IEC 61499 based Simulation Framework for Model-Driven Production Systems Development

Ingo Hegny, Monika Wenger, and Alois Zoitl
Vienna University of Technology, Automation and Control Institute
Gusshausstrasse 27-29 / E376
1040 Wien, Austria
hegny@acin.tuwien.ac.at

Abstract

In the design of plants and machines the complexity is permanently increasing. Model-driven design methodologies in the field of Industrial Automation and Control Systems, as proposed by the MEDEIA project, help to cope with current requirements like limited budget and reduced ramp-up time. Concurrent development in all involved domains is facilitated by the introduction of simulation in the development process. We propose the use of an IEC 61499 industrial automation runtime environment to simulate plants’ behavior. The concept for the simulation framework is based on the coupling of the control application and the plant simulation application via Instrumentation and Control points. The introduction of physical links in the plant simulation application enhances the behavior model of the plants, thus increasing the quality of the obtained simulation results.

1. Introduction

Today plants in the manufacturing and process technology domains face many challenges. Produced goods have to fulfill customers’ requirements in an efficient way. The Manufuture Platform identified agile production systems, which are able to adapt to future requirements, as a possibility to cope with these challenges [3]. Adaptive production systems rely on flexible and changeable mechanical plant structures. However, changes to the mechanical setup also require changes of the electrical setup as well as the control software, as stated in [8].

The international standard IEC 61131-3 is currently the basis for most control systems in industrial automation. Due to the device centric and monolithic engineering approach they are not well apt for regular changes of the executed control applications. Furthermore multiple engineering tools are required for adapting the control application, since many of today’s production plants are built with devices from multiple vendors. The increasing flexibility of the plants is also increasing the design complexity. Therefore the effort for engineering and commissioning is also growing. In the industrial sector, reusability of software across platforms is nearly inexistent. Each plant or system is built to order and can be treated like a prototype. Currently a huge part of the investments is spent for automation and control engineering. Ref. [11] shows that in the automotive sector the software related tasks (i.e., planning, programming, ramp-up) currently make up 55% of the total costs and are estimated to rise even higher. Furthermore experts from multiple domains (e.g., electrical, mechanical, control engineers) are required for the specification and implementation of the plant.

Friction losses caused by unclear or unspecified interfaces impose a high risk to keep schedules and budgets. Diverse methodologies and implementation technologies for the engineering of industrial automation and control systems are further obstacles for an efficient engineering. Especially the combination of machines and embedded devices from multiple vendors to build an automation solution is a hard task. Ref. [5] states that the integration of the development team from the beginning of the specification phase imposes a higher effort at the beginning, but bears the chance to reduce the overall project time frame and to implement a solution that is closer to a global optimum. The introduction of a meta-design architecture, as proposed by the MEDEIA project [16], helps to overcome the limitations of the high diversity of system programming and modeling approaches in the field of industrial automation.

Currently simulation is rarely used during the whole development and engineering cycle of production plants. An early control application development could be facilitated by the availability of a plant simulation framework. In the development of the model-driven MEDEIA approach the aspects of simulation have been considered. The specification of the interface between the plant and the control application has to be provided in a formal way. The development and validation of control applications for production systems are facilitated early in the engineering cycle, reducing ramp-up-time and downtimes for reconfiguration tasks. Control hardware and applica-
tions build up a time-discrete system. Therefore discrete event simulation is sufficient, which is provided by the MEDEIA simulation concept, based on IEC 61499.

This paper is structured as follows. The next section introduces the MEDEIA project and the model-driven development approach. Shortcomings of current simulation systems and requirements for the application of simulation in the development of production systems are identified in Section 3. Section 4 provides an insight on the proposed simulation concept, including IEC 61499 as basis for the simulation execution, detailing the involved models in the model-driven approach and the applied transformations. A prototypical implementation for a component of our lab-equipment is shown in Section 5. Finally the paper is concluded with a summary.

2. The MEDEIA Project

The MEDEIA project is a collaborative project in the 7th framework programme. Its main objective is the development of an integrative, model-driven engineering approach for the industrial automation sector [2]. The project team includes industrial partners from three diverse domains of the sector: manufacturing, robotics, and power generation. In all these domains the model-driven engineering approach, collaborative engineering and the possible reuse of components and engineering know-how is demanded. Information, including requirements and specifications, models, and existing components should be reusable across company borders.

2.1. Automation Component

The developed methodology is based on Automation Components (AC), which encapsulate embedded hardware and software. Even more a model of its functionality and interfaces, needed for the interaction with other ACs and composition, is held by the AC. A component oriented system design is applied to the functional structure of ACs. Current state-of-the-art design methodologies in the field of industrial automation are not compliant with the functional approach. Control system structures are chosen on the basis of economical reasons, implied by control system vendors.

Hierarchical aggregation and component based design are adopted for ACs. These principles are well known in mechanical design and process technology. Their aptitude for control application design was investigated and shown in [12]. A behavioral specification of the interfaces is an enabler for a hierarchical aggregation.

2.2. Integrative Engineering Approach

Domain experts (e.g., mechanical, process, control engineers) work on the same model. Starting from the first, rough specification of the plant architecture and ACs, a continuous refinement of the plant structure is performed. New, additional requirements are added and ACs are selected. The required functionality can either be fulfilled by a pre-existing AC, which will be reused, or a new AC is implemented during the engineering phase.

For the acceptance of the proposed modeling methodology it has to be easily understandable for the involved domain experts. They shall continue to use their domain specific engineering and development tools. The information, which is provided by these tools, will be extracted and transformed into the common Automation Component Implementation Model (ACIM) depicted in Fig. 1. The ACIM comprises the platform-independent Automation Component Model (including functional, behavioral specification) and the Execution System Model (providing information on the control platform).

Based on the specifications needed for clear, well-defined interfaces between the involved engineering domains, additional services can be added with low additional effort:

- **Integrated diagnostics:** The diagnostic model is described during the specification phase of the plant. Information on possible, diagnosable faults is provided. Diagnostic algorithms, which are based on these data, are included in the control application. Thus a fast and reliable identification of failure conditions and their causes is facilitated during operation. Faster recovery and increased availability of the heterogeneous plants are the expected results.

- **Verification:** Interface and behavioral specifications of the ACs are used for verification. The early detection of problems and errors helps to reduce the development time frame and the economic risk. Especially the aggregation of ACs can cause unforeseen and unwanted effects, which may be detected by verification.

- **Simulation:** The validation of the specified control behavior is an important step in the development cycle. The specification of the control behavior is the source for the automatic generation of control applications that are platform specific. Inconsistencies and missing requirements can be detected early, reducing the risks and the effort to fix these flaws and errors. All data that is necessary for simulation is provided very early in the design process. Thus the parallel development of the mechanical, electrical and control parts of a plant is supported.
3. Shortcomings and Requirements for the Simulation of Production Systems

Simulation is a powerful tool, which is used in multiple engineering fields to validate and compare proposed component or system designs. However, simulation is still used only to a limited extent in industrial automation. This is caused by various identified shortcomings of the available simulation tools and environments [4]:

**Limited spectrum:** Commercially available simulation tools are often tailored for the simulation of specific problems. However, industrial automation, even the automation of production systems, is a broad, integrative field dealing with multiple types of problems and aspects, for example assembly, machining, logistics, and process control.

**Specialized models and modeling techniques:** Depending on the type of simulation (e.g., continuous time vs. discrete event) the models and the modeling techniques differ. But even more within these classes the models differ to a large extent. The data, which is required for simulation, is determined by generalized, tool independent concepts, like world views [7] as well as by tool specific definitions and rules. Therefore the interoperability and model transfer between and/or the concurrent use of multiple simulation frameworks and tools may be necessary for the simulation of aggregated components.

**Specialized expert know-how:** Simulation is an emerging field in industrial automation and control. However, knowledge of control engineers on modeling and simulation is additional know-how that is often limited (e.g., specific tools, depth of knowledge). A composed plant could be validated by coupling multiple simulation tools. Co-simulation (i.e., multiple simulation tools, each using its own distinct model) is applicable, where it is neither possible to maintain a single model nor to use one simulation engine on multiple models [9]. However, setting up a co-simulation environment with recent tools requires expert knowledge on the coupled simulation tools and on the coupling itself [14].

**Maintenance:** Apart from specially trained personnel, having knowledge on multiple simulation tools, system integrators also need the technical infrastructure for these tools. This includes costly licences and maintenance of the simulation frameworks. Thus the simulation models, provided by the component vendors can be reused. But maintenance is also applicable to the provided simulation models, as newer versions of commercial simulation tools may also introduce changes to their model-structures.

3.1. Identified Requirements for Simulation in MEDEIA

The envisaged field of application for the simulation framework is the development of industrial automation systems. Although three diverse domains of the industrial project partners are analyzed and taken into consideration, the identified requirements may and will not fit to all possible applications in industrial automation.

**Targeted user group:** Simulation shall facilitate the development of control applications at an early stage, where the plant is not physically built up. The target user group are control engineers. Control engineers usually do not have thorough knowledge on modeling and usage of multiple simulation tools. Models, which have to be provided and interpreted by the control engineers, should follow familiar methodologies as used in their domain.

**Discrete event simulation:** Modern control hardware is based on digital microprocessors. Sensor signals are sampled and actuators are set at discrete time intervals. Therefore discrete event simulation is sufficient. In discrete event simulation the focus can be laid on different entities in the simulation model. Three world views are prevalent in the discrete event system simulation: event scheduling, activity scanning and process interaction [7]. Event scheduling and activity scanning seem to be more apt to simulate plant behavior.

**Transition from simulated to operational plant:** Four scenarios for the validation of plants have been identified [6]:

- offline simulation: the control application as well as the plant are simulated.
- “Reality in the Loop”: the real plant is connected to the simulated control system.
- “Soft-Commissioning”: the implemented control system is connected to the plant simulation, and
- Testing: the real plant is controlled by the implemented control system.

In the MEDEIA approach control engineers shall be enabled to validate the control systems/applications. Therefore the scenarios, which include operational control systems, are deemed important. Furthermore hybrid simulation is a promising scenario. Parts of the plant are still simulated, other parts are already operational. Thus the commissioning phase and the validation of physical reconfigurations of the plant can be facilitated. The classification of the required simulation scenarios is given in Fig. 2.

**Component based modeling:** Due to the component based structuring and specification of MEDEIA ACs also the simulation models of the plants have to follow this

![Figure 2. Classification of required simulation scenarios](image)
principle. Furthermore component based modeling is important for hybrid simulation. It is necessary to change the simulation model, replacing simulated components by interfaces to operational components. Such changes are harder to do on monolithic models than on component based models.

Physical interactions: Mechanical components, which build up a plant, have physical interactions that are not represented in the developed control applications. For example, the generation of sensor signals for incoming material, these interactions are relevant. For the MEDEIA target domains material flow and kinematic chains are important physical interactions that have to be modeled.

Framework availability: The reuse and the distribution of ACs across company borders is an important step in reducing the engineering effort. Since all simulation frameworks that might be used shall be available to all involved parties, the number of required simulation tools has to be low. Furthermore licence costs, employee training and maintenance are important factors, especially for SMEs, which are often in the role of system integrators.

4. MEDEIA Simulation Concept

Simulation is an integrated functionality within the MEDEIA project. There exist contact points with other functionalities, which also have requirements that influenced the commonly used MEDEIA models and the design flow. The central model for plant behavior simulation, the MEDEIA Plant Model (PM), is also used for the specification and generation of the integrated diagnostics. Therefore, a general model has been chosen for behavior specification, which is powerful enough to fulfill all requirements: timed state charts. By model transformation the PM is provided in a format that can be executed by the simulation execution tool.

4.1. Simulation Execution

Control Engineers have been identified to be the main target user group for the simulation functionality in MEDEIA. Therefore the executable simulation models shall be graspable to them. IEC 61499 provides concepts, which make it capable to be used as simulation execution framework [10]:

- application centric engineering,
- distributed execution,
- event based execution,
- components,
- aggregation, and
- adapters.

As basis for the simulation framework in MEDEIA FORTE and the integrated development environment 4DIAC-IDE have been chosen. Both, the runtime environment and the IDE, are provided as open source software by the 4DIAC project [1].

4.2. MEDEIA Plant Model

A PlantComponent comprises at least a PlantBehaviourModel and a PlantComponentInterface. Following the hierarchical aggregation approach, it can also include other PlantComponents as subComponents and functionalities or functionalities provided by external entities. External simulation tools can be integrated for a coupled simulation scenario, if they provide an appropriate API or communication interfaces. Discrete event simulation is enabled by the event based execution, which was introduced by the IEC 61499 standard into the field of industrial automation. Event based execution is provided by all IEC 61499 compliant runtime environments. However, there exist differences in the execution order and event scheduling [21]. Real-time execution, which facilitates hybrid simulation, is supported by the scheduling mechanism implemented in the 4DIAC runtime environment (FORTE) [20]. Hybrid simulation and the transition from simulation to operation is also facilitated by application centric engineering and distributed execution. Aggregation with Composite Function Blocks (CFB) and SubApplications facilitate structuring the simulation application. Furthermore the concept of Adapters is provided, which helps to structure the interfaces of FBs by aggregating event and data connections.

As basis for the simulation framework in MEDEIA FORTE and the integrated development environment 4DIAC-IDE have been chosen. Both, the runtime environment and the IDE, are provided as open source software by the 4DIAC project [1].
states. All states can contain actions, which are executed on entry and on exit of the state. Furthermore outgoing transitions are included in the states. Transitions are used, and the target state is activated, when the TransitionCondition is fulfilled. These conditions include time based DelayCondition elements and value based PortCondition. The DelayCondition elements contain parameters to specify randomness of the delay, to allow realistic modeling of the plant timing. PortElements are referenced to model transitions due to external events and data changes, including the control behavior and other connected Plant-Component elements.

4.3 IEC 61499 Plant Simulation Model

For the execution of the simulation on the IEC 61499 runtime environment, the specifications of the MEDEIA PM have to be provided as IEC 61499 application. The hierarchical PlantComponent structure is implemented as nested Sub-Applications and CFBs. A single entity’s interface comprises event and data inputs and outputs as specified within the PlantPort elements and adapter interfaces (plugs and sockets), which represent the material flow and kinematic ports. Adapters have been chosen for three reasons: First they encapsulate multiple event and data connections (bidirectionally), which reduces the engineering effort. Second due to the reduction of connections also the graphical representation is less scattered, thus increasing the readability of the applications. Finally adapter interfaces have a pre-defined direction and allow only a 1:1 connection, which is equivalent to the modeled physical links.

The behavior of the plant component is encapsulated in a CFB. Dependent on the description, either timed state chart or external behavior description, the internal structure of the FB is implemented. If external simulation tools shall be used, interfaces to the external simulation tool have to be included in this CFB. Otherwise the specification from the TimedStateChart is used. The state chart without timing information is transformed to the Execution Control Chart (ECC) of a Basic FB (BFB). ECCs implement finite state machines based on Moore machines. These do not support on exit actions. Therefore additional states will be added to the ECC. The timed transitions will be realized by additional E_DELAY FBs, which are triggered by the BFB containing the state machine implementation. The other transitions as well as the actions will be natively implemented within the BFB.

The behavior of the ACs, which is of interest for control engineers, can be validated by coupling the IEC 61499 Plant Simulation Model with the implementation of the control behavior. Available transformation rules are utilized to generate the IEC 61499 implementation from the BM [15]. Within the resulting control application the PlantPort elements are transformed into specialized CFBs—Instrumentation & Control Points (I&C-points). These provide the link to the controlled plant during normal operation or couple the control application with
the Plant Simulation Model. Previous implementations showed that I&C-points are a feasible way to switch from simulation to operation and vice versa [10]. However the tight integration of simulated I&C-points into the control application has a drawback. Adapter interfaces, implementing the physical ports, alter the interface of the CFB representing the I&C-point. Thus control engineers might get confused by the scattered view. Therefore local communication has been chosen to couple the I&C-points in the control application with the plant simulation application. A clearer separation of concerns is the result. Still both applications are executed on the same control device, avoiding unwanted effects from network communication (e.g., jitter, lost packets).

4.4. Model Transformations

For an automatic generation of a simulation model for an Automation Component we need on the one hand a model, which describes the plant as well as its behavior and on the other hand a structure description of such a model. A model’s structure description is called meta-model. We also need such a meta-model for our IEC 61499 simulation target system.

The meta-model’s structure is implemented within every new model. Consequently a model is an instance of the designed meta-model. Fig. 3 as well as Fig. 4 are therefore meta-model examples, which are an important part for the generation of a simulation project. Within MEDEIA all meta-models including the PM and the TSCM are combined by different constraints into one AutomationProjectManagementModel. An instance of this meta-model therefore also contains the desired PM. Since we have one input model, which describes the plant as well as its behavior, and we want to generate a plant simulation model executable within IEC 61499 we have to build a proper model-to-model (M2M) transformation.

The Eclipse Modeling Project [17] provides the Xpand Generator Framework, which offers textual languages for checks, code generation, as well as model transformation. All these languages operate on the same models, meta-models and meta-meta-models [18] and are built up on a common expression language and type system, which provides a uniform abstraction layer over different meta-meta-models like EMF Ecore (Eclipse Modeling Framework), XML Schema and others. Since the Xpand Generator Framework is specialized on code generation based on EMF models and the core meta-model in EMF is Ecore (Meta-Object-Facility like) we also used this description method for the MEDEIA meta-models. IEC 61499 already offers an XML Schema Definition as description for its models, which we can use for an automatic generation of our plant simulation model.

Fig. 5 illustrates the general structure of the transformation process. Within this process an implementation of PM shown in Fig. 3 is used as input model. The transformation engine parses this model according to the defined transformation rules. In our case these rules correspond to the relationships between the MEDEIA PM and IEC 61499 explained in the previous subsection.

Every transformation rule is built by an adequate transformation language. We use Xtend (eXtension language) since the Xpand Generation Framework offers this language for M2M transformations and because of good experience within other projects [19]. For the generation of the desired plant simulation model the transformation engine builds the contents according to the meta-models as well as the transformation rules.

5. Prototypical Implementation

For a first demonstration of our simulation concept we chose a small part from the Sorting Machine of our lab. This Sorting Machine sorts different parts, which are transported on palettes by a conveyor belt. An incoming palette passes through an identification station, which scans the transported part by a proper identification system. Dependent on the identification result the part is taken off the palette by one of two Handling Units or proceeds on the conveyor belt. During the part manipulation the palette is stopped on the conveyor belt by a stopper, which releases the palette afterwards. The schematic view as well as the component tree of this application is shown in Fig. 6.

The prototypical implementation described within this section only covers the machine’s Handling Unit. Our Handling Units consist of two pneumatic linear axes, which allow horizontal as well as vertical movements as
well as a vacuum gripper. The horizontal and vertical axes only provide end to end movements. Their position is indicated by the sensors (Sensor_X0, Sensor_X1). We assume that the movement from one to the other end position is completed within 500ms. The supposed movement time only represents the normal case.

Each Handling Unit is represented by a PlantComponent, which is hierarchically structured since its elements Horizontal Axis, Vertical Axis as well as Gripper are modeled by a subComponent. The plant behavior model of a Horizontal Axis implements the TSCM shown in Fig. 4. The graphical representation of the plant behavior model of the Horizontal Axis is illustrated in Fig. 7. The Horizontal Axis only changes its actual state, if it gets a move command in terms of a boolean value and the sensor of the desired position is not set. Before leaving the actual position its sensor is reset. After the movement time has elapsed the sensor of the desired position is set and the axis remains in the initial state HorizontalStop until a further move command is received.

The hierarchically modeled Handling Unit has to be made available for the proposed simulation framework as well as its behavior. Therefore it has to be transformed into an IEC 61499 application. During this transformation all PlantComponents containing subComponents shall be transformed into a SubApplication. Due to currently limited support for SubApplications, CFBs are used instead—limiting the execution to a single device. All PlantComponents without subComponents are transformed into a CFB. The CFB representing the Horizontal Axis and respectively the Vertical Axis is displayed within Fig. 8. The operation time of a plant component is represented by an E_Delay FB. In our example those delay FBs generate the needed time frame for the movement (in or out) of the linear axis. The three bottom FBs perform the calculation of the Denavit-Hartenberg parameters for the actual kinematic chain element. This part should only be understood as example since the kinematics of pneumatic components with only two postitions is quite simple.

The behavior of such an axis is represented by the BFB HorizontalAxisBehaviour, which contains the timed state chart of Fig. 7 in an ECC. Therefore the timed state chart of Fig. 7 is transformed into an ECC like displayed within Fig. 9.

The time delay for the axis movement is taken into account by the use of additional event inputs like the MoveInDelay as well as the MoveOutDelay. For initialization as well as deinitialization own states and algorithms are inserted during the transformation, which set the desired port values. During the execution of the OUT as well as the IN state not only sensor values are updated, but also the delay time of the connected E_DELAY FB is configured. If a position calculation is needed these state also set the Denavit-Hartenberg Parameter for a further calculation.

To keep the prototypical implementation rather simple neither error cases nor variations of the delay times (i.e., randomness parameters of the delay conditions) are used.

6. Conclusion and Outlook

Software related tasks in the development of automation systems make up a huge part of the total costs. The effort is expected to rise even more in the upcoming years. Concurrent development can help to reduce the implementation time and thus the related costs. The model-driven development approach, as proposed by the MEDEIA project, is promising to support an integrative, multi-domain development process.
The MEDEIA Simulation Concept, which is integrated in the overall MEDEIA development approach, was presented in this paper. It enables a more efficient control development for production systems. IEC 61499, an international standard for distributed control, which evolved from the industrially accepted standard IEC 61131, is used as basis for the simulation execution. The relevant MEDEIA meta-models (i.e., MEDEIA Plant Model and Timed State Chart Model) have been described. Furthermore the expected structure of the equivalent executable IEC 61499 Plant Simulation Model and the model transformation were presented. The applicability of the simulation concept for manufacturing systems was shown with a laboratory example. Further tests and refinements of the models will be conducted to support process engineering applications. The integration of external simulation tools in the simulation framework might be necessary for those applications. Further application domains might be considered for future revisions. To facilitate a wide-spread use and reuse of the MEDEIA approach, a common basis of meta-models is necessary. A standardisation of meta-models for the industrial automation domain is envisaged by the MEDEIA consortium.

Acknowledgment

The work leading to the invention described within this article has received funding from the European Community’s Seventh Framework Program (FP7/2007-2013) under grant agreement no. 211448 and the Austrian Federal Ministry of Science and Research. Further information is available at: www.mediea.eu

References