Prod. Eng. Res. Devel. (2010) 4:319–325
DOI 10.1007/s11740-010-0253-2

PRODUCTION MANAGEMENT

Modeling of orders in autonomously controlled logistic systems

B. Scholz-Reiter · St. Sowade · T. Hildebrandt · D. Rippel

Received: 29 January 2010 / Accepted: 9 June 2010 / Published online: 24 June 2010

Abstract The paper extends the Autonomous Logistics Engineering Methodology (ALEM) by a deeper understanding of immaterial logistic objects to trigger manufacturing processes. Further, a hierarchical modeling concept is introduced to split customer orders logically into partial orders, which run directly at the shop floor level. Each partial order consists of certain manufacturing steps. The amendments enable adequate modeling of autonomous manufacturing processes. The research is a further step to integrate autonomously controlled processes in logistics.

Keywords Production management · Autonomous logistic object · Customer order decomposition

1 Introduction

Today's manufacturing systems operate within a highly complex and dynamic environment. Several internal and external factors influence their performance continuously [1]. For this reason, changeable manufacturing and autonomous control are investigated in manufacturing systems research [2, 3]. The first concept compasses the achievement of changeability for products, processes, facilities and organizations. The latter concept focuses on control processes in particular. It aims to increase the robustness of manufacturing systems and to reduce the complexity of its processes by enabling local autonomy. Several simulation studies show the benefits of autonomously controlled manufacturing scenarios and of autonomous routing in transport logistics [4, 5].

B. Scholz-Reiter · St. Sowade (✉) · T. Hildebrandt · D. Rippel
BIBA, Bremer Institut für Produktion und Logistik,
University of Bremen, Hochschulring 20, 28359
Bremen, Germany
e-mail: sow@biba.uni-bremen.de

1.1 Research question

Within the Collaborative Research Center (CRC) 637 the Autonomous Logistics Engineering Methodology (ALEM) is developed in order to model logistic systems based on autonomous logistic processes [6, 7]. Currently, the methodology suffers from a vague understanding of immaterial logistic objects as well as from two subsequent modeling limitations. This paper aims to solve these three issues.

First, ALEM focuses on production logistics and its relevant physical logistic objects, like shop floor resources and half finished products. Although the methodology enables modeling of immaterial logistic objects as well, it lacks to explain the meaning and role of these objects. For instance, the current ALEM reference model contains several physical logistic objects, but only one immaterial logistic object. Hence, the paper seeks to clarify the understanding of immaterial logistic objects in manufacturing scenarios.

Second, ALEM requires mechanisms for processing and dispatching customer orders. Usually, companies obtain customer orders, which demand several (different) products. In contrast, ALEM assumes one order per product unit. As a result, the manufacturing system cannot process these orders directly. Thus, the production planning and control system (PPC system) has to decompose customer orders to manufacturing orders before releasing them to the shop floor. This process requires knowledge about product structures, manufacturing processes, and procedures to match customer orders to manufacturing orders. The paper focuses on the description of the decomposition method and the information required.

Third, manufacturing generally contains both: production processes and assembly processes. In contrast, ALEM is designed to model production processes only. This restriction eases modeling, but neglects an essential part of the real world. For this reason, the paper seeks to include modeling of assembly processes explicitly in ALEM. The integration of assembly processes leads to additional requirements, i.e. the synchronization of commodity flows.

1.2 Structure of the paper

After introducing the research question in Sect. 1, the following section explains the concept of autonomous control in logistics and the modeling methodology ALEM. The third section introduces the proposed amendments of ALEM. First, it describes the idea of modeling immaterial logistic objects. Second, the section explains the concept for order processing and dispatching using hierarchical order decomposition. Further it describes assembly step integration. Third, the paper presents the most important changes to the ALEM modeling framework. The last section summarizes the work and closes with an outlook for future research.

2 Modeling of autonomous control

High utilization, availability, productivity, and rate of in-time delivery, as well as low inventory, process costs, and short throughput times are important objectives in PPC systems [8–10]. These go along with demands for increased flexibility and product variants, as well as calls for adaption of different lot sizes and reduction of lead times [8, 10, 11]. Autonomous control promises to achieve these objectives in a decentralized way.

2.1 Autonomous control

The roots of the term autonomous control are located in biology and physics. These disciplines have tried to understand autonomy and self-organization for a long time. New research fields investigating these concepts are artificial intelligence and control theory [12].

In the context of logistic systems, 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" [3]. The system elements are called autonomous logistic objects and follow their own local objective system. However, there is no guarantee for a certain global behavior or performance [9]. Further, autonomous logistic systems require decision alternatives. These can be achieved by reserve machinery capacity or branches within product structures, for example [13].

2.2 ALEM modeling concept

The ALEM modeling concept consists of a notation, a view concept, and a procedure model. In addition, a modeling tool is developed by the CRC637 to enable logistic process experts to model their processes. The notation relies on standard UML diagrams, but is extended by certain elements and diagrams, e.g. knowledge maps and a layout diagram [14, 15].

System and process models usually imply a high degree of complexity. Hence, the modeling approach uses a view concept in order to handle the complexity of the model [8]. There are five views (Fig. 1). Each view focuses on specific aspects of the model [16]. Additionally, the concept distinguishes between static and dynamic sub-models. The former describes the structure of the modeled scenario; the latter describes its behavior. Another distinction is made between a micro view describing the systems elements' internals and a macro view describing the interaction between the autonomous objects.

The *structure view* describes the relevant logistic objects in a class diagram. Their relations are modeled by different types of associations. A layout-diagram of the manufacturing site supplements the abstract classification of logistic objects. The *knowledge view* organizes the knowledge, which is present at the logistic objects, to enable autonomous decentralized decision making. The *ability view* contains type and structure of the logistic objects. Abilities can be split into sub-abilities and form an abstract set of operations. The *process view* focuses on the logic-temporal sequence of activities and states of manufacturing control processes. The view uses activity and state diagrams. The *communication view* describes the interaction and the information exchange between logistic objects. Communication processes are modeled using sequence diagrams. The message content structure is modeled in class diagrams.

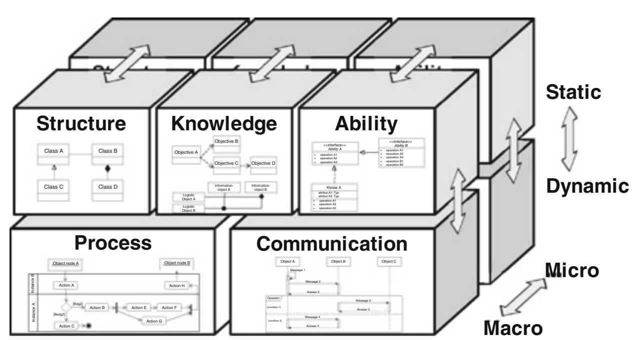


Fig. 1 ALEM view concept [17]

Scholz-Reiter et al. propose an eight step modeling procedure for ALEM to guide the user through the modeling process [18]. Steps one to seven refer to modeling of the autonomous system on an abstract level, while the eighth step is used to instantiate this model for a certain scenario, i.e. shop floor scenario. Although the steps show a straight sequence of modeling items (Fig. 2), the order of modeling can differ in certain cases. For example, an algorithm for decision making usually requires a specific set of abilities. Thus, a user shall model the abilities after modeling the process of the decision algorithm. Moreover, feedback flows indicate that modeling of successive steps can affect preceding modeling items.

2.3 ALEM structure reference model

Figure 3 depicts the ALEM structure reference model for autonomous systems. The logistic object is placed in the centre of the system. Commodity, resource, and customer order are specializations of the logistic object and inherit its attributes and methods. Each logistic object owns a set of objectives for decision making. Additionally, the objects are able to communicate via messages. A machine is a special type of resource that is able to execute production steps. If a machine employs a production step to a commodity, the commodity changes its type. The order item is a component part of a customer order. It is satisfied by commodities having certain types.

3 Enhancing ALEM to model autonomous control

The ALEM methodology already offers adequate means to model autonomous logistic systems. The current section clarifies three remaining issues. First, it characterizes the meaning of immaterial logistic objects at the customer

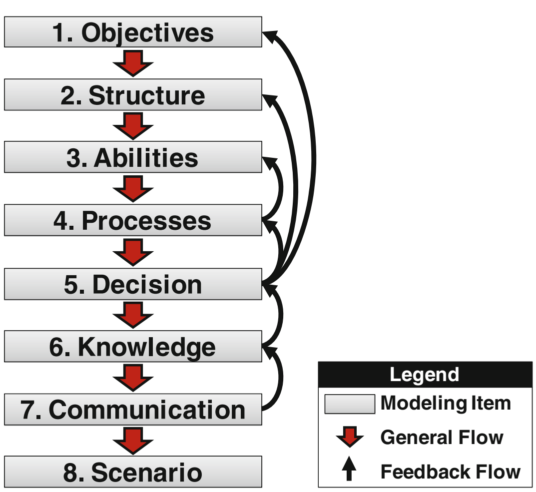


Fig. 2 ALEM procedure model [17]

order's example and points out their relation towards physical logistic objects. Second, a concept is presented to dispatch and process customer orders. Finally, the section describes the impact of the amendments towards the modeling methodology.

3.1 Understanding immaterial objects as logistic objects

ALEM supports modeling of any logistic object no matter if it is physical or not. While commodities or resources are physical objects, orders and messages are typical immaterial objects. The ALEM reference scenario contains both tight-knit object types: The customer order is an immaterial object and represents a physical commodity, while the message is an immaterial logistic object, which requires a carrier medium (not shown in the reference model). In the scope of autonomous control, orders resemble to intelligent logistic objects.

There are three necessary conditions to model both object types in a unitary way. First, an immaterial object supplements a physical object by information about its state and objectives. Second, resources modify this information, like they process physical objects. Thus, the immaterial object represents the current state of the corresponding physical object. Further, it includes information about future state demands of the physical object. Third, the immaterial object has ability "to process information, to render and to execute decisions" as well [19].

If these conditions are fulfilled, immaterial objects can be modeled as autonomous logistic objects in ALEM. For example, the conditions are true for customer orders and commodities.

3.2 Concept for hierarchical order decomposition

Processing of customer orders requires appropriate mechanisms to induce them into manufacturing systems. Thus, this section proposes product structure diagrams (PSD) and order decomposition as central elements for inducing orders.

Customer orders demand certain commodities and name amount, type, due date, and customer id. In contrast, manufacturers store product structure information in PSDs, which specify a certain product by hierarchical sequences of production and assembly orders (Fig. 4). Each manufacturing step processes an object by applying and manipulating its properties. Further, PSDs describe the states of necessary input and output objects for each manufacturing step. The order decomposition procedure requires three steps. The procedure is called hierarchical order decomposition with respect to the hierarchical structure of PSDs. First, manufacturing systems match each demanded product of a customer order to a certain PSD. Afterwards, the

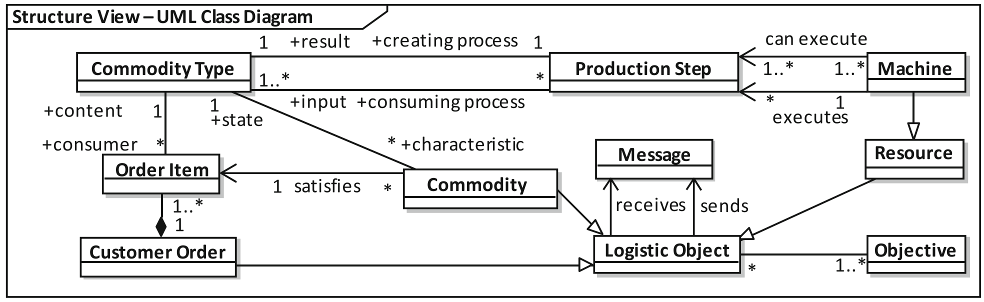


Fig. 3 ALEM structure reference model as an UML class diagram [6]

system decomposes the customer order into the manufacturing orders of the PSD. Finally, the manufacturing orders are associated to the respective raw materials for processing. The availability of product structure information is a precondition for hierarchical order decomposition.

In the case of an autonomously controlled system, the manufacturing orders are passed to the commodities, which initiate manufacturing subsequently. Thus, order and commodity are linked together while the order is manufactured.

3.2.1 Order decomposition and execution

There are three process steps required to transform customer orders into processable manufacturing orders. The decomposition procedure starts, when a customer order arrives. The first step of the procedure compares the demanded product to the end product of each PSD and selects an adequate diagram (Fig. 4a). Then, it decomposes the customer order into manufacturing orders, in accordance with the selected diagram (Fig. 4b). The manufacturing orders are grouped into assembly orders or production orders and are transferred to the relevant commodities finally. While assembly orders consist of one assembly step only, production orders combine one or more production steps. In contrast to customer orders, resources can process manufacturing orders directly.

Production orders consist of a structured graph of all production steps to be taken until either an assembly step,

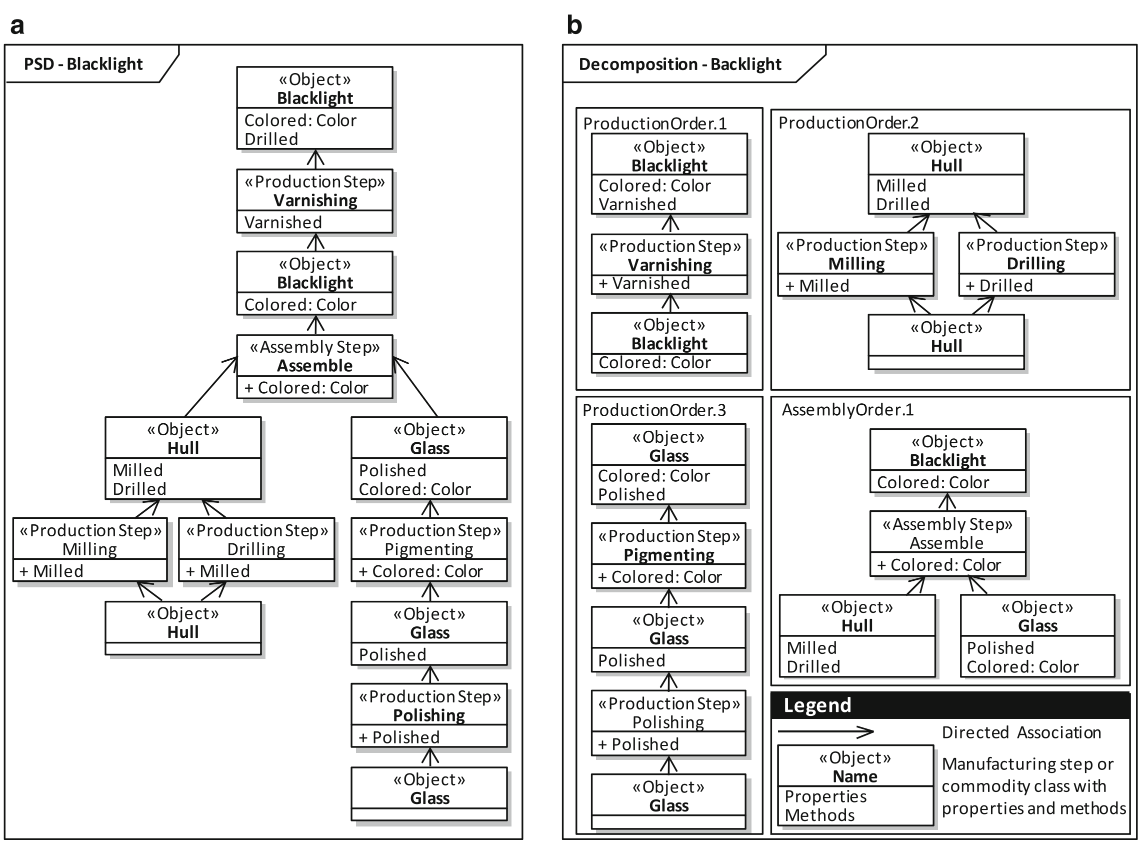


Fig. 4 Product structure model (**a**) and formation of manufacturing orders (**b**)

or the end product is reached. The graph supports modeling of concurrent manufacturing paths. Thus, PSDs offer the advantage to model decision alternatives in manufacturing. For example, ProductionOrder.2 allows the commodity to get drilled or milled in any order. Nevertheless, a commodity has to complete both paths before it can proceed. A joining node indicates this requirement (Fig. 4b). Furthermore, commodities are enabled to plan consecutive manufacturing steps flexibly and to increase the effectiveness of their production plan. However, in order to take this advantage, commodities need knowledge about subsequent manufacturing steps. On the contrary, assembly orders make use of one assembly step only to keep their complexity low. The complexity would rise strongly with an increasing number of object flows, which need to be coordinated and synchronized.

As mentioned, the process of order decomposition consists of the steps: mapping of customer orders to PSDs, decomposition of PSDs to manufacturing orders, and transfer of the manufacturing orders to commodities. In centralized logistic systems, PPC systems perform all of these steps. Contrary, intelligent logistic objects have to perform these steps in autonomously controlled logistic systems. Thus, such systems recognize all process steps as abilities and assign them to certain logistic objects. Therefore, a register is introduced as new logistic object. It decomposes the customer orders and assigns the resulting manufacturing orders to the relevant commodities upon request. Every time a commodity acquires new manufacturing orders, the register transmits the appropriate set of manufacturing goals. These goals represent the respective path in the PSD. For example, "Hull" receives the manufacturing order sequence ProductionOrder.2, AssemblyOrder.1 and ProductionOrder.1 (Fig. 4a). Logistic objects use these goals to autonomously negotiate the time which and when to use a certain resource.

3.2.2 *Synchronization of assembly steps*

All required flows of commodities have to join an assembly station at a certain time to be processed. Otherwise, assembly resource and commodity have to wait. Hence, the introduction of assembly steps leads to the problem of synchronicity. ALEM supports to model different concepts for synchronization. They range from central (resource-centered) to decentralized (commodity-centered) approaches. The following example describes briefly a reference synchronization process for the first case. In order to find an appropriate assembly station, logistic objects have to send a request to each resource. Each station collects the requests, until all necessary logistic objects have demanded a certain assembly step. Thereafter, the assembly stations transmit the results to the respective commodities. Now, each commodity ranks the assembly options and selects the best option by its own. It may happen that commodities, which have to be assembled together, prefer different assembly stations. To ensure a consistent decision anyway, each commodity sends its ranking and its evaluation procedure to the resource again. The assembly station uses the foreign objective functions and calculates a new set of options. The new result leads to a consistent decision at the commodities. For instance, related commodities accept the same resource, while rejecting all other resources.

3.3 Amendment of notation and procedure model

The application of order decomposition and assembly processes changes parts of the ALEM methodology. For example, new diagram types affect the notation and the view concept. Moreover, the ALEM procedure model must include modeling of the new information. Hence, this section describes the amendments of ALEM.

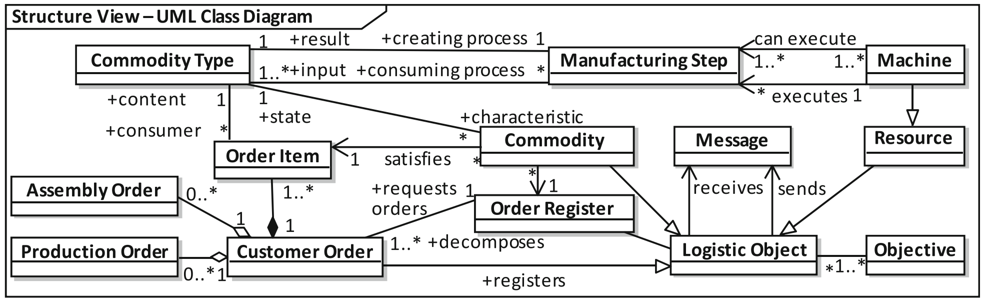
3.3.1 *Notation of product structure diagrams*

Product structure diagrams consist of a sequence of one or more manufacturing orders, which represent a complete manufacturing process (Fig. 4a). From the notation point of view, a PSD is a directed graph, pointing from the leaf nodes towards the root node. The root node is located on top and describes the end product. The leaf nodes represent the commodities or raw materials ("Hull" and "Glass"). A PSD uses three different kinds of nodes to define product structures. Object nodes represent commodities of a certain commodity type and consist of a set of properties describing the objects' actual state. Assembly nodes combine at least two object nodes to a new object node and add new properties ("Assemble"). Production nodes modify properties of an object ("Milling" or "Drilling"). Each object node, except for the root node, is succeeded by a sequence of one or more production nodes or by exactly one assembly node. A manufacturing node is succeeded by one object node, whose properties reflect the changes made by the last process step. While assembly steps replace most properties from the input nodes by new ones, production steps copy all properties. Both can add new properties, or change their values. PSDs belong to the ALEM knowledge view, because they represent knowledge about processes involved to create a product.

3.3.2 *Mandatory abilities*

The introduced concepts require the four abilities: mapping of customer orders, order decomposition, order transfer, and synchronization for assembly steps. The allocation of the abilities to logistic objects depends on the selected

Fig. 5 Class diagram of new ALEM structure reference model



processes and system architecture. For example, machines may be responsible for commodities' synchronization, while the other abilities are allocated at the order register. Of course, the order registers' abilities could be located rather decentralized as well. Further, the machines' ability to synchronize commodities could be transferred to the commodities themselves. The allocation depends on the modeler's intention.

3.3.3 Implications on the ALEM procedure model

Product structures are knowledge and describe a single product of a specific manufacturing scenario. Hence, PSDs are modeled in the eights step ("Scenario") of the ALEM procedure model. PSDs could be defined earlier as well, because of their importance for order decomposition. However, they neither offer necessary information to define the generic model, nor rely on information defined earlier. The semantic meta-model of the ALEM procedure model is not affected.

3.3.4 Amendments to ALEM structure reference model

The structure reference model employs an order register as a new intelligent object. It manages order mapping, decomposition, accounting, as well as goal-creation (Fig. 5). Further, customer orders are treated as intelligent objects, which consist of assembly and production orders. The abilities and knowledge of each order type are modeled in the respective views. The importance of the classes Commodity-Type and Manufacturing Step increase, because PSDs refer to them.

4 Conclusions and outlook

The research article describes the meaning and handling of immaterial logistic objects. For this, it employs the example of autonomous logistic processes and the customer order to answer three specific questions of modeling.

Firstly, the paper characterizes the meaning of immaterial logistic objects. As a result, the conformance to certain requirements allows modeling of immaterial logistic objects in the same way as physical autonomous logistic objects are modeled. Secondly, the modeling methodology ALEM is amended by a method to process and dispatch incoming customer orders. The method is called hierarchical order decomposition. It uses the a priori knowledge of product structure diagrams, integrates the knowledge into the ALEM knowledge view, and connects both with the order dispatch mechanism. Thirdly, assembly processes were introduced to amend the ALEM modeling concept. For each approach, the methods have been described towards their impact for the ALEM modeling methodology.

Further research will include modeling of an extended reference scenario as well as evaluation of different processes for synchronization, order decomposition and decision making. A simulation component will be integrated into ALEM to analyze the dynamic of autonomous processes. The component is a precondition to evaluate system architectures and infrastructure configurations as well.

Acknowledgments This research is funded by the German Research Foundation (DFG) as part of the Collaborative Research Centre 637 "Autonomous Cooperation Logistic Processes—A Paradigm Shift and its Limitations" (CRC637).

References

1. Westkämper E, Zahn E (eds) (2009) Wandlungsfähige Produktionsunternehmen, 1st edn. Springer, Berlin
2. ElMaraghy HA (2009) Changeable and reconfigurable manufacturing systems, 1st edn. Springer, London
3. Hülsmann M, Windt K (eds) (2007) Understanding of autonomous cooperation and control in logistics—the impact of autonomy on management, information, communication and material flow, 1st edn. Springer, Berlin
4. Scholz-Reiter B, Philipp T, De Beer C, Windt K, Freitag M (2006) Einfluss der strukturellen Komplexität auf den Einsatz von selbststeuernden logistischen Prozessen. In: Pfohl H, Wimmer T (eds) Steuerung von Logistiksystemen—auf dem Weg zur Selbststeuerung. Dt. Verkehrs-Verlag, Hamburg
5. Scholz-Reiter R, Rekersbrink H, Wenning BL, Makuschewitz T (2008) A survey of autonomous control algorithms by means of adapted vehicle routing problems. In: Proceedings of the 9th biennial ASME conference on engineering systems design and analysis, Haifa, Israel
6. Kolditz J (2009) Fachkonzeption für selbststeuernde logistische Prozesse, 1st edn. GITO-Verlag, Berlin

7. Scholz-Reiter B, Kolditz J, Hildebrandt T (2009) Engineering autonomously controlled logistic systems. Int J Prod Res 47/6: 449–1468
8. Scholz-Reiter B, Windt K, Kolditz J, Böse F, Hildebrandt T, Philipp T, Höhns H (2005) New concepts of modelling and evaluating autonomous logistics processes. In: Chryssolouris G, Mourtzis D (eds) Manufacturing, modelling, management and control 2004, IFAC workshop series. Elsevier Science, Amsterdam
9. Scholz-Reiter B, Freitag M (2007) Autonomous processes in assembly systems. CIRP Annals 56/2:712–729
10. Wiendahl HP (2005) Betriebsorganisation für Ingenieure, 5th edn. Carl Hanser Verlag, München
11. Luczak H, Eversheim W, Schotten M (eds) (2001) Produktionsplanung und -steuerung: Grundlagen, Gestaltung, Konzepte, 2nd edn. Springer, Berlin
12. Scholz-Reiter B, Höhns H (2005) Selbststeuerung logistischer Prozesse mit Agentensystemen. In: Schuh G (ed) Produktionsplanung und -steuerung: Grundlagen, Gestaltung, Konzepte, 3rd edn. Springer, Berlin
13. Scholz-Reiter B, Höhns H, Kolditz J, Hildebrandt T (2005) Autonomous supply net coordination. In: Proceedings of 38th CIRP manufacturing systems seminar, CD-ROM, Florianopolis, Brazil
14. Scholz-Reiter B, Kolditz J, Hildebrandt T (2007) UML as a basis to model autonomous production systems. In: Cunha PF, Maropoulos PG (eds) Digital enterprise technology: perspectives and future challenges, 1st edn. Springer, Berlin
15. Object Management Group: Unified Modeling Language (UML) V.2.2, Online
16. Scheer A-W (2001) ARIS—Modellierungsmethoden, Metamodelle, Anwendungen, 4th edn. Springer, Berlin
17. Scholz-Reiter B, Kolditz J, Hildebrandt T (2007) The autonomous logistics engineering methodology (ALEM). In: Proceedings of 4th international CIRP-sponsored conference on digital enterprise technology (DET2007), CD-ROM, University of Bath, Bath
18. Scholz-Reiter B, Hildebrandt T, Kolditz J (2008) Modellierung selbststeuernder produktionslogistischer Prozesse—die Modellierungsmethode ALEM. In: Mattfeld DC, Günther H-O, Suhl L, Voß St (eds) Informations und Kommunikationssysteme in SCM, Logistik und Transport. Teilkonferenz der Multikonferenz Wirtschaftsinformatik 2008. Universität Paderborn, Paderborn, pp 173–185
19. Windt K, Böse F, Philipp T (2008) Autonomy in production logistics: identification, characterisation and application. Int J Robot CIM 24/2: 572–578