Proceedings of the 2020 Winter Simulation Conference
K.-H. Bae, B. Feng, S. Kim, S. Lazarova-Molnar, Z. Zheng, T. Roeder, and R. Thiesing, eds.

MULTI-LEVEL OPTIMIZATION WITH AGGREGATED DISCRETE-EVENT MODELS

Simon Lidberg
Tehseen Aslam
Amos H.C. Ng

School of Engineering Science
University of Skövde
Högskolevägen, Box 408
Skövde, 541 28, SWEDEN

ABSTRACT

Removing bottlenecks that restrain the overall performance of a factory can give companies a competitive edge. Although in principle, it is possible to connect multiple detailed discrete-event simulation models to form a complete factory model, it could be too computationally expensive, especially if the connected models are used for simulation-based optimizations. Observing that computational speed of running a simulation model can be significantly reduced by aggregating multiple line-level models into an aggregated factory level, this paper investigates, with some loss of detail, if the identified bottleneck information from an aggregated factory model, in terms of which parameters to improve, would be useful and accurate enough when compared to the bottleneck information obtained with some detailed connected line-level models. The results from a real-world, multi-level industrial application study have demonstrated the feasibility of this approach, showing that the aggregation method can represent the underlying detailed line-level model for bottleneck analysis.

1 INTRODUCTION

Multi-objective optimization (MOO) of discrete-event simulation (DES) models representing manufacturing systems, also known as simulation-based optimization (SBO), is moving from a research topic to industrial applications at an increasing rate (Negahban and Smith 2014). Utilizing simulation and optimization to improve production processes gives companies a competitive advantage and can increase the success of planning and commissioning new production lines (Dudas et al. 2014).

The concept of the fourth industrial revolution, commonly referred to as Industry 4.0, has become important for companies to respond to rapidly changing markets and conditions (Goienetxea Uriarte et al. 2020). Industry 4.0 also incorporates smart manufacturing wherein factories can utilize diagnostic, predictive, and prescriptive analytics. Diagnostic analysis implies identifying root cause and effect on problems by analyzing past and current performance. Predictive analysis is the typical use of a simulation model where ”what if” scenarios are tested and compared, such as changing buffer sizes, new demand scenarios, or introducing new variants. Finally, prescriptive analytics offers a range of solutions to a problem and can help develop future courses of action (Jain et al. 2015). Offering accurate prescriptive advice to a decision-maker on the factory level, utilizing MOO with genetic algorithms, is the aim of this paper.

The highest level in the manufacturing hierarchy for this paper will be the factory level, here referred to as the factory. The factory contains production lines, each composed of interconnected workstations and/or machines forming a production flow to transform raw material into intermediate products and finished goods. Machine will be used to refer to both physical machines and workstations in the remainder of this paper.

There are issues with SBO, among others, the demanding computational time needed to complete an optimization run. The required computational time of the optimization is based on model running time, multiplied by the number of replications used to obtain statistically sound results, and multiplied by the number of iterations, or generations, of the specified genetic algorithm. Time needed to complete a simulation project, one of the grand challenges mentioned by Fowler and Rose (2004), increases time required for an optimization study. The field of model simplification is not well established, but interest is increasing over time, with special interest at the Winter Simulation Conference and in the semiconductor industry (van der Zee 2019; Robinson 2011). Approaches to decreasing or reducing the computational cost of a simulation model are studied by several authors, utilizing different techniques (Chwif et al. 2006; Lefeber and Armbruster 2011; Duarte et al. 2007). One approach is model abstraction or aggregation of the coded model, previously reported by Lidberg et al. (2018), enabling SBO on the factory level.

Through a real-world, multi-level industrial application study, this paper investigates if the same set of parameters identified in an aggregated model on the entire factory level matches the parameters identified with a simulation model that is a connection of multiple detailed line-level models. If the same parameters are identified with the two different modeling approaches, then it can demonstrate the feasibility of the aggregation approach and would allow a decision-maker to prioritize improvement areas of the entire factory, be assured of the efficacy of the improvements, without the need of resorting to the computationally expensive detailed line-level models to create the factory model.

To identify bottlenecks on the parameter level and prescribe the sequence in which to remove them, the SCORE method has been developed and studied in multiple applications and domains (Bernedixen et al. 2015; Ng et al. 2018). SCORE utilizes the Theory of Constraints by Goldratt and Cox (1984) and prescribes the removal of these constraints by utilizing SBO, most often with the genetic algorithm NSGA-II or NSGA-III (Deb et al. 2002; Deb and Jain 2014). The advantage of this method is the ability to identify, not just machines – as in Roser et al. (2002) – but which operating parameters, e.g., processing time per variant, as bottlenecks and ordering them in the sequence most beneficial for removal. By identifying parameters related to product variants as bottlenecks the production schedule is also considered. Each parameter is assigned either its original value or an improved value. The improved value provides an indication of the effect if the removal of the bottleneck, e.g., decreasing the processing time by 20%, has taken place. Even though it could be difficult to implement the improvement in practice, it is paramount information for the decision-maker if the optimization can show where the minimal improvement in the system can provide the largest leverage of the overall system performance. The number of improvements is set as a minimization objective in a multi-objective optimization study along with other objectives of interest. The parameters included in the Pareto-optimal front are then tallied resulting in prevalence and ranking. The most common parameter, i.e., with the highest frequency of inclusion in the non-dominated solutions, is considered to be the main bottleneck in the system.

The paper is organized as follows; Section 2 details the proposed optimization experiment and presents the application study and factory model, Section 3 presents the results of both factory and line-level bottleneck optimizations, and the paper concludes with conclusions and suggestions for further work while also summarizing the contributions to science and practice.

2 METHOD

This section beings with a brief explanation of the aggregation technique used in the optimization on the factory level is shown in Section 2.1. Several of these modules are used to build a high-level model of the industrial system, which are detailed in Section 2.2. Lastly, the optimization settings and setup are explained in Section 2.3.

2.1 Aggregate Model

To conserve computational resources and enable the optimization of several production lines in a factory, using simple line models is a requirement. Several methods are used in literature based on numerical, analytical, or modeling approaches applied to the conceptual model or coded model (Frantz 1995; Robinson

2011; van der Zee 2019). This study uses a model aggregation method where a small number of components and parameters are used to construct a generic model that is reconfigurable to model an arbitrary production line. A schematic description of the generic simulation model is shown in Figure 1, and the method is further detailed in Lidberg et al. (2019), Pehrsson (2013), and Pehrsson et al. (2015).

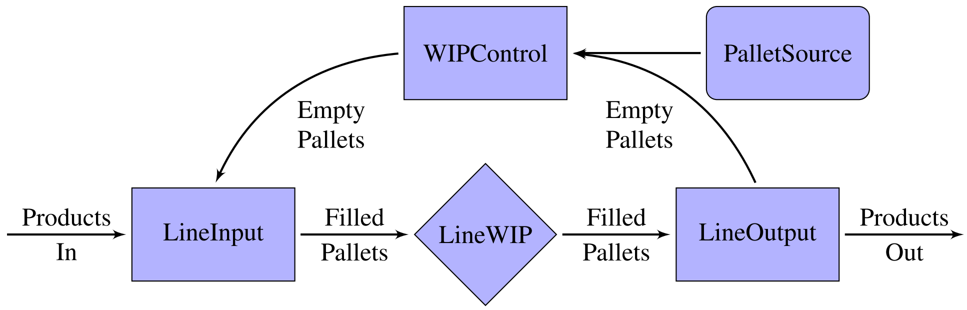

Figure 1: Schematic description of the aggregation technique, adapted from Lidberg et al. (2019).

The basic principle of the modeling technique is a closed-loop pallet system where pallets transport the products from entry to exit, called *LineInput* and *LineOutput* respectively, while keeping work in progress (WIP) at an average level. After releasing the product at *LineOutput*, the pallet enters a parallel processing object, *WIPControl*, acting as a delay mechanism. The processing time, i.e., return delay, in *WIPControl* is determined by an exponential distribution function with a mean value, or scale parameter, derived from Little's Law (Little 1961), shown in Equation 1. Following the time spent in *WIPControl*, the pallets are available for parts at *LineInput*. Additions to the technique, made in Lidberg et al. (2019), allows for variant based setup improving the accuracy of the technique. The data input required are *Availability*, mean down time *MDT*, maximum WIP *MaxWIP*, average WIP *AvgWIP*, processing time *PT*, and minimum lead time *MinLT*. The technique is built using generic modeling components of the selected DES software enabling the technique to be a drop-in replacement of production lines or other components without reconfiguration of the preceding or succeeding operations, and offers near-constant computational time irrespective of parameter settings.

$$\mathrm{E}[X] = \frac{(MaxWIP - AvgWIP) * ProcessingTime}{Availability} \tag{1}$$

2.2 Industrial System

The system studied is an automotive engine manufacturing plant with three production stages: machining, automated assembly, and final assembly. Each stage is separated by a finished goods inventory (FGI) with products to be delivered to the subsequent stage by means of automated transport. There is one FGI for each machining component, one shared for the automated assemblies, and one FGI after final assembly which has been omitted in the model. The industrial system used in the application study, the same as for Lidberg et al. (2018), has been extended with several new production lines, both for machining and assembly. This is due to the introduction of new product families, requiring the machining and assembly of additional unique parts. These extensions have increased the complexity of the model, as well as the complexity of the optimization problem.

Due to the historical organizational setup, the purchasing and development of each component type and assembly have been led by individual teams. Each team has overseen setting the cycle times, and availability levels, to achieve a yearly production target for the production lines they control. This has led to differences between the capabilities of the different production lines. Differences can be seen in Table 6 for the *PT* and *Availability* targets, which results in the need for different shift patterns. Access to the application study has been conditional on obfuscating input and output data to protect company assets, although all data relations have been preserved.

2.2.1 Industrial System Model

Each production line is modeled in detail and regularly updated with input data. From these detailed models, data can be extracted to populate the aggregated factory model. The updated and extended model for the industrial system application study is shown in Figure 2. The machining lines (ML) are identified according to the type of component produced $\{A,\ldots,D\}$, and numbered for each parallel line of the same component. The automated assembly lines (AA) and final assembly lines (FA) are numbered sequentially for each type of assembly. The inputs of raw material to the machining lines are infinite and only the main products are modeled in the lines, sub-assemblies are omitted. To allow for the introduction of new

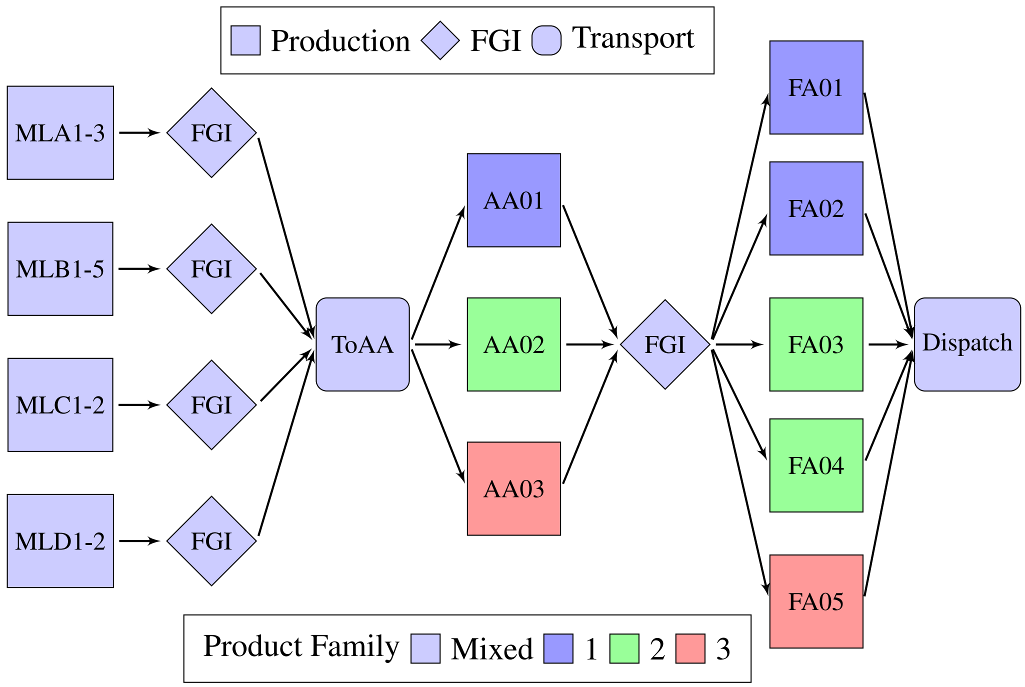

Figure 2: Layout of the model where machining lines are condensed. *ToAA* represents the automated transport to the different *AA*-lines and *Dispatch* is the output for the model. *AA* and *FA* lines are also dedicated to one product family compared to *ML* lines which are mixed.

product families additional production lines have been added. Component *A* has received the machining lines *MLA02* and *MLA03*, and *MLB03* and *MLB04* is added for component *B*. For component *C* and *D*, the machining lines have been reconfigured and extended for the new variants. To assemble the new products; one automated assembly line *AA03*, and one final assembly line *FA05* has been added while the others have been reconfigured.

The machining lines are modeled with one or several aggregated objects determined by two factors: size of the intermediary buffers in the line, and existing logical partitioning. The first factor where a machining line has a large internal buffer – intended for decoupling between shifts or decoupling several hours of production – the line will be modeled in parts. In the case of *MLA01*, three aggregation objects are used separated by large buffers, each aggregation object having separate parameters. The second factor, existing logical partitioning, divides a production line into several aggregation objects based on existing or arbitrary partitions of the physical system, e.g., team areas, production stages, data collection areas, production rate, or quality containment areas.

Several new variants have been added to enable the assembly of new products and the same variant can also be produced from multiple machining lines, thus adding additional complexity. Assignment of variants to machining lines is fixed. Each production line is running full output on every shift, with different shift patterns, see Table 6. The real industrial system can vary the production speed for different shifts to better match the requirements of the production plan. The production plan in the model is fixed. Improving the output of the whole plant is of larger interest for this study than matching the production plan. This also entails that production will continue, even though the production limit has been reached for that shift, in contrast to the real system where adherence to the production plan is highly prioritized.

2.3 Multi-Level Optimization

Optimizing on the factory level, to identify bottlenecks on the line-level, ensures that the improvements suggested are of benefit to the total system. The validity of the factory model is in part transitive, by using input data from valid detailed models of each production line to construct the aggregated models, and in part by comparison with historic plant output. In Section 1, the SCORE method of identifying bottlenecks in a production system was explained. The method relies on testing the effects of removing constraints by greatly improving certain parameters. The settings used for the optimization on factory level is detailed in Table 1, where each parameter, its abbreviation and symbol, the respective improvement value, and direction is shown.

The improvement to α is dependent on the original value of each production line shown in Table 6. For lines with $\alpha < 80\%$ the value is raised to 95%, otherwise, the value is raised to 98%. This is to mitigate the large differences in α between the different types of lines. Improving ζ, reducing the value by 20%, could result in a negative value in Equation 1 for cases where ε and ζ are close to equal, e.g., a production system with a pallet loop. Instead, by improving ζ, ε is also improved by the same amount. SCORE utilizes Boolean parameters where parameters are either improved, 1, or not improved, 0. Parameters used this way will be denoted with the hat operator, e.g., $\hat{\alpha}$, all subject to $\{0,1\}$. The two objectives used are $maxOut = max\left(\sum output\right)$ for maximizing output, and $minImp$ in Equation 2 for minimizing the number of improvements. The settings for NSGA-II are shown in Table 2.

Table 1: Improvements for each parameter shown with the abbreviation used and the direction of improvement.

Symbol	Parameter	Abbreviation	Improvement	Direction
α	Availability	Avb	10-15%	Higher
β	Mean Time To Repair	MTTR	10%	Lower
γ	Minimum Lead Time	MinLT	20%	Lower
δ	Processing Time	PT	10%	Lower
ε	Average WIP	AvgWIP	20%	Lower
ζ	Maximum WIP	MaxWIP	20%	Lower
η	Setup time	ST	20%	Lower

$$\begin{aligned} minImp = min\left(\sum_{i=1}^{n}(\hat{\alpha}_i+\hat{\beta}_i+\hat{\gamma}_i+\hat{\delta}_i+\hat{\varepsilon}_i+\hat{\zeta}_i+\hat{\eta}_i)\right), \quad \text{where} \\ n = \text{number of aggregation objects} \end{aligned} \tag{2}$$

3 RESULTS

This section presents the results for the factory level optimization in Section 3.1, choosing a line to further optimize in Section 3.2. After running a new optimization on the line-level, the bottleneck parameters identified are tallied and compared to the bottlenecks on the factory level in Section 3.3.

Table 2: Settings for the NSGA-II and NSGA-III optimization algorithms.

Parameter Name	NSGA-II	NSGA-III
Population size	120	200
Mutation type	Uniform	Uniform
Mutation distribution index	20	20
Mutation probability	0.0060241	0.0019342
Crossover type	Uniform Range	Uniform Range
Crossover distribution index	20	20
Crossover probability	0.9	0.9
Reference points		190

3.1 Factory Level Optimization

Following the SCORE methodology, the frequency of occurrence of a parameter in the non-dominated solutions are calculated for each improvement shown in Figure 3. The increases in output compared to the number of improvements made is shown in Figure 4, with diminishing returns clearly visible for $minImp > 5$. Comparison of the impact for different parameters is shown in Table 3, where six improvements are listed. The table shows improvements to the top bottlenecks identified in Figure 3.

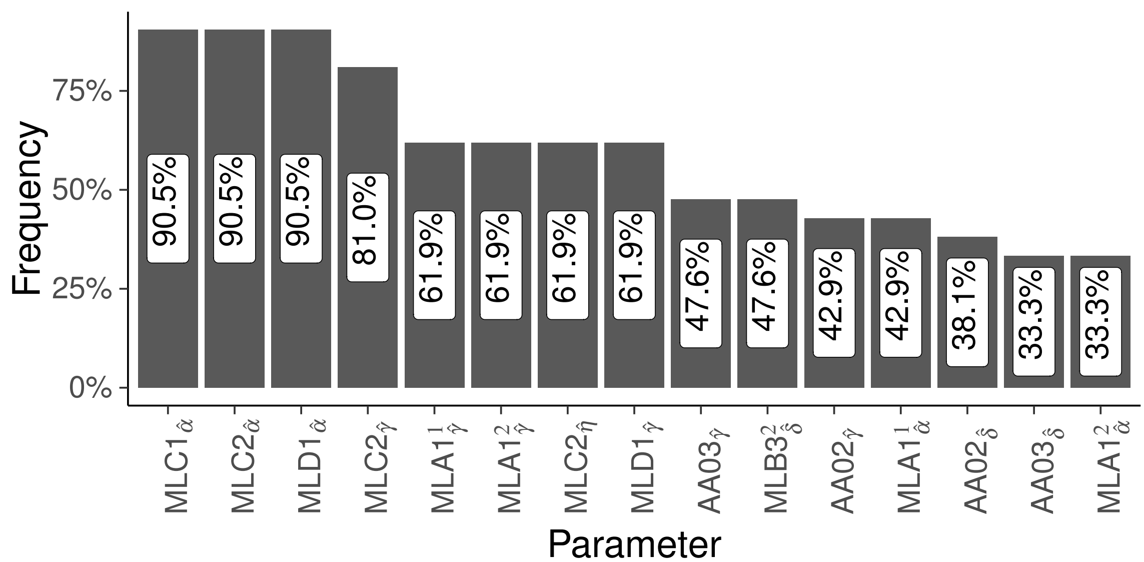

Figure 3: Top fifteen issues from the optimization on factory level, calculated from the first Pareto-optimal front. Frequency is the percentage of inclusion of the parameter in the solutions on the Pareto-optimal front. Sub-lines to each line are indicated by superscript.

The first four bottlenecks listed in the frequency chart have the most effect on the objective. For $minImp = 5$ and $minImp = 6$, the best solutions are not corresponding to the frequency chart. Instead, those parameters are located at position 11 and 13. The results prescribe an approach to improving the total manufacturing output from the factory, by priority and which parameter to improve. Compared to the baseline of zero improvements, removing five bottlenecks results in a possible improvement of over 12%.

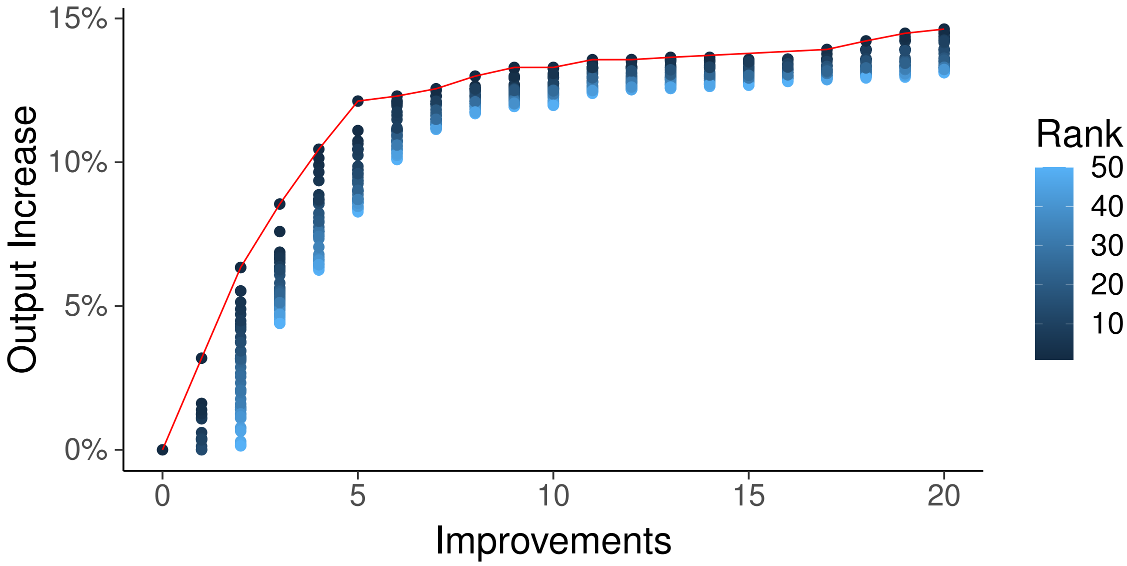

Figure 4: Results from the optimization on factory level showing the first 50 Pareto-optimal ranks with rank one solutions also shown connected by a red line. *Improvements* = *minImp* and Output is the total amount of products produced during the statistical collection period.

Table 3: Improvement to *maxOut* in percent, compared to baseline of zero improvements, for different combinations.

MLC1$_{\hat{\alpha}}$	MLC2$_{\hat{\alpha}}$	MLD1$_{\hat{\alpha}}$	MLC2$_{\hat{\gamma}}$	AA02$_{\hat{\gamma}}$	AA02$_{\hat{\delta}}$	minImp	maxOut
0	0	0	0	0	0	0	0.00%
0	1	0	0	0	0	1	3.19%
1	0	1	0	0	0	2	6.34%
1	1	1	0	0	0	3	8.55%
1	1	1	1	0	0	4	10.45%
1	1	1	1	0	1	5	12.13%
1	1	1	1	1	0	6	12.30%

3.2 Line-Level Optimization

The optimizations performed on the line-level models are configured differently from the factory-level optimizations. This is because, for the line level, the main interest lies in finding the improvements to a specific parameter already identified as significantly influencing on the factory level. *MLC2* had three parameters in the bottleneck ranking for the factory level and will be optimized with SCORE. *MLC2* is an automated machining line, where the machines are serviced by gantries and produces five different variants. Machining lines for component *C* require setup between variants, making them sensitive to production schedule changes and prone to running large batches, which in turn increases the need for a large FGI.

Three parameters from the factory level optimization points to *MLC2*; α, γ, and η. Improvements to α and η on the factory level are directly connected to improving the parameter of individual machines on the line-level. Improving γ for this line can be accomplished by the following actions: increasing availability of individual machines, lowering mean time to repair, decreasing buffer sizes, decreasing setup times, i.e., η, or changing the production schedule. Thus, the parameters affect each other, improving α or η would also affect γ.

A SCORE optimization with NSGA-III was setup with new settings, shown in Table 2, and objectives. An altered objective to determine the number of improvements made on the line-level is shown in Equation 3, with parameters shown in Table 4. The hat operator will also be used here to denote an improved parameter. The *maxOut* objective in the previous optimization is unchanged. To find results where also γ is improved, a new optimization goal is added as objective three, $minLT = min(\gamma)$. Improving η will benefit both objectives and is therefore not used as a separate objective.

$$
\begin{aligned}
minImp_{MLC2} = min\Big(\sum_{i=1}^{n}\sum_{j=1}^{m}\hat{\alpha}_{ij} + \sum_{i=1}^{n}\sum_{j=1}^{m}\hat{\beta}_{ij} + \sum_{i=1}^{n}\sum_{j=1}^{k}\hat{\delta}_{ij} + \sum_{i=1}^{n}\hat{\eta}_{i} + \sum_{i=1}^{s}\hat{\theta}_{i}\Big), \quad \text{where} \\
n &= \text{number of machines,} \\
m &= \text{number of failure profiles for each machine,} \\
s &= \text{number of buffers,} \quad \text{and} \\
k &= \text{number of variants for the system}
\end{aligned}
\tag{3}
$$

Table 4: Improvements for each parameter on the line-level shown with the abbreviation used and the direction of improvement. Subscripts for α and β denotes failure profiles, while for γ they denote product variants.

Symbol	Parameter	Abbreviation	Improvement	Direction
$\alpha_{1...j}$	Availability	Avb	15%	Higher
$\beta_{1...j}$	Mean Time To Repair	MTTR	50%	Lower
$\delta_{1...k}$	Processing Time Per Variant	PTV	25%	Lower
η	Setup Time	ST	25%	Lower
θ	Storage Capacity	SC	25%	Lower

After running the optimization for 40,000 iterations, the results are shown as two sets of figures, Figure 5 for improving γ, and Figure 6 for improving *maxOut*. Each result is ranked by only two objectives, *minImp* and either *maxOut* or *minLT*. The frequency charts are showing results for the first Pareto-optimal front, and the scatter plots are showing $minImp \leq 20$ in the first Pareto-optimal rank.

Summarizing the improvements to minimize γ, the most frequent are $\hat{\alpha}_2$, indicating the second failure profile designated for infrequent longer failures, and $\hat{\eta}$. Most notably, the lack of superscript in the parameters with $\hat{\alpha}_2$ shows that single sequential machines are the most affected. OP0160P1, which denotes the gantry serving machines in operation group 160, is sensitive to both $\hat{\alpha}_1$ and $\hat{\alpha}_2$; outages in the gantry affects the whole operation group. To maximize output of *MLC2*, many parameters are shared with the bottlenecks for minimizing γ, e.g., $\hat{\alpha}_2$. This objective indicates that improvements to δ, particularly for machines early in the production line, such as OP0005, are effective improvements.

MOO with conflicting objectives will provide a number of trade-off solutions, where none of the different solutions, in the same Pareto-optimal rank, are strictly better than another solution for all objectives. A summary of the most common parameters for the two extremes when applying ten improvements is shown in Table 5. The important improvements for *minLT* are $\hat{\alpha}_2$ and $\hat{\eta}$, while for *maxOut*, the most important improvement is $\hat{\alpha}_2$.

3.3 Line Improvements Applied to Factory Level

Removing a bottleneck, in SCORE terminology, implies an improvement of $10\% - 50\%$ depending on the parameter and setting. Achieving the exact improvement may not be feasible, economically or technically, and the ease of implementation can differ between improvement types. A small improvement was chosen by

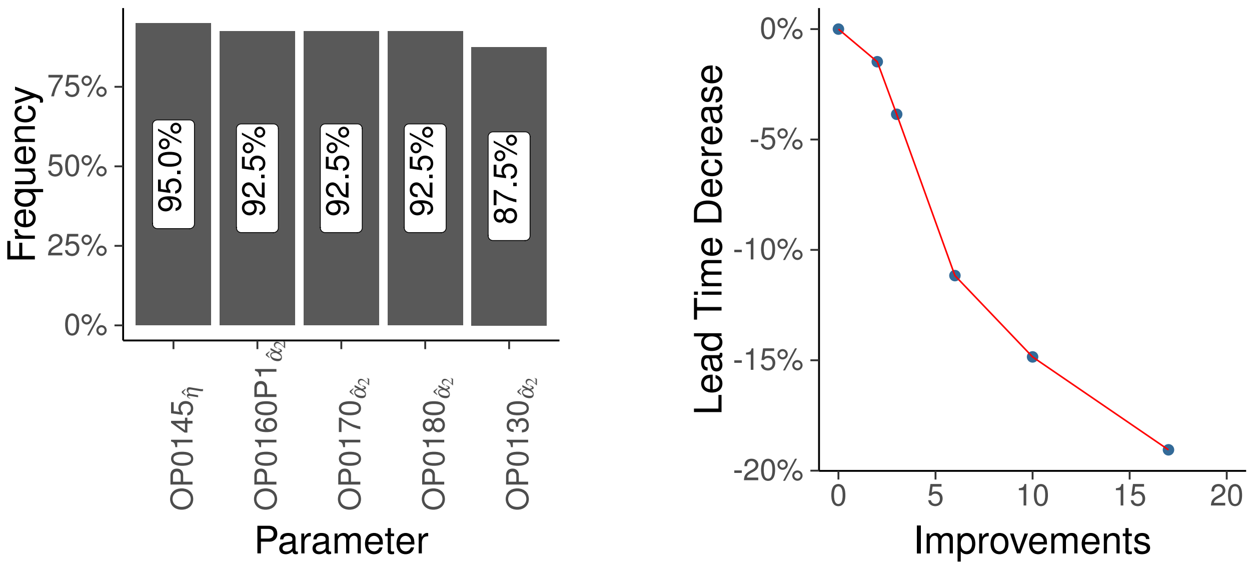

Figure 5: Frequency chart for the top five bottlenecks and a scatter plot showing the first Pareto-optimal front for $Improvements = minImp_{MLC2}$ cf. *minLT*, $minImp_{MLC2} \leq 20$.

Table 5: Best trade-offs for either objective after applying ten improvements summarized by type of improvement. Values are compared to the baseline of zero improvements.

$\hat{\alpha}_1$	$\hat{\alpha}_2$	$\hat{\beta}$	$\hat{\delta}$	$\hat{\eta}$	$\hat{\theta}$	maxOut	minLT
0	4	1	0	5	0	1.07%	-14.8%
1	5	2	1	1	0	1.83%	-11.4%
1	3	0	3	2	1	2.37%	-8.6%
1	4	2	1	1	1	4.02%	-7.4%
0	5	1	2	1	1	5.53%	-2.2%

increasing α by 2% and γ by 10%, delivering an increase of 5% in output for *MLC2*. Involving industrial engineers in the process of determining feasible improvements would increase the confidence in this value. On the factory level an improvement of 1.1% for *maxOut* was recorded. Although a small increase, the value of the finished products is higher than intermediate products in *MLC2*.

4 CONCLUSIONS AND FURTHER WORK

A multi-level MOO for an industrial application study has been performed where bottlenecks identified on the factory level has been improved in the context of the line-level, thus prescribing an improvement order beneficial to the entire system. Issues identified on the aggregated factory level was availability, lead time, and setup time for a particular line. These parameters were also found to be the most significant bottlenecks on the detailed line-level. This shows, albeit limited in scope, that an aggregated model on the factory level can identify relevant bottlenecks on the line-level and, therefore, gives a decision-maker on the factory level actionable improvement suggestions and a priority for investments without the need of connecting detailed line-level models. These improvements are also verified to be of benefit to the total industrial system avoiding sub-optimizations which can be introduced if only optimizing on the line-level.

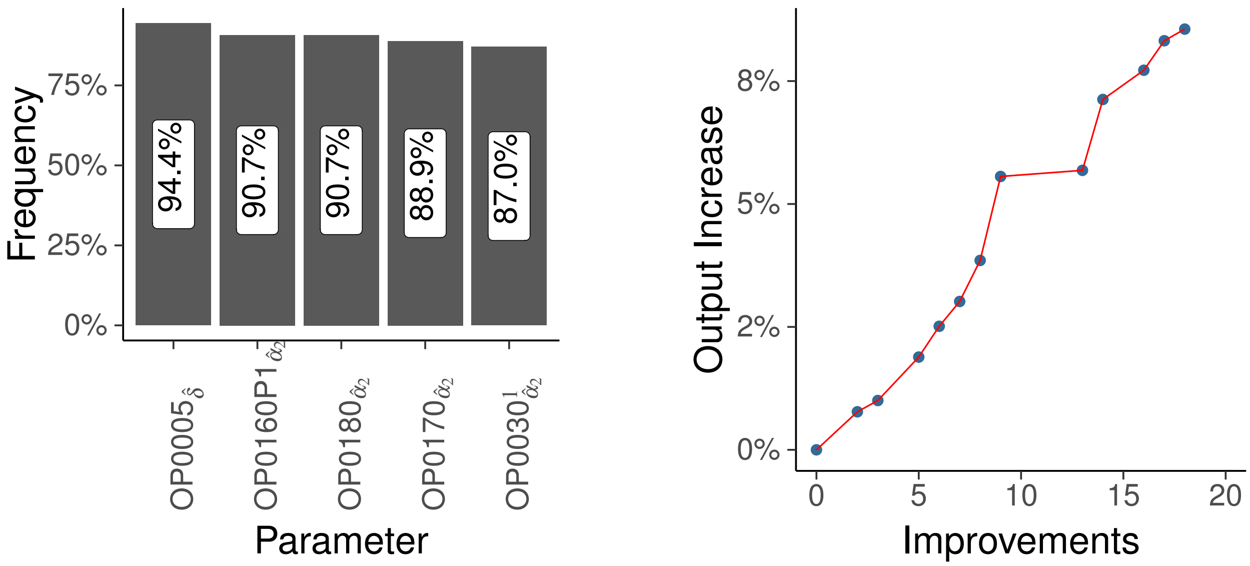

Figure 6: Frequency chart for the top five bottlenecks and a scatter plot showing the first Pareto-optimal front for $Improvements = minImp_{MLC2}$ cf. $maxOut$, $minImp_{MLC2} \leq 20$.

For further work, increasing the speed of the factory optimization, further enhancements to the aggregation model can be evaluated, and variance reduction techniques could be implemented to reduce the number of iterations needed. Another point of interest would be adding shifts combined with a variable speed parameter to simulate staffing requirements for a specific shift pattern. To improve the decision support to a decision-maker, adding economic parameters to the optimization, with costs for applying improvements on the line-level, would be beneficial. Automating the chain of generating data from detailed line-level models, populating or generating the aggregated factory model, running the multi-level optimization, and returning the improvements would increase the benefit for the industry.

ACKNOWLEDGMENTS

The authors would like to thank Leif Pehrsson, University of Skövde, for assisting with the setup of the research problem, and Jacob Bernedixen and Martin Andersson, University of Skövde, for help with the optimization setup and SCORE methodology. This research has been funded by Volvo Car Corporation and the Swedish Knowledge Foundation, the authors gratefully acknowledge their provision of research funding and access to the application study.

Table 6: Settings for each line in the factory level simulation with superscripts for sub-lines. Superscripts in the Line column shows the number of sub-lines. FA lines are identical. Shift parameters are inclusive and denotes, from 1 to 5: day, evening, night, weekend, and weekend nights. The data has been altered to protect company assets, although all data relations have been preserved.

Line	Shift	PT (s)	Avb (%)	MTTR (s)	AvgWIP	MaxWIP	MinLT (h)
$\mathrm{MLA1}^3$	4	30	85	300	400^1, 300^2, 350^3	600^1, $400^{2,3}$	$2^{1,2}$, 1^1
$\mathrm{MLA2\text{-}3}^3$	4	60	80^1, 85^2	900	436^1, 100^2	754^1, 200^2	4.5^1, 1^2
$\mathrm{MLB1}^2$	4	55^1, 53^2	90	380	250	300	2^1, 1^2
$\mathrm{MLB2}^2$	5	53	90^1, 85^2	380	250	300	2^1, 1^2
$\mathrm{MLB3}^2$	4	53	90	380	250	300	2^1, 1^2
$\mathrm{MLB4}^2$	5	190	95	600^1, 380^2	150^1, 250^2	200^1, 300^2	4^1, 1^2
MLC1	4	50	83	600	365	372	4
MLC2	4	45	83	600	350	500	5.1
MLD1	4	40	83	600	340	400	3
MLD2	4	60	83	600	540	600	5
AA01	3	45	85	300	120	135	1.3
AA02	3	40	85	300	110	125	1
AA03	3	45	85	300	120	135	1.3
FA	3	80	95	300	35	40	0.5

REFERENCES

Bernedixen, J., L. Pehrsson, A. H. Ng, and T. Antonsson. 2015. "Simulation-Based Multi-Objective Bottleneck Improvement: Towards an Automated Toolset for Industry". In *Proceedings of the 2015 Winter Simulation Conference*, 2183–2194. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers Inc.

Chwif, L., R. J. Paul, and M. R. P. Barretto. 2006. "Discrete Event Simulation Model Reduction: A Causal Approach". *Simulation Modelling Practice and Theory* 14(7):930–944.

Deb, K., and H. Jain. 2014. "An Evolutionary Many-Objective Optimization Algorithm Using Reference-Point-Based Nondominated Sorting Approach, Part I: Solving Problems with Box Constraints". *IEEE Transactions on Evolutionary Computation* 18(4):577–601.

Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan. 2002. "A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II". *IEEE Transactions on Evolutionary Computation* 6(2):182–197.

Duarte, B. M., J. W. Fowler, K. Knutson, E. Gel, and D. Shunk. 2007. "A Compact Abstraction of Manufacturing Nodes in a Supply Network". *International Journal of Simulation and Process Modelling* 3(3):115.

Dudas, C., A. H. Ng, L. Pehrsson, and H. Boström. 2014. "Integration of Data Mining and Multi-Objective Optimisation for Decision Support in Production Systems Development". *International Journal of Computer Integrated Manufacturing* 27(9):824–839.

Fowler, J. W., and O. Rose. 2004. "Grand Challenges in Modeling and Simulation of Complex Manufacturing Systems". *SIMULATION* 80(9):469–476.

Frantz, F. K. 1995. "A Taxonomy of Model Abstraction Techniques". In *Proceedings of the 1995 Winter Simulation Conference*, edited by C. Alexopoulos, K. Kang, W. R. Lilegdon, and D. Goldsman, 1413–1420. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers Inc.

Goienetxea Uriarte, A., A. H. Ng, and M. Urenda Moris. 2020. "Bringing Together Lean and Simulation: A Comprehensive Review". *International Journal of Production Research* 58(1):87–117.

Goldratt, E. M., and J. Cox. 1984. *The Goal: A Process of Ongoing Improvement*. Gower Publishing.

Jain, S., D. Lechevalier, J. Woo, and S.-J. Shin. 2015. "Towards a Virtual Factory Prototype". In *Proceedings of the 2015 Winter Simulation Conference*, WSC '15, 2207–2218. Piscataway, NJ, USA: Institute of Electrical and Electronics Engineers Inc.

Lefeber, E., and D. Armbruster. 2011. "Aggregate Modeling of Manufacturing Systems". In *Planning Production and Inventories in the Extended Enterprise*, Volume 151 of *International Series in Operations Research & Management Science*, 509–536. New York, NY, USA: Springer.

Lidberg, S., T. Aslam, L. Pehrsson, and A. H. Ng. 2019. "Optimizing Real-World Factory Flows Using Aggregated Discrete-Event Simulation Modelling: Creating Decision-Support Through Simulation-Based Optimization and Knowledge-Extraction". *Flexible Services and Manufacturing Journal*:1–25.

Lidberg, S., L. Pehrsson, and A. H. Ng. 2018. "Using Aggregated Discrete Event Simulation Models and Multi-Objective Optimization to Improve Real-World Factories". In *Proceedings of the 2018 Winter Simulation Conference*, edited by M. Rabe, A. A. Juan, N. Mustafee, A. Skoogh, S. Jain, and B. Johansson, 2015–2024. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers Inc.

Little, J. D. C. 1961. "A Proof for the Queuing Formula: L = λ W". *Operations Research* 9(3):383–387.

Negahban, A., and J. S. Smith. 2014. "Simulation for Manufacturing System Design and Operation: Literature Review and Analysis". *Journal of Manufacturing Systems* 33(2):241–261.

Ng, A. H., F. Siegmund, and K. Deb. 2018. "Reference Point Based Evolutionary Multi-Objective Optimization With Dynamic Resampling for Production Systems Improvement". *Journal of Systems and Information Technology* 20(4):489–512.

Pehrsson, L. 2013. *Manufacturing Management and Decision Support using Simulation-based Multi-Objective Optimisation*. Ph. D. thesis, De Montfort University.

Pehrsson, L., M. Frantzen, T. Aslam, and A. H. Ng. 2015. "Aggregated Line Modeling for Simulation and Optimization of Manufacturing Systems". In *Proceedings of the 2015 Winter Simulation Conference*, edited by L. Yilmaz, W. K. V. Chan, I. Moon, T. M. K. Roeder, C. Macal, and M. D. Rossetti, 3632–3643. Piscataway, New Jersey: IEEE.

Robinson, S. 2011. "Choosing the Right Model: Conceptual Modeling for Simulation". In *Proceedings of the 2011 Winter Simulation Conference*, edited by S. Jain, R. Creasey, J. Himmelspach, K. White, and M. Fu, 1423–1435. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers, Inc.

Roser, C., M. Nakano, and M. Tanaka. 2002. "Shifting Bottleneck Detection". In *Proceedings of the 2002 Winter Simulation Conference*, edited by E. Yücesan, C.-H. Chen, J. L. Snowdon, and J. M. Charnes, 1079–1086. Piscataway, New Jersey: Institute of Electrical and Electronics Engineers Inc.

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

AUTHOR BIOGRAPHIES

SIMON LIDBERG is an industrial Ph.D. student at Volvo Car Corporation and the University of Skövde. Graduating from the University of Skövde in 2011 with a M.Sc. in Industrial Informatics, he started working at Volvo Cars with simulation and optimization of production systems. In 2017 he started his industrial Ph.D. studies focusing on the creation of decision support by optimizing on the factory level using aggregated data and models derived from detailed DES models. He can be reached at simon.lidberg@his.se and simon.lidberg@volvocars.com.

TEHSEEN ASLAM holds a Ph.D. in industrial informatics from University of Skövde, Sweden. He has a M.Sc. degree in Manufacturing Management from Loughborough University and a B.Sc. in Automation Engineering from University of Skövde. His research interests include modeling, simulation and multi-objective optimization for the design and analysis of supply chains. His e-mail address is tehseen.aslam@his.se.

AMOS H.C. NG is a Professor of Production and Automation Engineering at the University of Skövde, Sweden. He holds a Ph.D. degree in Computing Sciences and Engineering. His main research interest lies in applying multi-objective optimization and data mining techniques for production systems design, analysis and improvement. His e-mail address is amos.ng@his.se.