

# Order Management in Non-Hierarchical Production Networks using Genetic Algorithms

Horst Meier, Tobias Teich, Harald Schallner

*Dept. of Mechanical Engineering  
Ruhr-Universität Bochum  
D-44780 Bochum  
Germany  
meier@lps.ruhr-uni-bochum.de  
schallner@lps.ruhr-uni-bochum.de*

*Dept. of Economics  
Technische Universität Chemnitz  
D-09107 Chemnitz  
Germany  
t.teich@wirtschaft.tu-chemnitz.de*

**Abstract:** The global economy of the 21st century will be characterized by competing corporate networks rather than by competing enterprises. Competition outside of and cooperation within supply chains will be the important factors. The tendency of companies to concentrate on their core competences causes a narrowing of their strategic focus, while at the same time these companies widen their focus by seeking alliances. Specialization and global networking are intensifying each other. The results are sinking costs in larger markets. The special research area 457 of the German Research Community is dealing with these topics. Another research project called INTERKON proposes an inter-organizational framework for collaborative manufacturing control in production networks. This paper combines both approaches. A short introduction gives a first insight into the problems that arise with the management and control of production networks. The purpose of this paper is to put forward a specific functionality to schedule production considering constraint networks.

**Key words:** production management, production networks, genetic algorithms

---

The original version of this chapter was revised: The copyright line was incorrect. This has been corrected. The Erratum to this chapter is available at DOI: [10.1007/978-0-387-35492-7\\_50](https://doi.org/10.1007/978-0-387-35492-7_50)

G. L. Kovács et al. (eds.), *Digital Enterprise Challenges*

© IFIP International Federation for Information Processing 2002

## 1. INTRODUCTION TO THE PROBLEM

A customer contacts a potential node of the network and releases a demand for the production of a specific good. The node has access to the front end of a model core and stimulates the genesis of a temporary network for the production of the good demanded. The information core is able to build directed graphs which stand for every adequate technological alternative for the production of the demanded good. For this purpose it uses information which is stored as fact- and rule-knowledge in an ontology-based database. Firstly, the different paths through the graph are built according to technological routings. Secondly, competence cells are assigned to the nodes of the network according to technological constraints. The objects of the graph are attributed. As a result, each node contains information on costs and times, while each directed edge contains information on logistics and quality.

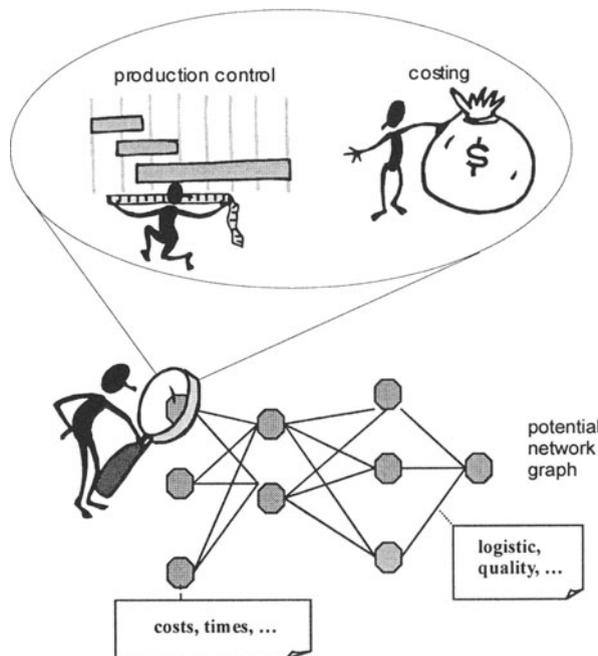


Figure 1. ASP-Functions

The task is to find a way through the graph which approximates to an unknown optimum for a time- and/or cost-based objective function. For the choice of efficient algorithms is necessary to know which class this problem belongs to. At first glance this may look like a Travelling Salesman Problem (TSP). A more detailed observation reveals that precedence relations between the nodes have to be modelled. Although sequencing problems can be

modelled as generalized TSPs in this case, the directed edges have to be weighed with distances which influence the value of the objective function. The calculation of these distances is difficult since all predecessors in the network influence the value of a directed edge. At the same time the nodes represent alternatives which influence the succeeding nodes. The problem described consists of elements of the CTSP and the JSP. Additionally, there is the problem of the alternative routings mentioned above. Thus it is an NP-hard problem. The routing within the network is managed by Ant-Colony-Optimization. This aspect will not be discussed in this paper. Instead of this, emphasis is put on the ASP-functionalities manufacturing control and costs (figure 1).

The routing must not be understood as a static optimization. It aims at getting process information on times and costs within a competence cell instead. In the following manufacturing control is focussed. Once the network of competence cell alternatives is generated, the nodes have to be attributed with times. At this point the network management should not only be allowed to ask for offers because this might result in an informational delay which might lead to the well-known bullwhip effect. The solution proposed here is a direct contact with the competence cell by means of a special ASP-functionality. The network management stimulates the reaction of a front end via standardized interfaces (like e.g. XMI). The front end schedules the order flow. This can be done by a virtual loading of the new request into the real production system. Fast intelligent technologies have to be used to compute the earliest due date. Genetic Algorithms (GA) could be one such method.

It is the task of shop floor scheduling to find sequences in which  $n$  jobs are to be processed on a set of  $m$  resources (machines, staff, special tools etc.) so that several constraints (technological, organizational etc.) are satisfied and a given objective function (e.g. mean flow time, mean lateness etc.) is maximized. Each job can consist of one or more operations. The technological sequence of the operations within a job is not necessarily set. Thus real-world shop floor scheduling comprises many different combinatorial optimization problems like Job Shop, Flow Shop, Open Shop, Parallel Machines Problems and many other. While models for these problems and heuristics to solve them had been developed in operations research, it was still not possible to handle shop floor scheduling as a whole. Even today it is mostly done manually. With the help of evolutionary methods and other meta-heuristics like Simulated Annealing or Threshold Accepting, there seemed to be hope for a change of this situation for the first time.

During the last ten years a number of attempts has been made world-wide to solve hard combinatorial optimization problems in the field of production control with the help of Genetic Algorithms (GA) [2, 3, 8, 13, 14]. However,

real-world applications in the field of shop floor scheduling are still rare, although the results shown in benchmark tests have been far better than those of the best priority rule heuristics. What is the reason for this?

There is the problem of a difference between theoretical research work and the development of a real-world application. However this is not the main point. More important is the fact that benchmark problems are rather small compared to real-world ones and that they do not contain those many, little extra problems and degrees of freedom which increase the complexity of real-world shop floor scheduling. This complexity is raised to the power of ten compared with e.g. the job shop benchmark problems that are often used for testing scheduling GAs. Regarding this high complexity the question of the time needed to calculate a satisfying solution is of highest interest for a practical use of EA-based scheduling tools. Often there are demands for calculation times no longer than five minutes, as shown in a study carried out by the authors [12]. There is no reason to believe that the solution an Evolutionary Algorithm delivers after five minutes could not be improved further. Therefore we decided to let the algorithm run continuously. There are two problems that have to be solved to do this.

Firstly, there has to be an organizational framework which allows companies to make use of a continuously running Evolutionary Algorithm for shop floor scheduling. Since in today's organizations scheduling is usually done just once at the beginning of a planning period, while afterwards only minor changes are allowed, companies are not prepared to handle new improved schedules delivered by an EA after the first schedule has been fixed and its sequencing information has been distributed to every worker. To change this, the authors developed an organizational framework in which these new improved schedules can easily be handled. However, since these matters do not lie within the field of Evolutionary Algorithms, they are not discussed further in this paper.

Secondly, there is the Evolutionary Algorithm itself. Since situations in production systems may change frequently, no set parameter constellation seems appropriate. Parameters should be the matter of a co-evolution process to make them adaptable. Last but not least, the computational resources companies can use for shop floor scheduling are often rather limited. Therefore a simple approach of parallelization using intelligent terminals of a plant data acquisition system (PDA) has been developed and tested successfully.

## **2. EVOLUTIONARY SCHEDULING ALGORITHM**

The shop floor scheduling EA introduced by the authors consists of two parts: a Genetic Algorithm for the schedule optimization and a Evolutionary

Algorithm for the parameter co-evolution. A GA, developed by the authors in 1998 for job shop problems, was used for schedule optimization [5]. It was built upon a genetic representation of permutations with repetition. This approach was developed by Bierwirth et al. in 1993 [1].

For each operation the corresponding job occurs within a chromosome. Thus it is possible to avoid invalid solutions without a complicated evaluation procedure. In order to build a semi-active schedule, the chromosome has to be read only from the beginning to the end. Each operation has to be placed on the necessary resources at the earliest possible starting date at which all resources have to be available. The predecessor of the operation within the same job must be finished. The latter condition could be weakened because sometimes there are operations whose sequence is not necessarily set. For crossover we used the Generalized Order Crossover (GOX) developed by Bierwirth et al. in 1993 and the Generalized Position Based Crossover (GPBX) developed by the author [12]. Therefore possible disadvantages of the use of just one crossover operator shall be overcome. Experiments carried out by the author [12] seem to justify this approach because the mean results reached by combined use of the two operators were better than the results reached by using single operators. Both GOX and GPBX are generalized versions of crossover operators which were originally developed by Syswerda [11] for the Travelling Salesman Problem. Syswerda's attempt to preserve order and absolute position information respectively is unfortunately partly undermined by Bierwirth's genetic representation. The limitation is that actual order/ position information of operations shall be saved, whereas the chromosome contains job numbers to avoid invalid solutions. This dilemma on the one hand and the high redundancy of the representation which increases the dimension of the search space by some additional powers of ten on the other hand make it worthwhile to use a more powerful scheduling GA in the future. Therefore the authors are currently comparing the concepts of van Bael [13] and Mattfeld [8] with regard to their suitability for real-world shop floor scheduling. As regards mutation the scheduling GA uses a reservoir of no less than six operators which differ in their power to destroy the genetic information. A shift operator which takes only one operation out of the chromosome and moves it to another position and a scramble operator which changes the whole chromosome at once are the two opposite ends of a scale these operators can be placed on. For mutation one of them is chosen according to the same principle as employed with the crossover operator. The same concept is applied with selection schemes. The idea behind the implementation of this variety of operators is to produce a robust algorithm which avoids a problem that GAs often have when using only a few operators. They do well with one problem but fail to deliver satisfying solutions with another one. With its variety of operators this algorithm

proves to be robust, yet at the cost of an inferior quality of solutions for benchmark tests. A first version of this algorithm was implemented in 1998 as a generational replacement GA. Results concerning this algorithm can be found in [5]. For easier parallelization it was later changed to a steady state GA. New solutions are accepted if they are better than the worst individual or they are accepted with a certain probability if they are worse. This concept was originally used in Simulated Annealing [7]. The probability is lowered during the course of the optimization according to what is called a cooling schedule in Simulated Annealing. The parameters which determine the cooling schedule can themselves be changed by co-evolution. The idea of linking the Simulated Annealing Concept with a steady state GA was developed because of the excellent results the authors reached with a simple Simulated Annealing procedure on the Fisher-Thompson benchmark problems for the JSP [6, 10].

The schedule optimization GA uses a weighed combination of mean lateness and mean flow time as objective function. Job priorities can easily be taken into account. For the simulation tests both objectives were weighed equally. The results are shown at the end of this paper. Parameters for the schedule optimization are represented as integer values within the chromosome which also carries parameters for the co-evolution process. The schedule optimization GA is run until there are no improvements within the last 400 iterations. Then its parameters are altered by a normal distributed mutation operator as part of a simple (1+1)-Evolution Strategy (ES). The scheduling GA is run again with the new set of parameters until there are no improvements throughout the last 400 iterations. Then the former parent and offspring are compared and the better one is chosen for a new mutation.

### **3. INTER-ORGANIZATIONAL FRAMEWORK**

The non-hierarchical structure of the production networks examined here demands a collaborative decision-making. In this case coordination is not carried out through hierarchical instructions, but is established through bi- and multilateral decision processes respectively. A multitude of competence cells with different interests and objectives takes part in these processes. Consequently, inter-organizational manufacturing control is carried out via collaborative agreements on detailed order conditions and across several manufacturing levels.

The order conditions (e.g. due date, used material, reserved capacities, and reserved stock etc.) are formulated as constraints that influence the manufacturing control of more than one enterprise. This means that bilateral constraints have to be communicated and committed by all decision-makers

involved. However, identifying and quantifying the implications of each decision that must be taken in conjunction with all but the smallest manufacturing level is not humanly possible. This makes an ASP-functionality necessary that performs "what-if" analyses for each competence cell before making local production decisions. The functionality also supports the interaction process with Internet as the medium among different companies.

A developed model editor offers intelligent support to help companies model their manufacturing process and the features they require. This approach is based on the "Reference Model for Shop Floor Production" according to the ISO-Standard (Workgroup TC184/SC5/WG1) [4]. This model comprises variables for each relevant manufacturing issue. Each process step is modelled on a very general level as an activity with the input or output of information, material, resource and control information. Thus there is the differentiation between the four generic processes of "transform", "transport", "store" and "verify". In addition, the relationships between manufacturing issues are formulated as constraints and the desired objective is defined. The ASP-functionality checks such factors like manufacturing capacities and parts orders to decide whether a new order can be inserted into the production schedule and still satisfies the resource constraints and the order conditions, e.g. the requested date. Scheduling is understood as a process of assigning activities to resources in time.

Besides, an interaction protocol has to be defined for each customer-supplier relation in order to coordinate the supply of materials with workflow and resources to optimize production. The manufacturing control is specified individually by the coordination strategy and the degree of integration.

The different degrees of integration classify five inter-organizational coordination patterns according to the customer order decoupling point, see [9]. The customer order decoupling point determines the temporal and structural limit between the production with a concrete customer order assignment and the production for an anonymous customer on a forecast basis. The limit is often selected according to the delivery times required on the part of the market. These must not be exceeded by the throughput times of the customer order assigned production and delivery processes. There is no direct customer's reference to the production steps before the customer order decoupling point. Therefore the maximum degree of integration of the customer-supplier relation - which can be selected sensibly - derives from the customer order decoupling point that depends on the individual organization structure. Thus we differentiate between the following five coordination patterns:

1. Integration degree: Deliver-to-stock  
Current and future requirements of the customer are communicated to the supplier so that an early planning of the demand coverage can take place. Thus the responsibility for the material availability is assigned to the supplier. Consequently, the customer can reduce the complexity of his order coordination since the material availability concerning bought-in parts is contractually guaranteed.
2. Integration degree: Make-to-stock  
It is the task of the classic customer-supplier relation of order-related delivery to agree upon order conditions concerning the product, the price, the delivery date, the quantity and quality as well as the place. The customer orders are mainly processed via stockpiled shelf inventories which have been produced beforehand, based on forecasting.
3. Integration degree: Assemble-to-order  
A closer involvement of the supplier is possible with the order-related final assembly since the final assembly of the components does not begin before the order placing. The necessary components are produced in preceding manufacturing steps on the basis of sales forecasts. They can be stored temporarily. Apart from the order conditions of the classic customer-supplier relation the reservation of capacities of the assembly can be committed to.
4. Integration degree: Make-to-order  
The synchronization of the order coordination with the order-related manufacturing goes a step further. Here a customer-specific manufacturing is possible, a reservation of the manufacturing capacities needed for the customer order can be committed to. The lot size that is to be produced is thus determined via the order quantity. The necessary material is provided through ordering processes or through the storehouse.
5. Integration degree: Source-to-order  
In this case bought-in parts are individually ordered, according to the customer orders. Alternatively, a reservation of bought-in parts can be arranged with the customer who also has the possibility to consign the material necessary for the production. Thus the supplier can be completely integrated into the manufacturing control. Therefore the customer arranges the complete order-specific manufacturing process as regards delivery dates, capacities and material.

With every degree of integration the transparency of the production processes increases. The possibility for a close-to-reality planning of intercorporation order management grows accordingly. At the same time the demand of information as well as an exceeded planning horizon and a greater

planning complexity increase. Figure 2 gives an example of a production network whose material flow is coordinated by different integration degrees.

The communication protocol for the exchange of bilateral constraints is modelled in the form of statechart diagrams. Message types of the international electronic data interchange standard (EDIFACT) are used and others are added in order to perform the whole range of constraint exchange. A statechart diagram models the business rules by defining possible sequences of states and actions as a result of the reaction to incoming messages.

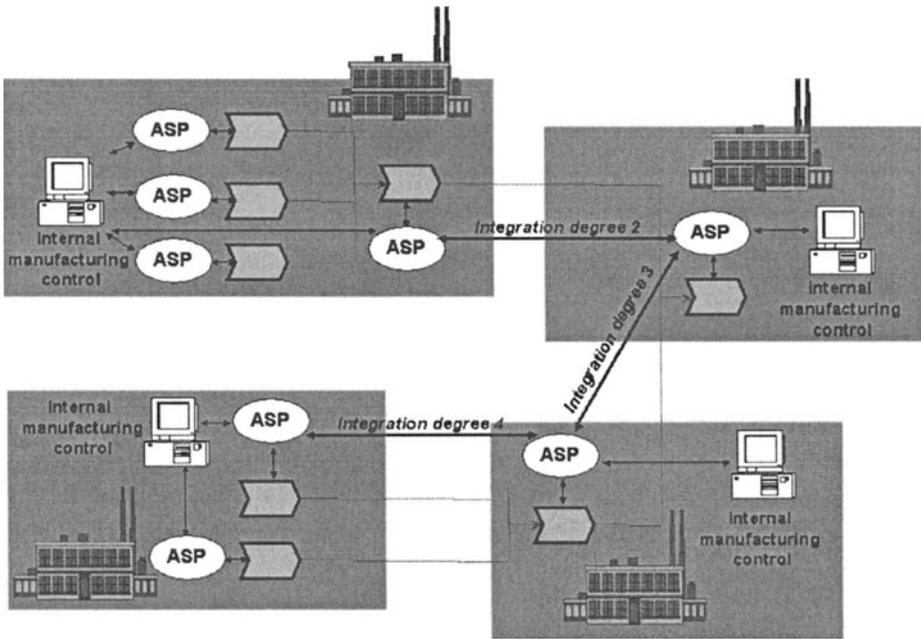


Figure 2. Production network (sample)

#### 4. AN INTRA-ORGANIZATIONAL FRAMEWORK

The next step is the integration of this software-system into an organizational framework. The authors carried out a study [6] to identify objectives used in companies for manufacturing control and to develop a ranking according to their importance. It turned out that it is primarily important to keep due dates for as many jobs as possible. A short mean flow time is less important. This latter objective is equivalent to other objectives like low inventory and high throughput.

Thus due dates and mean flow times seem to be the two most useful basic components for an objective function for schedule optimization. In order to calculate them the scheduling tool needs due dates for each job and each production area as input from the ERP-system. Sometimes this causes problems in companies with more than one production area where all areas need to have their separate scheduler. Many PPC-Systems calculate due dates with no regard to capacities. Others calculate virtually exact due dates for each operation but cannot distinguish where a production area ends and where the next one starts. Finally there are systems which do not keep records of the next job, for which one component is needed for. This makes it impossible for the scheduling system to derive a network of operations needed for the final product from the data that is kept within the ERP-/PPC-System. Therefore the rough capacity scheduling in most PPC-systems has to be changed to a certain extent to supply the data needed for manufacturing control.

During the installation process of the scheduling system, objective functions have to be adapted to the special needs of the company. The responsible person can choose from a set of objective constellations covering usual situations. When the scheduling system is started for the first time, all jobs currently in production or released for production are read from the ERP-system and distributed to different work stocks for the different schedulers - one for each production area. Then the schedulers are started. They are permitted to work for five minutes before their first solution is distributed to the PDA-terminals. The solution is set active.

The start of work has to be reported to the scheduler via the bi-directional PDA-system. Jobs which are currently being processed are no longer part of the scheduling process. Workers are allowed to choose the next job they want to work on from a number of  $n$  jobs. This is because up to now we are not able to build minimal setup time sequences due to the lack of data. Although the algorithm could easily handle sequence dependent setup times, there are only a few companies that record this data.

While the work process is running, the scheduler's EA is continuously searching for improved schedules. If a better schedule is found, it is set active and the planned work sequence is changed accordingly on all machines. This might cause a certain amount of disturbance within a manufacturing system because job floor workers usually prepare for their next job while the actual job is still in process. Therefore the  $n$  jobs appearing on the PDA-terminal are being "specified". This results in the fact that the sequence cannot be altered by the EA any more. Therefore the EA is given the initial five minutes to make sure that a satisfying solution is the base of the first specification of the jobs on a machine. If a job is completed on one machine or if some disturbance occurs, the worker has to report this via the PDA-

terminal. This information is sent back to the ERP-system where it is ready for further use. The scheduling algorithm is also informed and performs its task on this changed database. The same thing happens if a supervisor decides to move a job from one machine which is currently overloaded to an alternative machine or if new jobs are released within the ERP-system. Thus the whole system is working continuously allowing to use the power of the EA but also relying on intuition and experience to handle exceptions.

## 5. SUMMARY

It has been argued by the authors that modern shop floor scheduling tools based on Evolutionary Algorithms prove their potential for solving complex benchmark scheduling problems but failed to be used widely in real-world applications throughout the last ten years. One of the reasons identified is the need for organizational changes in manufacturing control caused by systems. The authors are proposing an organizational framework which includes bi-directional PDA-systems in which EA-based shop floor scheduling tools can be successfully implemented. A simulation study has shown the potential of the concept. A real-world implementation is currently being developed in cooperation with two German software companies and one mechanical engineering company.

In addition, such coordination patterns are proposed which make it possible to integrate interdependencies of manufacturing control between collaborating companies. An integration platform with ASP-functionality is being developed to link companies up to a non-hierarchical regional network. A model-based approach is chosen with which companies are able to configure their customer-supplier relations according to their specific organizational needs.

## REFERENCES

- [1] Bierwirth, C., Kopfer, H., Mattfeld, D., Utecht, T.: *Genetische Algorithmen und das Problem der Maschinenbelegung*. Universität Bremen, Lehrstuhl für Logistik, Bremen (1993)
- [2] Bierwirth, C.: A Generalized Permutation Approach to Job Shop Scheduling with Genetic Algorithms. *OR Spektrum* vol. 17 (1995) pp. 87-92
- [3] Davis, L.: Job shop scheduling with Genetic Algorithms. In Grefenstette, J. (ed): *Proceedings of the International Conference on Genetic Algorithms and their Applications*, Hillsdale, Lawrence Erlbaum Associates (1985) pp. 136-140
- [4] ISO TC184/SC5/WG1: *ISO Reference Model for Shop Floor Production Standards Part 1&2*. ISO, ANSI-NEMA, Washington DC (USA) 1989, (1990)

- [5] Käschel, J., Teich, T., Zacher, B.: An empirical study of manual shop floor scheduling in job shop environments. Working Report 30/2000. University of Technology of Chemnitz, Department of Economics, Chemnitz (2000)
- [6] Käschel, J.; Teich, T.; Meier, B., Fischer, M.: Real-World Applications: Evolutionary Real-World Shop Floor Scheduling using Parallelization and Parameter Coevolution? In: Banzhaf, et. al. (publ.), Proceedings of the Genetic and Evolutionary Computation Conference, Las Vegas, Nevada, July 8-12, 2000, pp. 697-701, Morgan Kaufmann
- [7] Kirkpatrick, S., Gelatt, C., Vecchi, M.: Optimization by Simulated Annealing. *Science*, Vol.220 (1983) pp. 671-680
- [8] Mattfeld, D.C.: Scalable Search Spaces for Scheduling Problems. In: Banzhaf, W. et al. (eds): Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'99), Morgan Kaufman (1999) pp. 1616-1621
- [9] Mertens, P.: Integrierte Informationsverarbeitung. Gabler, Wiesbaden (2000)
- [10] Muth, J. F., Thompson, G.L.: Industrial Scheduling. Prentice Hall, Englewood Cliffs, New Jersey (1963)
- [11] Syswerda, G.: Uniform Crossover in Genetic Algorithms. In: Schaffer, J.D. (ed): Proceedings of the Third International Conference on Genetic Algorithms and their Applications, San Mateo, 1989, Morgan Kaufman (1989) pp. 2-9
- [12] Teich, T.: Optimierung von Maschinenbelegungsplänen unter Benutzung heuristischer Verfahren. Verlag Josef Eul, Lohmar, Köln (1998)
- [13] Van Bael, P. et al.: The Job Shop Problem solved with simple, basic evolutionary search elements. In: Banzhaf, W., et al. (eds): Proceedings of the Genetic and Evolutionary Computation Conference (GECCO'99), Morgan Kaufman (1999) pp. 665-669
- [14] Yamada, T., Nakano, R.: Scheduling by Genetic Local Search with Multi-Step Crossover. In: Proceedings of the Fourth International Conference on Parallel Problem Solving from Nature (PPSN IV), Berlin (1996) pp. 960-969

## BIOGRAPHY

**Prof. Dr.-Ing. H. Meier:** Born 15. April 1951 in Pivitsheide, Germany. Mechanical Engineering studies at the TU Berlin; From 1977 to 1983 research assistant at the "Institut für Werkzeugmaschinen und Fertigungstechnik" in Berlin; 1981 Doctor of Mechanical Engineering by Prof. Dr.-Ing. G. Spur at the TU Berlin; From 1983 to 1995 leading position in the mid-size company "Schleicher & Co."; From 1995 to 1999 holder of the chair "Automatisierungstechnik" in Cottbus. Since 1999 holder of the chair "Production Systems" at the Ruhr-Universität Bochum.

**Dr. T. Teich:** Computer Science studies at the TU Chemnitz, Germany; Since 1994 research assistant at chair "Produktionswirtschaft und Industriebetriebslehre" (TU Chemnitz); 1998 Doctor "rer. pol." by Prof. Dr. J. Käschel.

**H. Schallner:** Born 22. September 1969 at Münster, Germany. Computer Science studies at the University Paderborn; Since October 1996 research assistant at the Automation Research Institute (Ruhr-Universität Bochum).