1

Introduction

Today's fast-changing global markets and decreasing product life-cycles are forcing a philosophical shift in manufacturing industries. The approach to manufacturing is quickly moving away from pure mass production to mass customization. In order to rise to the challenge of these emerging trends, new plant structures and production facility operation methods are necessary. Future production facilities need to be flexible, adaptable and allow fast, changes all at little cost. Today's flexible production systems already provide the mechanisms for producing different products in the same plant. However this flexibility comes with high cost. First of all such plants have higher initial costs compared to standard plants. But what is even more important, they incorporate a high operational cost because of down-times needed for adapting to new products or because of breakdowns in the plant (e.g., continuous processes with long ramp-up times). Therefore, means are necessary that provide flexibility without the need to stop the production.

As early as 1991, a visionary study by the Iacocca Institute (1991) demanded the overall support for reconfiguration of manufacturing plants with their *Agile Manufacturing* concept. The concept included reconfiguration on all plant levels, even the physical reconfiguration of production resources (e.g., adding a new machine or changing a machine's structure). In order to support this concept, the control software needs to be dynamically reconfigurable to both bring about and adapt to the physical changes. Whatever the solution, it must be simple, flexible and have limited space requirements. A too simple approach, for example, changing the parameters of software components, will not suffice, as this leads to large control programs that must contain any possible change in advance. Such large control programs are complicated to develop and maintain.

One of the most compelling reasons for reconfiguration is a change at the production site. The two main changes that require

reconfiguration are a change in the final products or a change in the production resources (e.g., machines). Changes in the products include adding or removing products or changing the volume of a product. The main reason for a change in the production resources is the breakdown or malfunction of production facilities; however, production resources could also be increased or reduced. Resources may be added to a plant in order to increase the overall throughput of the plant. This allows the plant to produce a higher volume of a certain product (e.g., because the market demands more products) or to produce new products. Resources may be reduced in order to respond to reduced orders (e.g., because the market does not accept the products as expected) or due to the removal of a product from the production line. A further reason for reconfiguration, which is not driven by disturbances at the production site, is the replacement of control algorithms in the case of a bug fix or in favor of a more sophisticated algorithm (e.g., add a friction compensating control algorithm in order to accommodate the normal wear to plant equipment).

The physical changes in production resources, such as those described above, means a need for dynamic reconfiguration at the control level. In the last fifteen years, several attempts have been made to turn such systems into reality. Two well-known efforts are the *Holonic Manufacturing Systems* (HMS) project (Deen 2003) and the *Reconfigurable Manufacturing Systems* (RMS) project (Koren et al. 1999). These attempts provided support for the dynamic reconfiguration of production systems at the higher control levels of an automation system. Both of these approaches stop at the higher level control devices. However, in order to support the highly agile systems that we are discussing, the lower level real-time control layer needs to be considered too. Currently, state-of-the-art industrial control systems used for the lower level control layer[1] do not offer sufficient reconfiguration and scalability (Xu et al. 2002). Furthermore, during a plant's full operation, the reconfiguration of control applications is completely ignored at the lower level control layers of industrial control systems. Without the ability to reconfigure the lower level control layers, the existing architectures are simply not designed to support the reconfiguration of control systems during execution time. In order to achieve the real-time reconfiguration demanded by today's changing market place, we need new software architectures and support from the execution environment.

Goal

The goal of this book is to provide an execution environment for the lower level, real-time control layer of automation systems that supports dynamic the reconfiguration of control applications. This includes the execution of real-time constrained control applications and support for changing the control applications while they are executing and controlling the plant. The result is control applications that can be changed without the need to stop the current control application's execution. This work is an important step towards zero down-time automation systems.

Conceptual Approach

In general, there are two main parts a system needs to possess in order to achieve reconfigurability while meeting the requirements of the lower level control layers of industrial control systems: a reconfiguration application and support in the execution environment.

1. A reconfiguration process is conducted by a reconfiguration application (RCA). An RCA is modeled with the same elements as normal control applications except that RCAs use basic reconfiguration services. These services are a set of commands that cause a small reconfiguration step within the control application. The RCA may interact directly with the control application under reconfiguration in order to react to certain control application states.

2. An execution environment ensures that real-time constraints will be met during the execution of control applications. This is a basic requirement in industrial control. As reconfiguration applications are of the same structure as control applications, they will also execute while fulfilling the real-time constraints of the reconfiguration process. This allows the executing of control applications and reconfiguration applications to run side by side without disturbing each other unless the reconfiguration process initiates the disturbance.

The application modeling language is the basis for both the control and reconfiguration parts described above. Industrial control sys-

tems have special requirements that this language needs to meet. The new International Electrotechnical Commission's (IEC) family of standards (IEC 61499) for modeling distributed control systems provides the basis for such a language. The standards are based on the widely adopted standard IEC 61131-3 (1993) and have been designed for distributed flexible control systems, therefore the IEC 61499 family of standards already defines the architectural elements necessary for reconfigurable control systems.

Task Definition

In order to achieve the goal discussed in the preceding paragraphs, the following tasks must be completed:

- Develop a model for the reconfiguration process. This model will consist of a reconfiguration application that interacts with the control application under reconfiguration.
  - Analyze the interaction between the reconfiguration application and the control application.
  - Derive a set of reconfiguration services that allow the reconfiguration application to change the control application. Specifically, derive a minimal set of reconfiguration services that can be implemented on small control devices, but still provides full reconfiguration control for the reconfiguration application.
  - Map the derived set of reconfiguration services to the existing limited reconfiguration support of IEC 61499 and specify the newly needed reconfiguration services.
- Develop a deterministic execution model for IEC 61499 applications.
  - Consider the requirements of the reconfiguration process demanded from the execution environment (e.g., modified execution times of control applications, new control applications, or removed control application parts).
  - Analyze the IEC 61499 execution elements.
  - Map the IEC 61499 execution elements to the execution means used in existing real-time scheduling theory.

  - Design a real-time scheduling method that provides for deterministic execution of event-triggered real-time constrained tasks.

- Analyze the impact of reconfiguration on the control application's execution behavior.

- Apply the results to the open-source IEC 61499 execution environment *Framework for Distributed Industrial Automation and Control* (4DIAC).

- Test the derived execution model with a representative real-time constrained control example and reconfiguration example (i.e., reconfigure the control example).

Notes

[1] Typically these are Programmable Logic Controllers (PLCs).