IN-FIELD SIMULATION FOR PROCESS TUNING IN INDUSTRY 4.0 APPLICATIONS

Michael Rathmair, Christoph Luckeneder, Marcus Meisel and Stefan Wilker

E384 - Institute of Computer Technology, TU Wien

INTRODUCTION

Modeling and simulation approaches are well embedded into today's development flows for industry 4.0 applications. Model-based design, virtual prototypes, and virtual commissioning enable cost-efficient verification and optimization during the design phase. Unfortunately, in many cases, the use of simulation processes ends as soon as a design is deployed to production. The approach of the presented work is to migrate simulations to the operation phase of a production process. Therefore, so-called in-field simulation methods, monitor the current process state and simulate its behavior for a limited future time period. The simulation results are then used for analysis and subsequent tuning of parameters with the goal to optimize the production process under a specific objective (e.g., efficiency, throughput). This described in-field simulation procedure has even more impact if besides classical functional simulation potentially occurring uncertainties are considered. For example, in industry 4.0 applications such uncertainties can be unpredictable changes in environmental conditions (temperature, humidity, etc.), reduced availability of controllers caused by variations of QoS parameters in communication channels, variations in the performance of electronic components, etc.

CHALLENGES OF THE APPROACH FOR INDUSTRY 4.0 APPLICATIONS

In this section, we identify and discuss some challenges and of course advantages of the introduced in-filed simulation approach for industry 4.0 smart factory applications.

Models and simulation software: Models used for the design process has to be migrated to in-field simulation. A critical part is the abstraction level of the models which directly reflects the complexity, finally affecting the performance of the in-field simulation. For this work, we use C/C++ based models derived from high level behavioral SystemC [3] descriptions of the application. Also, a model of the process environment has to be integrated into the simulation. Uncertainties are represented as parameter deviations in models which consequences that values in the system no longer are represented numerically but as ranges. However, modeling uncertain parameters is getting increasingly challenging due to the rising number of variabilities and complex uncertainty effects (e.g., dependencies). Thus, classical onte-Carlo based methods will be no longer sufficient, and new semi-symbolic and formal approaches may come into play [1].

Computation resources: As illustrated in Figure 1 in a smart factory application there are numerous computation cores integrated in the included automation components such as robots, PCs, Panels, etc. Some of the processing resources may be permanently or partially unused (indicated in red in the figure). These resources may be loaded with the proposed in-field simulation software. However, these resources may be very diverse and distributed.

Process optimization and decision-making: Optimization and decision-making algorithms are primarily essential for the performance of the full production process. Thus, optimization procedures based on the results obtained form in-filed simulation runs, have to be explicitly defined for a given application including a behavioral model of the system's environment [2]. However, innovative enhanced technologies such as machine learning, collaborative decision-making algorithms and blockchain techniques are potentially applicable.

Observability, controllability: A requirement for the smart factory process itself is that there is adequate sensor equipment installed. The current state of the production process has to be observed continuously. This data is a basis for the execution of simulation runs. Further, the process parameters have to be tunable in a way to maximize the performance even under the presence of uncertainty.

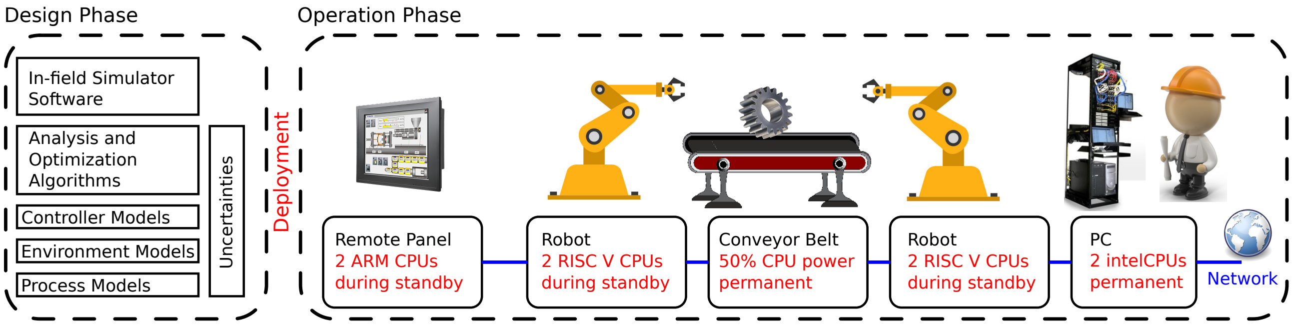

Figure 1: An industry 4.0 smart factory application including unused computation resources executing in-field simulation software

FIRST RESULTS AND DISCUSSION

For a first prototype implementation, we used the system simulator SystemC [3] in combination with a C++ library for semi-symbolic uncertainty modeling. Both is compiled for the ARM processor architecture. The simulation is then deployed on a Raspberry Pi platform. Until now our research focus is more on the management and distribution of the simulation software modules in available processing cores in the field. However, there is an ongoing experiment using the proposed in-field simulation concept for a smart lighting application.

CONCLUSION AND VISION FOR SMART CITIES

As described in the second section we highlight our operation-phase simulation for an industry 4.0 smart factory applications. However, our approach is extendable for smart cities. In such an architecture partially free computation resources are available nearly everywhere. All of them are in principle somehow connected to the internet. Our vision is to realize a collaborative simulation application for optimizing processes in the smart city. Such processes are for example smart power grid applications where the generation of energy has to be planned under the presence of uncertainties in the grid load. Besides the technical implementation, in smart cities also new business model for providing, managing and utilizing in-field simulation resources are possible, which potentially makes the approach very attractive for customers.

REFERENCES

[1] Grimm C., Rathmair M.: *"Dealing with Uncertainties in Analog/Mixed-Signal Systems"* Invited. 54th Annual Design Automation Conference 2017 (DAC'17), p. 35:1 - 35:6.

[2] Zvyagin L. S.: *"System analysis in optimization and decision making"*. 2017 IEEE International Conference on Control in Technical Systems (CTS), p. 161 - 164.

[3] http://www.accellera.org/downloads/standards/systemc