Limitations in Modeling Autonomous Logistic Processes

Challenges and Solutions in Business Process Modeling

Bernd Scholz-Reiter, Daniel Rippel, Steffen Sowade
BIBA – Bremer Institut für Produktion und Logistik GmbH
Bremen, Germany
{bsr, rip, sow}@biba.uni-bremen.de

Abstract—**The paradigm of autonomous control provides organizational means to face today's requirements for robust and flexible operating logistic systems. It delegates planning and execution competencies to logistic objects in order to decrease the control system's complexity. Specifically, the Autonomous Logistic Engineering Methodology is designed for modeling, simulation, and evaluation of autonomous logistic processes. However, challenges emerge concerning its scalability in case of analyzing large logistic networks. Hence, this article investigates the influence of the system's complexity and of companies' organizational independence on the methodology with the help of scenarios in production logistics and in supply chain management. It sketches possible solutions to cope with the scalability-induced effects and to enables modeling of cross-organizational logistic processes.**

Keywords—autonomous logistic processes; limitations in process modelling; scaleabilty induced system complexity; organizational independence

I. INTRODUCTION

Today's logistic systems become more and more complex. In this situation, high fluctuations in customer demand as well as unforeseen events decrease the predictability of their behavior and increase the system's dynamics and vulnerability [1]. While classical production planning and control systems are reaching their limits in dealing with these effects, the paradigm of autonomous control offers a solution [2]. It aims to increase a logistic system's robustness and flexibility by distributing planning and control competencies to logistic objects, e. g. to commodities, half-finished goods, resources, and orders. This approach relies on the logistic objects' local decision making and leads to a positive behavior of the overall system [2].

Logistic experts face the tasks to design, model, and evaluate autonomous processes in order to apply the paradigm of autonomous control to logistic systems. The development of autonomous logistic systems includes the specification of logistic processes, the logistic objects' abilities, decision-making strategies, and a description of the overall system. In order to guide logistic experts through the development process, a modeling methodology called Autonomous Logistic Engineering Methodology (ALEM) is being developed [3].

Although ALEM supports the development process, challenges may emerge in modeling large-scaled autonomous logistic systems. The first challenge can arise from ALEM's bottom-up modeling approach, which suggests modeling of a logistic object's knowledge, abilities, and decision-making strategies, before combining them into autonomous business processes. Due to the high level of detail, the effort of modeling increases rapidly with a system's size and complexity. Moreover, a growing number of modeled objects, their processes, abilities etc. complicates coordination and error tracking of models for logistic experts. The second challenge can arise in the context of modeling cross-company logistic networks. Information, being critical for the development of autonomous logistic processes, may be unavailable due to the organizational independence of network partners. In addition, partners might independently design their part of the network with respect to their own objectives. Both issues could lead to difficulties in modeling and evaluating of the overall system.

This article aims to identify challenges and limitations regarding the scalability of modeling autonomous logistic systems and proposes solutions to cope with both issues. It presents the paradigm of autonomous control and the corresponding modeling methodology ALEM in section two and three. In the following sections, the article discusses organizational, technical, and personnel issues by making use of two logistic scenarios. The first scenario represents a manufacturing system, while the second illustrates a supply chain. Thereby, the article sketches approaches to deal with these challenges and limitations.

II. AUTONOMOUS CONTROL

Hülsmann and Windt define autonomous control as "processes of decentralized decision-making in heterarchical structures. It presumes interacting elements in non-deterministic systems, which possess the capability and possibility to render decisions independently. The objective of Autonomous Control is the achievement of increased robustness and positive emergence of the total system due to distributed and flexible coping with dynamics and complexity" [2].

In order to enable autonomous control, selected logistic objects are equipped with the abilities to manage information, to process information (decision-making), and to execute decisions. The heterarchical interactions of these objects form autonomous business processes. Although it is impossible to predict the overall system's performance, simulation studies demonstrate positive effects of autonomous control in logistic

---

This research is funded by the German Research Foundation (DFG) as part of the Subproject B2 of the Collaborative Research Center 637 "Autonomous Cooperation Logistic Processes – A Paradigm Shift and its Limitations" (CRC 637).

978-1-61284-343-8/11/$26.00 ©2011 IEEE

systems, e. g. in terms of logistic goal achievement, flexibility, and robustness ([4], [5], [6], and [7]).

III. AUTONOMOUS LOGISTIC ENGINEERING METHODOLOGY

The methodology ALEM is designed to provide logistic process experts with tools and methods for the development of autonomously controlled logistic processes. The methodology spans four steps: First, it supports the specification of logistic processes and scenarios. Second and third, it employs a simulation concept to evaluate both. Finally, ALEM provides tools and methods for the configuration of the infrastructure that is necessary to enable the autonomous processes. Moreover, the methodology is expected to enable cost-benefit analyses in the future. ALEM's methodical part uses an extended UML-Notation [3] [8] to provide logistic process experts with a commonly known modeling language. It structures the models with the help of a view concept and proposes ALEM-P as modeling procedure for the modeling process.

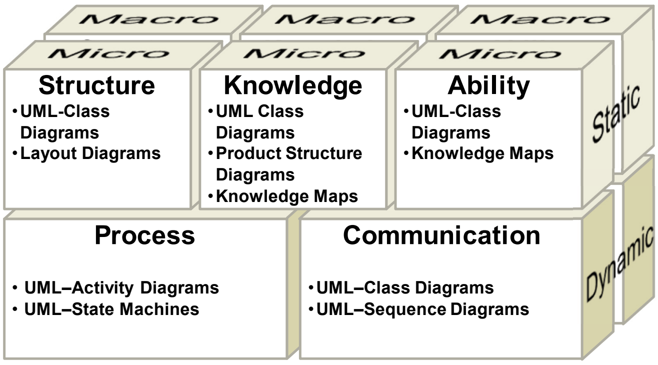

Figure 1. ALEM View Concept [9]

The view concept distinguishes ALEM models in five semantic views [9] (Figure 1.): a structure view (structural features), a knowledge view (knowledge aspects), an ability view (actions which can be performed by the logistic objects), a process view (processes), and a communication view (communication protocols and message contents). The views either describe static or dynamic features of the model. For example, static features define logistic objects existing within a model and detail their knowledge, abilities etc. Dynamic features cover the logistic objects' processes and communication behavior. Modeled aspects refer either to a micro or a macro perspective. The micro perspective presents object-internals, like decision-making strategies. The macro perspective describes object-externals, like interaction protocols.

The ALEM procedure model [10] conforms to the semantic views. It covers eight steps and follows the bottom-up modeling approach. Logistic experts are advised to model a logistic object's objectives, knowledge, abilities etc. before modeling the overall system (scenario). Feedback loops are allowed between the different modeling steps. Figure 2. presents the procedure model and depicts possible feedback flows between its steps. For example, a logistic expert has to ensure that the required knowledge is available to the logistic objects involved, if he/she designs a communication protocol.

The ALEM methodology has already been applied to small production logistic systems (e. g. [10]). In order to examine expected implication in case of larger scaled logistic systems, the next section briefly introduces two different sample scenarios and describes their differences.

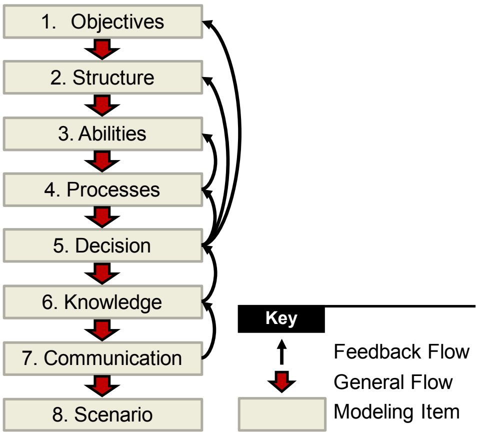

Figure 2. ALEM Procedure Model [10]

IV. SCENARIOS

This section introduces a flexible flow-shop manufacturing system and a supply chain scenario. It sketches the differences in modeling both scenarios. Subsequent sections use these scenarios to identify scalability-related challenges and limitations on the ALEM methodology.

A. Flexible Flow-Job System

Flow-shop systems usually employ different production stages. Commodities pass these stages in sequential order. According to Allahverdi et al. [11], flexible flow-job systems use parallel machines on each production stage. This article's example flow-shop scenario utilizes m production stages, each with n equal machines in parallel (Figure 3.). An example with two stages and two parallel machines could consist of, two turneries on the first stage and two sawmills on the second.

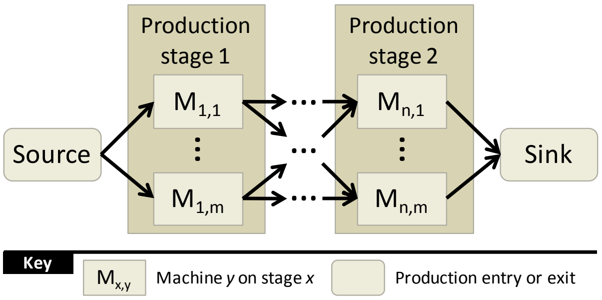

Figure 3. Flexible Flow-Job Scenario

In order to apply autonomous control to flexible flow-job scenarios, logistic experts first identify those logistic objects that will make and carry out local decisions. In this case, these objects are at least commodities and machines. Following the ALEM bottom-up modeling approach, logistic process experts model the objects' abilities, processes, and knowledge, as well as the data exchanged between them. Building upon this basic model of a system, logistic experts design decision-making functions, -strategies, communication protocols, and include additional, required knowledge. In a last step, logistic experts evaluate the model and refine it if needed.

B. *Supply Chain Scenario*

Value chains cover all activities that increase a product's value, i. e. manufacturing and transportation, as well as management activities within and around one particular organization [12]. In contrast, supply chains span the part of a value chain that focuses on the delivery process of products and on the involved partners. The supply chain example employed in this article originates from field studies conducted in the apparel industry [13]. The study covers the production of garments in Asia, their transportation to distribution centers in Europe, as well as their shipping to customers. The considered supply chain includes one production plant, one reloading point, three distribution centers, and several customers. Transportation between plant, reloading point, and distribution centers is performed either by ship or by airplane. Motor trucks serve the customers on the last hop. Different subcontractors perform the transportation task on each supply chain segment (Figure 4.).

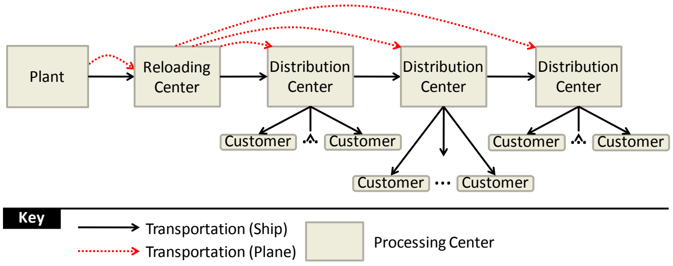

Figure 4. Value Chain Scenario

In the manufacturing plant, decision-making objects are commodities, (half-) finished goods and machines. The garments are able to arrange themselves to batches, e. g. packages, palettes, and containers. Further, the batches are able to decide on their own to decompose into smaller batches or even to single garments in distribution centers, as well as to recombine themselves as demanded in a customer order. Transportation devices, e. g. planes, ships, trucks, and trains, constitute additional decision-making objects. They perform routing decisions on behalf of (batches of) garments.

In order to apply autonomous control to the described value chain, logistic experts have to model all decision-making logistic objects, including their knowledge, abilities, decision-making strategies, processes, and data exchanges. Thereby, the organizational independence between manufacturing companies, transportation providers, and distribution center operators leads to difficulties in modeling all processes and objects in detail from a single point of view. Logistic experts, belonging to one manufacturing company, are unlikely to have insight into the corresponding transportation provider's processes, inventory, and decision-making strategies.

V. CHALLENGES ON THE SCALABILITY

As given in the introduction, challenges on the scalability of ALEM may emerge from issues related to a model's complexity and from the organizational independence between companies. This section investigates both issues with reference to the provided scenarios. It describes identified challenges and sketches methods to face them.

A. *Challenges resulting from Complexity*

The complexity induced by an increasing amount of objects, processes, and decision-strategies makes it difficult for logistic process experts to maintain a general view on all system aspects. Furthermore, they are likely unable to estimate the total impact of local modifications towards the overall system and to determine how parts of a model contribute to a specific behavior in the overall system. Thus, the effort for refining and testing of a model increases. These difficulties occur in particular in case of miss-specified model elements or if error tracing is demanded. Moreover, the time which is necessary for model refinement, error tracing, and simulation prolongs with an increasing number of objects, processes, and decisions. Last but not least, logistic objects interact frequently with each other and react to random environmental events. Thus, their behavior becomes more and more non-deterministic from a logistic process experts point of view, which impedes uncomplicated testing of specific system behaviors.

1) *Model Creation*

Compared to the production scenario, the amount of autonomous logistic objects is much higher within the supply chain example. Transportation, distribution, and repacking constitute additional autonomous processes, each related to different logistic objects and abilities. In more complex cases, the scenario might contain equally typed objects that possess varying sets of abilities and knowledge. For example, commodities in distinct manufacturing plants are likely to utilize different decision-making strategies and thus possessing different sets of knowledge. Additionally, each decision-making strategy could require a specific configuration of abilities at the plant's commodities and resources. Therefore, it might become necessary to model each type of logistic object individually for each plant. With an increase in the number of intelligent logistic objects, the amount of interactions between those objects rises as well. Communication protocols and processes have to regard each individual objects' abilities and knowledge. In case of multiple objects participating in one process, logistic process experts have to ensure that all of these objects are modeled and possess the required knowledge and abilities. The other way around, objects participating in multiple processes have to provide all the required abilities and knowledge. Hence, the effort in co-designing economically viable intelligent logistic objects and effective autonomous processes increases.

ALEM's bottom-up modeling approach constitutes an additional factor impeding design and coordination of a growing variety of objects and processes. In contrast to classical modeling approaches, ALEM shows individual processes and their related components separated from each other, but does not presents overall processes. Overall processes are composed of single intelligent logistic objects' local processes, activities, and communication protocols. During the system development, logistic process experts describe only single aspects, while the overall process emerges from the different behaviors' interactions. In order to acquire an illustration of the overall process, logistic process experts have to search manually through a variety of diagrams to identify main activities and their interconnections. With an increasing size of a logistic system, the number of diagrams and interconnections increases. Thus, it becomes harder to retain an overview of the complete model.

One possibility to face these challenges could be the integration of a process design perspective into ALEM's macro-process view. This diagram could enable the design of processes from a top-down point of view. It consists of interlinked process steps. Each process step refers to a logistic object's specific local processes or activities. Links indicate required communication protocols. By refinement, a modeler decomposes process steps, which cover activities of different logistic objects. These process design diagrams sketch those intelligent logistic objects involved in a process as well as the activities and processes those objects have to execute.

2) *Modification*

With an increasing size of the modeled system, the application of modifications becomes more difficult. Modifications might become necessary for purposes of error solving or comparison of different system architectures. While the introduction or removal of intelligent logistic objects requires a validation of modeled interactions and processes, modifications to exiting objects' abilities or knowledge have extensive consequences. With an increasing size of the modeled system, the number of interactions between logistic objects increases and objects become involved in a variety of processes. Therefore, modifications to a single object's abilities or knowledge affect a growing number of distinct processes and objects. Within the job-shop example, commodities proceeded trough manufacturing. A modification only affects those processes related to commodities and resources. Within the supply chain example, a modification to a commodity's knowledge or abilities might additionally affect processes related to transportation, packaging and repackaging, as well as distribution. Modifications might become even more serious if a scenario includes multiple, distinct manufacturing plants. Due to the growing interconnectivity, the estimation of a modification's effects becomes more difficult for a modeler. Each modification must regard the overall system's behavior as well as other objects' local processes' requirements. For example, if a modeler redistributes knowledge between commodities and resources of the supply chain example's manufacturing plant, he/she has to ensure that the commodities still possess all knowledge to perform (re-) packaging, transportation, and distribution.

The ALEM-Procedure model indicates relations between ALEM's different diagrams as feedback loops. If interpreted in a bidirectional way, those links provide hints on the implications of a modification. For example, a modification to an intelligent logistic object's abilities requires a validation of the objects processes, decision-making strategies, knowledge, and communication. Each successive modification requires additional validations. The integration of automated tests into the ALEM software tool could be another way to support modelers in modifying an existing model. In case of modifications, this tool checks, according to the procedure models feedback loops, if a modification affects other diagrams. Modifications to an intelligent object's knowledge might initiate a validation of the objects abilities, communication protocols, decision-making strategies etc. The validation checks if the modified piece of knowledge was referenced by those diagrams and presents affected diagrams to the modeler.

3) *Testing and Validation*

With an increasing system complexity, validating and testing of a model becomes more complex and time consuming. From a technical point of view, simulation runs prolong with the size of the system. Moreover, the systems behavior becomes more and more non-deterministic from the modelers point of view. This results from random events, occurring during simulation, as well as due to the non-determinism of specific autonomous control strategies (e. g. [14]). Furthermore, emergent effects infer with the predictability of autonomous systems. The increasing amount of interactions between logistic objects and the interconnection between processes faces modelers with the challenge to determine those objects being responsible for a particular behavior within the overall system. In case of a behavior or decision appears to be incorrect, a modeler has to identify the source of the problem. Therefore, it might be necessary to recreate the defective situation in order to identify possible solutions. For this purpose, the exact situation has to be restored as a scenario in ALEM by including all relevant object's states, positions, and knowledge at the given point of time. The isolation of the errors source contributes an additional challenge. With an increasing number of interactions, more and more distinct logistic objects might contribute to an error. For example, if a decision based on false information, this information might have been deduced wrongly be the object itself or it might have been provided wrongly by another object. The second case requires a modeler to further investigate other object's state and their information history. The task of isolating a source of an error becomes more complex with an increasing size of the modeled system. In addition, it requires a deeper insight into the interconnections, in particular if information is transmitted through several objects.

One option to support a modeler to master these challenges could be the introduction of a detailed logging functionality to the simulation. During simulation runtime, all deduced, transmitted and ascertained information is logged. In addition to logging, a tracing functionality could provide information traces. These depict all information used within a decision as well as the information's origins and transmission routes. This functionality supports the user in isolating the source of an error. Additionally, it supports recreation of one particular situation, as each object's state is visible at each point of time.

B. *Challenges resulting from Organizational Independence*

Companies organize their logistic processes in order to produce goods for consumers under the side condition of profit maximization. They perform the corresponding processes with respect to their technological and economical capabilities and concentrate on specific products and services. For this purpose, companies organize themselves in supply chains, production networks, or virtual enterprises whose members are organizationally independent of each other [15]. Their independence causes serious challenges for the modeling of autonomous logistic processes with ALEM.

1) Problem Areas and their Consequences

On the one hand, a company's logistic process experts have a limited view on partners' and subcontractors' processes. Their partners' control and information management strategies usually prevent them to obtain all information being required for modeling of logistic processes. Thus, they are unable to model the complete logistic system in high detail level. This knowledge-oriented problem results in a chance either to neglect useful information during the modeling process or to presume information which does not correspond to the system. On the other hand, ALEM assumes a centralized modeling of a highly decentralized system; i. e. a single modeling instance is proposed to create the complete model. This approach works fine in case of a small system that is under control of the modelers company. However, it fails in case of larger logistic systems where knowledge-oriented problems occur.

For instance, in the supply chain case study illustrated in Figure 4. , one company controls the plant and all distribution centers, while various service providers operate the reloading center and transportation links [13]. Each supply chain member hides his internal operational and organizational structure in order to keep business secrets, like politics and strategies. The members follow own objectives and are unable to decide on other partners' behavior. Hence, they are not responsible for a partner's behavior. This independence appears in the methods, standards, and tools used by them internally. Each member models their part of the supply chain for their own.

For this reason, suitable mechanisms have to be added to the ALEM methodology in order to deal with the lack of knowledge as well as to provide means for inter-organizational collaborative modeling of autonomous logistic business processes. The mechanisms have to support the complete system development cycle: system specification, infrastructure configuration, simulation, and evaluation. Further, they have to enable modeling of the complete model or parts of it in parallel, at different locations, at different times, and by different modelers or organizations.

2) Additional Views

Although the ALEM view concept provides seven distinct dimensions on its models, it excludes an illustration of the overall model to system modelers, process managers, or single autonomous logistic objects. Further, explicit collaborative views showing coordination aspects between modeling instances are not part of the ALEM view concept. However, this kind of a view is useful to oversee processes spanning large logistic networks from the viewpoint of one modeling instance [16], [17]. Hence, a collaborative view shall be part of mechanisms integrating the multi-user functionality. Black box and white box mechanisms help to generate an overview about the complete system for one particular modeler. In addition, a new dynamic tracing view can be helpful to trace selected logistic object on their path though a logistic system.

3) Collaborative Mechanisms for ALEM

The literature proposes several mechanisms that collaborative modeling methods have to comply with [16], [17]. The mechanisms contain different operational and organizational rules, processes, and structures and can be classified into five groups (Table 1).

TABLE I. CLASSES OF REQUIRED MECHANISMS (COMPLIES WITH [16])

• Spatial and temporal distributed modeling and simulation
• Parallel and partial modeling and simulation run execution
• Exchange of information of sub-models, models and simulation data
• Multi-user and multi-organization modeling
• Quality management functionality

Most important is a mechanism that is able to split models into sub-models and to recombine them later in a new model instance [17]. Therefore, standardized interfaces have to be defined which describe the linkage between ALEM model elements and ALEM sub-models. As a consequence thereof, mechanisms have to enable modeling in independent instances, e. g. at different companies at a time. For this purpose, a data format has to be defined for exchanging a complete model, selected sub-models, or model elements in order to present and integrate each of them in other modeling instances at different locations. In addition, a version control system shall be used to enable comparison and backtracking of different model versions. A user profile manager is required which handles the users' access rights towards model components. A commentary function could allow users to place asynchronously notes at the model workspace. Unknown parts of a model shall be handled as a black box that is characterized by its input and output characteristics. The black box integrates the concept of (sub-) model capsulation. The capsulation mechanism reduces the models' complexity and enables modeling of complete logistic systems although some information remains unavailable. A set of quality management functions shall ensure the overall model's consistency, completeness, as well as syntactic and semantic correctness.

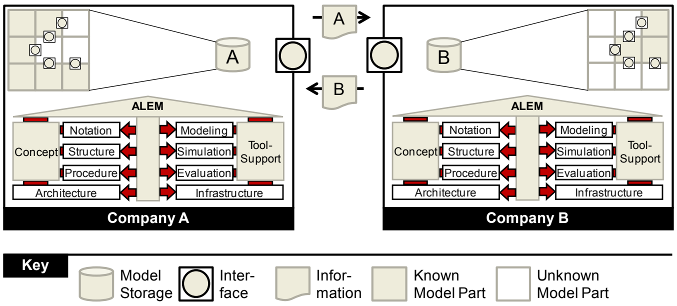

Figure 5. Collaborative-ALEM-Framework

Figure 5 exemplarily presents the exchange of sub-model information between two different companies using ALEM. It shows the interfaces between both methodology instances as well as between known and unknown sub-models. The interfaces enable the exchange of model information between different instances of the ALEM software tool and facilitate the capability for distributed simulations. For example, company A knows six of nine sub-model elements, while company B knows only three of them. Both companies are either able to exchange information about the sub-models content in order to learn more details of the overall model, or they exchange sub-model interface information. In the latter case, companies model their part of the logistic process independently. The interfaces, enabling interactions between different sub-models and ALEM processes, clearly have to specify input and output flows in terms of required and available information. While designing the overall process, each partner defines the information that is required for entering a process and that is provided on exit. The sub-model interface hides other sub-models. The respective company notices known model elements as white boxes, while recognizing unknown model elements and their behavior as black boxes.

VI. CONCLUSION

This article discussed several challenges concerning the scalability of the ALEM methodology at the example of a manufacturing and a supply chain scenario. It investigated the introduction's thesis that scalability related challenges emerge from the systems growing complexity as well as from the organizational independence of partners in supply chains. In order to overcome these challenges, the article proposed several modifications to the ALEM methodology and its software tool. TABLE II. summarizes the identified challenges and the proposed solutions. These solutions can support logistic process experts in modeling autonomous processes regardless of a scenarios' size. In particular, the paper introduced additional mechanisms applicable for ALEM which enable modeling of processes spanning different organizations.

TABLE II. SUMMARY OF CHALLENGES AND SOLUTIONS

Thesis	**Challenge**	**Proposed Solution**
Complexity	ALEM bottom-up modeling approach	➢ Process development view
	Model consistence	➢ Validations based on the ALEM-Procedure Model ➢ Automated tests
	Recreation (testing) of specific situations	➢ Extended logging
	Non-determinism	➢ Information tracing
Organizational Independence	Unavailable knowledge and low information transparency	➢ Interface design ➢ Tracing view ➢ Collaborative view
	Collaboration support	➢ Decentralized modeling ➢ Distributed simulation

However, a detailed requirements analysis and specification for particular ALEM amendments is necessary as first step for an integration of the proposed solutions into ALEM. Thereafter, the specified collaboration concept has to be implemented and tested in simple and complex supply chain scenarios. Upon the results observed thereby, further conclusion can be made towards the applicability of a collaborative ALEM methodology and the mechanisms employed.

REFERENCES

[1] W. Schäfer, R. Wagner, J. Gausemeier, R. Eckes "An Engineer's Workstation to Support Integrated Development of Flexible Production Control Systems," in *Integration of Software Specification Techniques for Applications in Engineering*. Berlin: Springer, pp. 48–48, 2004.

[2] M. Hülsmann and K. Windt, Eds., *Understanding of Autonomous Cooperation and Control in Logistics – The Impact of Autonomy on Management, Information, Communication and Material Flow*. Berlin: Springer, 2007.

[3] B. Scholz-Reiter, J. Kolditz, and T. Hildebrandt, "Engineering autonomously controlled logistic systems," *International Journal of Production Research*, vol. 47, no. 6, pp. 1449–1468, 2009.

[4] D. Armbruster, C. de Beer, M. Freitag, T. Jagalski, Ch. Ringhofer, "Autonomous Control of Production Networks Using a Pheromone Approach," in *Physica A*, vol. 363, no. 1, pp. 104–114, 2006

[5] B. Scholz-Reiter, M. Görges, T. Jagalski, and A. Mehrsai, "Modelling and analysis of autonomously controlled production networks," in *Proceedings of the 13th IFAC Symposium on Information Control Problems in Manufacturing (INCOM 09)*, Moscow, Russia, 2009, pp. 850–855.

[6] S. Dashkovskiy, F. Wirth, and T. Jagalski, "Autonomous control in shop floor logistics: Analytic models," in *Manufacturing, Modelling, Management and Control 2004*, G. Chryssolouris and D. Mourtzis, Eds. Amsterdam: Elsevier Science Ltd, 2005.

[7] H. Rekersbrink, T. Makuschewitz, and B. Scholz-Reiter, "A distributed routing concept for vehicle routing problems," *Logistics Research*, vol. 1, no. 1, pp. 45–52, 2009.

[8] Object Management Group, *Unified Modeling Language (UML) V.2.2*, Object Management Group Std. 2, Rev. 2. [Online]. Available: http://www.omg.org/technology/documents/modeling_spec_catalog.htm #UML

[9] B. Scholz-Reiter, H. Höhns, J. Kolditz, and T. Hildebrandt, "Autonomous Supply Net Coordination," in *Proceedings of 38th CIRP Manufacturing Systems Seminar*, Florianopolis, Brazil, 2005, CD-ROM.

[10] J. Kolditz, "Fachkonzeption für selbststeuernde logistische Prozesse," Ph.D. dissertation, Universität Bremen, Berlin, 2009.

[11] A. Allahverdi, C. Ng, T. Cheng, and M. Kovalyov, "A survey of scheduling problems with setup times or costs," *European Journal of Operational Research*, vol. 187, no. 3, pp. 985 – 1032, 2008.

[12] M. Porter, *Competitive Advantage: Creating and Sustaining Superior Performance*. N. York: The Free Press, 1985.

[13] B. Scholz-Reiter, S. Sowade, D. Rippel, M. Teucke, M. Özsahin, and T. Hildebrandt, "A Contribution to the Application of Autonomous Control in Manufacturing," *International Journal of Computers*, vol. 3, no. 3, pp. 279–291, 2009.

[14] K. Windt, T. Becker, O. Jeken, A. Gelessus, "A classification pattern for autonomous control methods in logistics," *Logistics Research*, vol. 2, no. 2, pp. 109-120, 2010.

[15] M. C. Cooper, D. M. Lambert, and J. D. Pagh, "Supply chain management: More than a new name for logistics," *International Journal of Logistics Management, The*, vol. 8, no. 1, pp. 1–14, 1997.

[16] K. Ryu and E. Yücesan, "Cpm: A collaborative process modeling for cooperative manufacturers," *Advanced Engineering Informatics*, vol. 21, no. 2, pp. 231–239, 2007.

[17] P. P. Gonzalez and J. M. Framiñan Torres, "Tools for collaborative business process modeling: Classification and evaluation," *Encyclopedia of Networked and Virtual Organizations*, vol. 3, pp. 1643–1652, 2008.