufacture the product in question. Simultaneous simulation with an OO model is used to verify that the schedule is good as well as to correct errors during the planning stage.

Keywords: Production Planning, Artificial Intelligence, Distributed Expert Systems, Heuristic Reasoning, Simulation

1. Introduction

The aim of this work is to explore the possibilities of automating the scheduling of production in petrochemical plants. The automatic production scheduling is done by using product-specific heuristic estimates. These estimates are calculated by simple sets of rules, which allow the definition of different scheduling strategies for each product. A prototype system for a phenol plant was developed in order to achieve this goal.

The prototype system was written using the object oriented (OO) programming language Smalltalk/V on Macintosh. The system allows the creation of graphical simulation models for most kinds of petrochemical plants as a simple drawing operation. This means that it is very fast to create a model at the level of detail necessary for the production scheduling.

2. Object oriented simulation models of petrochemical plants

Petrochemical plants are very complex systems that are often difficult to model. The models have to include large amounts of data as well as methods of calculation, which are hard to organize and handle in a consistent way. OO systems offer a natural solution to this problem of data and calculation management [Alasuvanto et al, 1988; Främling, 1990; Hammarström, 1990; Stephanopoulos et al, 1987], since the data needed for each operation is collected in the place where it is needed (the object), together with the calculations and/or operations which can be performed on this data (the methods). Hierarchical models with different levels of detail are a supplementary way to reduce the amount of data treated by the user at a time, which increases the usability of the model [Henning et al, 1988].

The three main properties of OO systems are inheritance, encapsulation and polymorphism. Inheritance is useful when modeling chemical plants for defining common properties of various kind of equipment (heat exchangers, distillation columns etc.), which reduces the programming effort required. Encapsulation means that the data of the model is completely attached to objects, which control the access to it, therefore preventing programming errors or corrupting data. The polymorphism means that it is possible to have generic messages that can be sent to almost all objects of the model, which will be able to handle them in their own way, therefore reducing the complexity of the software.

2.1. Simulation

Simulation allows the user to test different production plans in order to find an optimal one. What is actually simulated are the tank levels, since these are the critical indicators of the success or failure of the production plan. The plan has failed if a tank either runs empty or becomes full.

The starting time of the simulation depends on which tanks are simulated, as it has to be started from the oldest real level indication. Tanks with a newer level indication are not included in the simulation until their level indication time has been reached.

The simulation is a discrete time one, so a step length has to be defined. The step length does not, however, have any impact on the simulation precision, except that events happening during a time step might not be discovered. An example of such an event is that a tank goes empty, but is filled again during the same time step.

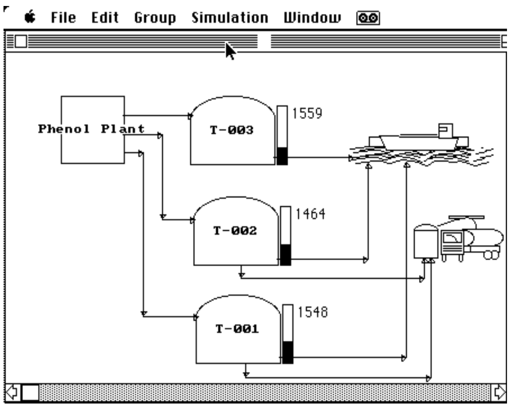

Figure 1. Terminated simulation.

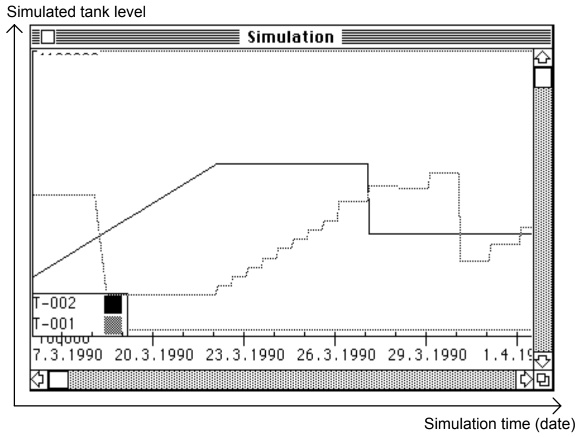

Figure 2. Simulation curves. The step length has been changed the 22.3 (12h) and the 26.3 (24h).

It is possible to change the step length during the simulation, which means that it is possible to focus on critical periods. The simulation can also be advanced step by step and backed up. The backing-up function is especially useful, as the production plan can be changed and re-tested without having to restart the whole simulation.

3. Production planning with the model

Artificial intelligence techniques have become quite a popular way of solving production planning problems [Stephanopoulos, 1987, 1988], partially replacing the traditional operations research techniques [Jensen, 1986]. Combining these techniques with simulation has also been found advantageous [Falster, 1987]. Machine learning has been considered in order to further improve the production planning [Realff and Stephanopoulos, 1990].

The production planning is done in a linear fashion (figure 3), meaning that the production plan is created from the start of the planning period towards the end [Lakshmanan and Stephanopoulos, 1988; Realff, 1989, 1990]. This approach has been selected mainly because of the need of continuous simulation. The scheduling principle is JIT, without idle periods, which means that some product is always manufactured if possible. Therefore the orders (shipments) are the main base for the scheduling. These shipments are considered as "batches", which have to be manufactured in time.

The production plan is created by combining the simulation with the planning module. The planning module consists of several product specific planning objects, which use a set of rules for calculating a heuristic estimate of how busy it is to produce the product. The product with the highest score (heuristic estimate) is the one produced.

A set of six margins is the base for calculating the heuristic estimates (figure 4):

- Margin 1: The lowest level allowed in the tank.
- Margin 2: The highest level desired in the tank.
- Margin 3: The time before the last possible starting time (LST) when the production should start.
- Margin 4: Security margin added to the model-estimated LST for producing the next batch.
- Margin 5: Used for deciding whether the second batch of a product is close enough to the first in time to be taken into consideration.
- Margin 6: Minimum quantity that should be produced when the production plan fails and is corrected.

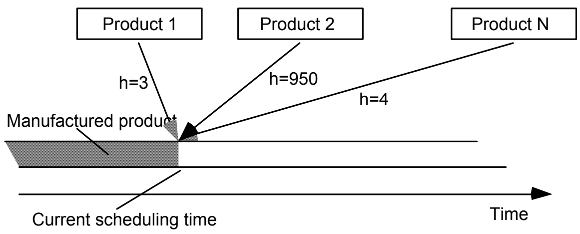

Figure 3. Heuristic linear planning with example heuristic values on a scale from zero to 1000. Product 2 is the one produced.

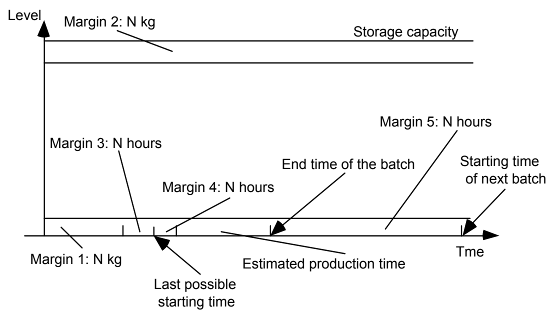

Figure 4. Security margins 1-5.

The time required for producing the batches is calculated by "asking" the model for an estimation of the delays necessary, with production stops, disturbances and margin four taken into consideration. Margin three is added to this time estimate so as to be sure to start the production in time. Unfortunately it is not possible to give an exact estimate of the production time due to the great amount of units concerned, setup times and other uncertainties. Mistakes in the production plan are, however, detected through the simulation and can be corrected (figure 5).

The product specific planning objects each have properties which define the planning strategy of the product. This means that one single rule set (currently containing twelve rules) can be used for all products, with different strategies e.g. JIT, produce to stock etc. being completely defined through the values of these properties.

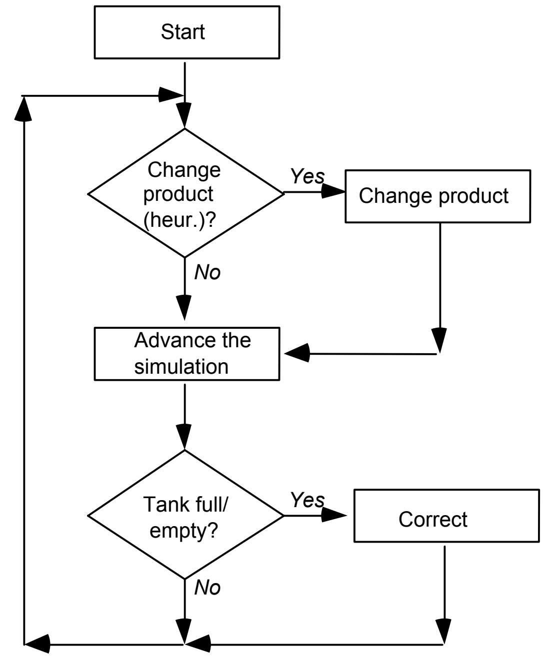

Figure 5. Production planning loop.

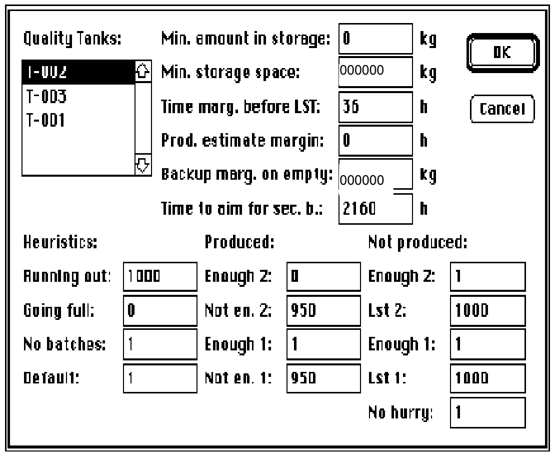

Figure 6. Dialogue for defining the production planning strategy.

The following properties have been defined:

- Product to which the strategy belongs.
- Tanks into which the product may be produced (unless they already contain another product).
- A set of batches of the product that should be manufactured.
- The margins one to six.
- Heuristic constants used in the rules (one to twelve and a default one).

Despite its simple structure, the rule set used expresses the knowledge of the human production planner very well. The rule set might, of course, be bigger or more complicated, but the experiments performed until now showed there is no need for this - the twelve rules are already sufficient. The rules are the following:

1. If the quantity in stock of the product is under the allowed minimum (margin 1), a very high constant **H1** is returned.
2. If the available storage space of the product is smaller than desired (margin 2), a very low constant **H2** is returned.
3. If no more batches are scheduled for the product, a low constant **H3** is returned.
4. If the product is already being manufactured, the second batch to be produced is close enough to the first (margin 5) and there is enough for both of them in stock, the constant **H4** is returned.

5. If the product is already being manufactured, the second batch to be produced is close enough to the first (margin 5) and there is not enough for both of them in stock, the constant **H5** is returned.
6. If the product is already being manufactured, the second batch to be produced is not close enough to the first (margin 5) and there is enough for the first in stock, the constant **H6** is returned.
7. If the product is already being manufactured, the second batch to be produced is not close enough to the first (margin 5) and there is not enough for the first in stock, the constant **H7** is returned.
8. If the product is not being manufactured, but there is enough of it for the next two batches, the constant **H8** is returned.
9. If the product is not being manufactured, and the production for the next two batches has to start in a certain number of hours (margin 3), a high constant **H9** is returned.
10. If the product is not being manufactured, there is enough in stock for the first batch but not for the second, a constant **H10** is returned.
11. If the product is not being manufactured, and the production for the next batch has to start in a certain number of hours (margin 3), a high constant **H11** is returned.
12. If the product is not being manufactured, there is not enough for the next batch, but it is not critical to start manufacturing it, a constant **H12** is returned.

The rules four to twelve are logically complete, which means that one of the rules is always applicable (unless one of the rules one to three has already been applied). This can be shown by the following reasoning:

The different situations are defined as follows:

- The quality is being produced **PR**.
- The second batch is close enough to the first to be taken into consideration **B2**.
- There is enough in stock for the two following batches **E2**.
- There is enough in stock for the first batch **E1**.
- The manufacturing of the product has to be started in order to be finished in time for the next two batches **P2**.
- The manufacturing of the product has to be started in order to be finished in time for the next batch **P1**.

The rules four to twelve can then be rewritten in the following way (~ means NOT):

4. PR & B2 & E2 => H4

5. PR & B2 & ~E2 => H5

6. PR & ~B2 & E1 => H6

7. PR & ~B2 & ~E1 => H7

8. ~PR & E2 => H8

9. ~PR & ~E2 & P2 => H9

10. ~PR & ~E2 & ~P2 & E1 => H10

11. ~PR & ~E2 & ~P2 & ~E1 & P1 => H11

12. ~PR & ~E2 & ~P2 & ~E1 & ~P1 => H12

The choice of taking only the next two batches into consideration is motivated by the production speed of the production facility, the storage capacity available and the size of the batches. Two batches seems to be enough as long as the storage capacity is between one and three times the size of the batches.

4. Correcting failures found through simulation

A failure of the production plan is indicated by a tank either running empty or becoming full during the simulation. Failures cannot be avoided completely due to the fact that the production plan is constructed based on estimates, which are much less exact than the simulation. This is especially true for the effects of setup times and production problems.

In the case of a tank becoming full there are three main possibilities of action. The first one proposed is to change the product (which could also be done automatically), the second one is to add a shipment (or do one earlier than planned) and the third is to stop the production. Stopping the production actually corresponds to changing the product because "Idle" is defined as a product with zero production speed.

If a tank runs empty the first reaction is to either insert a product change or advancing the last product change if the right product is already being manufactured. Another possibility is to delay a shipment, but this is usually quite difficult due to the potential problems caused to the client. Inserting a product change or advancing it is done through "brute force", which means that the change is advanced iteratively until the plan works or fails completely. A complete failure is indicated by a backing up past the starting time of the production planning.

A tank may run empty mainly due to two reasons. The first possibility is that there has been transports by lorries, that slowly empty the tank. This situation is detected through simulation, and the action taken is to insert a product change into the schedule in order to produce a certain quantity of the product. This quantity is margin number six (figure 7). The other possibility is that the estimated manufacturing time for a batch was too short and there has not been enough time to manufacture the needed quantity. In this case the missing quantity is calculated directly and the product change is backed up in order to be able to produce this quantity (figure 8).

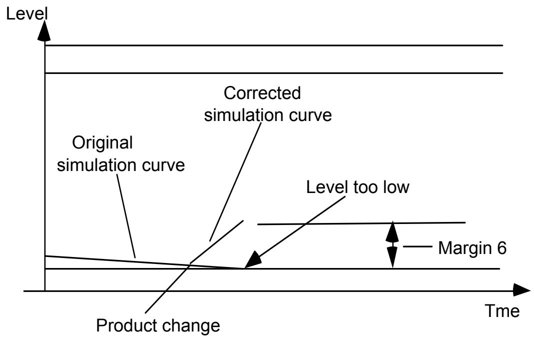

Figure 7. Margin six.

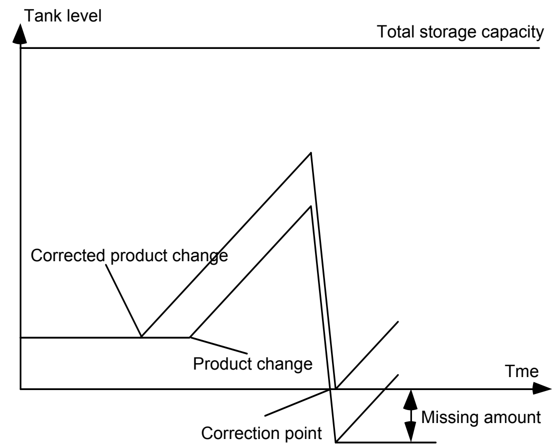

Figure 8. Correcting the production plan.

5. Results

The system was tested on several real situations from the past of the phenol plant, and managed to produce a well working production plan every time, which was quite similar to the one produced by the human planner. A good production plan was found without any corrections on the first try every time once the right margins and heuristic constants had been found. Even with badly chosen margins and constants the correction mechanism always managed to give a successful plan at the end.

A few small errors in the manually planned production schedule were also found because of the bigger simulation accuracy of the model compared to the production estimates used by the human production planner. The automatic production planning system managed to avoid these situations.

In figure nine there is shown a comparison between a manually produced (and then simulated) production plan and the production plan given by the automatic production planning. The simulation curves show the tank levels in two tanks during a period of about two and a half months. In the first picture (with the manually produced production plan), it can be seen that one of the tanks would become full after about two months. In the second picture, this problem is detected by the automatic production planning, and the product change happens earlier.

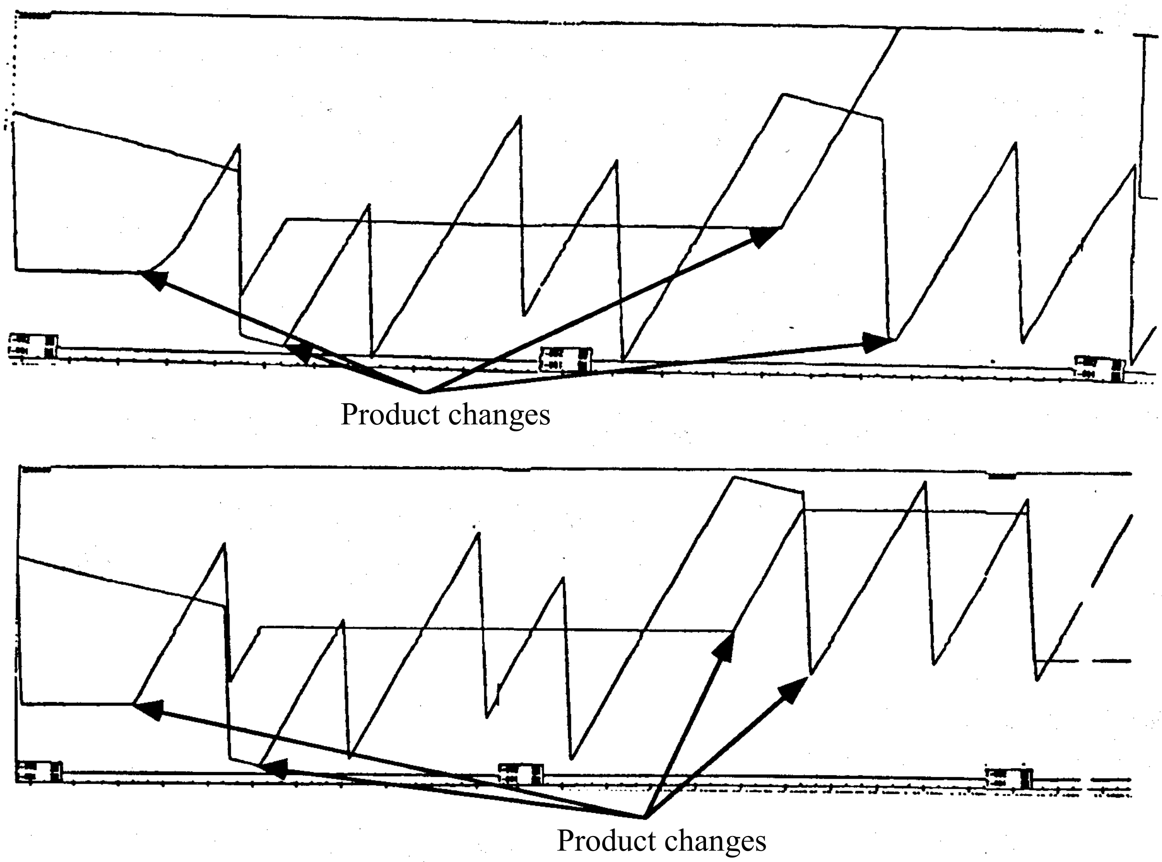

Figure 9. Comparison of production plans produced by the human production planner (above) and the automatic production planning module (below).

6. Conclusions

The combination of object orientation, simulation and production planning has been very helpful. These three techniques support each other very well, thus reducing the complexity of the final system compared to the complexity found in a traditional one.

Even though the plant tested is of the flow-shop type, the methods used should work quite well even in the case of several parallel production units (job-shop type). This aspect is already possible to take into consideration, since the effects of having parallel production units are treated by the simulation. Extending the methods to other kinds of production than chemical factories remains to be studied.

References

Alasuvanto J., Eloranta E., Fuyuki M., Kida T. Inoue I.: *Object oriented programming in production management - two pilot systems,* Int. J. Prod. Res. 1988, vol.26, no. 5, p. 765-776.

Falster P.: *Planning and controlling production systems combining simulation and expert systems*, Computers in Industry, nr.8 1987, p. 161-172.

Främling Kary. *Using object oriented models for the scheduling of production in petrochemical industries*, MSc. thesis : University of Technology in Helsinki, 1990. 86 p.

Hammarström Leif: *Dynamic Simulation of Chemical Engineering Processes by using Object Oriented Programming*, Neste Oy, Process Automation, Kullo, Finland 1990. 31 p.

Henning Gabriela P., Leone Horacio P., Stephanopoulos George: *Hierarchical Modelling Environment for Process Engineering*, Laboratory for Intelligent Systems in Process Engineering Technical Report, Massachusetts Institute of Technology, Cambridge, MA 02139, 1988. 14 p.

Jensen Paul A.: *Operations Research,* Holden-Day, Inc., Oakland CA 1986. 502 p.

Lakshmanan Ramachandran, Stephanopoulos George: *Synthesis of Operating Procedures for Complete Chemical Plants, Part I: Hierarchical, Structured Modelling for Nonlinear Planning*, Lispe Technical Report, Massachusetts Institute of Technology, Cambridge, MA 02139, February 1988. 51 p.

Realff Matthew: *Learning localized heuristics in batch scheduling*, Lispe Technical Report, Department of Chemical Engineering, Massachusetts Institute of Technology. May 1989. 37 p.

Realff Matthew: *An analysis of the chemical batch production problem and a detailed methodology for scheduling*, Lispe Technical Report, Department of Chemical Engineering, Massachusetts Institute of Technology. May 1989. 46 p.

Realff M. J., Stephanopoulos G.: *Machine Learning for the Improvement of Scheduling Algorithms*, Lispe Technical Report, Massachusetts Institute of Technology 1990. 26 p.

Stephanopoulos George, Johnston James, Lakshmanan Rama: *DESIGN-KIT: An Intelligent System for Planning Plant-wide Process Control Strategies*, Lispe Technical Report, Massachusetts Institute of Technology, Cambridge, MA 02139, March 1987. 10 p.

Stephanopoulos George: *The Scope of Artificial Intelligence in Plant-wide Operations*, Lispe Technical Report, Massachusetts Institute of Technology, Cambridge, MA 02139, September 1987. 46 p.

Stephanopoulos G., Johnston J., Kriticos T., Lakshmanan R., Mavrovouniotis M., Siletti C.: *DESIGN-KIT: An object-oriented environment for process engineering*, Comput. chem. Engng. Vol. 11, No 6 1987, pp. 655-674.