Towards Model-driven, Simulation-assisted Control Application Engineering
A Doctoral Research Path

Timo Vepsäläinen
Tampere University of Technology, Department of Automation Science and Engineering,
P.O. Box 692, 33101, Tampere, Finland

Keywords: Model-driven Engineering, Simulations, Model Transformations, Automation and Control.

Abstract: This paper, intended to the doctoral consortium of the Simultech 2013 conference, describes the problems, methodologies, objectives and status of doctoral research of the author. The research conforms to the constructive approach of design science within the application domain of industrial automation and control. The focus of the research is on model-driven, simulation-assisted engineering of automation and control applications with consideration of both basic control and safety-related control systems. The general research question to be answered is whether or not automation and control application development can benefit from model-driven engineering and the techniques enabled by it. The answer will be based on answers to smaller research questions related to industrial applicability of the general developed modelling approach, ability to simulate models at design-time and ability to include safety documentation in models.

1 INTRODUCTION

Complex industrial processes related to manufacturing, process and energy sector plants as well as machinery applications require continuous control, coordination and supervision to ensure their productivity and safety. Currently, the main implementation technology of such control systems is software. With dedicated hardware, e.g. PLC (Programmable Logic Controller) or DCS (Distributed Control System) platforms, buses and distributed I/O units, software has replaced earlier implementation technologies of control functions such as analogous and non-programmable electronics.

Software solutions are flexible and efficient by nature. They enable control and coordination of several devices and sub-systems from a single or few processing units. Software solutions can be further developed and updated even without changes to hardware. Software enables the use of sophisticated control algorithms that may require, for example, on-line optimization. Use of software also enables the reuse of existing solutions in form of libraries of re-usable classes, modules or function blocks. However, at the same time, the challenges related to software development have increased.

The development of a control application for an industrial process is a complex task that requires collaboration with and information from professionals of multidisciplinary backgrounds, e.g. process, control, hydraulics and electrical engineering. Control applications are often characterized with high real-timeliness, dependability, safety and other quality requirements. As a consequence, and because of the focal role of software applications, the efficiency of control application development process has become an essential competitiveness factor for industrial players in the domain. Exploring possibilities to improve efficiency of automation and control application development is also a profound objective of the doctoral research.

In general, the research is based on the assumption that efficiency of software development can be improved by reducing the total amount of required work per produced application. This can be accomplished by *re-using existing work* when possible as well as by reducing the amount of non-profitable and manual work. Re-use may be accomplished by, for example, designing and using re-usable controllers whereas reducing the amount

Vepsäläinen T. (2013).
Towards Model-driven, Simulation-assisted Control Application Engineering - A Doctoral Research Path.
In *Doctoral Consortium*, pages 12-19
DOI: 10.5220/0004637200120019

of non-profitable work by use of automated information transfer between development phases. The engineering process of applications should thus enable fluent re-use of both existing general knowledge and libraries of parameterizable solutions that can be applied to recurring design tasks. Applications and solutions should also be validated and verified as early as possible in order to minimize the effect of errors and design flaws to subsequent work and accordingly to the amount of repeated work. A means to assess the behaviour and characteristics of solutions early – preferably at design-time - is to apply simulation.

In this research, the foundation for seeking means to improve the efficiency of automation and control application development is Model-Driven Engineering (MDE). Related supporting techniques that are utilized in the research include UML, application domain specific profiles, especially UML Automation Profile (UML AP) as well as model transformations for partially automating the processing of models. Original introduction to UML AP is presented in (Ritala and Kuikka, 2007).

The idea of MDE and related approaches, e.g. MDA (Model-Driven Architecture) of OMG (Object Management Group, 2008), is the utilization of models during development of systems instead of, for example, documents. In MDE, models conforming to formally specified modelling languages (e.g. UML) can be processed with model transformations to create new or revise existing models and views. Automated model checks may reveal problems and inconsistencies in models and between modelled phase products. Model transformations can be used to automate importing information to models from models of preceding development phases and tools. Design models can be used for generating code or creating analysis models to be studied with domain specific, proven tools. Lastly, as have been already shown within the doctoral research, model transformations can be used to automate generation of simulation models that enable assessing and comparing the developed solutions at design-time.

In addition to basic automation and control, a special application area in the research is software development for safety systems. Safety-related systems require consideration of sophisticated standards including their requirements for development and documentation deliverables. As a consequence, if the focus of development work is shifted from documents to models according to the principles of MDE, models should be tailored to produce at least part of the required safety documentation. In addition to basic control, we regard simulations useful also in the development of safety systems although it is acknowledged that simulations are only a part of the solution. Like use of testing techniques, simulations cannot guarantee formal correctness of applications. That is why all the documentation required by safety standards must still be produced to be used in certification.

The rest of this paper is organized as follows. Section 2 presents and discusses the current state of the art related to the research area. In section 3, a more detailed presentation of the research problems and objectives will be provided. Section 4 focuses on the research methodology and the approaches to assess research results that are used in the doctoral research. Lastly, before conclusions, the current stage of the research and the results still expected to be achieved will be presented in section 5.

2 STATE OF THE ART

Utilization of models and model-based techniques for automation and control application development has been studied and proposed by several researchers. Most of the approaches have been targeted for IEC 61131-3 or IEC 61499 based development. IEC 61131-3 is currently also industrially widely used language - in addition to proprietary DCS systems and languages. Related to IEC 61131-3, such work has been presented in, for example, (Vogel-Heuser et al., 2005). Related to IEC 61499, which has drawn extensive research attention, such work has been carried out, for example, in MEDEIA project, by Tranoris and Thramboulidis and by Dubinin et al.

Vogel-Heuser et al. (2005), firstly, present an approach and mappings for generating IEC 61131-3 applications from UML models. The target code is in the approach in the forms of structured text and sequential function charts and can be imported to IEC 61131-3 tools using text import. The modelling approach utilizes class diagrams for presenting code structure and state charts for behaviour.

The MEDEIA (project) approach, is presented in (Strasser et al. 2008). The approach is based on Automation Components (AC) that are combinations of embedded hardware and software including integrated simulation, verification and diagnostics services. ACs can be deployed to hardware with code generation. The simulation approach of the project is discussed in (Hegny et al., 2010)

Dubinin et al. (2005) aim to support the development of function block (FB) applications conforming to IEC 61499 with an UML-FB language. The language is an extension of UML and based on its stereotype mechanism. The approach utilizes UML class (classes corresponding to IEC 61499 FBs), sequence, cooperation as well as state diagrams. Introduction to the code generation in the paper mainly focuses on class diagrams.

Thramboulidis and Tranoris have studied and developed a tool (Thramboulidis and Tranoris, 2004) and an engineering process (Tranoris and Thramboulidis, 2006) for distributed control applications using UML to present requirements and design before final implementations. In their development process, plain UML is used for describing all the features of function blocks and applications consisting of them. The process also utilizes model transformations for moving from analysis models to design diagrams and models.

Integrating *simulations* to MDE of control software is not generally a new idea. Embedded system and automotive control system industries, for instance, are in this respect ahead of industrial control domain. They have already benefitted from Model-in-the-Loop (MiL), Software-in-the-Loop (SiL), Processor-in-the-Loop (PiL), and Hardware-in-the-Loop (HiL) approaches. These general simulation approaches differ in terms of control system configuration used in the simulations. For example, in MiL only a model of the control system is used to control a process simulation whereas SiL utilizes generated software and HiL generated software with entire control system hardware. In the embedded and automotive system industries the HiL approach has been used in (Gietelink et al., 2009), SiL in (Canale et al., 2010) and MiL in (Plummer, 2006), to mention work of just a few researchers.

In the domain of industrial control, integrating simulations to MDE approaches has not been one of the first goals. However, such work has been presented at least in (Yang and Vyatkin, 2012) and (Hegny et al., 2010). In addition, (Ferrarini and Dede, 2010) presents a co-simulation approach for testing already implemented control application parts while simulating the rest. The approach is based on co-simulation (co-operative simulation) but does not necessarily restrict the development process to have MDE characteristics.

Yang and Vyatkin (2012) do utilize MDE techniques and model transformations. The purpose of their approach is to create IEC 61499 based FB models from Simulink models of the processes to be controlled in order to enable closed-loop simulations of the controlled processes with control application models. In the approach, the control applications are developed with IEC 61499.

Hegny et al. (2010), similarly, create IEC 61499 plant models to be integrated with control application models. However, their approach either transforms the plant models from MEDEIA (project) specific plant models that conform to a project specific timed state chart metamodel or uses external behaviour descriptions, i.e. external simulation tools. The approach to transform control application models to models of the processes to be controlled is - according to the knowledge of the authors - utilized only in our work, in the industrial automation and control domain.

Like integrating simulations to MDE in the domain, developing MDE support for *safety* systems and applications has not yet drawn extensive research attention. Perhaps most complete work in the area has been carried out in DECOS project (Huber and Obermeisser, 2007). In DECOS, IEC 61508 is used as a goal similarly to the work of the author. However, the modelling languages differ significantly. The DECOS approach utilizes e.g. Simulink and SCADE whereas the purpose of the author is to use UML and its extension to the automation domain: UML AP. In addition, as a difference to the work of the author, in the DECOS approach also hardware aspects are considered.

In addition, there are several approaches in various domains to integrate safety information to UML and SysML based modelling. (Guillerm et al., 2010) discusses the use of SysML to address requirements definition, traceability as well as verification and validation in system engineering process, which are all of importance in safety system development. The paper proposes the use of UML and SysML and extends the languages with stereotypes related to documenting risks of the underlying system.

In (Biehl et al., 2010), an attempt is made to integrate safety analysis to model-based development for automation industry. The paper presents an automated transformation from EAST-ADL2 to HiP-HOPS in order to automate performing safety-analysis on refined models. As such, the approach mainly focuses in automating safety analysis after changes to design models.

The UML Profile for developing Airworthiness-Compliant Safety Critical Software is presented in (Zoughbi et al., 2007). The work intends to extract the key safety-related concepts from RTCA DO-178B standard into a UML profile and to use them to facilitate the communication between different stakeholders in software development. One of the purposes of the profile is to make requirements more understandable to all stakeholders, which is a similar goal than those of the author. However, instead of focusing to requirements specification and occurrences of hazards, the profile aims to aid communication by enabling the use of concepts of the standard in models to characterize modelled parts of the systems.

Lastly, the safety analysis profile (Douglass, 2009) has been developed to support development of safety critical software so that safety professionals would not have to rely on disparate tools to capture requirements and design. According to the paper, UML can facilitate the development of safety critical systems in several ways. The means to facilitate the development work include: providing design clarity, modelling of architectural and low-level redundancy, creating safety-relevant views on design and requirements as well as aiding safety analysis. With the profile presented by Douglass, safety analysis can be accomplished in models and interconnected to requirements.

3 RESEARCH PROBLEMS AND OBJECTIVES

Currently, *modelling and simulation* is in automation and control domain used especially for describing and studying process dynamics and control algorithms. Models, however, are mostly based on mathematics (e.g. Matlab based Simulink) and do not suit well for software development. The modelling concepts of e.g. Simulink differ from those of software development and programming languages. As a consequence, although such models could be used for code generation, the resulting code would be difficult to maintain and integrate to manually developed parts of control applications. Another aspect is that the models should also support, for example, importing information from preceding development phases and producing documentation. And for such purposes, extendable modelling languages such as UML form a more natural basis.

In the doctoral research, the focus is in modelling and MDE techniques for software development in the automation and control domain. The basic research question to be answered is *whether or not automation and control application development can benefit from MDE and the techniques enabled by it,* e.g. integrated simulation. The answer will be based on answers of smaller research questions. It also produces research prototypes for evaluating the techniques and solutions. A more detailed, presentation of the smaller research questions will be provided below.

Automation applications are already often composed of parameterizable, platform specific components (e.g. function blocks) that are well-tested functional entities. This facilitates the development and improves the re-use of working, well-tested solutions but makes it difficult to re-use design work on other platforms. In this research, the purpose is to seek an approach for modelling of automation and control applications. The modelling approach should enable fluent re-use of parameterizable implementation blocks and use of design patterns for facilitating design and for documentation purposes. Additionally, the modelling approach should enable the use of model transformations that are necessary in applying MDE in all development phases.

Related to simulations, the research is aimed to answer the questions: how simulation integration can be supported in MDE environments, which general simulation approach (e.g. model-in-the-loop and co-simulation) should be used and what are the possible benefits compared to the present simulation support available in the industry. It is acknowledged that simulations are already supported by commercial PLC and DCS platform. However, the research asserts that it would be possible to obtain additional benefits from integrating simulations to the development process.

Related to the development of safety-related systems, the modelling concepts must take into account the requirements of safety standards. In this research, the essential functional safety standard to be used is IEC 61508 (IEC, 2010) which is also referenced by other, application domain specific standards in both process industry and machinery. The industrially significant question that needs to be addressed before applying MDE techniques to industrial safety system development is: how the strict requirements and documentation needs of safety standards could be fulfilled with MDE techniques. In the doctoral research, the aim is to prove possible to integrate risk and hazards analysis to the MDE process and to produce certification friendly information from models to support traceability, correctness and completeness within models. Traceability is an important characteristic for also basic control applications to facilitate inspection and review processes of the applications.

In addition to answering these research questions, an objective of the research is to produce prototypical tools for demonstrating the techniques and modelling concepts and to assess their possible benefits. The tools are aimed to be developed using open-source modelling and model transformation tools of the Eclipse platform as a basis. Such tools, e.g. Topcased and SmartQVT are based on standards of OMG including UML, MOF and QVT, and can be extended with new modelling concepts and to implement required model transformations.

As a whole, the research aims to complete the AUKOTON design process (Hästbacka et al., 2011) with simulation capabilities as well as support for re-use and processing safety information. The improved design process is illustrated in figure 1.

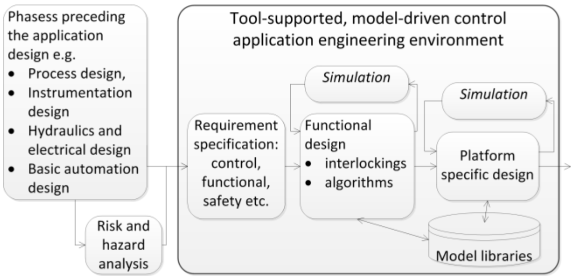

Figure 1: The results of the research will implement a simulation assisted, MDE process for automation and control applications based on open-source tools.

4 RESEARCH METHODOLOGY

The research included in the doctoral studies conforms to the constructive approach of design science. The application domain is industrial automation and control application development. In general, the research needs that the research aims to fulfil are based on interviews of industrial professionals and internationally reported challenges. In the Automation Software Engineering (ASE) research group to which the author belongs, the industrial partners include both industrial control system vendors as well as companies developing automation and control solutions for their machine products.

The research, as constructive design also in general, aims to meet the research problems and needs, with technical solutions that are defined and implemented as research prototypes. Solutions and their implementing prototypes are evaluated and further developed with *case studies* and industrial *assessment events*. Both methods enable collecting material and evidence on the possible benefits and disadvantages of the solutions. The research questions are then answered based on the results of the case studies as well as collected and analysed material, observations and feedback from the industrial assessments. In addition to gathering industrial feedback, assessments events also serve the purpose of transferring and demonstrating research results and new technologies to the industrial partners.

In the research of the ASE research group, the assessment events serve mainly collecting *qualitative* material on the suitability of the solutions and their implementations. For example, for (Vepsäläinen et al., 2010) we arranged a 1-day event with our research partners in university premises. In the event, 8 industrial professionals from 4 different companies developed a control application for a small-scale process industry process using research prototype tools developed by us during AUKOTON project. During the event, the professionals were observed and field notes were collected from technical and other kind of problems and challenges that they encountered in their work. After completing their design tasks the professionals, who were at that time already familiar with the techniques, were interviewed. The purpose of the interviews was to assess the suitability of the techniques to industrial use in the companies and the possible advantages and disadvantages in comparison to techniques used in the companies at the time of the interviews.

An example of assessing developed techniques, concepts and prototype implementations with case studies is presented in (Vepsäläinen and Kuikka, 2013a). The contribution of the paper is two-fold. The first one is a conceptual comparison of approaches to enable design-time simulations within MDE taking into account, for example, the amount of required model transformations and configuration work to execute simulation cases. The second contribution is an analysis of perceived benefits and disadvantages of the approach encountered with three published simulation experiments.

The benefits of using two kinds of methods to assess the research and developed solutions are significant. Case studies, firstly, can be targeted to academically interesting problems and to assess the research against the industrial state of the art and work of other researchers. Assessment events, on the other hand, are more related to industrial practice and enable both collecting industrial feedback and keeping the research focused to problems with practical relevance. With both assessment methods, the research has focused on qualitative material. Consequently, for example, possible net effect of the techniques to the amount of development work has not yet been estimated. However, such studies could be also arranged when the prototype tools are at a level on which the results would not be biased by usability challenges of prototype tools in comparison to commercial tools.

5 CURRENT STAGE OF THE RESEARCH AND EXPECTED RESULTS

Currently, the doctoral research is at a stage in which all the research questions have been addressed to some extent although most of them still require work and supporting publications. The modelling approach including use of library implementation blocks in code generation has been addressed in (Hästbacka et al., 2011). The results of industrial assessment of the approach, on the other hand, have been presented in (Vepsäläinen et al., 2010).

Use of re-usable simulation blocks and the general approach to transform functional models to a simulateable ModelicaML (Modelica Modeling Language) (Schamai, 2009) form have been addressed in (Vepsäläinen and Kuikka, 2013b). The design-time simulation of control applications is intended to cover all the common aspects of basic control systems including sequential control, interlockings as well as binary and feedback control. Of these aspects, interlockings as well as feedback control are covered in (Vepsäläinen and Kuikka, 2013b). Sequential control and binary control are also already supported by the approach; however, a paper addressing these issues is still to be published. A brief assessment of the research and perceived benefits is presented in (Vepsäläinen and Kuikka, 2013a). Additionally, the latter paper presents a conceptual comparison of possible simulation approaches for design-time simulation of models within MDE. The purpose of the comparison is to draw conclusions on which approach should be utilized in the domain.

Integration of safety, risk and hazard information to the MDE process and supporting traceability, correctness and completeness have been addressed in (Vepsäläinen and Kuikka, 2011). The article presents a set of meta-model additions to the modelling profile, see (Hästbacka et al., 2011) for introduction to the joint-work on UML AP, to enable modelling of risks and hazards using Fault Tree (FT) notation and logic diagrams for presenting detailed requirements

In (Vepsäläinen et al., 2012) the author has presented tool support for documenting architectures of safety systems and performing simple checks of consistency against safety standards. The work presents a set of concepts developed for describing safety system architectures including their requirements and components as well as used design patterns and made decisions. However, the work is based on an application lifecycle management platform (Polarion ALM) instead of UML profile assisted modelling. Part of porting this work to UML-based modelling, in addition to re-using conceptual design solutions – design patterns – and using them to aid documentation, is still under research and to be published. On the other hand, the research path related to safety applications lacks assessment of results, which will follow after finishing the basic work.

At this stage of the Ph.D. research, it is expected that the general question, whether or not automation and control application development can benefit from MDE and the techniques enabled by it, can be given a positive answer. According to the results of assessing the industrial applicability of the general approach (Vepsäläinen et al., 2010) the developed MDE and modelling process could be used for developing industrial DCS-based applications. Practically this could mean targeting platform specific work for a chosen platform and using code generation that would be required for each platform. A more open question is support for general solutions in form of automation domain specific design patterns, which is still to be addressed.

Perhaps in most complete stage are the research questions related to integrating design-time simulations to MDE. The approach to use model-transformations for creating closed-loop simulation models of controlled systems has been shown to enable simulation of all the common aspects of basic control systems. The approach is capable of both using libraries of existing blocks (corresponding to library implementations) and generating new simulation blocks based on logic diagram and automation sequence diagram presentations of application specific automation functions. (Vepsäläinen and Kuikka, 2013b)

Related to different simulation approaches, we have argued that simulations in MDE in the domain should focus on model-in-the-loop simulations. The most important reason for this recommendation is that industrial control system platforms already support later simulation approaches e.g. software-in-the-loop and hardware-in-the-loop simulations. Consequently, in order to obtain additional benefits from enabling simulations within the development process, simulations should be possible to perform earlier. Practically, this would mean MiL simulations. On the other hand, transforming control application models instead of plant models or using co-simulation may lead to fewer difficulties and less additional work with simulation cases, as presented in (Vepsäläinen and Kuikka, 2013a).

Related to perceived benefits of applying early MiL simulations, they have been found useful in comparing and prototyping alternative control and interlocking approaches, testing sequences as well as finding acceptable controller tunings. Simulations have helped finding missing implementations and requirements as well as testing exceptions that could be dangerous to test with the actual physical processes. Additionally, the simulation approach has been shown to scale to control applications of different industries and sizes. (Vepsäläinen and Kuikka, 2013a)

Integrating safety information to the development process is the least complete part of the research at present. However, some expected results can be still pointed out based on publications and on-going work of porting the results of (Vepsäläinen et al., 2012) to the modelling environment. Currently, the support for *traceability* illustrated in (Vepsäläinen et al., 2012) has already been implemented to complement the work presented in (Vepsäläinen and Kuikka, 2011). The current focus is on design patterns and partially automating their use.

In addition to facilitating development work, design patterns and pattern instance markings are expected to serve documentation purposes. Pattern instance markings enable pointing out where in design recommended solutions and patterns have been used. On the other hand, (some) design patterns could be given Safety Integrity Level (SIL) recommendations based on IEC 61508 (IEC, 2010). Such recommendations could then be used to implement simple checks of consistency against safety standards and modelled safety integrity level requirements of the applications being developed.

6 SUMMARY AND CONCLUSIONS

This paper is intended to present the research problems, methodologies, objectives, status and expected results of doctoral research of the author. The research included in the doctoral studies conforms to the constructive approach of design science within the application domain of industrial automation and control. The general research question to be answered is whether or not automation and control application development can benefit from MDE and the techniques enabled by it. The research question is to be answered based on answers to smaller research questions. These questions are related to modelling of automation and control applications, ability to integrate and gain benefits from integrating simulations to the development process and ability to include safety documentation in models.

Currently, the research is at a stage in which all the research questions have been addressed to some extent although most of them still require work and supporting publications. Generally, it is expected that the main research question can be given a positive answer. This expectation is based on the promising assessment results of the general development approach and the ability to generate closed-loop simulation models of controlled systems. The approach creates and integrates control system parts to existing simulation models of the process to be controlled based on UML AP software models. In our experiments, the approach has been found useful in many ways and it has scaled to applications of both machinery and process industry.

Open questions of the research are related to integrating safety documentation to models and automating utilization of design patterns. This work path that aims to utilize design patterns for both re-use and documentation purposes is still partially under development, to be published and to be assessed based on case studies or industrial assessments.

REFERENCES

Biehl, M., DeJiu, C. and Törngren, M. 2010. Integrating Safety Analysis into the Model-Based Development Toolchain of Automotive Embedded Systems. In: LCTES 2010, pp 125-132, New York, NY, USA, 2010. ACM.

Canale, M., Fagiano, L., Razza, V. 2010. Approximate NMPC for Vehicle Stability: Design, Implementation and SIL Testing, Control Engineering Practice, Volume 18, Issue 6, June 2010, Pages 630-639, ISSN 0967-0661.

Douglass, B. 2009. Analyze System Safety Using UML within the Telelogic Rhapsody Environment, White paper, IBM Corporation.

Dubinin, V., Vyatkin, V., Pfeiffer T. 2005. Engineering of Validatable Automation Systems Based on an Extension of UML Combined with Function Blocks of IEC 61499. Proceedings of the IEEE International Conference on Robotics and Automation, 2005 (ICRA 2005), pp. 3996–4001.

Ferrarini, L. and Dede, A. 2010. A Model-Based Approach for Mixed Hardware in the Loop Simulation of Manufacturing Systems", 10th IFAC Workshop on Intelligent Manufacturing Systems, 41–46, 2010.

Gietelink, O. J., Ploeg, J., De Schutter, B., Verhaegen, M. 2009. Development of a Driver Information and Warning System with Vehicle Hardware-in-the-Loop Simulations, Mechatronics, Volume 19, Issue 7, October 2009, Pages 1091-1104, ISSN 0957-4158.

Guillerm, R., Demmou, H. and Sadou, N. 2010. Information Model for Model Driven Safety Requirements Management of Complex Systems. In: First International Conference on Complex System Design and Management, Paris, France, October 27-29, 2010.

Hegny, I., Wenger, M. and Zoitl, A. 2010. IEC 61499 Based Simulation Framework for Model-Driven Production Systems Development, Emerging Technologies and Factory Automation, IEEE Conference on, 1–8, 2010.

Huber, B. and Obermaisser, R. 2007. Model-Based Development of Integrated Computer Systems: Modeling the Execution Platform. Proceedings of the 5th International Workshop on Intelligent Solutions in Embedded Systems (WISES'07), IEEE Computer Society Press: Washington, DC, USA, 2007; 151–164.

Hästbacka, D., Vepsäläinen, T., Kuikka, S. 2011. Model-driven Development of Industrial Process Control Applications, The Journal of Systems and Software, 84 (7), pp. 1100 – 1113.

IEC. 2010. IEC 61508: Functional Safety of Electrical/Electronic/Programmable Electronic Safety Related Systems. parts 1-7. 2010.

Object Management Group. 2008. Technical Guide to Model Driven Architecture: The MDA Guide. Version 1.0.1 (formal/2008-04-03) Edition.

Plummer, A. 2006. Model-in-the-Loop Testing, Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, Vol. 220, No. 3, 183–199, 2006.

Ritala, T., Kuikka, S. 2007. UML Automation Profile: Enhancing the Efficiency of Software Development in the Automation Industry, The Proceedings of the 5th IEEE International Conference on Industrial Informatics (INDIN 2007), Vienna, Austria, July 23-27, 2007, pp. 885-890.

Schamai, W. 2009. Modelica Modeling Language (ModelicaML) – a UML Profile for Modelica, Technical Report 2009:5, EADS IW, Germany, Linköping University, Institute of Technology.

Strasser, T., Sunder, C. and Valentini, A. 2008. Model-Driven Embedded Systems Design Environment for the Iindustrial Automation Sector. INDIN 2008 the 6th IEEE International Conference on Industrial Informatics, July 13-16, 2008, Daejeon, Korea.

Thramboulidis, K., Tranoris, C. 2004. Developing a Case Tool for Distributed Control Applications. The International Journal of Advanced Manufacturing Technology 24 (1), 24–31.

Tranoris, C., Thramboulidis, K., 2006. A Tool Supported Engineering Process for Developing Control Applications. Computers in Industry 57 (5), 462–472.

Vepsäläinen, T., Sierla, S., Peltola, J and Kuikka, S., 2010. Assessing the Industrial Applicability and Adoption Potential of the AUKOTON Model Driven Control Application Engineering Approach, Proceedings of International Conference on Industrial Informatics. Osaka, Japan, July 13-16, 2010.

Vepsäläinen, T. and Kuikka, S. 2011. Towards Model-Based Development of Safety-Related Control Applications. In 16th IEEE International Conference on Emerging Technologies and Factory Automation. Toulouse, France, September 5-9, 2011.

Vepsäläinen, T., Kuikka, S., Eloranta, V.-P. 2012. Software Architecture Knowledge Management for Safety Systems. In 17th IEEE International Conference on Emerging Technologies and Factory Automation. Krakow, Poland, September 17-21, 2012.

Vepsäläinen, T. and Kuikka, S. 2013a. Benefit From Simulating Early in MDE of Industrial Control. In 18th IEEE International Conference on Emerging Technologies and Factory Automation. Cagliari, Italy, September 10-13, 2013a. (Accepted)

Vepsäläinen, T., Kuikka, S. 2013b. Simulation-Based Development of Safety Related Interlocks. Simulation and Modeling Methodologies, Technologies and Applications, Springer Berlin Heidelberg pp. 165-182 2013b.

Vogel-Heuser, B., Witsch. D., Katzke, U. 2005. Automatic Code Generation from a UML Model to IEC 61131-3 and System Configuration Tools. International Conference on Control and Automation, 2005 (ICCA '05), vol. 2, June 2005, pp. 1034–1039.

Yang, C. H. and Vyatkin, V. 2012. Transformation of Simulink Models to IEC 61499 Function Blocks for Verification of Distributed Control Systems. Control Engineering Practice, Vol. 20 No. 12, 1259–1269, 2012.

Zoughbi, G., Briand, L., Labiche, Y. 2007. A UML Profile for Developing Airworthiness-Compliant (RTCA DO-78B), Safety-Critical Software. In: MODELS 2007. LNCS, vol. 4735, pp. 574–588. Springer, Heidelberg 2007.