Proceedings of the 2013 Winter Simulation Conference
R. Pasupathy, S.-H. Kim, A. Tolk, R. Hill, and M. E. Kuhl, eds

A HEURISTIC TO SUPPORT MAKE-TO-STOCK, ASSEMBLE-TO-ORDER, AND MAKE-TO-ORDER DECISIONS IN SEMICONDUCTOR SUPPLY CHAINS

Lisa Forstner

Supply Chain Management
Am Campeon 1-12
Infineon Technologies AG

Neubiberg, 85579, GERMANY

Lars Mönch

Department of Mathematics and Computer Science
Universitätsstraße 1
University of Hagen

Hagen, 58097, GERMANY

ABSTRACT

In this paper, we study Make-to-stock, Assemble-to-order, and Make-to-order decisions in semiconductor supply chains. We propose a genetic algorithm to support such decisions. Discrete-event simulation is used to estimate the profit-based objective function taking into account the stochastic behavior of the supply chain. We perform computational experiments with a simplified semiconductor supply chain model. It is shown that the proposed heuristic outperforms simple partitioning heuristics based on product characteristics.

1 INTRODUCTION

Semiconductor manufacturing processes are highly complex. A semiconductor chip is an integrated circuit consisting of a huge number of transistors. The manufacturing stages can be divided into two major segments. The frontend comprises wafer fabrication and wafer test while the backend is split into chip assembly and final test. One characteristic of the semiconductor industry is the reentrant material flow within wafer fabrication. In addition, the capital-intensive machines, long production cycle times, volatile demand, continuous cost and price pressure, high degree of product variants, fast changing up- and downturns as well as short product life cycles are typical for this industry (cf. Mönch et al. 2012).

To deal with this dynamic environment, semiconductor companies have to react quickly on the changing needs of the customers regarding product type and quantity, but at the same time, they also seek to keep costs as low as possible. Low inventory levels might lead to a poor delivery performance, whereas keeping inventory levels high increases the risk of obsolescence and leads to higher capital commitments. The main drivers to assign suitable production strategies to products are the characteristics of the semiconductor industry already described together with the aim to improve the performance in terms of costs and customer service. In this paper, we consider make-to-order (MTO), assemble-to-order (ATO), and make-to-stock (MTS) as production strategies. Products are produced forecast-driven until they are completely finished in the case of MTS. In the case of ATO, products are produced forecast-driven until the point right before it comes to the assembly. Starting from there, the production continues based on a customer order. The MTO strategy is characterized by an order-driven production (cf. Wemmerlöv 1984, Federgruen and Katalan 1999 amongst others).

While production strategy decisions are discussed to a certain extent in the literature for other industries like, for instance, the food processing industry, this is not the case for the semiconductor industry with the rare exception of the paper by Sun et al. (2010) where simulation is used to assess the performance of production strategies in a semiconductor supply chain. In this paper, we propose a genetic algorithm-based heuristic to assign a production strategy to each product, i.e., we do some partitioning for the

set of all products over a certain planning horizon. Discrete-event simulation is used to take into account the effect of partitioning decisions on the cycle time of the products.

The paper is organized as follows. We describe the researched problem in Section 2. In addition, related literature is discussed. Two partitioning heuristics are presented in Section 3. The results of extensive computational experiments are described and discussed in Section 4.

2 MAKE-TO-STOCK, ASSEMBLE-TO-ORDER, AND MAKE-TO-ORDER DECISIONS

2.1 Problem Setting

We consider a simplified semiconductor supply chain with P different products. It consists of a raw wafer storage, one frontend facility, one die bank (DB) to store semi-finished products, one backend facility, and finally one distribution center (DC) for finished products. The frontend and the backend facility have machines and operators as resources. Our supply chain model consists of two sections. The first section consists of a production in a frontend facility and a transport from the frontend to the DB, while the second section includes the transport from the DB to the backend facility, the backend production, and the transport from the backend to the distribution center. The overall situation is shown in Figure 1. In addition, the possible production strategies are also depicted in Figure 1.

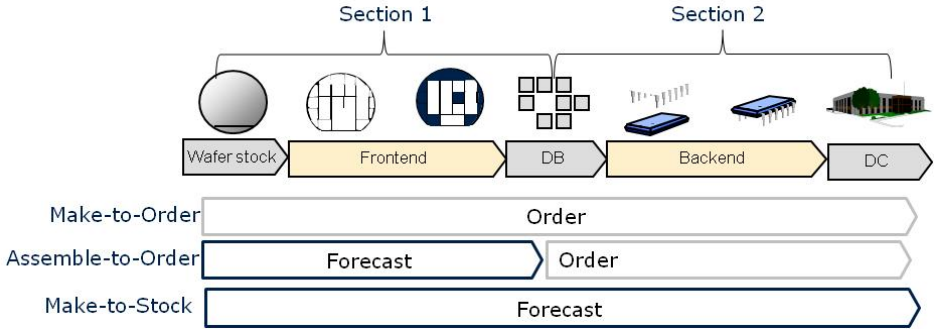

Figure 1: Base system of the simplified supply chain

Each product $1 \le p \le P$ has a production route, a transportation time, and a planned replenishment time for each of the two sections. The final demand $d_{pt}^{(f)}$ of product p is defined as the requested order quantity in the delivery week t. An individual order lead time l_p is assigned to each product p. The order lead time is the time span between the delivery week and the week in which the order was placed. We have a forecast $d_{pt}^{(r)}$ for each product p and for each delivery week t. The production cycle time (CT) is the period of time from releasing a lot until the delivery time into the DC. Based on the CT values, we also consider a planned lead time. We assume that safety stocks do not exist.

We consider a planning horizon of length T where the periods t are equal to one week. The number of lots that need to be started is calculated separately for each section. On the one hand, raw wafers are taken out from the raw wafer storage to start lots in the frontend facility. Here, we assume for the sake of simplicity to have an infinite supply available with zero costs. On the other hand, lots that need to be released from the DB depend on the replenishment of the first section. Since the replenishment times for each section are greater than one period, replenishment has to be triggered ahead of the due week according to the replenishment time. Each product has in each section a frozen interval. Operations and production quantities cannot be changed within this frozen interval. We assume the length of this interval to be equal to the replenishment time of the particular product.

The number of lots to be released into the corresponding sections is calculated at the beginning of each period. Information on demand, inventory, work-in-process (WIP), and backlog is taken into account. MTO products are only produced based on orders from raw wafer storage until the distribution center. ATO products are produced forecast-driven until the DB. But if orders exceed the forecast we allow to produce these orders as well. From DB, arriving customer orders enable to continue production until the end. MTS products are produced forecast-driven along the entire supply chain (see Figure 1). Here, we also allow the production of orders if they exceed the forecast. In this paper, we are interested in determining a partitioning

$$s := (S_1, \ldots, S_P) \tag{1}$$

of the products such that $S_p \in \{0,1,2\}$, $p = 1, \ldots, P$ and the profit

$$Z(s) := \sum_{p=1}^{P} \sum_{t=1}^{T} \left(rev_{pt} U_{pt}(s) - b_{pt} B_{pt}(s) - \sum_{k=1}^{K} h_{kpt} I_{kpt}(s) - c_{pt} M_{pt}(s) - \tilde{c}_{pt} U_{pt}(s) \right) \tag{2}$$

is maximized. Here, the setting $S_p = 2$ refers to a MTS strategy for product p, while $S_p = 1$ and $S_p = 0$ are used to model an ATO or MTO strategy, respectively. The following notation is used in expression (2):

rev_{pt}	:	expected revenue per chip of product p in period t
$U_{pt}(s)$	:	number of sold chips of product p in period t
b_{pt}	:	cost for one chip due to unmet demand of product p postponed from period t to $t+1$
$B_{pt}(s)$	:	backlog of demand for product p at the end of period t (in chips)
h_{kpt}	:	inventory costs for holding one chip of product p within period t in storage location k, $k = 1, \ldots, K$
I_{kpt}	:	inventory level of product p at the end of period t in storage location k, $k = 1, \ldots, K$ in chips
c_{pt}	:	manufacturing cost per chip of product p in period t
$M_{pt}(s)$	:	number of chips of product p that are in WIP in period t
$\tilde{c}_{pt}$	:	overall processing costs that are charged if one chip of product p is sold in period t.

Note that we have to take the stochastic behavior of the base system into account when we evaluate the $Z(s)$ value for a given s. We abbreviate the described partitioning problem by PP in the remainder of this paper.

2.2 Related Literature

There are several papers that address production strategy-related partitioning problems. Hoekstra et al. (1992) propose the customer order decoupling point (CODP) concept that focuses on market, product, and process-related factors to make decisions on the production strategy. The categorization of the factors and parts of the proposed concept have been used and extended by other researchers. Olhager (2003) points out the strategic importance of the decision whether products should be produced MTO, ATO, or MTS using the notion of the order penetration point (OPP). Hemmati and Rabbani (2009) use the analytic network process to make production strategy decisions. Similar factors as in Hoekstra et al. (1992) are taken into account. Interdependencies among these factors are considered.

Soman et al. (2004) discuss MTS and MTO partitioning decisions in the food processing industry. A decision support system for managers taking rough capacity constraints into account is described by van Donk et al. (2005). But congestion effects are neglected in this paper.

In contrast to this, congestion effects are considered explicitly in a series of papers that are heavily relying on queueing theory. For instance, Rajagopalan (2004) makes decisions whether products should be produced following a MTO or a MTS strategy. The production facility is modeled as a M/G/1 queue. A nonlinear integer programming formulation of the problem is provided. Because of the computational burden of this approach, an additional heuristic is proposed. Gupta and Benjaafar (2004) develop models to research the benefits and costs of delaying differentiation in series production systems where order lead times are load-dependent using results from queueing theory.

It is well-known that queueing theory has some limitations when it is applied in highly complex, reentrant manufacturing systems (cf. Shanthikumar et al. 2007). Therefore, it might be reasonable to apply discrete-event simulation to incorporate capacity constraints and hence to model congestion effects. Sun et al. (2010) investigate the problem of selecting appropriate production strategies in the semiconductor industry from a more strategic point of view using simulation. Based on their results, customer-requested lead time and the importance of on-time delivery are the main drivers for this type of decisions. Demand pattern and process variability are less important. A hierarchical decision support framework is proposed to offer recommendations on a more conceptual level. Consequently, concrete partitioning decisions are not derived for a given demand scenario.

In the present paper, we propose to make such decisions using a genetic algorithm (GA). A detailed discrete-event simulation model of a simplified supply chain is used to determine the corresponding objective function value in the presence of machine breakdowns. To the best of our knowledge, such an approach is not described in the literature so far.

3 PARTITIONING HEURISTIC

3.1 Reference Approach

We have to compare the results provided by the GA with results obtained by other methods. Because there is no specific approach for semiconductor manufacturing available in the literature (see the discussion in Subsection 2.2), we compare our results against the general-purpose approach proposed by Olhager (2003). Two major factors are considered to make the MTO, ATO, and MTS decision. The first one is the production lead time to order lead time (P/D) ratio, while the second one is the relative demand volatility, also called coefficient of variation (CoV). Each of these two factors is divided into two subcategories. Figure 2 depicts the possible categories with the recommended strategy.

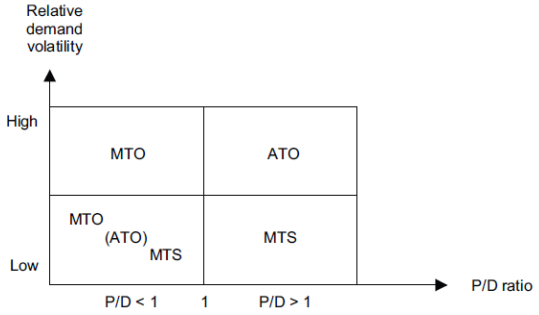

Figure 2: Matrix for partitioning decisions according to Olhager (2003)

A CoV of 0.1 and of 0.25 are considered as a low value, whereas a CoV of 0.5 refers to a high relative demand volatility. We can see from Figure 2, for instance, that when the production lead time is smaller than the order lead time and the demand volatility of this product is high then the MTO production strategy is chosen for the corresponding product. This method has the drawback that we need information on the product lead times that are load-dependent, hence it is hard to estimate them. On the other hand, the dependencies between the different products are not taken into account.

3.2 GA-based Method

A GA is a population-based metaheuristic (cf. Michalewicz 1996). GAs are successfully applied to solve large-scale, hard optimization problems in manufacturing. A GA maintains within each generation, i.e. iteration, a population of chromosomes where each chromosome represents a solution to the PP. We use the vector of size P from the right-hand side of equation (1) to represent a solution. Hence, each gene of the chromosome represents the production strategy of a specific product. The values of a gene, i.e. the alleles, are from $\{0,1,2\}$ and indicate the different production strategies. An initial population is created to start the GA by randomly selecting chromosomes. A fitness value derived from the objective function (2) is assigned to each chromosome of the population. We use discrete-event simulation to compute the objective function value for a specific partitioning. The simulation model is described in detail in Subsection 4.1. We assign a production strategy according to the chromosome to each of the products in the simulation model.

Variation and selection operators are used to modify the individuals of a population from iteration to iteration. All the chromosomes of the current population are available for mating. We start by describing how the crossover operator works. Two parent chromosomes are selected from the population according to a selection scheme. A coin is flipped with the crossover probability pc to determine whether a crossover is performed or not. In the first case, the parent chromosomes are copied directly as offspring. In the latter case, we use the one-point crossover operator to create the new offspring. We select $z \sim DU[1, P]$, where $DU[a,b]$ denotes a discrete uniform distribution over $\{a,\ldots,b\}$ for $a,b \in IN$. The two parent chromosomes are then crossed at position z to create two new offspring. Mutation is applied to the offspring by an operator called flip mutator. A coin is flipped for each gene with the mutation probability pm. If mutation needs to be applied to a gene, the allele is switched to any value from $\{0,1,2\}$. Mutation is used to avoid a premature convergence of the GA towards a local optimum.

We use the roulette wheel method (cf. Michalewicz 1996) for selecting chromosomes that are used for mating. In this approach, a probability is assigned to each chromosome that is directly proportional to the fitness value of the chromosome. The higher the fitness value, the higher is the probability to be selected.

A steady state GA with overlapping population is applied. The amount of overlap is specified by the replacement probability pr. A temporary copy of the current population is derived, and newly generated offspring is added. The worst chromosomes with respect to their fitness values, either from the offspring or from the current population, are discarded and the population size shrinks back to its initial size. The GA is finished if a termination criterion is fulfilled. In this research, we simply use a prescribed number of generations as termination criterion.

4 COMPUTATIONAL EXPERIMENTS

4.1 Simulation Model and Design of Experiments

We use the MIMAC-I data set (cf. MASM 1997) for the frontend and a slightly modified version of the backend model briefly described by Ehm et al. (2011) as the base simulation model of our simple supply chain. The model contains around 280 machines that form 83 work centers. Each product has around 250 processing steps. One DB and one DC are added. We increase the amount of products from two to ten, while we keep the two original production routes. An individual production strategy can be assigned to each product. Among the ten products five have a planned CT of eight weeks, while the remaining products have a planned CT of seven weeks.

A simple planning logic is implemented to incorporate MTO, ATO, and MTS production strategies. Determining the gross demand works differently for the three production strategies. The gross demand is the quantity that is requested to fulfill either the forecast and/or the order of a customer, where the net

demand is the quantity that needs to be replenished after WIP and already available inventory is subtracted from the gross demand. If the production strategy of a product is MTO, the gross demand is calculated based only on orders for the frontend and backend section. The gross demand for the backend section is based on orders in case of ATO, while the gross demand for the frontend section is based on forecasts in the ATO setting. For MTS, the gross demand is calculated based on forecasts. The release schedules for the frontend facility is determined based on a backward calculation scheme taking into account the planned lead time.

The planned bottleneck in the frontend consists of several stepper machines. We assume that the mean-time-to-failure (MTTF) and the mean-time-to-repair (MTTR) are exponentially distributed. Each lot in the frontend facility contains 45 wafers, while the lot size in the backend facility is again 45 wafers, however, lot splits and merges are possible in the backend facility.

We expect that the performance of our GA-based heuristic depends on the utilization of the frontend and the demand pattern used. The latter factor is characterized by the CoV of the final demand. The design of experiments used is summarized in Table 1. Here, we denote by $x \sim U[a,b]$ a random variable that is uniformly distributed over $[a,b]$. Totally, we consider 24 different simulation scenarios.

Table 1: Design of Experiments

Factor	Level	Count
average utilization of the bottleneck work center in the frontend facility (in [%])	65, 78, 91,96	4
expected revenue rev_{pt}	$\sim U[8,12]$, $\sim U[15,20]$ each range is for 50% of the products	1
inventory holding cost h_{kpt}	0.2/52 rev_{pt}	1
manufacturing cost c_{pt}	0.1/52 rev_{pt}	1
backlog cost b_{bt}	$2\,h_{kpt}$, $5\,h_{kpt}$	2
overall processing cost $\tilde{c}_p$	0.8 rev_{pt}	1
CoV of final demand	0.1, 0.25, 0.5	3
order lead time l_p in periods (weeks)	$\sim DU[0,8]$ for products with a planned CT of 8 weeks, $\sim DU[0,7]$ for products with a planned CT of 7 weeks	1
forecast $d_{pt}^{(r)}$	$\sim U\left[0.5E\left(d_{pt}^{(f)}\right),1.5E\left(d_{pt}^{(f)}\right)\right]$	1
total factor combinations		24

The final demand $d_{pt}^{(f)}$ is normally distributed, i.e., we have $d_{pt}^{(f)} \sim N\left(\mu_p,\sigma_p^2\right)$, where μ_p is the expected value and σ_p is the standard deviation of the final demand of product p. Note that both μ_p and σ_p are determined by the bottleneck utilization and the CoV value, respectively. We assume an equal product mix for the sake of simplicity.

The values for the product characteristics are randomly generated for the simulation experiments as shown in Table 1. Each single product has its individual order lead time, planned CT, forecast quantity, and expected revenue. The realizations of the corresponding random variables are summarized in Table 2 because we will show that these product characteristics influence the selected production strategy to a certain extent.

Table 2: Product characteristics

Product	P1	P2	P3	P4	P5	P6	P7	P8	P9	P10
l_p	6	5	1	6	3	2	7	3	6	6
CT	8	7	8	7	8	7	8	7	8	7
$d_{pt}^{(r)} / E\left(d_{pt}^{(f)}\right)$	1.48	0.56	1.16	0.95	0.99	1.39	1.40	1.44	0.66	0.93
rev_{pt}	10.40	11.30	9.20	9.40	8.70	17.40	19.00	16.20	19.20	18.80

4.2 Implementation Issues and Parameter Settings

The commercial simulation package AutoSched AP 9.0.1 is used as the simulation engine. It is a class library written in the C++ programming language. The GA is implemented using the GALib framework (cf. Wall 2013). It is a C++ class library. The GA is the steering program that calls AutoSched AP.

We use the following parameter settings for the GA. The number of generations is 50. The population size used is 60. The crossover probability is $pc = 0.8$, while the mutation probability is $pm = 0.2$. Finally, the replacement probability is $pr = 0.6$. These values are determined by some preliminary testing based on a trial and error strategy.

We use $T = 30$ weeks in all experiments. The simulation model is already initialized with appropriate WIP settings to avoid warm-up effects. Three independent replications per simulation run are performed to evaluate the fitness values of a single chromosome. The planning approach to determine lot release schedules is the begin of each planning period taking feedback from the base system into account. All the simulation experiments are performed on a PC with a 3.0 GHz and a 2.99 GHz processor and with 16 GB RAM. The average computing time for a single run of the GA is between 18 and 45 hours depending on the demand pattern used where a larger bottleneck utilization leads to longer computing times.

4.3 Results and Discussion

The P/D ratio as well as the CoV value are used to determine the production strategy according to the reference approach of Olhager (2003) discussed in Section 3.1. Table 3 shows the performance improvement that is reached with the GA taking into account the stochastic behavior of the supply chain as well as the different product characteristics. We provide the values for $100\%\left(Z(s_{GA}) - Z(s_{ref})\right) / Z(s_{ref})$, where s_{ref} and s_{GA} denotes the partitioning determined by the reference approach and the GA, respectively. The average improvement that the GA reaches is 9.7%. The range of improvement for individual scenarios is between 3.4% and 17.2%. Instead of comparing all the scenarios individually, the scenarios were grouped according to factor levels such as utilization or level of backlog costs in Table 3.

Table 3: Performance improvement reached with the GA for different scenarios

Factor	Level	Average Improvement
utilization	65%	9.0%
	78%	8.8%
	91%	10.0%
	96%	11.1%
CoV	0.10	11.1%
	0.25	12.1%
	0.50	5.9%
backlog	2	11.1%
	5	8.4%

The GA works especially well in the case of a high utilization, a low demand variability, and a low backlog penalty.

The different bottleneck utilization values are shown together with the relative frequency of the strategies that are selected by the GA during all scenarios in Figure 3.

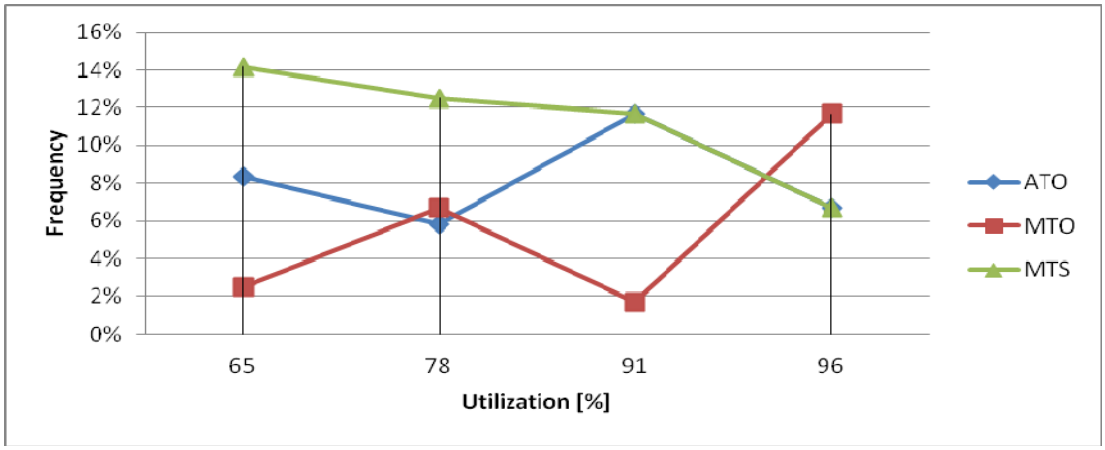

Figure 3: Influence of different bottleneck utilizations

In case of low utilizations, MTS production seems to be more appropriate. One might think that with a low utilization level, the real production CT values decrease and hence demands with a short order lead time can be met. But especially very short order lead times cannot be met even if the real production CT values are decreased. There are lots waiting to be batched in front of machines in this situation. High utilizations whereas, drive the decision more towards the MTO direction in order to reduce the overall load situation.

The influence of the degree of penalty for not meeting a customer demand, i.e. for backlog, is as expected as shown in Figure 4. The higher the penalty is, the less the products are categorized as ATO. The decision moves then towards the MTS strategy.

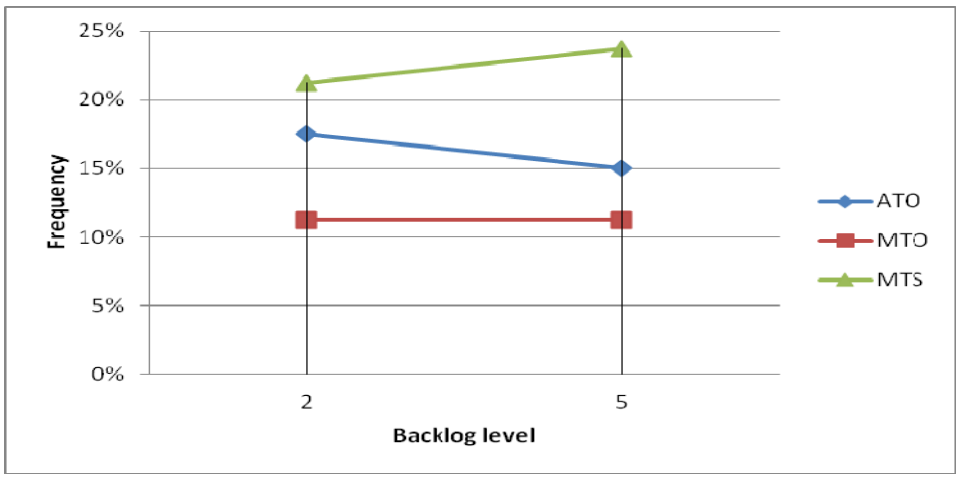

Figure 4: Influence of different levels of backlog penalties

Figure 5 shows the impact of different CoV values on the selection of the production strategy. For a low CoV the preferred strategies are MTS and ATO. The amount of products that are produced by a MTO strategy increases slightly for larger CoV values.

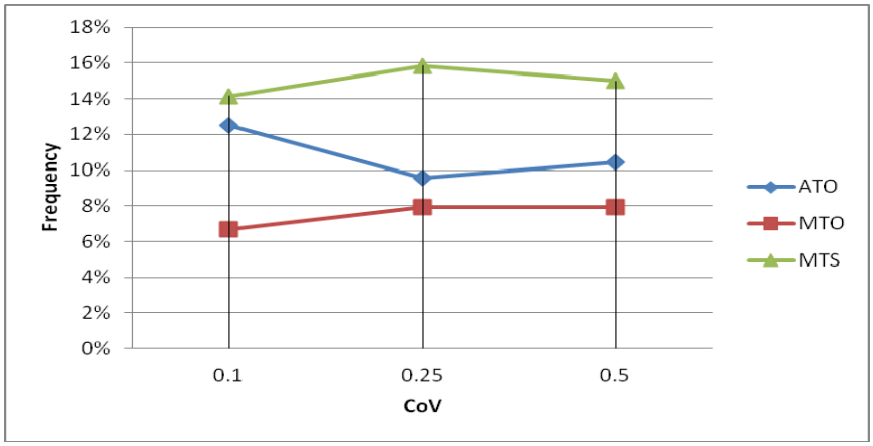

Figure 5: Influence of different CoV values

The different product characteristics affect the selection of the production strategy as well. The relative frequency of the strategies that are selected by the GA for each product during all scenarios is depicted in Figure 6. Each product has different characteristics such as expected revenue, order lead time, and the over and under estimation of the final demand by the forecast. For simplicity reasons, each characteristic is divided into two subcategories. The expected revenue x is categorized as low if $x \in [8,12]$ and as high if $x \in [15,20]$. The category short is assigned to order lead times with $l_p < 4$, the remaining order lead times belong to the category long. If the forecast exceeds the final demand, it is categorized as overestimated, otherwise it is underestimated. Product P1, for instance, has a low expected revenue, a forecast that overestimates the final demand, and a long order lead time (cf. Table 2).

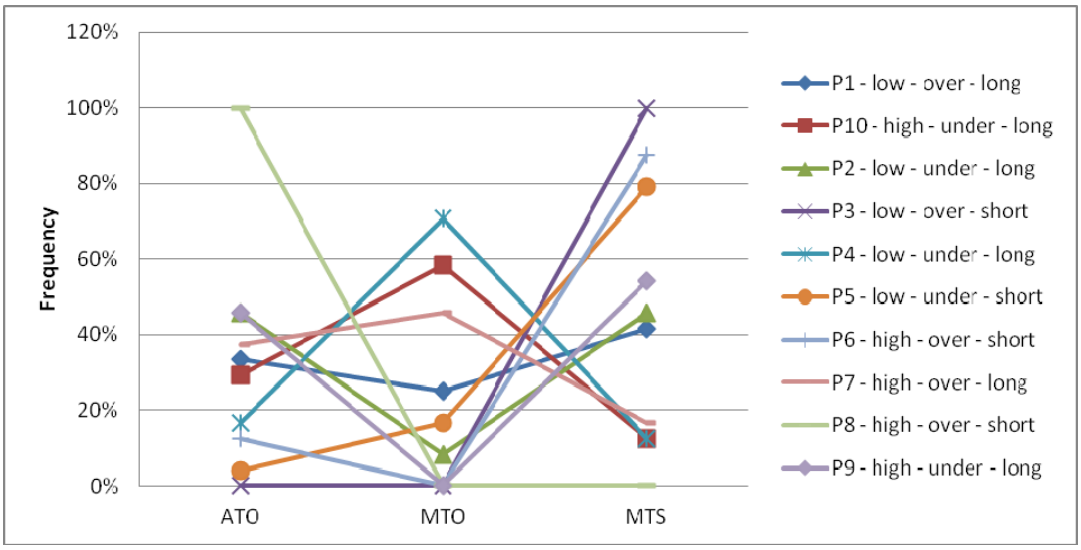

Figure 6: Influence of different product characteristics

Products with a short order lead time tend to be produced using the MTS or the ATO strategy, while products with a low expected revenue are more likely produced using MTS compared to products with a high expected revenue.

Obviously, for products P8 and P3 there is only one strategy selected independently of the different scenarios. The short order lead time $l_8 = 3$ and the highly overestimating forecasts lead in all scenarios to an ATO decision for product P8. All scenarios result in the MTS strategy for P3 due to the very short order lead time $l_3 = 1$. For the remaining products the decision varies in the different scenarios. All strategies can be found for products with long order lead times that are close to the production lead time. These products have a high improvement potential considering different load scenarios.

In summary, the solutions that the GA finds are comprehensive to a certain extent and confirm the effects described in the literature. The GA is also able to adapt its solutions to the different factors as shown in the design of experiment.

The simple reference heuristic is based on exogenous lead time estimates. However, the CT values are load-dependent. The load is to a certain extent a consequence of the partitioning decisions. As a result, it is non-trivial to predict them. In contrast to the reference heuristic, the GA is able to consider load-dependent CT values taken from the simulation.

5 CONCLUSIONS AND FUTURE RESEARCH

In this paper, we studied the problem to determine an appropriate production strategy for a given set of products with demand and forecast information for a certain horizon. We proposed a GA to tackle this problem. Since the GA requires several runs with a detailed simulation model of the supply chain to assess the fitness of a single chromosome, this is a time-consuming procedure. However, it turned out that the GA is able to outperform straightforward general-purpose partitioning strategies that are proposed in the literature.

There are several directions for future research. First of all, we have to decrease the huge simulation burden by using reduced simulation models as proposed by Hung and Leachman (1999) or by Ehm et al. (2011). In addition, we are also interested in the reduction or better utilization of the independent simulation replications using optimal computing budget allocation (OCBA) techniques (cf. Chen and Lee 2011). Much more computational experiments with a larger number of products, different planned CT settings, and more realistic supply chain models are necessary. Finally, we believe that it is worth and possible to extend the proposed GA in such a way that it makes simultaneously decisions on appropriate safety stocks and product partitioning. However, carrying out all the necessary details is part of future research.

REFERENCES

Chen, C. H., and L. H. Lee. 2011. *Stochastic Simulation Optimization – an Optimal Computing Budget Allocation.* World Scientific Publishing, Singapore.

Ehm, H., H. Wenke, L. Mönch, T. Ponsignon, and L. Forstner. 2011. "Towards a Supply Chain Simulation Reference Model for the Semiconductor Industry." In *Proceedings of the 2011 Winter Simulation Conference*, 2124-2135.

Federgruen, A., and Z. Katalan. 1999. "The Impact of Adding a Make-to-order Item to a Make-to-Stock Production System." *Management Science*, 45(7): 980-994.

Gupta, D., and S. Benjaafar. 2004. "Make-to-order, Make-to-stock, or Delay Product Differentiation? A Common Framework for Modeling and Analysis." *IIE Transactions*, 36: 529-546.

Hoekstra, S., J. Romme, and S. M. Argelo. 1992. *Integral Logistic Structures: Developing Customer-oriented Goods Flow*. Industrial Press Inc.

Hemmati, S., and M. Rabbani. 2009. "Make-to-order/Make-to-stock Partitioning Decisions Using the Analytic Network Process." *International Journal of Advanced Manufacturing Technology,* 48:801–813.

Hung, Y.-F., and R. C. Leachman. 1999. "Reduced Simulation Models of Wafer Fabrication Facilities." *International Journal of Production Research*, 37(12): 2685-2701.

MASMLab. 1997. "Test Data Sets." *http://www.eas.asu.edu/~masmlab*.

Michalewicz, Z. 1996. *Genetic Algorithms + Data Structures = Evolution Programs*. 3rd ed., Springer, New York.

Mönch, L., J. W. Fowler, and S. J. Mason. 2012. *Production Planning and Control for Semiconductor Wafer Fabrication Facilities: Models, Algorithms, and Systems.* Springer, New York.

Olhager, J. 2003. "Strategic Positioning of the Order Penetration Point. *International Journal of Production Economics*." 85(3): 319-329.

Rajagopalan, S. 2004. "Make to Order or Make to Stock: Model and Application." *Management Science*, 48(2), 241-256.

Shanthikumar, J. G., S. Ding, and M. T. Zhang. 2007. Queueing Theory for Semiconductor Manufacturing Systems: a Survey and Open Problems. *IEEE Transactions on Automation Science and Engineering* 4(4): 513–522.

Soman, C. A., D. P. van Donk, and G. Gaalman. 2004. "Combined Make-to-Order and Make-to-Stock in a Food Production System." *International Journal of Production Economics*, 90: 223-235.

Sun, Y., D. L. Shunk, J. W. Fowler, and E. S. Gel. 2010. "Strategic Factor-driven Supply Chain Design for Semiconductors." *California Journal of Operations Management, 8*(1): 31-43.

van Donk, D. P., C. A. Soman, and G. Gaalman. 2005. "A Decision Aid for Make-to-Order and Make-to-Stock Classification in Food Processing Industries." In *Proceedings EurOM International Conference on Operations and Global Competitiveness*.

Wall, M. 2013. "GALib – a C++ Library of Genetic Algorithm Components. " *http://lancet.mit.edu/ga/*.

Wemmerlöv, U. 1984. "Assemble-to-Order Manufacturing: Implications for Materials Management." *Journal of Operations Management*, 4(4): 347-368.

AUTHOR BIOGRAPHIES

LISA FORSTNER is a Ph.D. candidate in the Department of Mathematics and Computer Science at the University of Hagen, Germany. She received her master's degrees in production and information technologies from the EPF-Ecole d'Ingénieurs, Sceaux, France and the University of Applied Sciences, Munich, Germany. Currently, she works as a supply chain management professional at Infineon Technologies AG and is interested in supply chain management, discrete-event simulation and metaheuristics. Her email address is <lisa.forstner@infineon.com>.

LARS MÖNCH is Full Professor in the Department of Mathematics and Computer Science at the University of Hagen, Germany. He received a master's degree in applied mathematics and a Ph.D. in the same subject from the University of Göttingen, Germany. In addition, he also earned a habilitation degree in information systems from the Technical University of Ilmenau. His current research interests are in production planning and control of semiconductor wafer fabrication facilities, applied optimization and artificial intelligence applications in manufacturing, logistics, and service operations. He is a member of GI (German Chapter of the ACM), GOR (German Operations Research Society), SCS, INFORMS, and IIE. His email address is <Lars.Moench@fernuni-hagen.de>.