Combining Multi–Agent Systems and Constraint Techniques in Production Logistics

Hartwig Baumgärtel, Stefan Bussmann, Marcus Klosterberg
Daimler-Benz AG
Research for Systems Technology
D-10559 Berlin, Alt-Moabit 96a, Germany
{baumg, bussmann, klosterb}@DBresearch–berlin.de

Abstract

The paper describes objectives and results of a research project conducted by Daimler–Benz AG. Intention of this project is decision support for a logistics division. In particular, the production flow control (PFC) in a factory has to be optimized by a decision support system. PFC is a sophisticated process influenced by complex factory structures and high dynamism of manufacturing processes. In order to support PFC we developed a system based on concepts of multi–agent systems and constraint techniques. With this system it is possible to simulate arbitrary production scenarios and to compute their effects on necessary production settings. The user thus receives decision support which may be used during operational planning in logistics as well as in development of new logistic strategies.

1. Application Domain and Problematic Nature

The production flow control (PFC) department is part of a logistics division responsible for superordinate planning and coordination of the manufacturing process at a particular plant of Mercedes–Benz. PFC planning results in a production strategy. This strategy encompasses quantitative quotas for every manufacturing center in the factory to be entered in a table called daily production plan (DPP). Additionally, corresponding production settings have to be determined for the manufacturing centers. The general aim of production strategy is to achieve a production goal, that is a given number of products.

PFC is a complex and sophisticated process influenced by a number of factors. Those have strong consequences on the manufacturing process and in so far on the logistic planning itself (see table 1 for an overview). Various factors and their effects on production and planning processes are described below:

- number of manufacturing lines in the factory: a product can be manufactured in several lines which are closely interconnected (see figure 1). This structure results in a multitude of alternatives during production planning and controlling. In addition, the number of alternatives is restricted by technical reasons and organizational requirements.
- number of manufacturing steps: the entire manufacturing process is divided into distinct steps. Each step can be performed at several manufacturing centers which are organizationally separated (cost centers). The installation of cost centers supports cost–optimal manufacturing on a local basis. However, decentralized planning and controlling leads to suboptimal global conditions since no sufficient global coordination is provided.
- manufacturing of different series variants, and individuality of every particular order due to a wealth of optional equippment: each product is at last an individual piece.

Another aggravating effect towards the planning process within the logistics department is caused by the dynamic character of the manufacturing process. First, dynamism is yielded by a series of rebuildings at the present and in the future. Permanent reconstructions lead to serious alterations in factory structures. Due to this, alternatives for formulating a production strategy keep changing. Planning and controlling processes in PFC have to be adapted and rearranged.

Second, it is typical for the manufacturing process to be disrupted by varied technical disturbances. Proneness to disruptions results in frequent plan deviations, requiring short–term plan modifications. This kind of dynamism results in high planning uncertainty and lack of time to formulate new production strategies.

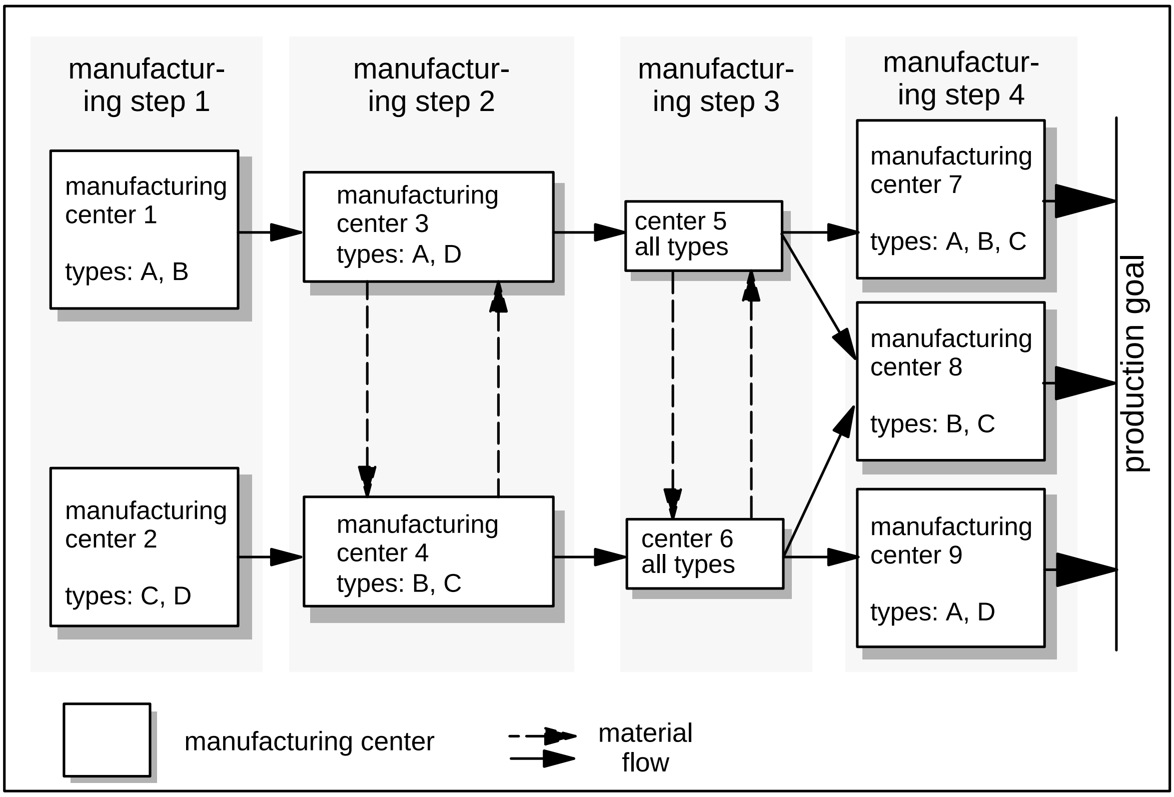

Figure 1: factory structure

Size, degree of distribution and dynamism within the factory and the manufacturing process create a complexity that proves planning in logistics is no longer possible without support by information technology. For this reason, the planning of a production strategy should be carried out by means of a simulation–based decision support system. High requirements to PFC result in sophisticated demands on the supporting system.

TABLE 1: Problematic Nature of Production Flow Control

Situation in the factory	Consequences on manufacturing process	Consequences on logistical planning
Multiple manufacturing lines in the factory	Several alternatives to manufacture a product	Multitude of decision– respectively planning–alternatives
Different series variants and a wealth of optional equippment	many different product types and complex manufacturing process	Complex, complicated and iterating planning process
Organizational seperated manufacturing centers with local optimization strategies	Decentralized local planning and insufficient coordination in manufacturing control	Decentralized planning processes and global suboptimal planning and controlling
Permanent modifications of the factory structure by reconstructions	Irregular changings of manufacturing process and of strategy alternatives	Dynamism in planning process leading to frequent rearranging of planning procedures
Proneness to disruption and variety of disturbances	Plan deviations and short termed plan modifications	High planning uncertainty lack of time for new planning processes

2. System Task and Requirements

System task is to support computation of a production strategy. This includes determination of quantitative quotas within DPP as well as calculation of appropriate production settings for the manufacturing centers.

System requirements could be derived from problem description and system task as detailed below:

- The system has to comprise a model of the entire factory structure and manufacturing process.
- The model must represent different manufacturing centers and their local optimization strategies as seen by the logistics department.
- The user needs an easy way of incorporating changes in factory structure into the model.
- System calculations should be adaptable to the factory's current state by flexible parametrization. Particularly, disturbances in form of breakdowns or capacity reductions must be taken into account. The user needs an intuitive way to do parametrization.
- After having been given the number of products to be manufactured and selecting some alternatives (production settings), the system has to calculate production quotas of DPP and determine the remaining settings for the manufacturing centers (and therefore the consequences of the decision).

Currently, no existing standard application is able to represent complexity, distribution, and autonomy of the manufacturing centers, and the given dynamism within the factory. The following approach has been pursued in order to develop a new system which meets these requirements.

The general approach consists in providing the logistics department with a decision support system that

- reveals consequences of chosen alternatives by simulation and
- determines a DPP and corresponding production settings for a given production goal.

3. Combining Multi–Agent Systems and Constraint Techniques

The approach chosen is based on concepts of multi–agent systems [cf. eg. 3, 12]. Multi-agent systems consist of active, concurrent software modules called agents. Each agent has its own goals, plans, and knowledge. Additionaly, agents have the ability to communicate with each other. Communication is structured according to defined protocols and enables the agents to coordinate their performance. The idea of agent-oriented design allows software engineers to analyse, design, realize complex systems on a higher level of abstraction, as for example object-orientation.

The manufacturing centers are modelled as autonomous agents (manufacturing agents). This facilitates an individual and independent representation of the entities and their local optimization strategies, and thus increases modularization of the system.

Constraints are used to express relations between arbitrary entities of models, i.e. variables, structures, or partial models. Evaluation of constraints is multidirectional: the value of every entity of a constraint can be computed or restricted dependent on values or value restrictions of all other entities. Evaluation is incremental too: value changes may be propagated if they occur.

A set of constraints with common entities is called a constraint net. If the value of a variable changes, propagation will be done along all constraints which contain the variable.

In order to fulfill the production quota of a manufacturing step, manufacturing agents communicate their free resource capacities as constraints. A constraint–solver is used for coordinated allocation of sub–quotas to the centers. After collecting the constraints, the constraint–solver creates a corresponding network, performs constraint propagation, and enumerates the solutions [cf. eg. 4, 6, 8].

If planning of a manufacturing step fails because the corresponding constraint network has no solution, the system returns to preceding planning steps by backtracking and searches for a different solution.

Since manufacturing in the particular plant at Mercedes–Benz has the character of flow production, every agent exchanges material with its direct neighbours only. That is, intermediate products which are delivered by an entity must be taken over by subsequent centers. Consequently, result of the planning process is a set of production settings and production quotas for every manufacturing center that must also be globally consistent. By that, the material need of a manufacturing step is equal to the production quota of the preceding step. In order to fulfill this global planning condition, a special agent (DPP–agent) has been introduced. Its task is to coordinate production settings along the material flow. The sequence of planning steps are exactly opposed to the manufacturing process. The last manufacturing steps are planned first and vice versa.

With help of a system based on the concepts presented above it is possible to simulate arbitrary production scenarios and to compute their effects on the DPP and necessary production settings. The user thus receives decision support which may be used during operational planning in logistics as well as in supporting development of new logistic strategies.

4. System Description

The implementation of the presented approach comprises a system architecture and a description of the general planning algorithm. The architecture provides a survey of modules to be implemented and necessary information processing. The planning algorithm describes the overall planning procedure and the co–operation among the outlined entities.

4.1 System Architecture

Essential components of the architecture can be derived from the rough draft of the multi–agent system. It remains to specify the manufacturing agents, the DPP–agent, and the constraint–solver in detail.

Manufacturing agents are used to model the structure of the factory and the individual manufacturing centers. Different manufacturing entities are represented by distinct manufacturing agents. Figure 2 shows a hierarchical representation of the agent types used for modelling the whole factory structure.

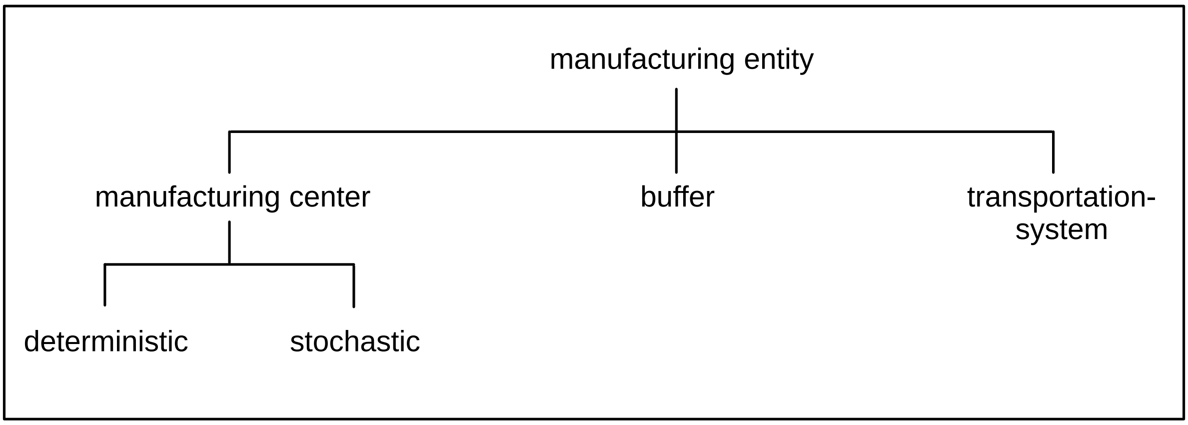

Figure 2: agent types representing manufacturing entities

We need agent types to represent transportation systems, buffers, and manufacturing centers. The latter are subdivided depending on the way a production quota is determining the actual production volume. For example, in repair centers production volume stochastically depends on given quotas for other manufacturing centers.

Every manufacturing agent has a knowledge base containing facts about possible production settings, related capacities, and its need for resources, especially for workers. For assessing different production settings, agents use given preferences. Additional facts describe the current state of the center. From the view of the logistics, production settings, preferences, and states are parameters which can be modified. With respect to a simulation run they can be both variable and fixed. Manufacturing agents have the intention either to fill the capacity derived from a given production setting and the current state or to adapt settings according to requested capacity. Manufacturing agents receive and pursue their goals by interacting with the DPP–agent. Therefore, agents possess procedural knowledge describing how to carry out the communication and to calculate reasonable production settings.

The DPP–agent has the task of coordinating production settings between manufacturing centers. It intends to carry out coordination according to the production goal provided by the user. The knowledge of the DPP–agent describes which manufacturing agents exist, their agent type, connections among them, the product types to be produced, and agents suitable to process these types. Moreover, it has knowledge about the necessary sequence of manufacturing steps. By its procedural knowledge, the DPP–agent knows how to communicate with manufacturing agents. In addition, this knowledge describes how to proceed in coordinating production settings.

The constraint–solver is employed in planning a single manufacturing step. Its task consists of computing coordinated allocation of eligible resources with respect to the production quota. It receives resource capacity constraints posted by manufacturing agents expressing their currently available capacities referred to product types. Next, it assembles a network consisting of production quota and resource capacity constraints. After performing constraint propagation, possible solutions are determined by enumeration and conveyed to the DPP–agent.

4.2 Planning Algorithm

The planning algorithm evolves from the interaction of participating agent’s procedural knowledge. The algorithm represents a comprehensive view on the distributed planning components. Essentially, the planning algorithm deals with two tasks: tuning of agents within each manufacturing step and coordination of manufacturing steps along the material flow. For that, following planning phases are processed (see figure 3):

a) ascertaining the production quota: First in planning the DPP–agent has to determine the number of products to manufacture. In succeeding planning steps, the production quota corresponds to the material need of the respective preceding step.
b) In the following phase called pre–inspection, manufacturing agents which are suitable and ready to use are determined. A call is sent to those selected.
c) The manufacturing agents calculate their free resource capacities. They either start out from a production setting fixed by the user or calculate the settings using their internal preferences. The capacities are communicated as constraints.
d) The DPP–agent collects the posted constraints and evaluates them first in order to test whether solutions to the constraint net are possible. If the pre–test is positive the constraints are passed to the constraint–solver.
e) The constraint–solver creates a network of constraints from the replies of the manufacturing centers and the production quota. It then performs constraint propagation, determines the next solution, and returns it to the DPP–agent.

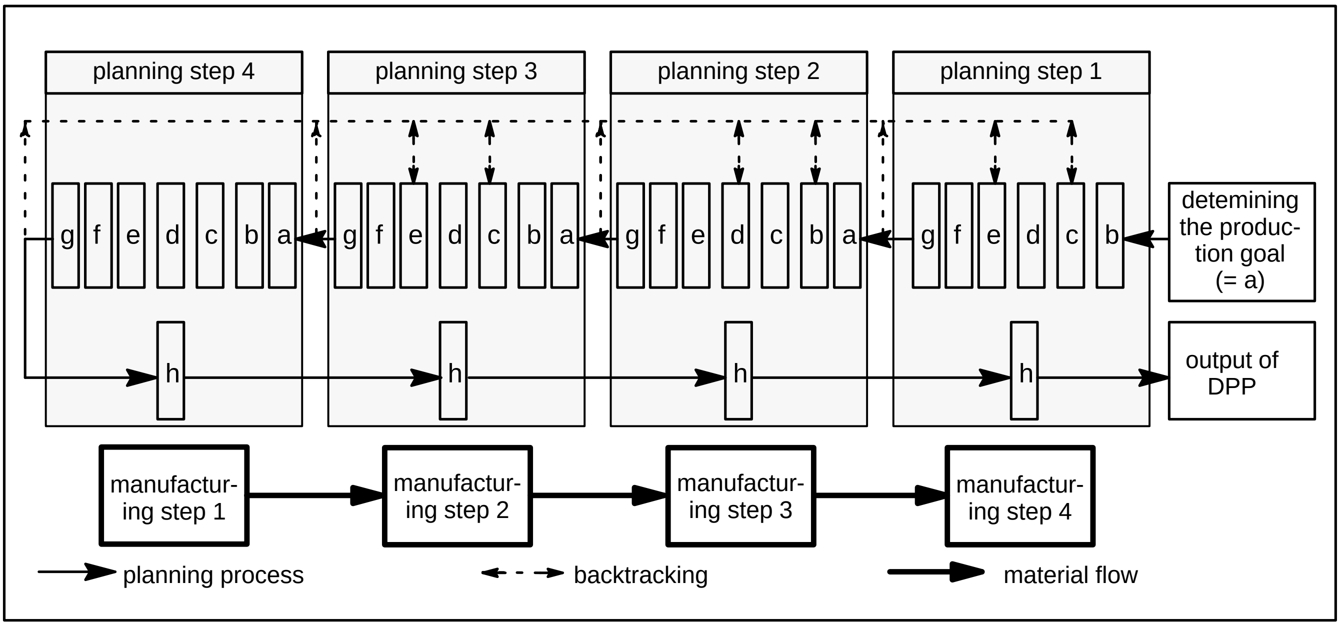

Figure 3: diagram of the planning algorithm

f) According to the solutions, the DPP–agents makes reservations at the manufacturing agents.
g) The manufacturing agents accept reservations and calculate their need of intermediate products.

Phases a) to g) are repeated for each manufacturing step. If a solution for every manufacturing step has been found the DPP–agent commits the manufacturing agents (phase h). If a planning step fails due to constraint violations, reservations made in preceding steps are cancelled and a different solution is searched by backtracking. Suitable phases for backtracking are enumeration of solutions (e) and calculation of the manufacturing agents (c).

Interaction between DPP–agent, manufacturing agents, and constraint–solver follows a standardized sequence (see figure 4). This sequence is repeated for the planning of every manufacturing step.

5. Related Work

Ayel describes an approach in which task of global coordination is distributed to local controllers [1]. These controllers coordinate underlying manufacturing cells using a blackbord data structure for communication. That is, coordination according to a global criterion is distributed, causing a high additional communication overhead and implementational burden. Due to these severe practical disadvantages, we did not follow her approach.

A formal and generic description of synchronizing multi–agent plans by a single intelligent agent is described by Rosenschein [9]. In his framework, Rosenschein determines primitives for inter–agent communication. In our system, we chose to replace communication primitives by constraints in order to get statements which instantly can be used for computing an allocation of production quotas.

Many approaches of manufacturing control by distributed AI are using negotiation–based techniques such as contract–net. These systems are mostly used for shop floor control [11]. We decided not to design a system based on the contract–net metaphor because PFC is a slightly different problem. Focus of PFC is not just the fulfillment of the production goal. Another important aim is to achieve equal distribution of production volumes to manufacturing centers. Therefore, manufacturing agents do not post bids but entire intervals of free resource capacities.

Planning is also an application field of pure constraint techniques. Most approaches use either constraint logic programming (CLP) languages [cf. eg. 6, 8] or special purpose constraint libraries for standard programming languages [5]. In both cases underlying algorithms are standard constraint solving methods as for example propagation of known values or intervals [cf. 4]. These algorithms are limited in case of over–constraint problems, where conflict set detecting and constraint relaxation are required. CLP approaches are more flexible in problem representation then special purpose libraries. Otherwise, they do not support problem– and software structuring in a natural way as agent–oriented techniques.

An interesting approach to handle conflict set detection in overconstraint problems has been developed by Bowen and Bahler [2]. Their work is based on combination of constraint techniques and truth maintainance. This facilitates efficient conflict handling, e.g. identification and relaxation of constraints responsible for conflicts. The resulting constraint programming language was applied to concurrent engineering. Although a planning application is not known this approach is significant for our system for various reasons, e.g. employment of constraint negotiation.

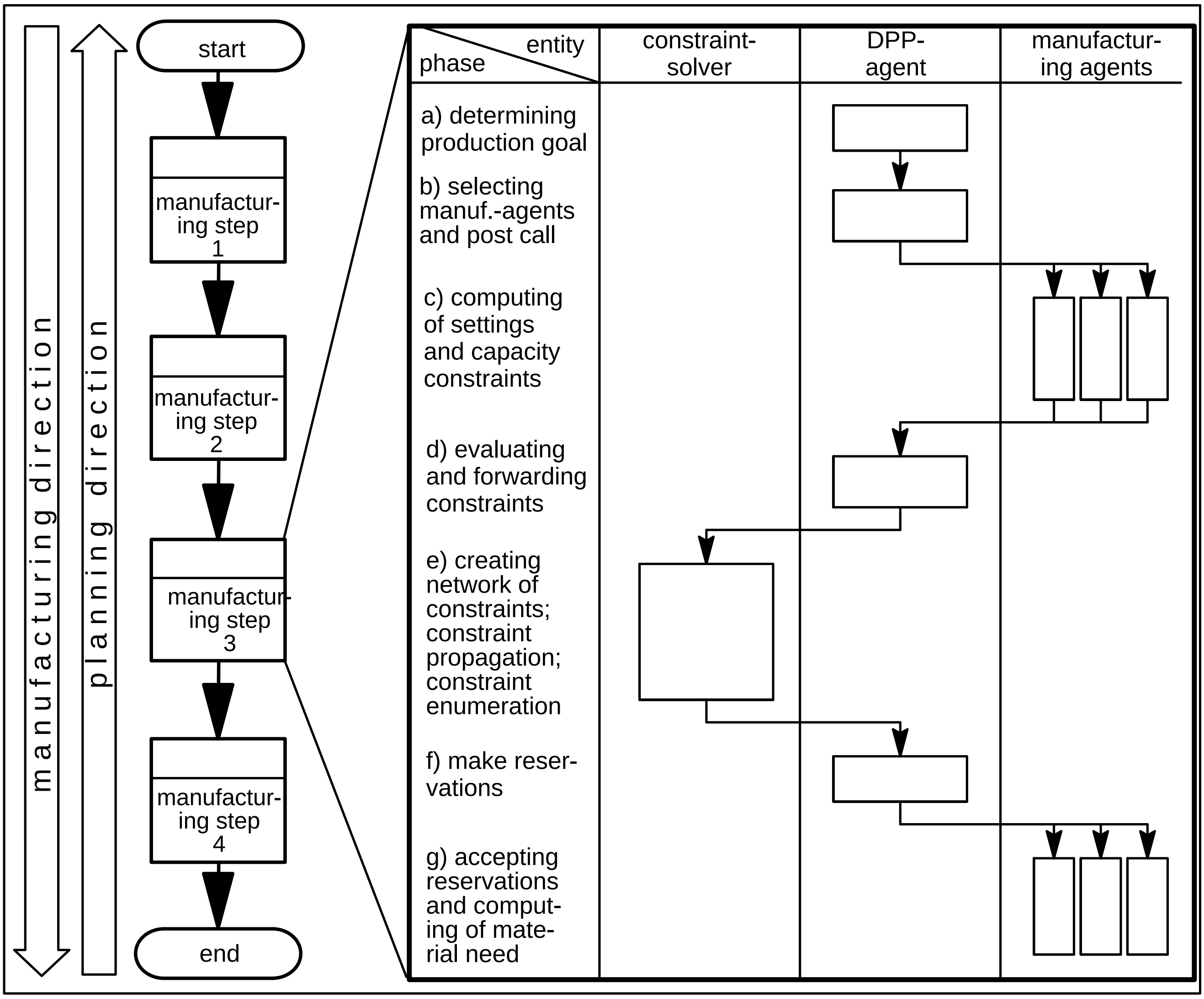

Figure 4: diagram of interaction

6. Evaluation and Future Work

In this paper, we presented a decision support system for the logistic planning of a factory that is based on concepts of multi–agent systems and constraint techniques. This approach meets the requirements put forward in section 2.

The factory structure and the manufacturing process are modelled by a multi–agent system representing each manufacturing center as an agent. This approach allows to encapsulate within an agent the structure of a center and its specific optimization strategy, and thus increases the modularity of the system. Furthermore, the use of a constraint logic as the basic language for communicating dependencies between manufacturing centers keeps the interactions of agents independent of the specific structure of their centers. Finally, the constraint logic chosen is, on the one hand, powerful enough to express dependencies between centers and, on the other hand, tractable so that the DPP–agent can use a standard constraint–solver in order to resolve the dependencies encountered during the coordination process.

The design of the decision support system as a multi–agent system with a one–to–one relation of manufacturing centers and agents also facilitates the incorporation of changes of the factory structure. A change in a manufacturing center, respectively in the connections of centers, only requires a change within the corresponding agents. Due to the use of a constraint logic, such a change does neither affect the overall system architecture nor the interactions of agents. Moreover, because of the correspondence between centers and agents, the agent system can be created automatically from production data.

An adaptation of the system's state to the current situation of the factory is supported by knowledge–based techniques. The explicit representation of factual and in particular procedural knowledge allows a flexible parametrization of the system. And by this, state changes can be incorporated faster than with conventional programming. Partly, these changes can be performed by a user.

All in all, the agent–oriented system meets the requirements stated for the PFC planning. Currently, a prototype is implemented that will be tested at the production site of Mercedes–Benz. For this test, the prototype will receive access to a real–life production data base. The test itself will be carried out by the logistics division.

The final decision support system also has to handle disturbances of the production process, such as for example breakdowns of transfer lines. In case of a disturbance, the system must determine alternative production strategies that will compensate the capacity loss and try to reach (as much as possible) the production goal. Even though the basic principle of the PFC planning process can be adopted for the disturbance handling, the agents must behave more reactively due to the dynamic nature of the production process. Our future work in this project will focus on developping appropriate methods for incorporating the disturbance handling into the current design.

References

[1] Ayel, Jacqueline: Decision Coordination in Production Management, in: Castelfranchi, Cristiano; Werner, Eric (Eds.): Artificial Social Systems, 4th European Workshop on Modelling Autonomous Agents in a Multi–Agent World (MAAMAW '92), Berlin 1994

[2] Bowen, J.; Bahler, D.: Conditional Variable Existence in Generalized Constraint Networks, in: Proceedings of 9th National Conference on Artificial Intelligence (AAAI '91), Volume I, Anaheim (CA) 1991

[3] Burmeister, Birgit; Sundermeyer, Kurt: Cooperative problem solving guided by intentions and perception, in: Demazeau, Yves; Müller, Jean–Pierre (Eds.): Decentralized A.I. 3, Proceedings of the Third European Workshop on Modelling Autonomous Agents in a Multi–Agent World (MAAMAW '91), Amsterdam 1992

[4] Dechter, R.: Constraint Networks, in: Shapiro, S.C. (ed.): Encyclopedia of Artificial Intelligence, Second Edition, p. 276–285, John Wiley & Sons, Inc., USA, 1992

[5] ILOG Solver Reference Manual Version 1.2, 1993, ILOG S.A., Gentilly Cedex 1993

[6] Jaffar, J.; Lassez, J.–L.: Constraint Logic Programming, in: Proceedings of the Fourteenth ACM Principles of Programming Languages Conference, Munich, January 1987

[7] Liu, JyiShane; Sycara, Katia: Emergent Constraint Satisfaction through Multi–Agent Coordinated Interaction, in: Ghedira, Khaled; Sprumont, Francois (Eds.): Pre–Proceedings of the 5th European Workshop on Modelling Autonomous Agents in a Multi–Agent World (MAAMAW '93), Universite de Neuchatel 1993

[8] Müller, T.; Popov, K.; Schulte, C.; Würtz, J.: Constraint Programming in Oz, DFKI Oz Documentation Series. Programming Systems Lab, German Research Center for Artificial Intelligence, Saarbrücken 1995

[9] Rosenschein, Jeffrey: Synchronization of Multi–Agent Plans, in: Bond, Alan; Gasser, Les: Readings in Distributed Artificial Intelligence, San Mateo (CA) 1988

[10] Sycara, Katia; Roth, S.; Sadeh, N.; Fox, M.: An Investigation into Distributed Constraint–directed Factory Scheduling, in: Proceedings of the 6th Conference on Artificial Intelligence Applications (CAIA '90), Los Alamitos (CA) 1990

[11] Van Dyke Parunak, H.: Distributed AI and Manufacturing Control: Some Issues and Insights, in: Demazeau, Yves; Müller, Jean–Pierre (Eds.): Decentralized A.I., Proceedings of the First European Workshop on Modelling Autonomous Agents in a Multi–Agent World (MAAMAW '89), Amsterdam 1990

[12] Wooldridge, Michael; Jennings, Nicholas: Agent Theories, Architectures, and Languages: A Survey, in: Wooldridge, Michael; Jennings, Nicholas (Eds.): Intelligent Agents, Workshop on Agent Theories, Architectures, and Languages (ECAI '94), Berlin 1995