Proceedings of the 2003 IEEE
International Conference on Robotics & Automation
Taipei, Taiwan, September 14-19, 2003

# Efficient Selection of Scheduling Rule Combination by Combining Design of Experiment and Ordinal Optimization-based Simulation

Bo-Wei Hsieh
Dept. of Electrical Engineering
National Taiwan University
Taipei, Taiwan, R.O.C.
bwhsieh@ac.ee.ntu.edu.tw

Shi-Chung Chang
Dept. of Electrical Engineering
National Taiwan University
Taipei, Taiwan, R.O.C.
scchang@cc.ee.ntu.edu.tw

Chun-Hung Chen
Dept. of Syst. Engrg. & Oper. Res.
George Mason University
Fairfax, VA 22030, USA
cchen9@gmu.edu

***Abstract* – In a fab with heterogeneous machine groups, the number of scheduling policies grows in a combinatorial way because each machine group has its specific dispatching rules. In this paper, we design a fast simulation methodology by an innovative combination of the notions of *ordinal optimization* (OO) and *design of experiments* (DOE) to efficiently select a good scheduling policy for fab operation. Instead of finding the exact performance among scheduling policies, our approach compares their relative orders of performance to a specified level of confidence. The DOE method is exploited to largely reduce the number of scheduling policies to be evaluated by the OO-based simulation. Simulation results of applications to scheduling wafer fabrications show that most of the OO-based DOE simulations require 2 to 3 orders of magnitude less computation time than those of a traditional approach, and the speedup is up to 7,000 times in certain cases.**

## I. INTRODUCTION

Major fab scheduling problems include how wafers should be released into a fab and how they should be dispatched among machines for processing. A popular practitioners' approach for scheduling the production is to select from the many empirical scheduling rules available for IC fabs. Scheduling rules for each machine group should are designed based on the specific characteristics and operation goals of the machine group. Empirical or heuristic rules are collected for individual machine groups. The industry of wafer fabrications has indicated a strong need for an efficient simulation tool for selecting a good scheduling rule from the existing library.

Rule selection by using the traditional simulation approaches is not fast enough in computation for short-term scheduling of fab operations. Recent research has shown that comparing *relative orders* of performance measures converges much faster than the performance measures themselves do. This is the basic idea of *ordinal comparison* (OC). OC can be used as a means for solving scheduling rule selection problems if our goal is to find a good scheduling policy rather than to find an accurate estimate of the performance value of a scheduling rule. A technique called *optimal computing budget allocation* (OCBA) that can further reduce computing time when used in conjunction with OO is adopted in our simulation methodology. Hsieh et al. [5] has applied OO and OCBA techniques to dynamic selection of scheduling rules for fabs and has shown its potential for real applications. However, a homogeneous set of dispatching rules among machine groups is assumed in [5]. As the number of candidate policies grows in a combinatorial way with the number of machine groups, a brute-force application of the OC and OCBA method to selecting a good scheduling policy is still infeasible.

The method of *design of experiments* (DOE) has been effective in reducing the number of options to be evaluated. DOE methods are used to experiment with various combinations of the important design factors for the purpose of identifying the particular combination that optimize certain design criteria or performance measure. An efficient class of DOE methods, *fractional factorial design of experiments* (FFDOE), has been proposed for handling experiments with several design factors simultaneously [3, 7]. With the assumption that high order interaction effects on a performance function of interest caused by several factors are not significant and negligible, performance evaluation of a large portion of redundant options can be saved.

In this paper, we design an innovative combination of the DOE and OO methods and investigate its application to efficiently selecting good rules for scheduling wafer fabs. Section II describes and formulates scheduling rule selection problems. The notions of OC and DOE that can significantly reduce the required simulation time are described in Section III. An efficient simulation methodology, which combines the OC technique and the DOE method is designed in Section IV. Rule selection experiments among machine groups are conducted in Section V. Section VI concludes this paper.

## II. SCHEDULING RULE SELECTION PROBLEM

A fab scheduling/operation policy is a combination of a wafer release policy and dispatching rules for individual machine groups over a specific time horizon. Due to the diversity of equipment in a fab, scheduling rules for each machine group should be designed based on the specific characteristics and operation goals of the machine group. Empirical or heuristic scheduling rules are collected for individual machine groups. Such rule collections are built into a scheduling rule library of a fab. Suppose that there are $M$ machine groups in a fab and wafer release and the dispatching of each machine group have $D$ candidate rules, there are a total number of $(D)^{M+1}$ scheduling policies, which grows in a combinatorial way with the number of machine groups. To capture the inherent complexity of a production line, simulation is often adopted to evaluate performance measures of operation policies. However, it takes a formidable amount of computation time to select a good policy by using brute-force simulation to evaluate these scheduling policies.

Suppose we want to compare a total of $R$ policies, which are indexed by $i$, $i = 1, 2, ..., R$. Denote $h(\Theta_i; w)$ as the performance measure of a policy $i$, where $\Theta_i$ is a vector of design parameters of policy $i$ and $w$ is a random vector that represents uncertain factors in the system. Because of the inherent complexity of the system, discrete event simulations are adopted to estimate $\mathrm{E}_w[h(\Theta_i; w)]$ by taking $n$ independent replications of the simulation and approximating the expectation as

$$\hat{\mathrm{E}}_w[h(\Theta_i; w)] \equiv \frac{1}{n}\sum_{j=1}^{n} h(\Theta_i; w_j),$$

where $w_j$, $j=1,...,n$, are the $j$-th sample of $w$, and $h(\Theta_i; w_j)$ is obtained from $j$-th simulation replication.

In general, the rate of convergence for estimate $\hat{\mathrm{E}}_w[h(\Theta_i; w)]$ by using traditional simulation methods is at best $O(1/\sqrt{n})$. The large $n$ required for a good approximation implies that each policy must be simulated with a large number of replications, which translates to long computation time. Moreover, the number of policies $R$ grows in a combinatorial way with respect to production control factors. Even the number of simulation replications $n$ for each policy is small, the simulation time required for the enormous number of policies is formidable.

## III. ORDINAL COMPARISON AND DESIGN OF EXPERIMENTS

Motivated by the deficiencies of applying traditional simulation approaches to scheduling policy selection problems, the notions of *ordinal comparison* (OC) and *design of experiments* (DOE) are adopted which are known methods for significantly reduction of the required simulation time.

### A. *Ordinal Comparison and Optimal Computing Budget Allocation*

Let option $b$ be defined as

$$b \equiv \arg\min_i \hat{\mathrm{E}}_w[h(\Theta_i; w)].$$

*Definition 1.* Define *correct selection-1* ($\mathrm{CS}_1$) as the event that the selected option $b$ is actually the best option. Define the *confidence probability* $P\{\mathrm{CS}_1\} \equiv P\{$ The current top-raking option $b$ is actually the best option $\}$.

Instead of finding the optimal option, the approach of OC compares the relative order of performance among options to a specified level of confidence. By using OC, the probability $P\{\mathrm{CS}_1\}$ may converge at an exponential rate while $\hat{\mathrm{E}}_w[h(\Theta_i; w)]$ may converge slowly. A critical issue is the estimation of $P\{\mathrm{CS}_1\}$. Using a Bayesian model, [1] developed an effective estimation technique, where

$$P\{\mathrm{CS}_1\} \approx \prod_{i=1, i\neq b}^{k} P\{\tilde{J}_b < \tilde{J}_i\} \equiv \text{Approximate Probability of Correct Selection-1 } (APCS_1)$$

$\tilde{J}_i$ denotes the random variable whose probability distribution is the posterior distribution of the expected performance for option $i$ under a Bayesian model. We shall use $APCS_1$ to approximate $P\{\mathrm{CS}_1\}$.

To further enhance the efficiency of OC, the technique *optimal computing budget allocation* (OCBA) intelligently determines the best number of simulation replications among different options as simulation proceeds. Intuitively, to ensure a high confidence probability, a larger portion of the computing budget should be allocated to options that are potentially good options. A critical issue is the determination of a set of "promising" options. The "promising" options are options that can maximize the improvement of $APCS_1$ if they are further simulated. Chen et al. offer a simple and effective way to solve the problem [2].

### B. *Design of Experiments*

In this paper, an efficient DOE method, *fractional factorial design of experiments* (FFDOE), is adopted to handle the combinatorial complexity of scheduling rule selection problems. With the assumption that high order interaction effects on a performance function of interest caused by several factors are insignificant, the FFDOE method exploits orthogonal arrays (OAs) to reduce the number of necessary options to be evaluated [3, 7]. The first step is to identify factors that affect the performance function of interest and choose different levels of design parameters for each factor. An OA is then used to determine a small set of test samples for evaluation by simulation. Denote an OA as $\mathrm{OA}(n_S, n_F, n_L, s)$, where

$n_S$ is the size of the test sample set to be performed, $n_F$ is the number of factors, $n_L$ is the number of levels of each factor, and $s$ is the number of columns where we are guaranteed to see all the possible combinations of levels an equal number of times and is called strength [4]. Simulation runs are conducted for the set of test samples. A performance estimation model of all options is built by using the simulated performance measures of the OA samples. Details of performance estimation are described as follows.

A regression model is commonly used to estimate the performance surface for a factorial design. Since factors of scheduling rule selection have discrete choices without quantitative relationship, general regression models cannot be directly applied. Let us define some notations.

| | |
|---|---|
| $h_{ij} \equiv h(\Theta_i; w_j)$ | the $j$-th sample of option $i$ |
| $\bar{h}_i \equiv \bar{h}(\Theta_i; w)$ | the sample mean of option $i$, $\bar{h}_i = \frac{1}{n}\sum_{j=1}^{n} h_{ij}$ |
| $\hat{h}_i \equiv \hat{h}(\Theta_i; w)$ | the estimated performance measure of option $i$ |
| $S(n_S)$ | the orthogonal set of test samples |
| $\theta_{ij}$ | the design parameter of factor $j$ in option $i$ |

With the assumption that third-factor and higher interaction effects are negligible, the mathematical model of the performance measure is thus represented as

$$h(\Theta_i; w) = h_0 + \sum_{p=1}^{n_F} h_p(\theta_{ip}; w) + \sum_{p=1}^{n_F} \sum_{q=p+1}^{n_F} h_{pq}(\theta_{ip}\theta_{iq}; w) + \varepsilon,$$

where $h_p(\theta_{ip}; w)$ is the performance effect (main effect) of $h(\Theta_i; w)$ caused by factor $p$, $h_{pq}(\theta_{ip}\theta_{iq}; w)$ is the two-factor interaction effect caused by factors $p$ and $q$, and $\varepsilon$ is the error term. The method of least squares is then used to fit the model with the simulated performance measures of OA samples.

## IV. DESIGN OF OO-BASED DOE SIMULATION

We now present an innovative combination of the aforementioned OO and DOE methods into an efficient methodology for the scheduling rule selection problem.

### *A. Design Overview*

As shown in Figure 1, our methodology consists of two phases. Phase I exploits DOE technique to estimate performance measures of all policies with a small set of policies simulated. Top-ranking policies are screened for further evaluation under a new OC criterion. Phase II directly applies OC+OCBA techniques to top-ranking policies screened out by phase I and identifies a good policy for fab operation. This two-phase methodology will be referred to as the OO-based DOE method hereafter.

In the phase I application of DOE, to shorten the simulation time for each scheduling policy in the OA, we define a new correct selection criterion, $CS_2$, to screen top-ranking policies for further simulation. Different from $CS_1$, $CS_2$ considers policies of which the performances are not worse than a fraction of that of the best policy. The compromise in optimality may lead to substantial reduction in simulation time.

The probability of $CS_2$, $P\{CS_2\}$, determines how accurate the performance estimates of the OA policies should be. Generally speaking, the more simulation replications for the OA policies the better in performance estimates of all policies. $P\{CS_2\}$ is derived based on the probability distributions of the estimated performance measures of all policies. Detailed design of the option-screening criterion is described as follows.

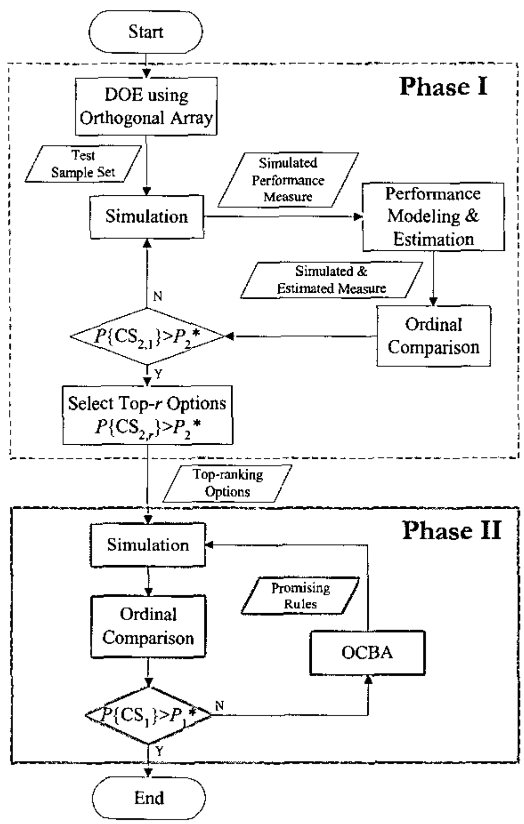


Figure 1. Flowchart of the OO-based DOE algorithm

*B. Design of Option-screening Criterion*

Let us first give the formal definitions of the new OC criterion $CS_2$ and $P\{CS_2\}$.

*Definition 2.* Define *correct selection-2* ($CS_2$) as the event that the true performance of the observed rank-$r$ option is not worse than $\beta$ fraction of the performance of the true best option. Define $P\{CS_2(r,\beta)\} \equiv \Pr\{ CS_2 \text{ occurs} \}$.

Similar to the estimation of $P\{CS_1\}$, $P\{CS_2(r,\beta)\}$ is estimated by

$$P\{CS_2(r,\beta)\} \approx \prod_{i=1, i\neq r}^{R} P\{\tilde{J}_r > \beta\tilde{J}_i\} \equiv APCS_2(r,\beta),$$

where the posterior distribution for option $i$, $\tilde{J}_i$, is estimated by estimated mean, $\hat{h}(\Theta_i; w)$, and estimated variance, $\hat{S}_i^2$. $P\{CS_2(r,\beta)\}$, which is approximated by $APCS_2(r,\beta)$, is used as the option-screening criterion.

Value estimates of both mean and variance of performance measures are required parameters for $APCS_2(r,\beta)$. Our simulation study shows that the variance of a scheduling policy depends highly on the scheduling rules. The frequently used common variance assumption [3] is not applicable. Without attempting to find accurate estimates of variances, a conservative estimation approach is taken, where the variance of an un-simulated policy is estimated by the maximum residual variation of simulated policies so that the confidence probability is not overestimated.

The residual variation of a performance model is composed of two parts, that due to pure error among test samples (replications) and that due to model lack of fit [3]. The variance of $\hat{h}_k$ is then estimated by

$$\hat{S}_k^2 = S_k^2 + \frac{n(\bar{h}_k - \hat{h}_k)^2}{n-1} \qquad \text{for } k \in S(n_s),$$

where $S_k^2$ is the sample variance of policy $k$. The variance of an un-simulated policy $i$ corresponding to the estimated performance measure $\hat{h}_i$ is estimated by

$$\hat{S}_i^2 \approx \max_k \hat{S}_k^2 \qquad \text{for } i \notin S(n_s).$$

## V. APPLICATION: SCHEDULING POLICY SELECTION

A full-scale single product fab model (named FAB) modified from Lu et al. [6] is adopted in the simulation experiments for evaluating the OO-based DOE method. It involves 12 failure-prone machine groups, each having one or more identical machines. There are 60 operation stages in the entire process flow. Processing times at each station, times between machine failures, and times to repair are all exponentially distributed. A lot consists of 24 wafers and is the unit of processing and transportation. With a release rate of 0.52 lots/hour, utilization of the machine groups are mostly greater than 90%, which captures the loading situation of wafer fabs. Bottleneck machine group in this model is Station 1, whose utilization rate is 94.2%. Station 8 corresponds to a furnace machine group and is a batch-processing machine group, where the loading capacity is 6 lots per batch. We also deduce that Station 1 corresponds to a photolithography machine group, which encounters the most complicated re-entrant flow among all stations. Due to quality issues, once the photolithography machine of the first layer of a lot is decided upon, the following layers of the lot must be limited to exactly the same machine of Station 1. Detailed model parameters of FAB are given in Table 1.

In our experimental study, three performance indices are considered: mean cycle time (MCT), variance of cycle time (VCT), and smoothness of a fabrication line (SM), which are among the most frequently used fab performance indices. There are four sets of dispatching rules and a set of wafer release policies, each having four options, as listed in Table 2. There are therefore $4^5$ (=1024) scheduling policies. The four sets of dispatching rules are lot assignment policies for photolithography (LAP_PH), dispatching rules for general machine groups (LDR), dispatching rules for furnace machine group (LDR_F), and dispatching rules for upstream of furnace machine group (LDR_UF). LAP_PH policies dedicate a lot to a photo machine for the purpose of workload balancing. LDR_F rules select a stage with certain criteria and choose a batch of lots from the stage. SAF1 and SAF2 are dispatching rules designed for avoiding furnace machines from starving.

The number of simulation runs for test samples in an OA is set to 10. The number of promising options, $r$, is set to 16, the fraction parameter, $\beta$, is set to 0.97, and the confidence probabilities are all set to 0.9.

Table 1. Plant Data of FAB

| Station | Machine Type | # of Machines | # of Visits | MPT[1] | MTBF[2] | MTTR[3] | Batch Size | % Util |
|---|---|---|---|---|---|---|---|---|
| 1 | Photo | 4 | 14 | 0.500 | 150 | 5 | 1 | 94.2% |
| 2 | General | 3 | 12 | 0.375 | 200 | 9 | 1 | 82.3% |
| 3 | Implanter | 10 | 7 | 2.500 | 200 | 5 | 1 | 93.4% |
| 4 | General | 1 | 1 | 1.800 | 200 | 1 | 1 | 94.1% |
| 5 | General | 1 | 2 | 0.900 | 200 | 1 | 1 | 94.1% |
| 6 | General | 2 | 3 | 1.200 | 200 | 1 | 1 | 94.1% |
| 7 | General | 1 | 1 | 1.800 | 200 | 1 | 1 | 94.1% |
| 8 | Furnace | 4 | 8 | 4.800 | 150 | 5 | 6 | 86.4% |
| 9 | General | 1 | 3 | 0.580 | 200 | 5 | 1 | 92.9% |
| 10 | General | 9 | 5 | 3.000 | 130 | 5 | 1 | 90.4% |
| 11 | General | 2 | 3 | 1.100 | 200 | 5 | 1 | 88.2% |
| 12 | General | 2 | 1 | 2.500 | 200 | 5 | 1 | 67.4% |

[1] MPT: Mean Processing Time (by hours)
[2] MTBF: Mean Time between Failures (by hours)
[3] MTTR: Mean Time to Repair (by hours)

Table 2. Scheduling rules

| Rule Set | Symbol | Description |
|---|---|---|
| WRP | DET | Interarrival times of lots are constant. |
| | WR($C$) | Workload regulation release for one bottleneck system. When the expected work in fab for bottleneck machine drops to $C$ hours, then release a new lot. |
| | DET-BH($B$) | Interarrival times of batches of lots are constant, where the batch size is $B$. |
| | WR-BH ($C$, $B$) | Workload regulation release for one bottleneck system. When the expected work in fab for bottleneck machine drops to $C$ hours, then release a batch of lots, where the batch size is $B$. |
| LAP_PH | LTWP | Choose a machine with least dedicated WIPs in the re-entrant line. |
| | LTWL | Choose a machine with least total workload in the re-entrant line. |
| | LGWP | Choose a machine with least dedicated WIPs among photo machines. |
| | LGWL | Choose a machine with least workload among photo machines. |
| LDR | FSMCT | Choose the lot with smallest $(n_p/\lambda_p + C_p - \zeta_i)$, where $p$ represents the index of product type, $n_p$ is the sequence number of the lot under consideration, $C_p$ is the mean cycle time, $\lambda_p$ is the throughput rate, and $\zeta_i$ is the estimate of the remaining cycle time from buffer $i$. |
| | LDF | Choose a stage with the largest deviation of completed moves from the desired moves. Then choose from the stage a lot which is released into the fab the earliest. |
| | OSA | Choose a stage $i$ according to the following priorities:<br>I: $N_i(t) > \overline{N}_i$ and $N_{i+1}(t) < \overline{N}_{i+1}$;<br>II: $N_i(t) < \overline{N}_i$ and $N_{i+1}(t) < \overline{N}_{i+1}$;<br>III: $N_i(t) > \overline{N}_i$ and $N_{i+1}(t) > \overline{N}_{i+1}$;<br>IV: $N_i(t) < \overline{N}_i$ and $N_{i+1}(t) > \overline{N}_{i+1}$,<br>where $N_i(t)$ is the WIP at time $t$ at step $i$, $\overline{N}_i$ is the average WIP at step $i$. Choose a lot with the same priority which is released into the fab the earliest. |
| | FIFO | Select the lot which arrived at the station the earliest. |
| LDR_F | LAS | Choose a stage with least average slack time of lots in queue and then choose a batch of lots from the stage with least slack time. |
| | LLS | Choose a stage that has the lot with least slack time and then choose a batch of lots from the stage with least slack time. |
| | LNGQ | Choose a stage with longest queue and then choose a batch of lots from the stage with least slack time. |
| | LLA | Choose a stage that has the lot, which arrives the machine group at the earliest time, and then choose a batch of lots from the stage with least slack time. |
| LDR_UF | SAF1 | If the queue lengths of all downstream furnace stages are smaller than a threshold, select a stage that has the largest sum of lots at current stage and downstream furnace stage, and then choose a lot from the stage using FIFO. If not, use FIFO. |
| | SAF2 | If the queue lengths of all downstream furnace stages are smaller than a threshold, select a stage that has the most lots at downstream stage, and then choose a lot from the stage using FIFO. If not, use FIFO. |
| | FSMCT | As FSMCT in LDR. |
| | FIFO | As FIFO in LDR. |

*A. Long-term Performance Analysis*

This set of experiments is designed to study the long-term MCT, VCT, and SM performance of individual scheduling policies. A simulation run of 8.2 years starting with the fab empty is conducted for each scheduling policy. Figure 2 shows the main effects of the five factors that affect the three performance indices. FSMCT dispatching rule along with LTWL lot assignment policy are superior to other rules in MCT reduction. Dispatching rules have significant influence on VCT and SM across all machine groups.

*B. Effectiveness of the OO-based DOE Simulation*

To assess the effectiveness of the OO-based DOE method, we estimate the long-term MCT, VCT, and SM performance of individual scheduling policies. Table 3 summarizes the ranks of simulated performances of the estimated top-10 policies. When modeling with only main effects, one of the estimated top-10 policies is within the simulated top-10 policies under MCT and VCT performance criteria. Great improvement of the estimate on relative performance is observed when two-factor interactions are modeled. It is inspirational that 6 of the estimated top-10 scheduling policies are within the simulated top-10 polices under MCT performance index. Four of the estimated top-10 policies are within simulated top-10 policies under SM performance index with only main effects modeled. This indicates that two-factor interaction effects of SM are indeed negligible.

*C. Computational Efficiency*

To examine the efficiency of the OO-based DOE simulation, 8 sets of short-term rule selection experiments are conducted and traditional simulation approach serves as a benchmark. Computation times of the two simulation approaches are listed in Table 4. Time saving factor is defined as the ratio of simulation replications of traditional approach to that of the OO-based DOE approach. In Exps. 1 and 2, traditional simulation approach requires up to 7000 times more computation time than the OO-based DOE approach. But in Exps. 5 and 6, the time saving factors are about 20. This is because MCT performance of policies converge faster and require fewer simulation replications to obtain a good approximation. Most of the OO-based DOE simulations require three to four orders of magnitude less computation time than the traditional approach.

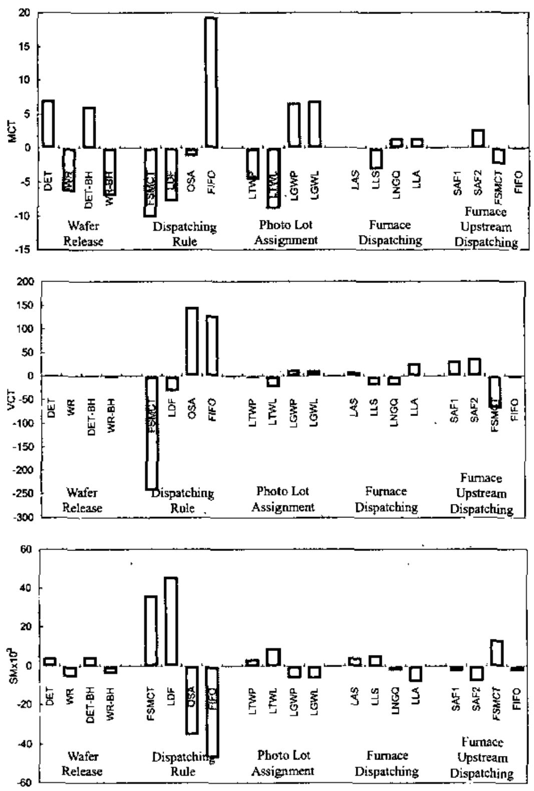


Figure2. Main effects of rules among machine groups

Table 3. True rankings of estimated top-10 policies

| Est. Rank | Modeling with main effects | | | Modeling with main effects and two-factor interactions | |
|---|---|---|---|---|---|
| | MCT | VCT | SM | MCT | VCT |
| 1 | 18 | 5 | 2 | 1 | 1 |
| 2 | 15 | 35 | 5 | 5 | 15 |
| 3 | 43 | 25 | 15 | 13 | 16 |
| 4 | 27 | 23 | 13 | 12 | 9 |
| 5 | 29 | 40 | 56 | 6 | 28 |
| 6 | 77 | 44 | 62 | 15 | 23 |
| 7 | 80 | 37 | 42 | 160 | 29 |
| 8 | 106 | 49 | 63 | 144 | 59 |
| 9 | 71 | 99 | 32 | 135 | 113 |
| 10 | 136 | 104 | 93 | 176 | 120 |

Table 4. Efficiency of the OO-based DOE simulation

| Exp. ID | Performance Index | Simulation Replications: Traditional (approximated) | Simulation Replications: OO-based DOE | Time Saving Factor |
|---|---|---|---|---|
| 1 | VCT | 7,023,865 | 972 | 7,226 |
| 2 | VCT | 5,520,734 | 783 | 7,051 |
| 3 | SM | 1,099,417 | 576 | 1,909 |
| 4 | SM | 1,994,606 | 435 | 4,585 |
| 5 | MCT | 22,524 | 858 | 26 |
| 6 | MCT | 20,583 | 1,053 | 20 |
| 7 | SM | 381,547 | 546 | 699 |
| 8 | SM | 476,888 | 417 | 1,144 |

## VI. CONCLUSIONS

Motivated by the problem of scheduling policy selection for semiconductor wafer fabs, we have designed in this paper a two-phase methodology of fast simulation. The methodology is an innovative combination of ordinal optimization techniques, OO+OCBA, in phase II, and design of experiments in phase I. The innovation of phase I lies in a new definition of correct selection for finding a set of good enough orthogonal array and a regression model for estimating the performance surface. Top ranking policy options screened out by phase I are then further simulated by OO+OCBA in phase II. Not only is the number of simulation options reduced to a manageable level but also the simulation time for evaluating selected options largely shortened. Simulation results of applications to scheduling a fab show that most of the OO-based DOE simulations require 2 to 3 orders of magnitude less computation time than those of a traditional approach, and the speedup is up to 7,000 times in certain cases. The methodology developed by this paper is complementary to that of [5].

## ACKNOWLEDGEMENT

This work was supported in part by the National Science Council of the Republic of China under Grants NSC 89-2212-E-002-040 and NSC89-2213-E-002-119, Semiconductor Research Corporation Grant 090E1092, Taiwan Semiconductor Manufacturing Company Grant 91-S-B14, NSF Grants DMI-9732173 and DMI-0002900, Sandia National Laboratory Grant BD-0618, and the George Mason University Research Foundation.

## REFERENCES

[1] C.-H. Chen, "A lower bound for the correct subset selection probability and its application to discrete-event system simulations," *IEEE Transactions on Automatic Control*, vol. 41, no. 8, pp. 1227-1231, Aug. 1996.

[2] H.-C. Chen, L. Dai, C.-H. Chen, and E. Yucesan, "New development of optimal computing budget allocation for discrete event simulation," in *Proc. 1997 Winter Simulation Conference*, Dec. 1997, pp. 334 -341.

[3] R. E. DeVor, T.-h. Chang, and J. W. Sutherland, *Statistical quality design and control: contemporary concepts and methods*. New York: Macmillan, 1992.

[4] A. S. Hedayat, N. J. A. Sloane, and J. Stufken, *Orthogonal arrays : theory and applications*. New York: Springer, 1999.

[5] B.-W. Hsieh, C.-H. Chen, and S.-C. Chang, "Scheduling semiconductor wafer fabrication by using ordinal optimization-based simulation," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 5, pp. 599-608, Oct. 2001.

[6] S. C. H. Lu, D. Ramaswamy, and P. R. Kumar, "Efficient scheduling policies to reduce mean and variance of cycle-time in semiconductor manufacturing plants," *IEEE Transactions on Semiconductor Manufacturing*, vol. 7, no. 3, pp. 374 -388, Aug. 1994.

[7] D. C. Montgomery, *Design and analysis of experiments*, 5th ed. New York: John Wiley, 2001.