Computers in Industry 60 (2009) 195–203

Contents lists available at ScienceDirect

Computers in Industry

journal homepage: www.elsevier.com/locate/compind

The performance of product-driven manufacturing control: An emulation-based benchmarking study

Rémi Pannequin *, Gérard Morel, André Thomas

Research Centre for Automatic Control of Nancy (CRAN - UMR 7039), CNRS, Nancy Université, F-54506 Vandoeuvre les Nancy, France

ARTICLE INFO

Article history:
Available online 21 January 2009

Keywords:
Intelligent manufacturing systems
Product-driven control
Benchmarking
Simulation
MAS

ABSTRACT

Product-driven control may enable manufacturing companies to meet business demands more quickly and effectively. But a key point in making this concept acceptable by industry is to provide benchmarking environments in order to compare and analyze their efficiency on emulated large-scale industry-led case studies with regard to current technologies and approaches. In this paper, a benchmarking protocol is defined, in order to provide R&D practitioners with benchmarking services in a product-driven implementation project. A component-based generic architecture is proposed to support this protocol, enabling to model and compare various control architectures. This benchmarking protocol is applied to an automotive-industry case study in order to evaluate the impact of making the products interact with the local decision centers. Finally the experiments show that product-driven control can perform as good as traditional centralized control, and that its robustness depends mainly of the local decision-making processes.

1. Introduction

There is a large consensus in the IMS community between holonic control, production management and virtual enterprises [1] that the combination of agent and infotronics technologies may permit meeting flexibility and adaptability issues as required by the increasing customization of goods and services. Indeed, on one hand, agent technology brings forward new fundamental insights on decentralized coordination and auto-organization, enabling new manufacturing decision-making policies and on-the-fly reconfiguration capabilities (''plug-and-produce''). On the other hand, infotronic technologies address mainly the issue of synchronization between physical objects and their informational representation, which was acknowledged to be one of the major issues in production and inventory management [2].

This holonic-modeling paradigm, mainly developed by the Holonic Manufacturing Systems community [3] can be applied into product-driven control. Product-driven control is a way to exchange the hierarchical integrated vision of plant-wide control for a more interoperable/intelligent one [4], by dealing with products whose information content is permanently bound to their material content and which are able to influence decisions made about them [5]. This approach is applicable at the supply chain level to improve products information management, with applications of the system in tracking and logistics control [6].

Research interest in product-driven control is currently growing, as numerous research projects appear. For instance the European project PABADIS'PROMISE,[1] involving researchers and software vendors, aims at building a product-centered manufacturing execution system, by developing new kind of devices capable of embedding an agent representing a product, and ERP modules and process controllers capable of interacting with these product agents.

Improvements in observation of products, thanks to identification technologies (such as RFID, bar codes, etc.) are therefore driving major changes in the way control architectures are organized. One of the key organizational issues is business to manufacturing (B2M) interaction; holonic products integrating information from both the business and manufacturing point of view are able to solve interoperability issues [7]. They may also make possible to have seamlessly coexisting mature centralized decision systems, such as MRP2 [8] with newer distributed control approaches. Fig. 1 summarizes this novel organization centered on active products.

Nevertheless, as noted in recent work by Marik and Lazansky [9], there is still a long way to go to make these heterarchical architectures efficient in real industrial environments. Among

* Corresponding author.
E-mail address: remi.pannequin@cran.uhp-nancy.fr (R. Pannequin).

[1] Plant Automation Based on Distributed Systems Product Oriented Manufacturing Systems for Re-Configurable Enterprises (http://www.pabadis-promise.org).

0166-3615/$ – see front matter © 2008 Elsevier B.V. All rights reserved.
doi:10.1016/j.compind.2008.12.007

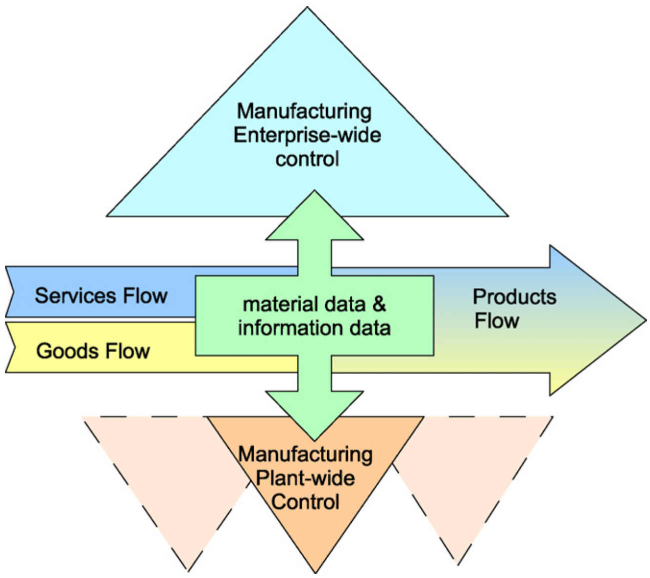

Fig. 1. B2M interaction centered on holonic products [4].

many issues to be solved, embedded devices, as well as agent technologies, are not yet sufficiently reliable and powerful to handle the scalability problems so that decision-making can be fully shared.

Another issue is then to identify the correct balance between centralized and distributed control capabilities of decision-making agents, so that the process can be digitally interactive in both directions, from the operators down to the products and back. A modeling and testing environment for analyzing and comparing alternative decision-making scenarios with traditional ones is required. The experimental feedback on design decisions provided by such a benchmarking environment would permit the firm grounding of product-driven architectures, and could discover new issues.

Therefore, this paper proposes an architecture based on generic components, which can be used to build emulation-based benchmarking models. This environment enables researchers or practitioners to easily build industry-scale executable models of factories in order to test control systems, and to model and execute various organizational schemes of the decision-making system.

After stating in Section 2 the requirements for a benchmarking environment in the context of product-driven control, we will define the proposed components-based architecture; Section 3 deals with building an emulation model of an industrial shop-floor system. This modeling and testing environment is then applied in Section 4 to an industrial case study where a manufacturing line of an automotive industry subsidiary is considered. In Section 5 we address the interpretation of the experimental results; Section 6 presents some perspectives into ongoing works at CRAN for industrial transfer in the area of product-driven logistics for the natural-fiber industry.

2. Problem statement

An empirical approach seems the most feasible way to evaluate new control architectures. By *empirical*, we mean that generic properties of the control architecture are logically induced from concrete applications of the control system to many particular cases. Theoretical approaches, such as evaluation of computational complexity [10] may generate useful results, but cannot cope with the complexity of distributed systems. Indeed, the behavior of distributed systems is often based on the *emergence* of global properties from local behavior and interactions. But mathematically formalizing emergence is not easy [11], and modeling the system as a whole may result in models more complex than the theory can actually solve. So, by using an analytical approach, researchers might have to simplify reality, or to choose only simple situations.

We distinguish three ways for conducting such concrete performance measurements: namely analytical methods such as queuing theory, Monte Carlo methods such as simulation or emulation, and physical experimentation such as lab platforms or industrial pilot implementations.

In the context of product-driven control, analytical methods are impractical because the mathematical models corresponding to a realistic case are often too complex to be solved. Physical experimentation has the disadvantage of technical- and cost-related limitations [12]. Simulation seems the only recourse for modeling and analyzing performances in large-scale industrial cases.

Comparison between antagonistic control modes such as market-based and hierarchical control [13] or planning-based and reactive control [14] have been carried out using specifically developed test beds, but more generic evaluation tools are needed, enabling us to store, share and compare test cases.

The development and definition of such generic evaluation tools has drawn a great deal of interest. An online benchmarking utility has been defined by IMS-NoE special interest group 4 [15,16], and this enables collection and sharing of a wide range of industrial test cases. Until such a generic service—one is under development at KU Leuven—is available, simulation-based benchmarking of complex manufacturing systems remains the way to establish proof of the efficiency of plant-wide-control organizational strategies before their deployment for practical purposes [17].

In fact, there is consensus on the architecture of benchmarking environments putting the emphasis on modularity between the control system *CS* and the shop-floor system *SF*. With these notations, an experimental run consists in verifying the assertion [18] $SF \wedge CS \supset G$, where G is the required performance level. The real shop-floor system may be replaced by a model, an *emulated* shop floor. Likewise, a model of the control system can be used instead of the real one. Therefore, four experimental situations can be defined, using either models or real systems [19].

Nevertheless, modeling issues remain. For instance, it can be difficult to distinguish between the shop floor and control systems. The borderline between the two systems depends of the level of the control functions being tested. Another issue is to build the emulation model itself, as currently available simulation software does not offer emulation-capable modeling components. A third issue is to be able to develop easily used product-driven control systems, in order to compare various architectures or to validate decision-making policies.

This paper aims at solving these modeling issues by defining a modeling and testing component-based framework. Moreover, we propose two experimental steps, to enable iterative development of the product-driven execution system.

In the first step, a model of the control system is applied to the emulated shop-floor system ($SF_m \wedge CS_m \supset G$). Under the hypothesis that product-embedded data are available to decision centers, this approach can be used to define what kind of data should be embedded into products and how decision algorithms should use them. The second step uses the real distributed control system (e.g. a multi-agent system) that controls the emulated process ($SF_m \wedge CS \supset G$). This testing approach enables us to consider software-related issues such as how products and decision-making centers should interact to exchange data.