Available online at www.sciencedirect.com
ScienceDirect
IFAC PapersOnLine 50-1 (2017) 1217–1224

Automatic analysis and adaption of the interface of automated material flow systems to improve backwards compatibility

T. Aicher*, D. Schütz*, M. Spindler**, S. Liu*, W. A. Günthner** and B. Vogel-Heuser*
*Institute of Automation and Information Systems, Garching near Munich,
*85425 Germany; e-mail: {aicher, schuetz, stud_liu, vogel-heuser}@ ais.mw.tum.de.
**Institute for Materials Handling, Material Flow, Logistics, Garching near Munich,
**85425 Germany; e-mail: {spindler, kontakt}@ fml.mw.tum.de.

Abstract: Today's industrial automation production systems (aPS) have to deal with the requirement to produce many different products. Hence, not only a flexible modularized aPS, but also a flexible modularized material flow system (MFS) is required in order to improve modifications to the system, i.e. extending, reducing or modifying parts of the system. To reduce the engineering effort involved in setting up modularized MFS and to deal with the complexity of the system, a definition of the architecture, behavior and (pre)defined interfaces of such modules is provided by a model-based engineering approach, which is based on an appropriate meta model. In addition, the modules can interact with neighboring modules to execute specific logistic functionalities by using (pre)defined interfaces. Since the development of automated material flow modules (aMFM) can be considered as an evolutional process, the aMFM are frequently redeveloped using current-state-of-the-art technologies, which, in some circumstances, also require newly-developed interfaces. Thus, an inconsistency between older and newer modules containing different version numbers occurs when different (pre)defined interfaces are used, leading to a reduction in system flexibility. To implement the backward compatibility of newer modules, the (pre)defined software interfaces of the newer modules, or even of its neighboring modules, usually have to be adapted manually, which increases the engineering effort and propensity for errors.

This paper introduces a model-based approach to analyzing and adapting the interface of aMFM automatically in order to enable interaction with neighboring modules across different version numbers, i.e. modules containing a different version number. Hence, the engineering effort and error-proneness for the adaption can be reduced. For instance, the automatic analysis and adaption of the interface of a newly-developed QR Code Scanner is considered here.

Keywords: flexible automation, control engineering, model-based control, material systems

1.INTRODUCTION

Based on the increasing demand for customized and highly-individualized products, but also taking into account shorter product life cycles, automated production systems (aPS) need a new level of flexibility to deal with these requirements. Due to this trend, a single plant has to be able to produce a wide range of different products requiring an adaptable material flow within the plant and control software architecture that is able to accomplish the changing production environment and the associated manufacturing planning and scheduling, which also frequently changes (Dennert et al., 2013). Thus, the demand for a flexible, automated material flow system (aMFS) within an aPS, including flexible adaptable control software, is also increasing. Considering the modification of aMFS, i.e. extending, reducing or modifying parts, focused on present-day software engineering, a control software is commonly reused by means of copying, pasting and modifying existing control code, which involves a propensity for errors and a high degree of effort for the engineering. To reduce the static structure of aMFS as well as the error proneness and engineering effort, a modularization of the software architecture into small encapsulated automated material flow modules (aMFM), e.g. a conveyor belt including peripheral sensors and actuators, is introduced. This allows every aMFM to execute specific tasks and possibly interact with neighboring modules or higher-level systems. Variable assembling of the different aMFM to larger aPS creates flexibility in this architecture. To improve the compatibility of the different aMFM, (pre)defined interfaces are commonly developed between different aMFM. Since these (pre)defined interfaces consider only modules with the same version number, which are commonly constructed with the same state-of-the-art technologies, the interface must be redeveloped, e.g. with different or further variables, for different modules possessing a different version number to interact with one another.

As a result, in the case of a module possessing a lower version number, e.g. a barcode scanner, being exchanged for one possessing a higher version number, e.g. a Quick Response (QR) scanner, firstly the software engineering has to analyze the different interfaces of the modules manually before adapting, i.e. wrapping, the newer module's interface to be backwards compatible and able to interact with the neighboring modules. Both tasks, the analysis of the different interfaces as well as the adaption of the interface, involve a

10.1016/j.ifacol.2017.08.345

high level of effort by software engineering and increases the propensity for errors. Thus, the demand for automatic analysis and adaption of the interface of aMFM to improve backward compatibility is increasing.

A common approach to reducing the effort and decreasing the propensity for errors is the introduction of model-driven engineering (MDE), which is suggested by Bohner and Mohan (2010) as methodology based on the results of their survey. To develop aMFM using an MDE approach, a meta model for automated material flow modules, – the AutoMFM –, which enables the encapsulated description of logistic modules and their related information as well as the composition of multiple modules to a system model is introduced in previous work (Aicher et al., 2016). Applying AutoMFM as modelling language, this paper presents an approach for the automatic analysis and adaption of the interface of the modules to reduce the effort involved in developing backward compatibility for aMFM. Therefore, due to the automatic analysis and adaption of the interface, the changeability of modules possessing a different version number is improved, which is also an essential challenge to realize the current trend of automation and data exchange in manufacturing technologies (Vogel-Heuser and Hess, 2016). Considering the control of aMFM, programmable logic controllers (PLC), which execute IEC 61131-3 compliant code on a cyclical basis, are typically applied. Since the approach presented in this paper should be applicable for industrial systems, the restrictions regarding the limited analysis and adaption of the interface, i.e. architecture of the interface, number of data types or potential methodology for the adaption of the interface, have to be taken into account. To apply the aMFM model instances for different PLC hardware, the code generation tool can be adapted.

The main contribution of this paper is the automated model-based analysis of aMFM possessing a different version number to detect differences, e.g. due to modifications, as well as the control code generation to adapt different interfaces of the modules to enable the "automatic backwards compatibility" capability of aMFM.

This paper is organized as follows: In the next section, the requirements addressed by our approach for wrapping aMFM are specified. Subsequently, in section III, state of the art technologies and approaches are evaluated against these requirements. Based on this analysis, we show our solution for a model-based development and the associated architecture in order to improve backward compatibility of aMFM in section IV. The implementation of our approach using the software Eclipse is presented in section V. An evaluation based on exchanging a barcode scanner for a QR code scanner is shown in section VI. In addition, an experiment with a test person and a survey of experts in the field of material flow systems were conducted to reveal the current situation in industry and to identify the added value of the introduced approach. This paper concludes with a summary and a discussion of results and future work in section VI.

2. REQUIREMENTS

To increase flexibility and enable automatic analysis and adaption of the interfaces of MFM, several requirements regarding system and software architecture have to be considered by the model-based development approach.

2.1 Backward compatibility of automated material flow modules across different versions (R1)

To improve the modifications to aMFS, parts of the system, e.g. single aMFMs, should be easily exchangeable and with less engineering effort involved. Hence, the exchange of an older module with a newer module, which often exhibits different behavior and has a different interface description, which should not affect the collaboration of this module and the neighboring module. Instead, based on backward compatibility, newer modules, i.e. modules with a higher version number, should be able to imitate the interface of the old module. This means that the neighboring modules do not have to be adapted when another module is changed, but the necessary adaption can be limited to the newly-exchanged module. In addition, the interface with higher systems, e.g. a general material flow control system, as well as to global information, e.g. the access to variables stored on shared data blocks, should also be taken into account.

2.2 Centralized control architecture for industrial applicability (R2)

Considering current research on the topic of material flow systems, many approaches to improving flexibility and reusability are based on self-controlling, e.g. agent technology, decentralized/distributed control or autonomy (ten Hompel et al., 2008; Schröppel, Weber and Wehking, 2013; Regulin et al., 2016). These approaches introduce completely new control strategies and paradigms, which could just be applied in fewer, and well-chosen, industrial applications. Existing concerns such as the missing quantifiable detection of the added value for the company as well as missing experiences with regard to these new control strategies and paradigms are the cause of the limited application. In addition, the lack of tool support for developing self-controlling systems further increases the industry's uncertainty regarding the use of agent control systems. Thus, most manufacturers and operators of aMFS do still rely on centralized control strategies, e.g. by using a PLC, controlling whole areas of the system. Since our approach should support the current state of technology in the industry, but not require a complete revolution of the control system, centralized control strategies should be supported in order to improve flexibility and reusability.

2.3 Model-based design of wrappers to adapt the interfaces of aMFM (R3)

To decrease error proneness and handle the complexity, the model-based design of control software is increasingly applied for mechatronic products and plants, e.g. aMFS. A meta model that considers both the module functions and the wrapper characteristics for adapting the interfaces is required to ensure compatibility among several modules. In addition, logistic functionalities, which can be separated into standard functions but also combined to achieve more complex characteristics, have to be supported by the meta model (VDI, 1990;