

Model Adaptation for Dynamic Flotation Process Simulation

Markku Ohenoja, Mika Ruusunen, Matias Hultgren, and Antti Remes

Abstract— Dynamic production processes in mineral beneficiation have complex flowsheets and non-linear, time-varying behavior. They have also limited measurement capabilities. Hence, virtual models are seen as important tools for assisting in design, planning and operation. The development of operational dimension requires not only a suitable software architecture and virtual model, but also continuous validation or adaptation of the virtual model. In this paper, a framework establishing a digital twin for a flotation process is presented. The model adaptation is treated as a trajectory matching problem and realized with a Differential Evolution algorithm. The results are demonstrating the applicability of the presented approach in simulation environment together with a discussion on additional challenges foreseen in implementations to the real processes.

I. INTRODUCTION

The fourth industrial revolution, driven by the rapid development of information and communication technology, involves the utilization of big data, machine learning and cyber-physical systems [1]. Currently, simulation tools can efficiently be applied in various engineering problems such as in process design and production planning. The next wave in simulation, bringing the tools to real-time utilization, is a digital twin [2]–[4]. Hence, the digital twin is a key enabling technology for the realization of cyber-physical systems and Industry 4.0, seen for example as centralized analysis and control of the manufacturing processes [5]. Moreover, digital twins are expected to provide more integrated data analysis, prediction abilities and data visualization [6]. In process industries, such as mineral beneficiation, it is expected that the virtual models will be benefiting the operational dimensions such as advanced process control, fault diagnosis, self-recovery, intelligent soft sensing, decision making and value-chain optimization [1].

In order to facilitate the real-time decision making via the digital counterpart, the process model representing the physical system (digital model) must be dynamically updated. Based on the classification in terms of level of integration [7], with the up-to-date model, a digital shadow is established, whereas a digital twin requires automatic feedback to the process as well. The model's prediction ability can be compromised due to changes in the modelled process, such as sensor or equipment fouling, changing process conditions and inherent time variant and non-linear behavior of the process. Model adaptation is understood as automatically maintaining the model performance over time.

The adaptation strategies may range from simple off-line identification methods to machine learning approaches [8]. In

froth flotation applications, the attempts for the adaptive virtual models include e.g. PI controller-based adaptation [9] and recursive estimation techniques [10]. However, these cases assume direct pairing of process measurements and corresponding model parameters. Another approach based on trajectory matching is taken in [8], where the froth flotation adaptation is treated as a multivariable identification problem and the initial results were presented. In this paper, the framework of the real-time adaptation mechanism for non-linear process models presented in [8] is taken forward and the simulation results in sliding time windows are presented together with discussion on implementation to the real process.

II. ADAPTATION FRAMEWORK

Simulation of detailed physical process models has sufficiently high calculation costs. However, parallel computing and cloud-based systems can nowadays facilitate the utilization of such models in a trajectory matching method, as the simulation trials can be divided into multiple process models with reasonable effort. In addition, the length of the required simulation period to adapt the model has a significant effect on the calculation cost.

The adaptation framework taken in this study is illustrated in Fig. 1. The adaptation is performed in sliding windows to track the model parameters continuously using parallel process models. The adaptation algorithm aims to minimize the error between the estimated variables of the physical process model and the measured process variables. These adapted model parameters are then distributed also to the physical process model acting as a digital shadow. Finally, the adapted process model could then be utilized in decision support system (DSS), process optimization or process control fulfilling the requirement on automatic feedback and therefore realizing the digital twin.

A. Adaptation strategy

A high number of stochastic search methods, especially evolutionary optimization algorithms have been developed to solve the engineering optimization problems with high number of variables and/or non-linear functions, where the traditional gradient-based (derivative) methods often fail [11]. These methods aim to map the whole search space to reach a global optimum. It is also important to notice, that the global solution in practical problems is unknown [12]. In addition, the algorithms need to stop after a fixed number of function evaluations. Thus, the 'global optimum' usually refers to a solution outperforming its alternatives when the maximum number of function evaluation is reached. The evolutionary algorithms share the idea of mimicking the behavior of natural

Markku Ohenoja and Mika Ruusunen are with the Control Engineering Research Group, University of Oulu, P.O.Box 4300, 90014 Oulu, Finland (corresponding author: Markku Ohenoja, phone: +358-29-448-2473; e-mail: markku.ohenoja@oulu.fi).

Matias Hultgren and Antti Remes are with the Outotec Oyj, Rauhalanpuisto 9, P.O.Box 1000, 02230 Espoo, Finland. (e-mail: forename.surname@outotec.com).

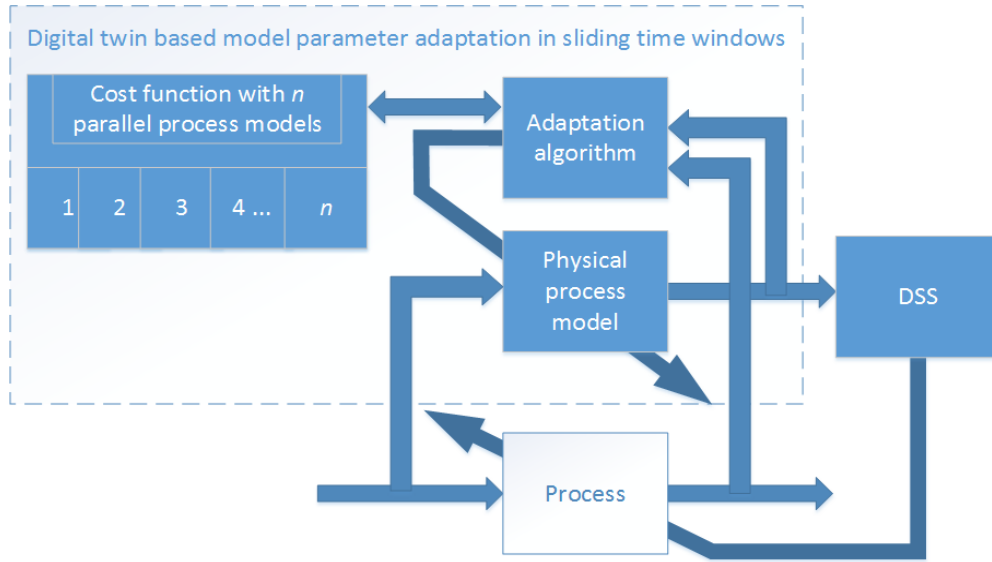


Figure 1. Real-time digital twin adaptation framework consisting the process, the adapted physical process model and its utilization in process via decision support system (DSS).

biological evolution and/or the social behavior of species [11]. They are popular especially among practitioners due to relatively easy implementation, available codes and less high-level mathematical preparation in comparison to deterministic optimization methods [12].

The applied optimization method for the adaptation algorithm is a Differential Evolution (DE) [13], [14]. DE is a global, stochastic search method suitable for mathematically difficult or real-valued functions. The main principle of this method is to utilize parameter differences in vector form for evolving candidate populations. These groups of candidates are then refined further using mutation and combination operations with some probability as with evolutionary algorithm in general. If newly formed vector of real valued parameters improves the existing solution, it is accepted into population. Optimal solution is typically expected after several generations.

The DE algorithm has only several hyperparameters; NP describes the size of population, G is for number of repetitions, F determines how the parameter vectors are mutated or changed, and CR how probably those are combined. The convergence properties can be controlled also with value to reach (VTR), which basically determines the ending point for the optimization.

The cost (objective) function (CF) applied in this case is the cumulative sum of the error between the simulated outputs $\hat{y}(k)$ and the measured process outputs $y(k)$ in a sliding window:

$$CF = 10^5 \cdot \sum_{i=1}^n \sum_{k=1}^m |\hat{y}_i(k) - y_i(k)| \quad (1)$$

where n and m represents the number of outputs and length of the sliding window, respectively. The purpose of the

multiplier in front of the cost function is to help with the visualization of the results.

B. Software architecture

The digital twin adaptation was implemented according to Fig. 2. The application consists of process simulation engines, a model parameter optimization module, a reference data depository and a centralized framework for coordinating the application components. The central framework was implemented using the Outotec® ACT platform [15]. The target process is simulated with the HSC Chemistry® software [16], and parallel simulation engines were used in order to increase computational performance. The parameter optimization with the DE algorithm is performed in Matlab.

The validation of the digital twin adaptation was carried out with simulated reference data in this paper. However, the software architecture allows a direct implementation of the application to the real process environment. In the present approach, design parameters of interest were varied with time, and the process was simulated with a dedicated HSC engine to generate the reference data. The adaptation then tries to replicate these parameter changes.

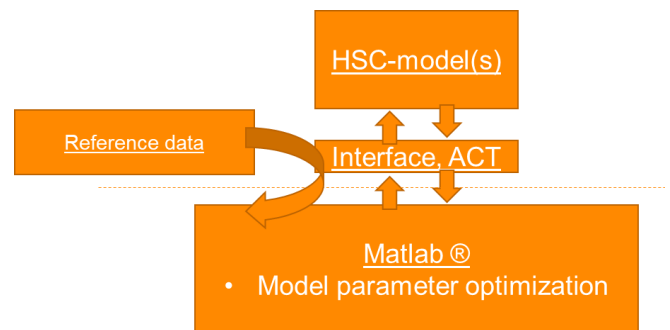


Figure 2. Conceptual figure for the data communication between the parameter optimization module, the process simulation engines (HSC) and the central coordinating framework (ACT).

C. Simulated process

The target process is an open-loop rougher flotation circuit for copper, depicted in Fig. 3. The circuit consists of four flotation cells, a conditioner tank, a pump sump and associated unit operations. The flotation feed was a bulk mixture of chalcopyrite (CuFeS_2), pyrite (FeS_2) and quartz (SiO_2). In the process, the flotation feed first enters the conditioner tank, and then goes through the stage-wise flotation process, where material is distributed between concentrate froth and tailings in each stage. The dynamic model for the adaptation was constructed with the HSC Sim flowsheet module of HSC, where the flotation bank units were described using the corresponding unit operation models. A detailed description of flotation bank modeling principles and the flotation unit operation models in HSC Sim is provided in [17]. Flotation cell and pump sump levels were regulated by adequately tuned PI controllers, with the outflow valve as manipulated variable for the flotation cells, and the pump frequency setpoint for the pump sump.

Four (bounded) design parameters were selected from the flotation bank model to be adapted based on output data: k_{Cp} , k_{Py} , k_{Qtz} and $WaterRec$. These parameters were deemed to be the most likely to drift during the process operation, as they are typically fit based on experimental data. k_{Cp} , k_{Py} and k_{Qtz} are rate constants (1/min) in the fast flotation kinetic equations of the chalcopyrite, pyrite and quartz minerals. These equations determine the mineral recovery as a function of the kinetic rate constant and the mass fraction of floatability component (see Equation (3) in [17]). $WaterRec$ is a tuning parameter in the water recovery equation of the flotation bank (see Equation (10) in [17]), where a linear dependency between the water recovery and the solids flowrate was assumed (parameter $b=1$). Six (measured) output variables were assumed to be available from the HSC Sim model outputs

for the parameter adaptation: the overall copper, iron and sulfur contents (wt-%) in the concentrate stream, the overall copper and iron contents (wt-%) in the tailings stream, and the overall copper recovery (%) in the concentrate. When validating the adaptation results, the solid material flowrates and stream compositions of the concentrate and the tailings were monitored more extensively to determine the impact of the chosen outputs.

D. Performance evaluation

The performance of the presented model adaptation strategy is primarily based on comparison of estimated and real output values. Especially in practical problems, this is the most important criterion as the real parameter values are not known. Hence, the tracking performance/modelling error is the most important evaluation criteria. This is also captured in the cost function (see (1)). Additionally, the comparison between the adapted model parameters and the real model parameters is made, as the results in this paper are from simulations.

III. RESULTS AND DISCUSSION

A. Functional testing

The functional testing was conducted in order to demonstrate the convergence abilities of the selected adaptation approach. The simulation scenario involved the simulation model described in Section 2, and constant parameters. The adaptation was triggered from the beginning of simulation and a window length of 10 seconds was used. The DE algorithm parameters applied to this test are depicted in Table I. The model parameter search space is presented in Table II. The selected algorithm parameters should allow the adaptation to proceed to (unnecessarily) low cost function values during functional testing.

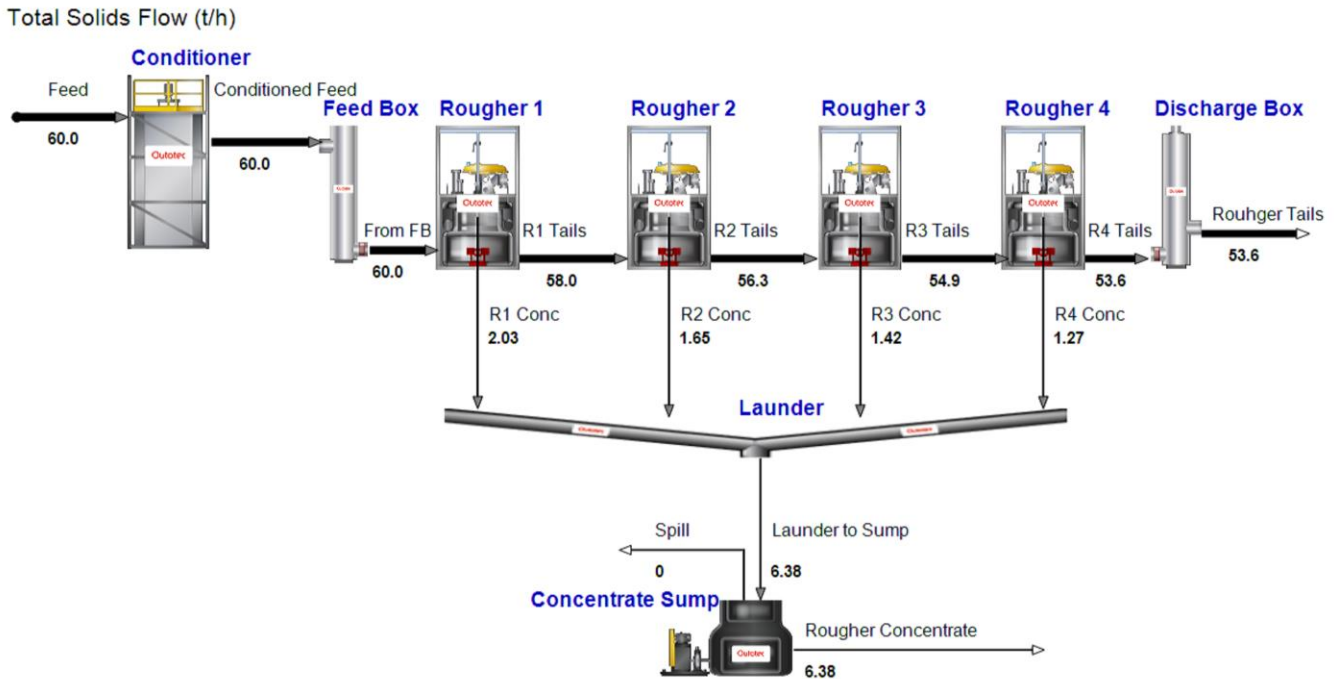


Figure 3. Simulated flotation process.

TABLE I. ALGORITHM PARAMETERS. NP – NUMBER OF MEMBERS IN POPULATION, G – MAXIMUM NUMBER OF GENERATIONS, F – STEPSIZE, CR – CROSSOVER PROBABILITY, VTR – COST FUNCTION VALUE TO REACH (STOPPING CRITERION)

Case	NP	G	F	CR	VTR
A	40	100	0.8	0.8	1e-4
B		60			7500

Fig. 4 shows the cost function value as a function of optimization generations. The algorithm shows a good convergence performance when no measurement noise is assumed (solid line). In the second case, the measurement is corrupted with SNR 30 dB (dashed line). Also, in this case the solution is converged after 50 generations, but with higher cost function values due to the measurement noise. The estimated parameter values for the cases without and with measurement noise are presented in Table II. Only the parameter k_{Qz} shows a high relative error whilst the other parameters have relative error smaller than 5% in the second case. In the noise free scenario, the parameter values have negligible relative errors.

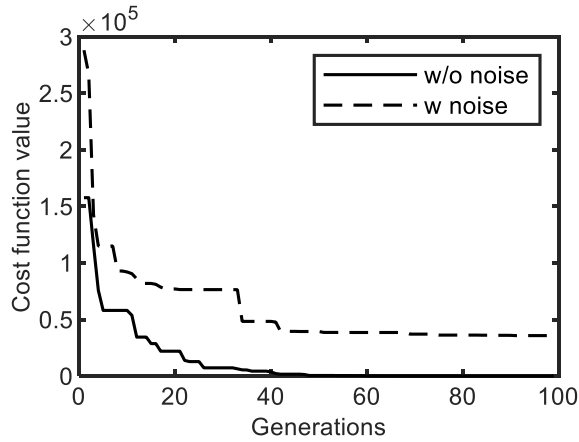


Figure 4. Convergence abilities of the adaptation. The solid line and dashed line represent the cases without and with the measurement noise, respectively.

TABLE II. THE SEARCH RANGE AND ADAPTATION PERFORMANCE WITH AND WITHOUT THE MEASUREMENT NOISE.

Parameters	WaterRec	k_{Ccp}	k_{Py}	$k_{Qz} * 10^3$
Real value	2.0	0.3	0.02	1.0000
W/O noise	2.0000	0.3000	0.0199	0.9999
W noise	1.9994	0.2998	0.0191	1.2777
Lower bound	1.4	0.1	0.005	0.1
Upper bound	3.4	0.5	0.45	10.0

B. Adaptation in sliding window

Next the adaptation was performed continuously in sliding windows. This time the simulation scenario involved changing parameter values; A 5% reduction in the parameters k_{Ccp} and k_{Py} take place at simulation time $t=36$ s. The adaptation is performed with a window length of 10 seconds. The simulation period was 70 seconds, therefore having seven adaptations performed during the simulation. The DE

parameters used are presented in Table I. The maximum number of iterations (generations) was decreased based on the results in functional testing. The stopping criterion was similarly altered in order to avoid excessive number of generations. No measurement noise was assumed. The sampling rate for the outputs was one second.

The behavior of estimated parameter values as a function of simulation time are presented in Fig. 5. It can be seen that the parameter values (crosses) are distributed along the real values (dashed line), and that the selected adaptation approach can follow the changes in the two parameters taking place during the simulation. The relative errors of parameter values are depicted in Fig. 6. It can be observed that the errors are negligible for two first parameters. The third and fourth parameters (k_{Qz}) have larger relative errors, which are still less than 5%.

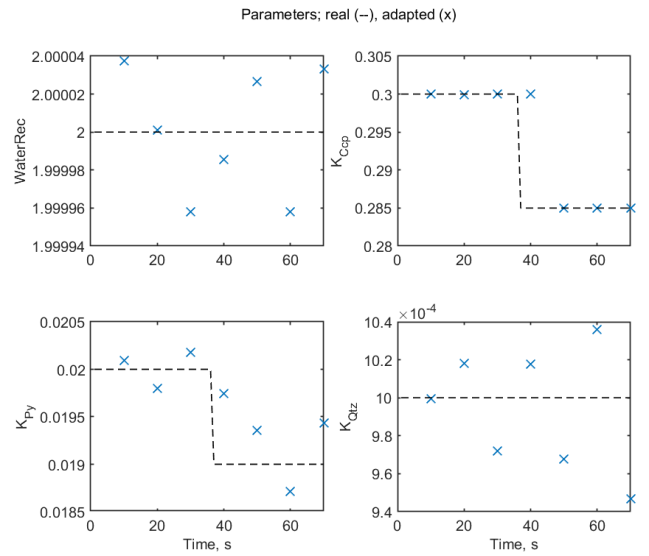


Figure 5. Parameter values in sliding time windows.

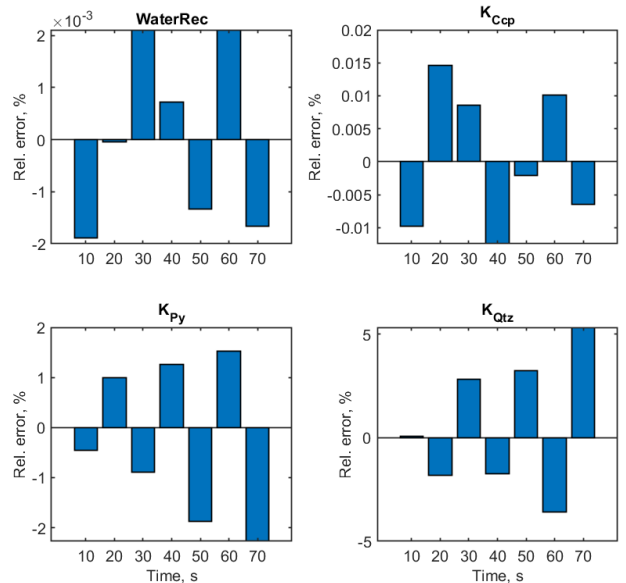


Figure 6. Relative error of parameter values in sliding time windows.

Finally, the comparison between the estimated data with adapted parameters and the real data is presented in Fig. 7. The figure represents the dynamic behavior of one process output, namely solids percentage in concentrate stream (Solids% Conc) with and without the adaptation together with the real values. Hence, deviations are expected once the parameters change at $t=36$ s. The simulation was continued for a period of 140 s, but the adaptation takes place only between $t=1\dots70$ as indicated in Fig. 5 and 6.

The results in Fig. 7 show that the adapted simulation model is able to follow the real simulation with a small error. Without adaptation, the error increases rapidly. In general, the relative errors were very small (less than 0.3%) for all outputs in this short simulation period used.

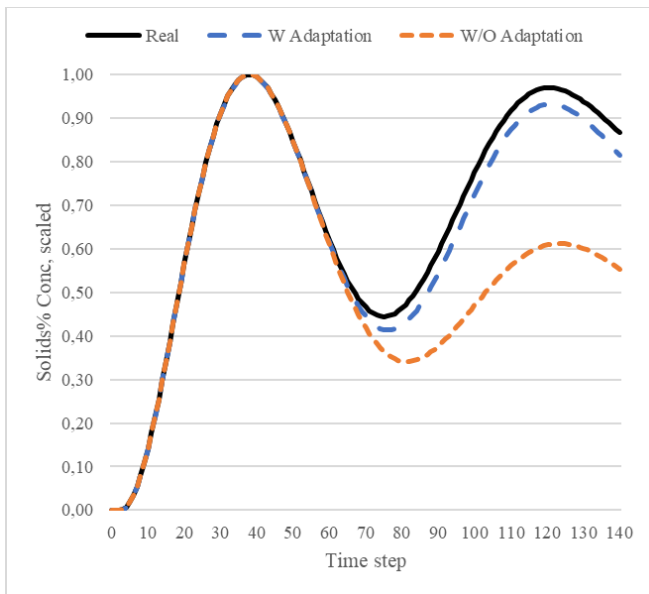


Figure 7. Dynamic behaviour of process output.

C. Considerations to real process and future work

The presented results demonstrate the applicability of the proposed framework for the adaptation of a dynamic, nonlinear flotation simulation model. Both the algorithm and software architecture are stressed, being able to turn the digital model into digital shadow. However, the framework requires additional insight before being implementable to the real process as a digital twin.

In order to realize the automated feedback between the model and the process, the communication architecture needs also to include prognosis capabilities. This requires additional data communication tools as discussed already in [10]. With regard to the model adaptation, it was here assumed that all six measured variables have equal sampling rates with high sampling frequency in comparison to the real process. Such a situation cannot currently be realized in a mineral beneficiation plant, where the elemental grades (mineral concentration measurements) typically involve on-line XRF-analyzers with a sampling interval of 10-30 minutes [18]. In fact, the analyzers are typically scheduled to sample multiple streams, therefore having different sampling rates for different streams and irregular sampling times. The longer sampling time in the real process means that the adaptation cannot capture changes with as fast dynamics as presented in

simulations above, although, the development efforts in rapid elemental analyzers [18] may facilitate the use of shorter adaptation windows in the future.

The second limitation to consider in the present work is the simulated reference data. As the flotation circuit model does not contain significant elements of pure delay like pipe transfer, bank parameter changes have immediate, albeit minor, effects on the simulated outputs (e.g. 1st order dynamics without delay). The results readily illustrated how even these small changes could be captured with the DE adaptation, which validated the suitability of the algorithm for detecting even the fastest changes in the flotation dynamics. However, this aspect should be considered when selecting the adaptation time window in future industrial applications, as the window should exceed the estimated minimum delay of the process, and also ensure a sufficient length of the adaptation timeseries to prevent adapting process parameters to measurement noise. The main contribution of the present results is thus to serve as a proof-of-concept for implementing the DE adaptation for industrial minerals processing plants: the adaptation was able to detect the changes in the process dynamics and correctly modify the model parameters based on a sufficiently small amount of output data. These outcomes are promising for creating effective industrial applications.

Another interesting aspect is the selection of process outputs and adapted model parameters in a large-scale system. In this case, expert knowledge was used to determine four adjustable parameters. However, with large-scale models the selection may not be straightforward. One option would be to screen the parameters and the measured variables using, for example, Monte Carlo simulations and statistical sensitivity analysis [19]. Also, the SNR of the different measurements need to be accounted in optimization. These aspects will be considered in more detail in an ongoing project with different types of process models and long-term simulations.

IV. CONCLUSION

The presented study demonstrated the applicability of the selected software architecture and the model adaptation approach enabling a digital twin for a simulated flotation process. The results were gained in simulation environment and some of the foreseen challenges in actual implementations were discussed. The development will continue in an ongoing project and be stressed with different types of models.

ACKNOWLEDGMENT

This work was developed under the financial support received from The EU Framework Programme for Research and Innovation – H2020-SPIRE-2015 (IbD® - Intensified by Design. GA-680565) and Business Finland project APASSI.

REFERENCES

- [1] B. Sun, S.-L. Jämsä-Jounela, Y. Todorov, L. E. Olivier, and I. K. Craig, "Perspective for equipment automation in process industries," *IFAC-Pap.*, vol. 50, no. 2, pp. 65–70, Dec. 2017, doi: 10.1016/j.ifacol.2017.12.012.
- [2] S. Weyer, T. Meyer, M. Ohmer, D. Gorecky, and D. Zühlke, "Future Modeling and Simulation of CPS-based Factories: an Example from the Automotive Industry," *IFAC-Pap.*, vol. 49, no. 31, pp. 97–102, Jan. 2016, doi: 10.1016/j.ifacol.2016.12.168.

- [3] T. Gabor, L. Belzner, M. Kiermeier, M. T. Beck, and A. Neitz, "A Simulation-Based Architecture for Smart Cyber-Physical Systems," in *2016 IEEE International Conference on Autonomic Computing (ICAC)*, Jul. 2016, pp. 374–379, doi: 10.1109/ICAC.2016.29.
- [4] H. Zhang, Q. Liu, X. Chen, D. Zhang, and J. Leng, "A Digital Twin-Based Approach for Designing and Multi-Objective Optimization of Hollow Glass Production Line," *IEEE Access*, vol. 5, pp. 26901–26911, 2017, doi: 10.1109/ACCESS.2017.2766453.
- [5] T. H.-J. Uhlemann, C. Lehmann, and R. Steinhilper, "The Digital Twin: Realizing the Cyber-Physical Production System for Industry 4.0," *Procedia CIRP*, vol. 61, pp. 335–340, Jan. 2017, doi: 10.1016/j.procir.2016.11.152.
- [6] Y. He, J. Guo, and X. Zheng, "From Surveillance to Digital Twin: Challenges and Recent Advances of Signal Processing for Industrial Internet of Things," *IEEE Signal Process. Mag.*, vol. 35, no. 5, pp. 120–129, Sep. 2018, doi: 10.1109/MSP.2018.2842228.
- [7] W. Kritzinger, M. Kärner, G. Traar, J. Henjes, and W. Sihn, "Digital Twin in manufacturing: A categorical literature review and classification," *IFAC-Pap.*, vol. 51, no. 11, pp. 1016–1022, Jan. 2018, doi: 10.1016/j.ifacol.2018.08.474.
- [8] M. Ohenoja *et al.*, "Towards Mineral Beneficiation Process Chain Intensification," *IFAC-Pap.*, vol. 51, no. 21, pp. 201–206, Jan. 2018, doi: 10.1016/j.ifacol.2018.09.418.
- [9] M. Friman and P. Airikka, "Tracking Simulation Based on PI Controllers and Autotuning," *IFAC Proc. Vol.*, vol. 45, no. 3, pp. 548–553, Jan. 2012, doi: 10.3182/20120328-3-IT-3014.00093.
- [10] J. Pietilä, J. Kaartinen, and A.-M. Reinsalo, "Parameter estimation for a flotation process tracking simulator," *IFAC Proc. Vol.*, vol. 46, no. 16, pp. 122–127, Jan. 2013, doi: 10.3182/20130825-4-US-2038.00048.
- [11] E. Elbeltagi, T. Hegazy, and D. Grierson, "Comparison among five evolutionary-based optimization algorithms," *Adv. Eng. Inform.*, vol. 19, no. 1, pp. 43–53, Jan. 2005, doi: 10.1016/j.aei.2005.01.004.
- [12] Y. D. Sergeyev, D. E. Kvasov, and M. S. Mukhametzhanov, "On the efficiency of nature-inspired metaheuristics in expensive global optimization with limited budget," *Sci. Rep.*, vol. 8, no. 1, pp. 1–9, Jan. 2018, doi: 10.1038/s41598-017-18940-4.
- [13] R. Storn and K. Price, "Differential Evolution – A Simple and Efficient Heuristic for global Optimization over Continuous Spaces," *J. Glob. Optim.*, vol. 11, no. 4, pp. 341–359, Dec. 1997, doi: 10.1023/A:1008202821328.
- [14] A. K. Qin, V. L. Huang, and P. N. Suganthan, "Differential Evolution Algorithm With Strategy Adaptation for Global Numerical Optimization," *IEEE Trans. Evol. Comput.*, vol. 13, no. 2, pp. 398–417, Apr. 2009, doi: 10.1109/TEVC.2008.927706.
- [15] Outotec, *Outotec(R) ACT*, 2019. <https://www.outotec.com/products-and-services/technologies/analyzers-and-automation/act-advanced-process-control/> (accessed Jan. 24, 2020).
- [16] A. Roine, *HSC Chemistry®*, 2019. www.outotec.com/HSC (accessed Jan. 24, 2020).
- [17] P. Seppälä *et al.*, "Development and calibration of a dynamic flotation circuit model," *Miner. Eng.*, vol. 96–97, pp. 168–176, Oct. 2016, doi: 10.1016/j.mineng.2016.07.004.
- [18] I. Kejonen, O. Haavisto, J. Martikainen, V. Suontaka, and B. Musuku, "Improving grade control efficiency with rapid on-line elemental analysis," *Miner. Eng.*, vol. 124, pp. 68–73, Aug. 2018, doi: 10.1016/j.mineng.2018.05.002.
- [19] I. M. Sobol', "Global sensitivity indices for nonlinear mathematical models and their Monte Carlo estimates," *Math. Comput. Simul.*, vol. 55, no. 1, pp. 271–280, Feb. 2001, doi: 10.1016/S0378-4754(00)00270-6.