Using representative process flows for simulation model simplification

Igor Stogniy
Institute of Applied Computer Science
Technische Universität Dresden
Dresden, Germany
igor.stogniy@tu-dresden.de

Wolfgang Scholl
Operational Panning & Controlling
Infineon Technologies Dresden
Dreden, Germany
wolfgang.scholl@infineon.com

***Abstract*— Infineon Technologies Dresden has been using a long-term simulation model to support production planning for a long time. There is a need to reduce efforts to create and maintain models. There are two ways of doing this: using representative process flows and substituting tool sets for constant delays. This paper considers both approaches as well as their combination. The main idea is to evaluate them and find the appropriate level of model complexity in terms of model accuracy. Therefore, a gradual simplification is used. The MIMAC dataset 5 was the data source for this study.**

Keywords — simulation model simplification, representative process flows, semiconductor manufacturing simulation

I. INTRODUCTION

This paper considers discrete-event simulation and in particular the problem of simulation model simplification. The simplification of simulation models is still rather a "green field" [1]. One of the simplification methods is to apply representative process flows. This means that instead of modeling all routings, only individual representatives are used (the demand of a group of routings is aggregated for one of them). Surprisingly, despite its usage in practical work, this topic is not well covered in scientific papers. This is probably due to the fact that the use of representative process flows implies a rather complex simulation model as a research object, which is difficult for an academic researcher to work with. On the other hand, simulation modeling practitioners are usually not interested in carrying out a large number of experiments to compare the things that are obvious to them. However, even the seeming obvious approaches require scientific proof and explanation. In addition, as we have learned from our research, the use of representative process flows is coupled with model calibration, which is still more an art than a science.

This work is a continuation of our previous studies, in which we studied tool set substitutions for constant delays [2] and various accuracy measurements [3]. In this paper we compare two scenarios: original process flows (γ_1) and representative process flows (γ_2). For both scenarios, we apply a simplification by substituting tool sets for constant delays according to the sieve functions ($\zeta_1 \ldots \zeta_{10}$). For each of the sieve functions 83 simplification experiments were carried out in order to assess the degree to which the accuracy of the simulation model deteriorated with gradual simplification. We consider both scenarios for two cases: steady state and transient state.

The MIMAC 5 data set [4] was chosen as the basis for the research, as it includes the largest number of process flows, which allowed using 11 of them for the scenario γ_1 and 3 for γ_2. AutoSched AP from Applied Material, version 11.5 was used as a simulation tool.

II. RELATED WORK

Reference [1] provides a good overview of the simulation model simplification with a huge amount of scientific sources. The using of representative process flows is mentioned as a part of entity aggregation. [5] reflected on the importance of having the right level of modeling complexity using the examples of an automated material handling system. They reviewed several practical cases and came to interesting conclusions, one of which was "simpler is not always better". In our study, we consider factory simulation and compare quantitatively different simplified models. [6] presented the routing aggregation using representative flows, but they did not compare the results with all process flow scenario. In addition, they used t-test as an accuracy measurement, which we criticize in [3]. [7] presented abstractions for product variants using effective processing time, but they investigated mostly transport simulation. [8] discussed the importance of simplification for supply chain modeling in the semiconductor industry. They built a supply chain simulation model. They also introduced an accuracy measurement "similarity between two histograms", which is divided by two Summarized Absolute Divergence (SAD) we use. Unfortunately, they used the MIMAC 1 dataset with only two products and did not work with representative process flows. [6] described several features of the simplification, one of which used representative process flows, but presented the results of the working model, rather than comparing different options. [10] presented criteria that should be taken into account when selecting representative process flows: sufficiency of historical data and similarity in terms of the recipe. However, the authors presented only the results of the modeling, not the comparisons of the scenarios under consideration.

As we can see from the overview above, most studies focus on comparing an already simplified model using representative process flows with reality. Only [6] added to the simplification substitution of tool sets for constant delays. In this paper, we expand the view of simplification by comparing two γ_1 and γ_2 scenarios, and by using delay substitutes.

III. DESIGN OF EXPERIMENTS

In this work, an earlier developed automated experimental environment was used. It allowed a significant number of experiments to be carried out: 4000 runs of the detailed model to calculate the sieve functions and base line data for calibrations; 85 and 1700 runs for calibration experiments (steady and transient state cases); 16600 runs to evaluate the accuracy of model simplification. We used only 5 seeds (runs) for each experiment, due to the huge amount of planned experiments that are beyond the scope of this paper, and rather the exploratory nature of this research.

We considered two cases: steady state and transient state. Each simulation run was 114 weeks. The warm up period was determined by the Welch's procedure [11] and was set at 10 weeks. Tool set substitution for constant but not random delays was considered. For an explanation of this decision, see [2]. Furthermore, we chose only process step based delays because they showed better accuracy than tool set based [3].

A. Demand and representative process flows

Representative process flows were selected based on their structure (similarity of processing steps) and demand for the relevant products. The routing that had the highest demand was chosen out of all the similar ones. Among 11 process flows, we chose 3 as representatives (black, red and green line in Fig. 1). Demand for other process flows was summed up and assigned to the representatives. Thus, two scenarios were obtained: original process flows (γ_1) and representative process flows (γ_2).

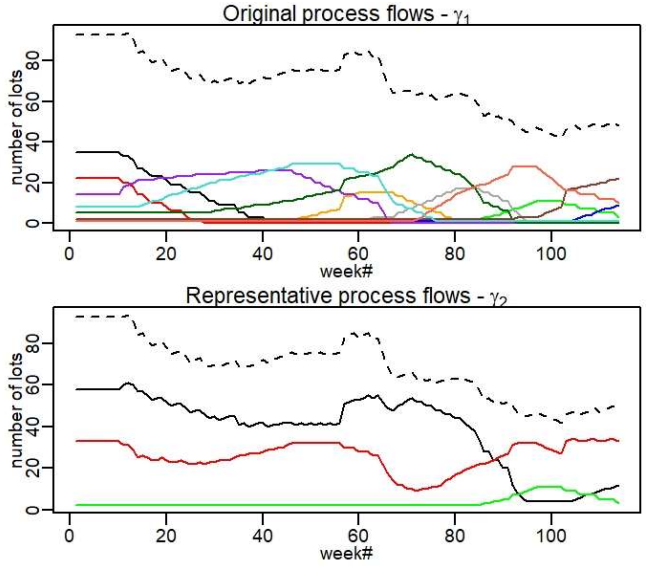

Fig. 1. Two scenarios: original process flows – γ_1; representative process flows – γ_2;

B. Sieve functions ζ

A sieve function is a mechanism for ordering tool sets to substitute them step by step for constant delays (at each step one more tool set is replaced). In other words, the sieve function is the tool set criticality index. The value of a sieve function was calculated based on the following parameters of the weekly simulation tool set reports: *IDLE%/IDLE#* – the idle state; *PROC%* – the processing state; BS_{AVG} – the average batch size; BS_{MAX} – the maximum batch size; QT_{AVG} – the queue time; QL_{AVG} – the queue length; PT_{AVG} – the processing time; CT_{AVG} – the lot cycle time; CT_{SD} – the standard deviation of the cycle time; and TH_{AVG} – the throughput. The following formulas were used: $\zeta_1 = IDLE\%$; $\zeta_2 = IDLE\% + PROC\% - PROC\%(BS_{AVG} / BS_{MAX})$; $\zeta_3 = (100 - IDLE\%) / IDLE\#$; $\zeta_4 = QT_{AVG}$; $\zeta_5 = QT_{AVG} / PT_{AVG}$; $\zeta_6 = QL_{AVG}$; $\zeta_7 = QL_{AVG} / BS_{MAX}$; $\zeta_8 = CT_{SD}^{total}$; $\zeta_9 = CT_{SD}^{total} / CT_{AVG}^{total}$; and $\zeta_{10} = TH_{AVG}$. Each of the sieve functions represents one experimental series consisting of 83 experiments to substitute tool sets for delays.

C. Accuracy measurements

After the study of accuracy measurements [3], the most suitable two of them were used for this work: the Summarized Absolute Divergence (SAD) and the Kolmogorov-Smirnov (KS) test.

SAD is a very simple measurement based on absolute divergence between two probability density functions. The value of SAD is the area between two probability density functions. It is changed from 0, when two distributions are identical, to 2, when two distributions have nothing in common [2] (compare with "similarity between two histograms" in [8]).

To calculate SAD, we consider probability density function (or frequency distribution) of two variables X_1 (base case) and X_2 (experiment data). We can devide the range of the variables into k intervals (bins) and denote $n^{X1}{}_j$ and $n^{X2}{}_j$ the number X_1's and X_2's in the j^{th} interval, then:

$$SAD = \sum_{j=1}^{k} \left| n_j^{X2} - n_j^{X1} \right| \quad (1)$$

To calculate the Kolmogorov-Smirnov test, we use cumulative distribution functions: F_{X1} and F_{X2} for variables X_1 and X_2 respectively:

$$KS = \sup_x |F_{x2}(x) - F_{x1}(x)| \quad (2)$$

The value of KS is changed from 0 to 1. 0 means the two distributions are identical. 1 means two distributions have nothing in common.

IV. Experiments and Results

The use of representative process flows has caused the necessity to calibrate the simulation model for both stationary and transitional cases. This is due to the fact that we change the model when aggregating demand for representative process flows. Below we describe at first a calibration procedure, and then the simplification experiment results for each case.

A. Steady state case

After the demand was aggregated for the three selected representative process flows (see Fig. 1), it turned out that the model was no longer stable: bottlenecks were produced in the system (see step 1 in Fig. 2). Therefore, it was necessary to reduce the load on the bottlenecks (in our case these are workcenters w14 and w2).

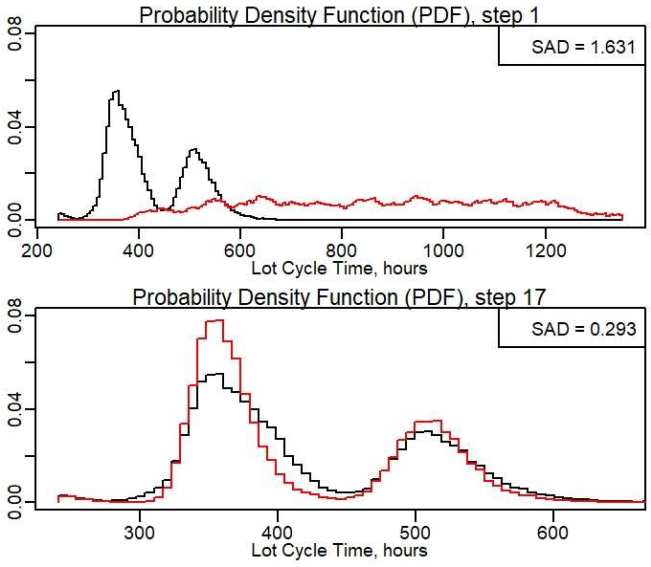

Fig. 2. Steady state case calibration: step 1 and step 17: black – base line (detailed model data, γ_1), red – detailed model data, γ_2.

The load reduction was done by using the processing efficiency factor. The factor is a parameter by which the processing time for a given tool set is divided. First 11 calibration steps were necessary to compensate for the increased load for bottlenecks. Further steps (12-17) were intended to bring the lot cycle time Probability Density Function (PDF) of γ_2 closer to the baseline (γ_1). To do this, it was necessary to increase the load on some other tool sets (table I). It should be noted that as a result of calibration, we can not get a PDF that is fully consistent with the base line, because we are dealing with a modified production system. Therefore, we used the minimization of SAD as a calibration criterion. Another important point is the desire to make changes to the model as little as possible. Therefore, we limited ourselves to only 7 machines.

TABLE I. STEADY STATE CASE. CALIBRATION PARAMETERS.

Calib. Steps:	Processing efficiency factor for tool sets						
	w14	*w2*	*w18*	*w4*	*w46*	*w32*	*w34*
1	1.0	1.0	1.0	1.0	1.0	1.0	1.0
2	1.03	1.0	1.0	1.0	1.0	1.0	1.0
3	1.04	1.0	1.0	1.0	1.0	1.0	1.0
4	1.04	1.02	1.0	1.0	1.0	1.0	1.0
5	1.04	1.03	1.0	1.0	1.0	1.0	1.0
6	1.05	1.03	1.0	1.0	1.0	1.0	1.0
7	1.05	1.04	1.0	1.0	1.0	1.0	1.0
8	1.05	1.05	1.0	1.0	1.0	1.0	1.0
9	1.06	1.05	1.0	1.0	1.0	1.0	1.0
10	1.06	1.06	1.0	1.0	1.0	1.0	1.0
11	1.06	1.07	1.0	1.0	1.0	1.0	1.0
12	1.06	1.07	0.70	1.0	1.0	1.0	1.0
13	1.06	1.08	0.70	1.0	1.0	1.0	1.0
14	1.06	1.08	0.70	0.80	1.0	1.0	1.0
15	1.06	1.08	0.70	0.80	0.80	1.0	1.0
16	1.06	1.08	0.70	0.80	0.80	0.75	1.0
17	1.06	1.08	0.70	0.80	0.80	0.75	0.9

After the calibration was completed, we started the simplification experiments. The essence of the experiments is to estimate the accuracy of the simplified model for two scenarios of γ_1 and γ_2. As a result of each experimental series we obtain 83 SAD and KS values, which can be represented as a line on the diagram: x axis - experiment number, y axis - accuracy measurement value (see Fig. 3).

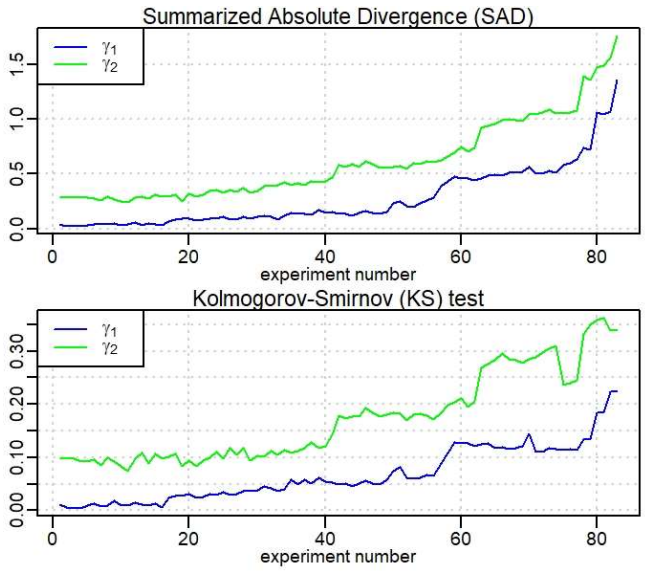

Fig. 3. Steady state case. SAD and KS for experiment series ζ_1.

Fig. 3 shows that the accuracy measurement value for γ_2 is considerably higher than that for γ_1. The higher the value of SAD and KS, the worse the accuracy of the simplified model. For example, a significant simplification of γ_1 (experiment #55) leads to the same accuracy as γ_2 without simplification. On the other hand, it should be noted that the overall accuracy measurement curves increase insignificantly for the first 40 experiments. This means that substitution of even 40 tool sets for delays does not result in a significant loss of model accuracy.

Let us consider the difference in accuracy on the example of one experiment #60. Fig. 4 illustrates the γ_1 and γ_2 PDFs for the experimental series ζ_1. It can be seen that the peaks of the γ_2 curve are higher, and the peak width are narrower. This is a consequence of using representative process flows, because we reduce the variability of the model. This can be especially seen in the left side of the PDFs. For γ_1 (blue line) we see two peaks (one big and one small), which corresponds to different process flows. For γ_2 (green line), we see only one peak, as this part of the PDF is related to only one representative process flow. From this we can conclude that γ_2 will always be worse matched with the base line than γ_1 for this part of the PDF. This is what we see in Fig. 3, where the γ_2 curve is higher than the γ_1 curve, i.e. the accuracy of γ_2 is worse than the accuracy of γ_1.

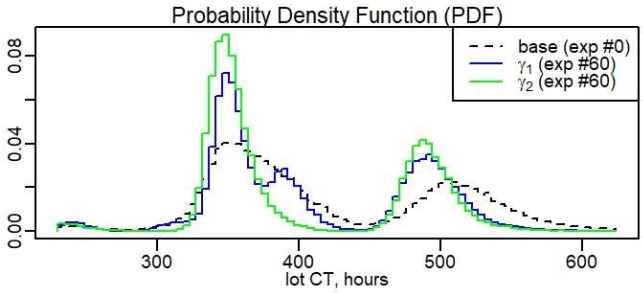

Fig. 4. Steady state case, ζ_1. Comparison γ_1 and γ_2 (exp #60).

Unfortunately, in this paper it is not possible to provide diagrams of all performed experiments because of the large amount of data. Therefore, to present the whole set of 10 experimental series and to estimate which of the sieve functions allows to get the best accuracy value, the SAD and KS values for 83 experiments were summed up and summarized in the table II. Bold font in this table indicates the minimum values for the column.

TABLE II. STEADY STATE EXPERIMENT RESULTS

Sieve functions	Sum of accuracy measurements			
	SAD γ_1	*SAD* γ_2	*KS* γ_1	*KS* γ_2
ζ_1	22.70	50.75	5.59	14.35
ζ_2	18.77	49.15	4.97	13.38
ζ_3	22.99	49.98	4.90	13.19
ζ_4	17.65	46.86	**4.04**	12.60
ζ_5	15.96	42.60	4.18	11.95
ζ_6	**15.87**	39.28	4.06	11.45
ζ_7	15.93	**39.22**	4.27	11.68
ζ_8	19.15	47.76	5.42	11.83
ζ_9	23.90	51.72	4.79	**11.28**
ζ_{10}	41.03	70.53	8.93	15.70

It can be seen that the minimum values correspond to different sieve functions. However, in practice it is preferable to use one approach. Such approach could be the use of $\zeta 6$ (marked in yellow background); because it gives values of

accuracy criteria close to the minimum ones. It should also be noted that the criterion most often used in practice (utilization – ζ2) gives values worse than those of ζ6.

B. Transient state case

For the transient case (changing demand), calibration is a very complex task. In reality, calibration is mostly based on the gut feeling of the expert and his or her ability to evaluate the future. In our case, it was facilitated by the presence of a simulation model as a baseline, i.e. we knew exactly the future for the model being calibrated. On the other hand, we intentionally chose a sufficiently variable demand to highlight the difficulties faced by the simulation modeling expert in calibrating the model.

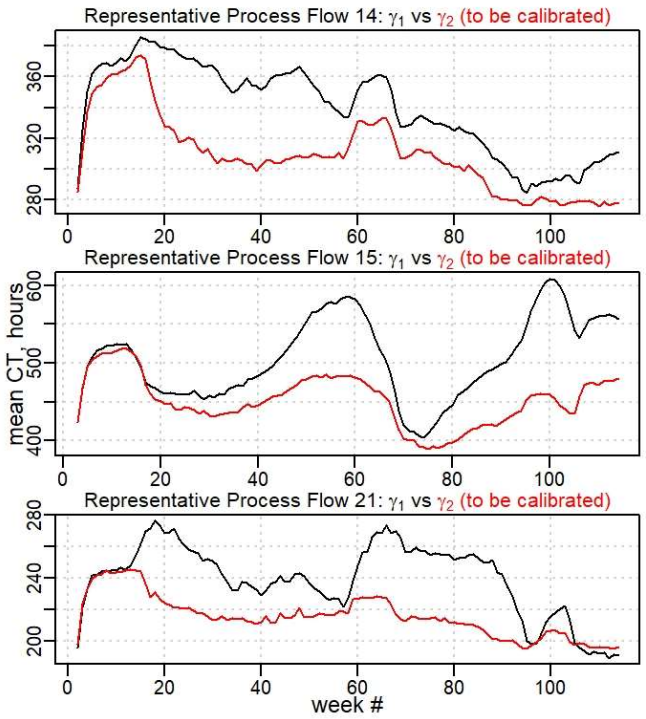

Fig. 5. Transient state case. Additional calibration step 1.

For the transient state case, we used a model already calibrated for the steady state case. Fig. 5 shows the dependence of the mean cycle time for representative process flows on the week number. The first 14 weeks the values are close enough. Then one can see a significant difference. This means that additional model calibration is required for the transient case.

By coincidence, we also needed 17 steps to calibrate the γ_2. Although we also wanted to minimize the number of tool sets to be calibrated, we had to adjust the processing efficiency factor values for 23 of them. In addition, it was not possible to set these values constant, but it was necessary to make them time dependent. Another challenge was the fact that a large number of runs were required to assess the accuracy of a single calibration step. Since the production plan in some weeks on certain items contains only one lot, it was decided to carry out 100 runs to get for such a situation at least 100 values of the cycle time. For comparison, 5 runs were enough in a steady state case. In addition to the diagrams presented here, we used time-dependent WIP graphics for each of the tool sets to visualize the calibration process more accurately. However, the calibration criterion was the matching of PDFs.

During the calibration process, the interdependencies of different representative process flows were discovered. For example, the mean cycle time of the representative process flows 14 and 21 in the period 35-60 weeks depend on the same tool sets. However, a more accurate calibration for 14 results in an overestimated mean cycle time for 21. It was decided to consider this offset for representative process flow 21 as acceptable, since up to 95 weeks it is limited to one lot per week. Thus, the result shown in Fig. 6 was obtained at step 17 of calibration. In Fig. 6, it can be seen that the lines do not fully match. Nevertheless, we considered this calibration acceptable. In order to understand the reason for this decision, we should consider it in detail at the PDF level for one week.

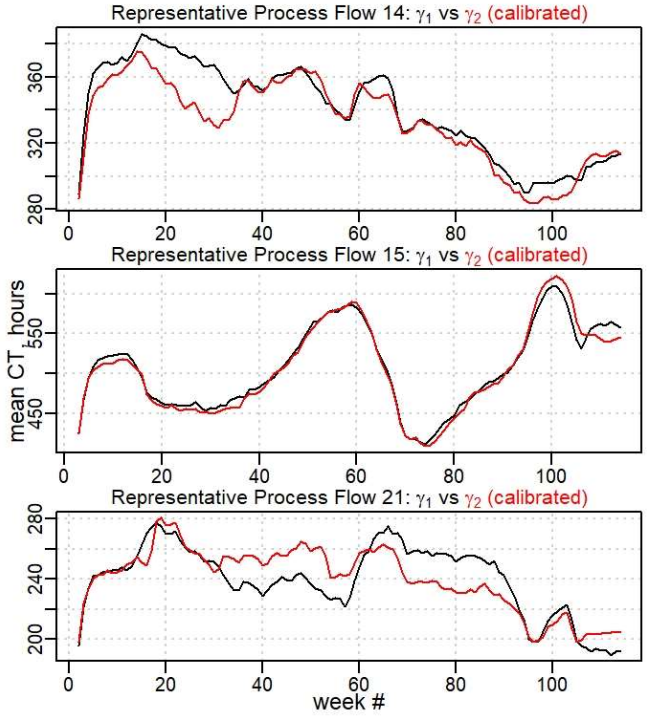

Fig. 6. Transient state case. Additional calibration step 17.

Fig. 7 shows the PDF for week #50. Here we can see more clearly, what happens to the model in a specific period. In step 1, a red line shows the model calibrated for a steady state case. If comparing Fig. 2 (Step 17) and Fig. 7 (Step 1), one can see that the γ_2 model does not respond to changing demand.

There is another important fact when calibrating: how to evaluate PDF coincidence. We have indicated above that it is preferable to use SAD for this purpose. It should be noted that a simple comparison of cycle times means is not valid. For example, if we look at the mean curves in Fig. 6 for the representative flow 14 for week #50, we see that the red line is slightly above the black line. However, it doesn't make sense to calibrate the process flow 14 downwards as this will only lead to an even bigger divergence in the PDF, as we can see in Fig. 7 for step 17 (300 – 400 hours).

It is necessary to recall that during the calibration process when using representative process flows it is not possible to obtain fully coincident PDFs. In addition, for the transient state case, this fact is even more obvious. For step 17 in Fig.7, one can see a good match on the right side of the PDF. However, for the middle part, the baseline is represented by a two-peak distribution (black line, 300-400 hours), which cannot be achieved using only one-peak distribution (red line,

300-400 hours). Therefore, the SAD value, which was achieved during the calibration process, is rather high (0.417). However, compared to the initial value (1.327), we were able to achieve a certain level of success.

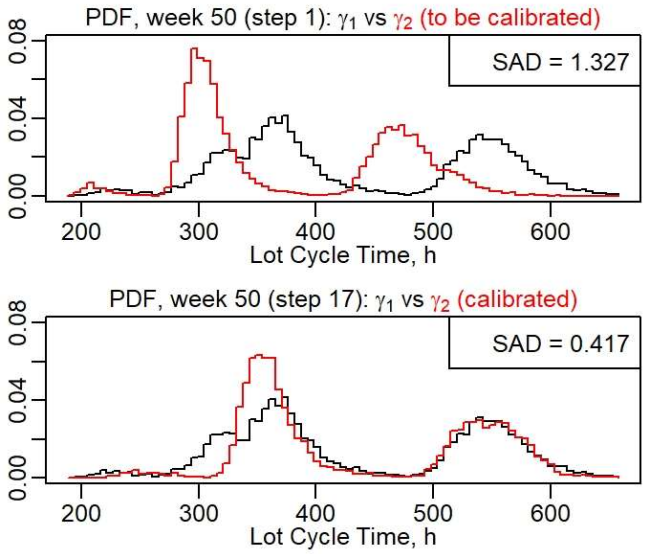

Fig. 7. Transient state case. Calibration step1.

Fig. 8 shows the initial (step 1) and final (step 17) SAD values of γ_2 for the different weeks. It can be seen that it varies considerably from week to week. However, because of situations similar to Fig. 7, these values are difficult to make better. The nonlinear dependence of the cycle time on the processing efficiency factor causes additional complexity. As soon as we approach the limit load for a given tool set, the cycle time starts to grow exponentially (operative curve).

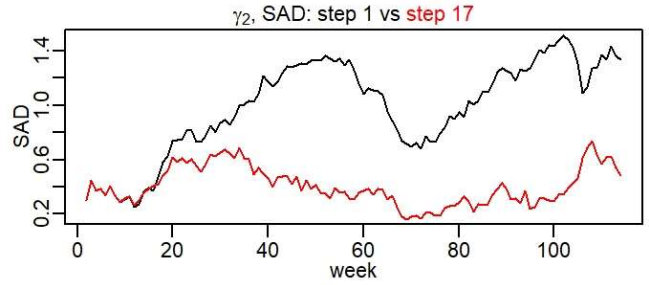

Fig. 8. Transient state case. Comparison step 1 and step 17.

After the calibration was completed, simulation experiments similar to those performed for the steady state case were carried out. The same accuracy measurements (SAD and KS) were estimated. The same accuracy measurements (SAD and KS) were estimated. However, in the case of the transient case, the values for each week were first calculated, and then the mean value for all weeks (104) was derived to evaluate the accuracy of this experiment. Eventually the sum for each experiment series was calculated and the results are summarized in table III. In this table, the minimum values are in bold. We suggest using ζ_6 as a sieve function similar to the steady state case (marked in yellow background). This means using queue length as a criterion for ordering machines to substitute for constant delays. It should be noted that queue length as a criterion works better than utilization (ζ_2), as in the case of the steady state (see table II). It should be noted that queue length works better than utilization as a criterion (ζ_6). This result shows that only a detailed study of simulation model simplification contributes to finding the best sieve functions.

TABLE III. TRANSIENT STATE EXPERIMENT RESULTS

Sieve functions	Sum of accuracy measurements			
	SAD γ_1	*SAD* γ_2	*KS* γ_1	*KS* γ_2
ζ_1	65.20	87.89	17.80	25.50
ζ_2	66.53	90.44	18.58	26.40
ζ_3	67.20	89.59	18.56	25.92
ζ_4	65.48	90.37	17.62	25.90
ζ_5	64.34	88.82	**17.04**	**25.06**
ζ_6	**63.64**	**85.58**	17.25	25.10
ζ_7	65.50	85.64	18.00	25.44
ζ_8	64.83	89.55	17.47	25.57
ζ_9	67.03	89.90	17.82	25.33
ζ_{10}	71.41	96.48	18.83	26.62

The accuracy measurements values in table III deserves special attention. Comparing the accuracy measurement values in table II and table III, it can be seen that the values in table III are more than twice as big as in table II. The main reason for this is a small number of model runs (5), which resulted in a small number of variables used to build distribution functions. Fig. 9 illustrates this by the example of week number 50.

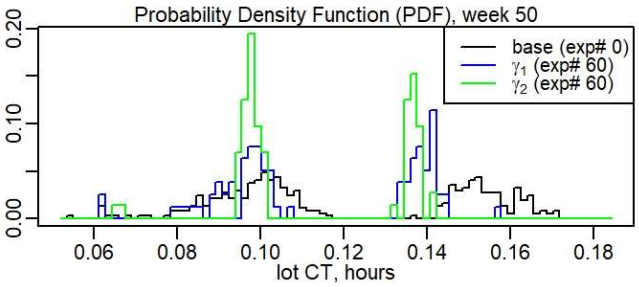

Fig. 9. Transient state case, ζ_1. Comparison γ_1 and γ_2 (exp #60, week #50).

Comparing Fig. 7 and 9 we can see a significant difference in the smoothness of PDF curves. This is due to the fact that 100 model runs were carried out for Fig. 7 and only 5 for Fig. 9. The reason for this is the time of model run together with postprocessing (about 2 minutes) and the number of necessary experiments (83 only for one series). If we perform 5 runs for one experiment, then we get 83*5*2 = 830 minutes (half a day) for one series, and if 100 runs, then 16600 minutes (11.5 days). We had to make 40 series of experiments for this study. On the other hand, we could allow 100 runs for calibration experiments (transient state), since only 17 steps were performed. This study is more of an exploratory activity with the purpose of technology testing and cutting off the known poor sieve functions. We plan to carry out more detailed research in the future.

V. DISCUSSION

Obviously, the most debatable issue in this paper is the quality of our calibration. We considered only the least time-consuming method of calibration – calibration with the help of process efficiency factor. Another more accurate method is calibration of processing time for each individual processing step. In practice, both approaches are used.

However, our task was to show how representative process flows are used to simplify the simulation model and to highlight the need for calibration itself, as the section II papers

overlook this fact. Since usually the reasons for calibration are referred to as inaccurate source data and modeling itself.

A further controversial issue may be the choice of criteria for assessing the accuracy of simplification. Here we refer to our paper [3] where we discussed this problem in detail.

Regarding the assessment of accuracy of measurements for a transient case, we consider it quite acceptable for our task – exploration study. However, for more accurate conclusions it is certainly necessary to perform more runs. We suppose that 100 runs should be enough. In particular, the calibration of the transient case confirmed this assumption.

We also would like to emphasize that according to the results of our study, it is more appropriate to choose the sieve function ζ_6 (queue length) rather than ζ_1 or ζ_2 (utilization). This fact allows us to look at the bottlenecks and simplification from a different perspective. Most papers on this issue argue that utilization is the criterion for assessing the bottleneck. On the other hand, some simulation experts suggest using queue length. However, since many papers have already established an understanding of the bottleneck as a tool set with the highest utilization, as well as other papers linking the simplification of simulation models to the bottlenecks, we propose to avoid the term "bottleneck" and instead use, for example, "tool importance" when considering the simplification of simulation models. This will allow researchers to look at the problem more comprehensively and perform more diverse experiments. We have considered ten different sieve functions in this paper based on these considerations.

VI. Conclusions

In this paper, we considered using representative process flows and constant delays to simplify the simulation model. The following conclusions can be drawn.

- When using representative process flows it is necessary to calibrate the simulation model. For a highly variable demand (transitional case), the calibration task is very difficult.
- Calibration using the process efficiency factor only is simple and suitable for exploration studies. However, for deeper investigations it is necessary to use additional calibration based on processing steps.
- Using representative process flows leads to less accuracy than using constant delays only.
- Substituting more than half of the tool sets for constant delays does not lead to significant deterioration of the simulation model accuracy.

In future research we plan to add processing time in addition to the process efficiency factor when calibrating the simulation model and to consider other dispatching rules (Critical Ratio, Earliest Due Date, Operation Due Date). We will expand our approach by using a forecasting procedure to evaluate sieve functions in near-reality conditions where there is no knowledge of the future. Eventually, these approaches will be implemented in a real Infineon simulation model.

Acknowledgment

A part of the work has been performed in the project iDev40. The iDev40 project has received funding from the ECSEL Joint Undertaking (JU) under grant agreement No 783163. The JU receives support from the European Union's Horizon 2020 research and innovation programme. It is co-funded by the consortium members, grants from Austria, Germany, Belgium, Italy, Spain, and Romania.

References

[1] Van der Zee, D. J. 2019. "Model Simplification in Manufacturing Simulation – Review and Framework". Computers & Industrial Engineering, 127: 1056-1067.

[2] I. Stogniy and W. Scholl. 2019. Using delays for process flow simplification. In Proceedings of the 2019 Winter Simulation Conference, 2372 – 2383. Piscataway, New Jersey: IEEE.

[3] I. Stogniy and W. Scholl. 2020. Using accuracy measurements to evaluate simulation model simplification. In Proceedings of the 2020 Winter Simulation Conference. Piscataway, New Jersey: IEEE. *(in press)*

[4] MIMAC Datasets. http://p2schedgen.fernuni-hagen.de/index.php, accessed 20th November 2020.

[5] S. Rank , C. Hammel, T. Schmidt, J. Müller, A. Wenzel, R. Lasch, and G. Schneider. 2016. "The Correct Level of Model Complexity in Semiconductor Fab Simulation – Lessons Learned from Practice". In Proceedings of the 27th Annual SEMI Advanced Semiconductor Manufacturing Conference (ASMC) Saratoga Springs 16-19 May 2016, 133-139. Saratoga Springs, New York: IEEE.

[6] R. Piplani and S. A. Puah. 2004. "Simplification Strategies for Simulation Models of Semiconductor Facilities". Journal of Manufacturing Technology Management 15(7): 618-625.

[7] M. U. Moris, A. H. Ng, and J. Svensson. 2008. Simplification and aggregation strategies applied for factory analysis in conceptual phase using simulation. In Proceedings of the 2008 Winter Simulation Conference, 1913-1921. IEEE.

[8] H. Ehm, 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, 2119-2130. IEEE.

[9] W. Scholl, M. Mosinski, B. P. Gan, P. Lendermann, P. Preuss, and D. Noack. 2012. A multi-stage discrete event simulation approach for scheduling of maintenance activities in a semiconductor manufacturing line. In Proceedings of the 2012 Winter Simulation Conference, 2263 - 2272. IEEE.

[10] M. Mosinski, T. Weissgaerber, S. L. Low, B. P. Gan, and P. Preuss. 2017. An easy approach to extending a short term simulation model for long term forecast in semiconductor industry. In Proceedings of the 2017 Winter Simulation Conference, 3636-3645. IEEE.

[11] Law, A. M. 2015. Simulation Modeling and Analysis. 5th ed. New York: McGraw-Hill Education