2

Reconfigurable Process Control Research

2.1 Introduction

In this chapter we collect together a number of different developments which lay the foundations for the distributed, reconfigurable process control (DRPC) approach we are proposing. We begin by positioning process operations within the spectrum of industrial production approaches and in particular provide a contrast between continuous process and discrete manufacturing. (This is important when reviewing existing work in distributed reconfigurable control.) We then examine the evolution of process control and in particular developments which have dealt with reconfigurability challenges and their limitations. The second part of the chapter then goes on to introduce distributed coordination methods in process control and then to provide a comprehensive review of the way in which distributed coordination has been applied in other industrial domains.

2.2 Classification of Manufacturing Systems

Manufacturing industries involve a range of production operations and operating conditions. Based on the physical layout of production processes these can be split broadly into: discrete parts manufacturing (automobile, semiconductor industries) and continuous processes (polymer, pharmaceuticals, petroleum industries).

In a discrete process, the individual parts are produced first using various discrete, loosely coupled operations such as machining, drilling, grinding *etc.* These parts are then pieced together in an assembly line to create the main end-product. Often a large number of parts may be involved (*e.g.*, in a car engine) with parts, being physically stable in nature, can be stored or transferred between lines. Unfinished orders can be pre-empted or transferred for more important orders where facility exists.

Table 2.1. Production control in discrete and continuous processes

	DISCRETE	CONTINUOUS
Physical Layout	Jobshop/flowshop with parallel machines	Line / series of equipment
Objective	Part or job centered	Product or recipe centered
Coupling	Intermediate buffers due to conveyors,AGVs	Tightly coupled with piping network
	Time/schedule based	Product based (non-mixing)
	Stable intermediate forms of parts	Possibly unstable chemistry
Controlled Variables	Due date, arrival time, processing time	Process values / set-points, product quality
Control Freedom	Machine assignment, route flexibility	Equipment operational modes, route Flexibility
Control Strategy	Discrete on-off logic (using PLC)	PID/multivariable control (using DCS/PLC)
Example	Semiconductor, Automobile	Petrochemicals, Polymer

A continuous process instead involves continuous flow of materials (such as bulk chemicals) and utilities through process units interconnected via piping streams. New property values are added to these streams as they pass through process units. Normally, an interim form of the end-product is first produced using one or more reaction operations. The un-reacted raw-materials are then separated and re-used while the interim product is purified and processed to bring into final form. The interim product can be mostly unstable and may not sustain long storage. Therefore, pre-empting or transferring of unfinished orders is not normally possible.

These physical differences between discrete and continuous processes lead to their use of different production goals and control methods as summarised in Table 2.1. In a discrete process, the target is to identify a routing of discrete parts across shop-floor and assign appropriate tasks to machines and define their scheduling. In a continuous process the routing remains normally fixed, and the goal instead is to identify the local operating settings of process units and their combinations across the plant that meet the required quality and throughput of the end-product.

A misconception generally prevails, particularly in the research community, that continuous processes are primarily long-term, steady-state operations. This is strictly not true however. By shortening the range and horizon of operations, a continuous process can be made to behave as discontinuous or discrete as in a batch process. Fig. 2.1 depicts the spectrum of discontinuous operations that can be found in process industries. As Keller & Bryan (2000) note, almost half of the production tonnage in process industry comes from discontinuous processes – the proportion which is only likely to grow in future.

Particularly important to this monograph is the so-called *semicontinuous class* of processes in Fig. 2.1, which similar to a continuous process also involve continuous flow of materials and utilities, however the plants are not operated

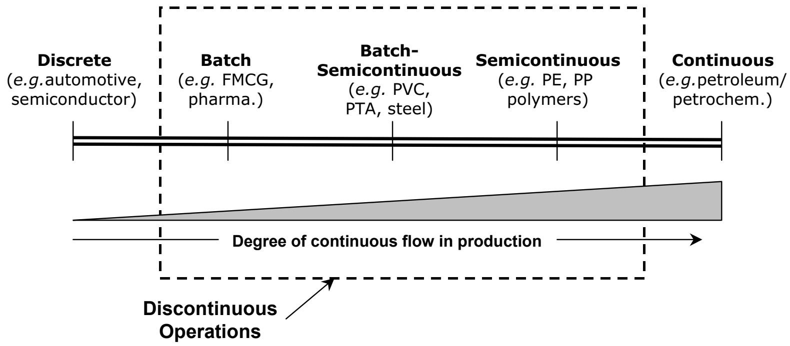

Fig. 2.1. Discontinuous operations in process industry

in a purely steady-state mode. Instead, so-called *campaign* mode of operation is often used (Papageorgiou & Pantelides 1996). The overall planning horizon in a campaign operation is split into multiple product campaigns, each associated with a different product, product grade and/or raw-materials. Subsequently, a campaign for any one product is first produced for a defined period. The production conditions are then changed and a separate campaign for another product is produced using the same set of equipment. Thus, although each campaign operates in a continuous mode, the sequence of campaigns over a certain period results in discontinuity of operations.

The key rationale for the move towards discontinuous operations has been to increase the re-use of equipment particularly when a number of products are to be produced in typically small amounts that do not justify the use of stand-alone plants. The level of re-use required and therefore the plant design may vary depending on the number and type of products to be produced and the variations expected in market demands. In a so-called *multiproduct* design the process is organised such that all products follow the same path and use the same equipment with typically one product produced at a time. In a more flexible *multipurpose* design each product may take one or more distinct processing paths with possibly more than one products produced together where necessary. Multiproduct designs are thus suitable for conditions when the products and processes to be used are known in advance but the quantities or time scales are not, whereas multipurpose designs are suitable when none of these is known and the plant is constructed to contain equipment suitable for certain unit operations with a range of parameters (Mah 1990).

The management of (dis)continuous operations involves a great deal more control operations than purely continuous processes in order to define which products to be produced, when and how. To understand this role of control more clearly, we next discuss the evolution in process control and the structure of modern process control systems.

2.3 Industrial Process Control Systems

The domain of industrial process control encompasses a range of activities to produce products of right quality, type and specification, and importantly, at the right time. To understand how a process control system meets these targets, we discuss in this section a brief history of the field of process control over last few decades. The structure of modern control systems in terms of the information and control functions involved is described later in the section. The final subsection then explores the emerging needs of reconfigurability in process control to put the present work in an appropriate context.

2.3.1 Evolution in Industrial Process Control

Modern process control systems came into existence after various phases of evolution, with each phase having a distinct impression on a particular aspect of the system design. The early designs were governed by then-current business drivers, or indeed the inhibitors such as energy crisis, but in recent years numerous other factors such as IT and communication technologies have played their role in changing the perception of process control in the industry.

The early control systems developed in the 1950's or before were focussed on regulatory control, *i.e.,* the PID controller was the key building block of control. The advent of mechanical and pneumatic devices at this time and subsequently electronic controllers in the early 60's allowed a level of remote control to be achieved but the scope of control was limited to a single or at most few process variables. Coordination of unit systems was mainly the operator's responsibility.

The first major shift in process control occurred in the 1960's when digital circuits and computers were introduced. In a so-called *Direct Digital Control* (DDC) application, a computer was used to replace analog controllers and panelboard displays. This was a pioneering change in control as it allowed advanced strategies, such as sampled data control, to be employed as part regulatory functions. However, the centralised role of computer was a risk of failure with possible complete loss of control. The costs and skills required to deploy computers were also prohibitive (around $30,000-250,000) and proved difficult to justify against simple, PID control in most cases (Smith 1970).

By the early 1970's it became clear that putting computers (or the optimal controllers developed based upon them) in direct control of physical processes is neither convenient nor necessary when a two-level scheme is employed comprising a supervisory (optimising) controller and the bottom-level regulatory controllers. The supervisory computer would focus on key variables (such as reactor yield) and provide regulatory controllers with set points for implementation through analog or digital hardware. Since the computer did not replace underlying hardware, its failure was not critical (Edgar 2004). The use of supervisory control was a conceptual change in the design of control

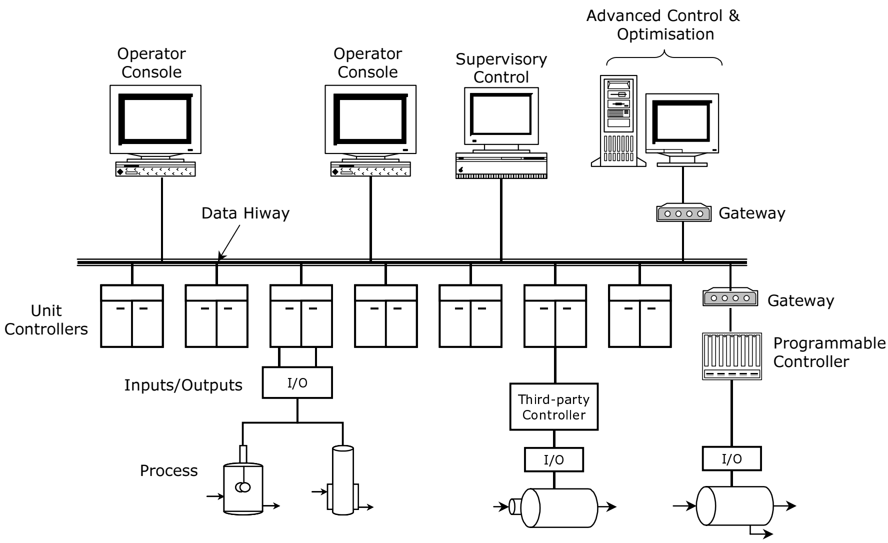

Fig. 2.2. Architecture of a distributed control system (DCS)

system architectures that led to the birth of so-called *distributed control systems* as shown in Fig. 2.2. The significant reduction in size and costs achieved by DCSs and the parallel development of so-called *programmable logic controllers* (PLCs) led to the widespread acceptance of these new architectures in the industry (Samad, McLaughlin & Lu 2007).

A number of events occurred in the 1970's and 80's that changed the perception and hence the structure of process control in the industry. The energy crisis in the mid-70s had a profound influence in this as energy was no longer available cheaply or easily, nor was it easy to sustain long-term demand as new suppliers entered in the markets from countries having access to oil reserves. Sunk with overcapacity and costs many enterprises, particularly in western world, were forced to restructure their businesses. While the economies of scale and scope still remained the dominant means for cutting costs, it became clear that further reduction could only be achieved by reducing the material and energy consumptions and importantly, from making improvements in process control. New control functions such as planning and scheduling, statistical process control, optimisation *etc.*, thus started to take shape as part of the mainstream components of production control and have remained so for the time since then (Chandler 2005).

The period of 1990's saw process control moving one step further from that of managing individual plants to managing enterprise-wide functions. The integration of online enterprise data consisting commercial and financial information with the real-time functions of planning, scheduling and control has become significant (Shobrys & White 2002). Equally significant has been

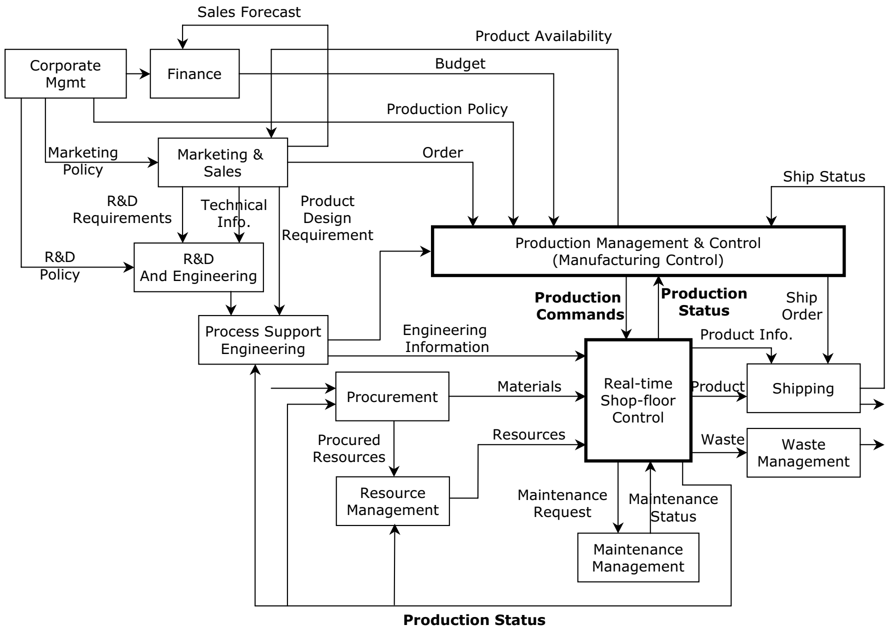

Fig. 2.3. Process control interface to other enterprise functions

the need to create *open* system architectures that enable the system integrators to mix-and-match control components from different suppliers and technologies in a seamless fashion. The adoption of open technologies such as OLE for process control, fieldbus networks and Commercial-off-the-Shelf tools (*e.g.*, based on Microsoft Windows platform) have become the norm in many cases for developing modular system designs that can be rapidly engineered and reconfigured (O'Brien & Woll 2005).

In summary, the past six decades of history have seen process control grow from a primitive regulatory mechanism to a function central to an enterprise that provides the means necessary to deliver the emerging business goals in changing times.

2.3.2 Key Features of Modern Process Control Systems

Today's state-of-the-art process control system includes a variety of tools and techniques to control plant(s) comprising multiple, interconnected unit operations. The control system also interfaces with numerous other enterprise functions shown in Fig. 2.3. In this section, we now discuss the key structural aspects of the modern systems to examine the underlying information flows in coordinating the plantwide operations.

Structure of Process Control Hierarchy

The structure of modern process control systems is based on a hierarchical approach developed as part of wider Computer Integrated Manufacturing (CIM) initiatives in the early 80's. A system hierarchy was preferred as a suitable, and at times a necessary, mechanism to deal with the growing complexity and size of process systems that involved control problems spanning hundreds of variables.

The design of a hierarchical control system has been structured around a *functional* hierarchy that decomposes the business goals defining which products to produce, when and how, to lowest-level set points for regulatory control. Multiple levels of decomposition may be used with each level fixing certain key variables. The implementation of the resulting goals is then carried out via an *aggregation* hierarchy that, in most cases, parallels to the physical decomposition of a plant into its constituent elements (*i.e.,* area, cell, units *etc.*). Fig. 2.4 depicts the two forms of hierarchy using a so-called *Purdue Reference Model* (PRM) employed widely in the industry (Williams 1989).

In both forms of hierarchy, the controllers at successively higher levels cover the larger and broader but relatively slower aspects of overall system behaviour to provide the visibility to global, long-term operations. The decision-time horizon of higher levels also remain longer than those of lower levels. To limit the size of problem formulations, the higher level problem descriptions are generally less structured, involving more uncertainties, than those for lower levels (Mesarovic *et al.* 1970).

The control of production in a hierarchical system under both normal and abnormal conditions is governed by hierarchical communication. When situations are normal, the business goals are propagated to lower levels where decisions at each level are made based on the fixed parameters. When an error occurs, the controller responsible for that level attempts to resolve the uncertainty. If this is not achievable, the higher level controller is invoked to alter the decisions on these fixed parameters. If in turn, the error can still be not resolved, the problem passes up a further level and so on. Hierarchical control thus provides a level of visibility in production operations when conditions are planned and stable, and a level of flexibility in decisions when contingencies arise. Historically, these attributes have underpinned the success of hierarchical control in mass production environment where operations remain largely steady-state and the focus of production control is to economise the production costs through planned, stable, long-term operations.

Information Flow and Coordination

Looking further in detail at Fig. 2.4, the control functions in a hierarchical system can be split into two categories: levels 4 and 3, referred to collectively as *manufacturing control*, and levels 2, 1 and 0, referred to as *real-time*

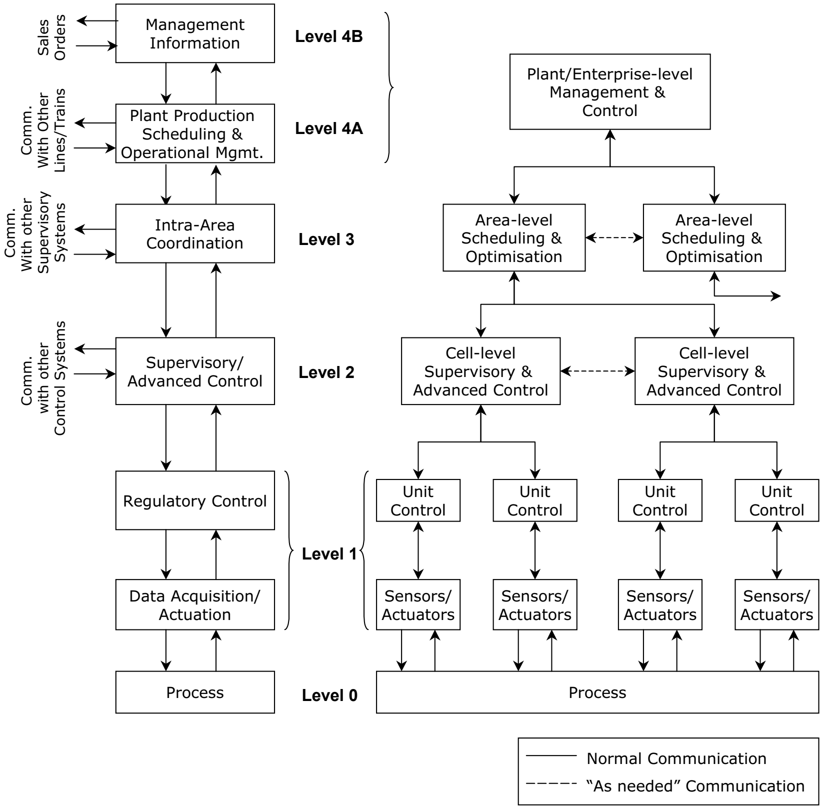

Fig. 2.4. Hierarchical control architecture

control. The manufacturing control levels are responsible for decisions on production management (*i.e.,* which products to produce and when) and the coordination of product flows (*i.e.,* how to produce these products), while the real-time control levels are responsible for executing the outcomes of these decisions onto physical process. These functions also feedback the necessary plant information back to higher levels.

The research in this monograph is mainly focussed on manufacturing control levels, *i.e.,* levels 3 and 4 and the interface between the two. Fig. 2.5 shows the different categories of information involved at these levels, which can be described as follows (adapted from ANSI/ISA 2003):

- *Product Definition Information:* This covers the information on product production rules and the bill of materials and resources required in production. The production rules are abstract and only define how a chemist would produce the product on a laboratory scale, *i.e.,* the information about materials involved in unit operations and their operating conditions.

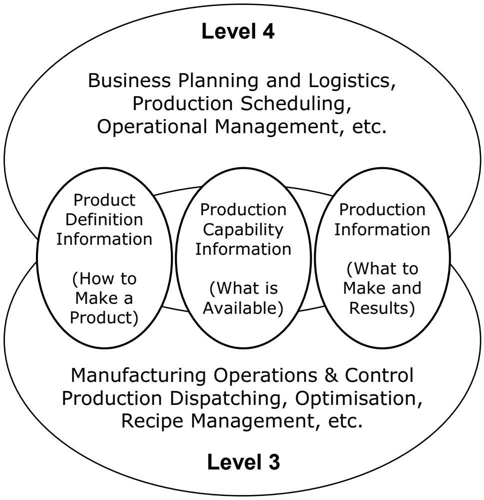

Fig. 2.5. Areas of information exchange at levels 3 and 4

- *Production Capability Information :* This represents the capability information for production resources available on the plant, such as equipment, materials, personnel, energy, consumables *etc.* The information may include details about their design and operational attributes, their current maintenance status and the capacity scheduled for near future.
- *Production Information:* This defines the information necessary to facilitate the actual production. It may cover the areas such as production history, in-process inventory, scheduling of equipment, and the detailed operating procedures *etc.*

The product definition information (how to make a product) is interpreted in terms of the production capability information (what is available) to define the production information (what to make and results). In traditional or legacy systems this integration may be carried out by an operator or a planning and scheduling function operating offline and using spreadsheet tools and/or human knowledge. In more modern systems the standards such as ISA-S88 (ANSI/ISA 1995) and ISA-S95 (ANSI/ISA 2003) are employed to speed up the process and support rapid integration. The key to rapid integration in both standards is the separation of production rules (so-called *recipes* in ISA-S88) from production capabilities of equipment and other resources in the plant. The separation enables creating site-independent, generic recipes that can be deployed across different sites, situated perhaps in different countries and/or having access to different types of resources. As discussed later in Section 2.4, this principle of separating product (recipe) information from that of the processing operations has also been identified as being key for enhancing reconfigurability elsewhere – for example in the holonic and agent-based industrial control research (van Brussel, Wyns, Valckenaers, Bongaerts

& Peeters 1998, Chirn & McFarlane 2001). The separation, in turn, also forms one of the key principles in developing the DRPC approach.

2.3.3 Reconfigurability in Process Control

We now revisit the main topic of this monograph – reconfigurable process control – to understand the incentives for enhancing the reconfigurability of process operations and the factors that characterise reconfigurability in terms of underpinning system requirements.

In a dictionary sense, the term *reconfigurability* of a (computing) system can be defined as its ability to adapt to a new task by altering its configuration (based on Oxford English Dictionary (2005) definition of *to reconfigure*). In the context of production control, reconfigurability then refers to the ability of the control system to adapt to emerging changes (*e.g.*, introduction of new products, processes, raw-materials, utilities, technologies) or disturbances in production operations (*e.g.*, changes in market demands, prices, failure of a process unit, loss or raw-material or utility supplies).

The term *reconfigurable process control* (in short RPC) defines a paradigm in the design of process control systems where reconfigurability forms an essential criteria of the design process. Intuitively, it translates to a facility in the design method with which the control elements can be (i) decoupled, (ii) reorganised, and (iii) recoupled into a new configuration in a possibly smooth and transient manner. The type and nature of reconfigurability required may depend on the ultimate needs of the specific application and the trade-offs that it may have with other design goals. An RPC approach for control design thus provides a layer of additional design decisions that combined with other design criteria and fundamental technical principles should lead to a required level of reconfigurability in the design of control operations.

To develop a new RPC approach, we must therefore understand the motives for introducing reconfigurability in process control. In broad terms, these can be divided into three categories: (i) business needs, (ii) engineer and design needs and (iii) operational needs from end-users.

Business Needs for Reconfigurability

The business needs for reconfigurability emerge from the changing structure of global process industry, *i.e.,* the increased attention on product customisation and globalisation in recent years with a move towards service-centric operations.

As generally true, the process industry sits in the middle of wider supply chains (such as in semiconductor, automotive, consumer goods *etc.*) and faces the impact of technological growth, not just within its own, but also in other industries. With the emergence of new manufacturing technologies and increased pace of technological change (*e.g.*, in electronics industry), the demand patterns of consumers of process industries have been constantly changing. For example, the inventions in mobile phones, computing, audio/visual equipment, home appliances and consumer goods, *etc.*, all nowadays require new varieties of basic products (*i.e.,* polymers, plastic) with additional features, high product quality and better service life. Against this increased variety, the demand for conventional products and commodities has also been sustained or even increased over the past few years as a result of the growing demand from emerging economies in the developing world (Cefic 2006).

However, as Shah (2005) rightly points out, production systems or supply chains in process businesses have yet to catch up with these changing trends or the widening scope of operations. Performance benchmarks for process supply chains generally do not compare well with other sectors (*e.g.*, automotive), for example:

- the stock levels in the chain amount to $30 - 90\%$ of annual demand, with usually $4 - 24$ weeks' worth of finished good stocks in 'pipeline';
- the supply chain cycle times (time elapsed between raw-materials entering and products leaving the chain) tend to lie between 1000 to 8000 hours, of which only 0.3-5% actually involve value-adding operations;
- the material efficiencies tend to be low or below average with only a small proportion of materials entering the chain end up as final products (in case of fine chemicals and pharmaceuticals this figure can be as low as 1-10%).

Clearly, there are incentives to improve here, but large improvements cannot be achieved simply by changing the logistics or transactional processes in supply chains. Rather, some fundamental changes are necessary, particularly at the process and plant level and at the interfaces between various constituents of the value chains (Shah 2005). To a manager responsible for a process enterprise, this means some new challenges for reconfigurable operations:

- shorter product life cycles, with shorter time from innovate-to-market;
- diverse product portfolio with a drive to deliver specialty products at commodity prices;
- enhanced relations with suppliers and customers in global supply chains.

Engineering and Design Needs for Reconfigurability

Even if the business demands of today have been the same as they were twenty years ago, still there are reasons for building reconfigurable process designs from an engineering and design point of view, especially with having the benefits of all technical knowhow gained over the years.

As stated earlier in Chapter 1, the peril of conventional design techniques, both in process systems engineering and control (see, for example, Douglas

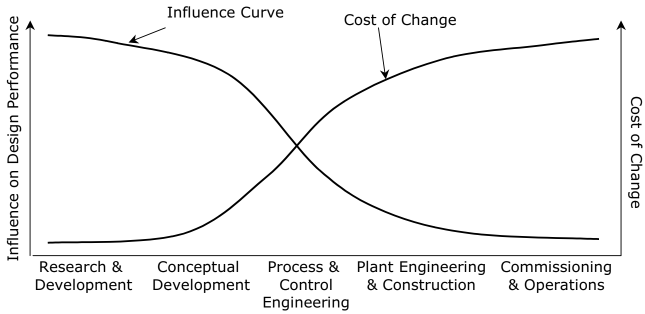

Fig. 2.6. Cost of change across process life cycle

1988, Biegler, Grossmann & Westerberg 1997), comes from their use of a top-down method for scoping the end-user requirements and building from that a conceptual design that forms the basis of further developments. While this approach certainly aids in visibility to the subsequent design phases, the process or control designs built as a basis of conceptual design can become customised and susceptible to change as the design progresses as shown by the 'cost of change' curve in Fig. 2.6. Instead, a combined approach of top-down decomposition of requirements followed by bottom-up integration of standardised components would be preferred as it can support the design modifications at any stage in the life-cycle.

The use of standardised, reusable designs is also preferred to more customised or bespoke designs by the developers of process and control components (Schug & Realff 1996). While customised designs match the requirements of specific applications and incur sale (*e.g.*, in replacing an existing kit), they also need repeating the same design effort and regulatory approval time and expenses. This can be cumbersome in safety or quality critical applications (such as in nuclear, chemicals and pharmaceuticals industries) where standardised designs may be preferred as they can be re-used with shorter lead times and lower engineering costs.

With the increasing pace of technological advances, there also remains a scope for introducing new technologies, *e.g.*, IT and communications, to avoid obsolescence. Often, the new technologies are also more efficient, cheap to procure and easy to build. However, the benchmarks in this case also do not compare well against, for example, to those in automotive or semiconductor industry. In producing chemicals and plastics, the capital and raw-materials cost as much as $50 - 60\%$. Because the plants cost so much, they are usually run for many years and only upgraded when obsolete. This often means lost opportunities. Instead, many lessons can be learnt from experiences in the automobile industry where the use of cheap sensors and on-board computers has transformed motor cars into more comfortable and reliable machines that are also economical to build (Anderson 1997).

In summary, the engineering and design needs for reconfigurability are:

- support for design modifications, during and after the design life-cycle
- use of standardised, re-usable designs with shorter lead-times
- support for technological advances

Operational Needs for Reconfigurability

With increased emphasis on material and energy conservation, it has been a common practice in recent years to design plants with reduced losses, *i.e.,* the use of recycles and heat integration has been norm for a while. While such measures do work in practice and deliver the end results of reduced investment and inventory, they also add up to the operational costs because fluids need to be pumped around constantly. More importantly, they lead to stronger interactions between process units that often cause operational difficulties particularly during transients (Lenhoff & Morari 1982). To maintain satisfactory performance, the plants are hence designed with tighter margins and run in steady-state modes for longer periods. In practical situations, with increased emphasis on product and process variety, the design efficiency can however be a secondary concern. The primary concern instead is to make processes flexible, operable and controllable to handle product/process changeovers or internal and external disturbances such as changes in demands, market prices or arrival of new opportunities (Shah 2005). Many of these require invariably some changes in conventional practices.

On reliability of operations, it has also been a practice to assume that process components are unreliable and that operational upsets are likely to occur, hence redundancy is considered by default (Koolen 1998). Although this helps keep the plants running unattended, it also means the inclusion of spare equipment, devices and sensors. More often this can be avoided if equipment functions are simplified and combined into multipurpose equipment (such as reactive distillation) or broken down into manageable, modular functions that can enhance transparency of operations without compromising on reliability (Schug & Realff 1996).

But, as with any other system, failures do occur, *e.g.*, a process unit fails or becomes bottleneck due to its age or frequent use. Whilst plants or control systems built with redundancy can tackle failures better, there always remains a scope for a level of built-in *fault-tolerance*, *i.e.,* the ability to provide graceful degradation of performance, and where possible, support easy recovery or replacement of failed component. This also is a reconfigurability issue as the losses from a failure or recovering from a failure can sometimes outweigh the cost of the equipment or control system itself.

To summarise, the reconfigurability needs from the perspective of an end-user responsible for operating a process plant are:

- transparent design that is easy to comprehend and operate
- flexible, operable design that supports easy changeover management and disturbance handling

		Diversity	Modifiability	Responsiveness	Fault-tolerance
Business Needs	• Shorter product life-cycles	**✓**	✓	✓	
	• Product customisation & differentiation	**✓**	✓		
	• Enhanced supply chain relations	**✓**		**✓**	
Engineering & Design Needs	• Ease of design modifications		**✓**		
	• Standardised, re-usable designs		**✓**		✓
	• Support for technological advances	**✓**	**✓**		✓
Operational Needs	• Transparent operations		**✓**	✓	✓
	• Support for changeover/disturbance handling	✓	✓	**✓**	
	• Graceful degradation of performance during failures		✓	✓	**✓**

Fig. 2.7. System requirements for reconfigurable process control (the shaded labels show a major link for all four system properties)

- fault-tolerant design with graceful degradation of performance when failure occurs

Summary

Focussing particularly on process control, the reconfigurability needs identified in this section can be summarised into four key system properties as shown in Fig. 2.7 and defined below:

- **Diversity**: The ability to introduce new products and processes including raw-materials, utilities and product recipes;
- **Modifiability**: The ability to support ready integration of new components or the reorganisation of existing components;
- **Responsiveness**: The ability to provide a timely response to product changeovers or disturbances or to adapt to new plant conditions;
- **Fault-tolerance**: The ability to tolerate failures or disturbances and when necessary provide graceful degradation of performance.

While diversity and modifiability are more static properties that concern with the underlying architecture and information flows between control elements, responsiveness and fault-tolerance are both static and dynamic measures and relate to how well the control system is able to cope with dynamic changes, disturbances or failures. We believe a process control system that possesses the above properties should have a high degree of reconfigurability. It is for this reason that we focus this work on distributed coordination methods – which are reviewed next.

2.4 Distributed Approaches in Control

This research presents a distributed approach to reconfigurable process control. In order to understand the rationale for taking such an approach, we now discuss the general concepts behind distributed control approaches developed in the past and in particular, examine in the so-called *holonic manufacturing* and *agent-based* control fields.

2.4.1 Understanding Distributed Control

The concept of *distribution* in control, sometimes referred to as *decentralised* control, is rooted in large-scale and complex systems such as power networks, communication networks, markets and organisations. In such large systems, the standard presupposition for control that *information about the system, or calculations based upon it, are available centrally in a single location* does not often hold. In some cases it may be impossible to collect all information centrally (*e.g.*, in case of markets, the companies may prefer not to disclose their internal details to others) or in other cases the information transfer may have an economic or reliability cost which cannot be ignored (Šiljak 1991). In general though, it remains important that the system is flexible and robust enough to absorb various and sudden changes and be able to accommodate graceful failures in components where a centralised decision system can easily fail (Androulakis & Reklaitis 1999).

A distributed control or decision-making system circumvents this information constraint of a large-scale or complex system by spreading the control calculations or decisions directly to the locations where information exists. The process of distribution generally follows three key principles:

i. *Decomposition:* The overall system is split into multiple subsystems such that variables local to any subsystem are strongly coupled while those among subsystems are only weakly coupled; the term *coupling* here may refer to the impact that a change in any variable has on other variables;
ii. *Local decisions:* Each subsystem is associated with a local decision-making agent or controller that possesses the knowledge of its own subsystem plus at most a partial knowledge of its neighboring subsystems; these local controllers may work towards their individual control objectives or to a team objective or to a combination of both;
iii. *Coordination:* The impact of local actions of any controller on other subsystems is assessed and where necessary, coordinated via some form of communication to solve the local problems or a common, global problem or a combination of both; the communication may be either *direct* (through communication links) or indirect (via observing the perturbations from other subsystems).

Process plants, in one sense, can be perceived as a form of large-scale, complex systems because of their highly interconnected nature. A process control problem, if cast as a computational problem, would exhibit this large-scale behaviour in terms of its model coefficients, *e.g.*, a large number of model elements referring to piping connections between process units would be either small or zero in value. This suggests that a process control problem might be decomposed and solved – in principle – in a similar distributed manner. In modern DCS architectures this assertion has been used – at least partially – to implement the bottom regulatory control level in a distributed form. A similar interest is also growing to distribute the other levels in the hierarchy (see, for example, Camponogara, Jia, Krogh & Talukdar 2002, Lu 2003, Venkat, Hiskens, Rawlings & Wright 2006) and the planning and control problems concerning process supply chains (Perea-López, Grossmann, Ydstie & Tahmassebi 2001).

2.4.2 Solution Techniques for Distributed Control

The solution approaches developed in the past for distributed control – while all follow the above-mentioned three principles – differ in the way the local problems are defined and coordinated across the system. Based on the type of coordination mechanism used for problem solving these can be split broadly into so-called *hierarchical coordination* and *distributed coordination* techniques.

Hierarchical Coordination

In a hierarchical, or so-called *multi-level* scheme, the coordination is achieved by a separate higher level controller. Each local controller receives a freedom to choose its control actions based on its local system model and cost criterion, both derived from a simplification of the overall model and cost criterion. In order that these independently arrived choices are coherent, a separate higher-level controller or so-called *coordinator* is used which incrementally adjusts the individual models or criteria such that the combined cost for the whole system improves. The interactions thus repeat between two levels until a form of convergence is achieved.

Research in hierarchical coordination received wide interest in the 60's and 70's when it was difficult to solve large-scale linear programs using limited computing facilities available then. The first known coordination or so-called *decomposition* algorithm is due to Dantzig & Wolfe (1961) where distribution was used to solve large-scale planning problems via coordination. A *dual* method was suggested therein where the coordinator adjusts Lagrange multipliers or so-called *marginal costs* for coupling constraints associating the local subsystems. Benders (1962) proposed the first *primal* algorithm for linear programs that was later generalised by Geoffrion (Geoffrion 1970, Geoffrion 1972) for wider class of non-linear problems. In a primal scheme the coordinator directly fixes the coupling variables connecting the local subsystems so as to incrementally refine the bounds within which the local controllers can choose their actions. Numerous coordination algorithms and solution techniques have been developed since this early work for applications in operations research and later in systems theory and control engineering. See (Mesarovic *et al.* 1970, Findeisen, Bailey, Brdys, Malinowski, Tatjewski & Wozniak 1980, Jamshidi 1983) for detailed overviews.

Application of hierarchical coordination in process applications has been scattered throughout the years. The early references include (Brosilow & Lasdon 1965, Lasdon 1970, Morari, Arkun & Stephanopoulos 1980). More recently, Katebi & Johnson (1997) considered a dual method for optimising control of chemical processes. Jose & Ungar (2000) applied the so-called *Slack Auction* method to process optimisation where a purpose-built auction mechanism was used to coordinate the interaction variables associated with piping connections between process units. Grothey (2001) proposed a mixed primal-dual technique in a *fixed-and-price* algorithm for more general class of process control problems of nonlinear form. Hou (2001) applied a dual method for coordinating large-scale neural network problems arising in optimal control.

It is worth noting that the above *multi-level* schemes are different than *multi-layer* schemes used in conventional control hierarchy (Fig. 2.4). In a multi-layer scheme, the higher-level controller solves the same plantwide problem, but at an aggregate level, to fix certain key variables. In a multi-level scheme the coordinator is not required to solve any such problem. This has an advantage that modifications required in any part of the system are only made at the local level. The coordinator, being a centralised function, however still poses a threat of single point of failure. Also, the process of coordination is a synchronous process and can be limiting as all local solutions problems must be communicated to coordinator before it can adjust local models or cost criteria. The computational speed of the overall problem can thus be limited by the slowest or busiest local processor among all.

Distributed Coordination

In a distributed coordination scheme, the role of coordinator is removed. Instead the coordination is achieved by the decision-making controllers themselves (called below as *agents*). The agents interact in a distributed mode and are guided by some form of global rule that leads them to converge towards a consensus.

Central to distributed coordination is the information that agents exchange in making local decisions. Agents may not communicate at all and still reach concensus by using some form of min-max strategy of choosing local decisions that satisfy the worst-case physical interactions. Problems of these form have been studied in the fields of *decentralised control* (Šiljak 1991, Sandell, Varaiya, Athans & Safonov 1978) and *game theory* (Basar & Olsder 1995) and applied to large-scale industrial problems (Samyudia, Lee & Cameron 1994, Samyudia, Lee, Cameron & Green 1995, Guo, Hill & Wang 2000). The lack of communication naturally results in a suboptimal global performance.

This can be improved if the agents can be allowed to communicate. In the setting of dynamical control, the agents can be made to communicate various forms of information, for example: (a) the abstraction of their local dynamic models, (ii) the predictions of their future interactions, (iii) the *cost-to-produce* and *cost-to-respond* to incoming and outgoing interactions, *etc.* (Tenney & Sandell 1981). With increased availability and reliability of communication tools, such communication based structures, in particular those based on prediction, have found application in distributing so-called model predictive control calculations (see, for example, Camponogara *et al.* 2002, Venkat *et al.* 2006, Keviczky, Borrelli & Balas 2006).

A large body of work on distributed algorithms that also uses communication as part of problem solving belongs to so-called *relaxation* techniques from optimisation and operations research literature (Bertsekas & Tsitsiklis 1989). In simple terms, the relaxation methods build upon a principle that, if problem structure permits, the optimisation step in a centralised technique, *e.g.*, a gradient step $x(t+1) = x(t) - \gamma \nabla F(x(t))$, can be split and distributed among agents responsible for subsets of variables. The agents iteratively solve their local problems and communicate these local solutions in some form. The overall solution is made to converge by imposing a global constraint such as the order in which their local problems are solved. See (Bertsekas & Tsitsiklis 1989) for an extensive overview of this class of algorithms. The concept of dynamic programming also provides a communication-based method for solving multi-stage problems such as in process synthesis (Jackson 1964b, Jackson 1964a, Rudd & Watson 1968) and process modelling (Kisala, Trevino-Lozano, Boston, Britt & Evans 1987, Westerberg, Hutchison, Motard & Winter 1979, Alkaya, Vasantharajan & Biegler 2000).

An important class of distributed solution techniques based on so-called *nested decomposition* concept have remained dormant over the years (Glassey 1973, Ho & Manne 1974, O'Neill 1976), however, as shown later in this monograph, these can provide an excellent tool for solving distributed control problems arising in multi-stage networks such as process plants. The word *nested* refers to a sequential solution of multiple, two-level coordination problems, each associated with a junction (or link) connecting two or more agents or subsystems. Starting from the root of the network, each agent in the sequence coordinates its own actions plus those of its predecessors and passes relevant information down to its successors. The interactions repeat across the network whereby agents incrementally build and refine abstractions of cost objectives and feasible regions and utilise this information in solving the global problem. See Chapter 6 for further details on nested decomposition.

Discussion

Both coordination methods described above offer improved benefit of reconfigurability over conventional methods because the formulation of local controller problems are distributed and can be easily modified. However, both coordination methods also need a separate mechanism for coordinating the local solutions to guarantee coherent global operations. Historically, coordination is perceived as a complex process difficult to implement within industrial process control due to: (a) the process problems can be complicated due to the use of material and energy recycles and (b) the problem formulations used at higher-levels, *e.g.*, in planning and scheduling problems, remain generally monolithic. The use of coordination in this context for problem solving can lead to slower convergence and may not work reliably due to the reliance placed on communication tools. However, with the advances in communication and computing technologies in recent years, these issues have remained less of a concern nowadays. As discussed earlier in the previous section, if the complexities of recycles and heat integrated are treated secondary to the reconfigurability of operations then the benefits offered by coordination methods, in particular those based on distributed coordination, can provide attractive alternatives for building modular control architectures that also support such rapid integration and reconfiguration (Backx *et al.* 2000, Samad *et al.* 2007).

2.4.3 Distributed Paradigms for Reconfigurable Manufacturing Control

As mentioned earlier, distributed approaches have been used previously in developing greater reconfigurability in distributed manufacturing control. The driver for such development was the business pressures felt by manufacturing industries in the early nineties. The increased attention on product customisation and diversification led to many researchers tackle the problem of manufacturing agility by seeking inspiration from other man-made or natural systems where adaptability to change has been key to their survival. Some examples of new paradigms include *fractal factory* (Askin, Ciarallo & Lundgren 1999), *bionic manufacturing*, (Ueda 1992, Tharumarajah, Wells & Nemes 1996), *holonic manufacturing* (Christensen 1994, Seidel 1994) *etc.* Although motivational and insightful, many of these new approaches failed to make an impact due to their rather radical nature. The two concepts which did succeed namely, holonic and agent-based manufacturing, led to major research interests internationally. We give in this section a brief overview of the research in these fields with an aim to identify the background concepts relevant to this work. More comprehensive overviews can be found in surveys (McFarlane & Bussmann 2000, Mařik, Fletcher & Pěchoucček 2002, Babiceanu & Chen 2006, Shen, Hao, Yoon & Norrie 2006, Shen, Wang & Hao 2006). Industrial deployment of these technologies has been reviewed in (Mařik & McFarlane 2005, Pěchouček & Mařik 2006).

Holonic Manufacturing Systems

The concept of *holon* was proposed by Koestler (1967) in his studies on the evolution in biological and social systems. The word holon, a combination of *holos* (meaning 'whole') and *-on* (meaning 'part'), describes a self-reliant element of a system that is able to exist on its own as an autonomous entity and also is able to integrate with other such elements in the system to create a larger system *i.e.,* a holon demonstrates the dual characteristics of *autonomy* and *co-operation* at the same time. The holonic concept was brought to manufacturing by Suda in his work (Suda 1989, Suda 1990) where he observed that properties analogous of holons in a biological or social system would be desirable in a manufacturing environment when faced with the challenges of customisation and global competition. To motivate the analogy, he proposed the concept of *manufacturing holons* and the associated manufacturing model as *holonic manufacturing systems.* Suda's work led to a number of research efforts promoting the holonic concept as the paradigm for next generation manufacturing systems. The motivation behind these developments was to create a distributed manufacturing architecture that is made up of a modular mix of (semi-)autonomous manufacturing holons that can make stand-alone decisions and are able to collaborate among themselves to produce goods. A bottom-up integration of manufacturing holons, achieved through reconfigurable, distributed interactions is then considered a rational approach to building manufacturing systems of the future.

Agent-Based Manufacturing Control

In parallel to holonic research, the concept of agent-based control also emerged as a paradigm to address similar challenges in manufacturing. An agent, by definition, is a flexible, computational element possessing a level of intelligence to operate independently (Wooldridge 2002). A multi-agent system, comprising multiple interacting agents, is considered to provide the *intelligence* necessary to create a dynamically reconfigurable and to an extent self-organisable design of manufacturing elements.

The agency concepts, while studied previously in computer science, were largely untested in manufacturing and led to bringing together the researchers from holonic and agent communities, with the former providing a physical platform for building agent-based manufacturing systems (Fischer 1999, Mařik *et al.* 2002, Giret & Botti 2004). The concepts of *pro-activeness* and *reactiveness* from agency research are since used widely in holonic and agent research to define the various coordination issues such as communication protocols, decision-making strategies and the planning and scheduling algorithms (Mařik *et al.* 2002, Shen, Hao, Yoon & Norrie 2006).

Holonic and Agent Research in Discrete Manufacturing

The mainstream holonic or agent research, while focussed on discrete manufacturing, has followed the so-called *low and late commitment* principle from the theory of flexibility (Valckenaers & van Brussel 2005), which suggests that to enhance flexibility a designer should commit to a design decision as late as possible and the severity of the commitment should be kept as low as possible, *i.e.,* (a) where possible, the design decisions should be postponed or avoided by providing alternatives and (b) the design process should avoid building "inertia" that makes it harder to rectify the errors at a later stage (Wyns 1999).

In a make-to-order environment, the principle of late commitment has been employed to provide the support for customisation and diversification of products. The concepts of so-called *product holon* and *resource holon* are introduced – the former representing the recipe knowledge on 'how to produce a product' for a specific order and the latter as the production capabilities in terms of machines and other resources available on the shopfloor (van Brussel *et al.* 1998, Chirn & McFarlane 2001, Leitão & Restivo 2006). These two aspects are separated in the design and only integrated during run-time operations via distributed interactions between product and resource holons. By delaying their integration, the developers of the recipe knowledge or the machine control receive a freedom to choose design solutions that best suit the local conditions. Equally, the most recent status of conditions on the shopfloor is taken into account before assigning tasks that fit with the order requirements. As a result new orders can be dynamically introduced or the existing orders shuffled to better utilise the resources.

The principle of low commitment is also extended to engineering and design of control system so as to suggest a method of *top-down decomposition, bottom-up integration.* A bottom-up method is preferred for integration as it avoids the pitfalls of initial global design which can be restrictive (van Brussel *et al.* 1999). In the proposed method, the decomposition of end-user requirements still occurs top-down however little or no design choices are made enroute. Resulting bottom-level requirements from the decomposition are then associated with appropriate holons from a set of pre-identified holon types. Selected holons are then designed and implemented in a bottom-up manner such that their final designs are reusable, preferably of multifunctional nature. To support the identification of holons, a number of different classifications have been suggested in the form of so-called *reference architectures.* Some prominent examples of these include PROSA (van Brussel *et al.* 1998), HCBA (Chirn & McFarlane 2001), ADACOR (Leitão & Restivo 2006) and MetaMorph (Maturana & Norrie 1996, Shen, Maturanan & Norrie 2000). Internal design of holons that supports this architectural research has also received vivid interest. Some key references include (Christensen 1994, Rannanjärvi & Heikkilä 1998, Heikkilä, Järviluoma & Juntunen 1997, Fischer 1999, Brennan, Fletcher & Norrie 2002).

The holons operate in a distributed mode and share information to reorganise their operations and coordinate associated decisions. The functionality of conventional hierarchy is loosened and distributed among holons; holons solve related planning, scheduling and control problems in a distributed form. Development of coordination techniques to support these interactions has formed an essential part of research, not just to define the problem solving mechanisms but also to provide an ontological description of the interactions that are used to standardise the communication protocols used by holons and their internal designs. The key solution concepts considered include contracting (Smith 1980), lagrangian decomposition (Gou, Luh & Kyoya 1998), market programming (Váncza & Márkus 2000) and behaviour-based techniques (Valckenaers, van Brussel, Kollingbaum & Bochmann 2001, Tharumarajah & Wells 1996). Associated applications in control cover holonic planning (Deen 1993), scheduling (Gou *et al.* 1998, Sousa & Ramos 1998) and execution control (Heikkilä *et al.* 1997). See (McFarlane & Bussmann 2000, Tharumarajah 2001, Shen, Hao, Yoon & Norrie 2006) for recent overviews.

Holonic and Agent Research in Process Applications

Research on holonic or agent-based based systems or similar principles has been scarce in the process industry. One of the early interests was in agent applications to support design and engineering of process plants purely to perform mundane tasks such as collecting the data and checking different design alternatives. (Jennings, Faratin, Norman, O'Brien, Odgers & Alty 2000, Batres, Asprey, Fuchino & Naka 1999). More technical use of agents has been found in distributed fault diagnosis (Seilonen, Appelqvist, Halme & Koskinen 2002, Eo, Chang, Shin & Yoon 2000, Maturana, Tichý, Slechta, Staron, Discenzo & Hall 2003). The agents here represent and monitor one or more pieces of equipment. During a fault scenario, they build and postulate possible hypothesis of the fault scenarios and communicate results to eliminate unlikely possibilities. Ultimately they recognise the nature and extent of the fault and advise the operator of potential remedies for repair. On a different front, Chokshi, Matson & McFarlane (2000) considered a holonic framework for batch re-scheduling in a steel-making. The concept of *partial global planning* (Durfee & Lesser 1991) was considered from distributed AI research to define the coordination of start and end-times of batch tasks and the movement of *ladles* between unit operations.

More recently, agent-based research has found a surge of interest in the coordination of process supply chains. Among them the key references include (García-Flores, Wang & Goltz 2000, Julka, Srinivasan & Karimi 2002, Julka, Karimi & Srinivasan 2002, Gjerdrum, Shah & Papageorgiou 2001, Aldea, Bañares Alcaántara, Jiménez, Moreno, Martínez & Riaño 2004). Backx *et al.* (2000) gave an interesting insight on the need for *intentionally dynamic, supply-chain conscious* process operations. They showed that a decentralised design of process plants operating in a so-called *cooperative* mode will be essential to support the future requirements. Their initial results defining the control algorithms for market-oriented optimisation and scheduling of process operations are reported in (Tousain 2002, Tousain & Bosgra 2006).

Diversity	Modifiability	Responsiveness	Fault-tolerance	Property	
✓	**✓**	✓	✓	• Modular, multipurpose, re-usable design	Architectural Properties
✓	✓	✓		• Separation of product recipe information	
✓	**✓**		**✓**	• Top down decomposition, bottom up integration	
✓	✓	**✓**	✓	• Low and late commitment	Operational Properties
✓		**✓**	✓	• Dynamic integration of product recipe information	
✓		**✓**	**✓**	• Distributed decision-making and control	
✓		**✓**	**✓**	• Proactive and reactive behaviour	

Fig. 2.8. Satisfaction of reconfigurability requirements using a distributed approach (the shaded labels show a major link)

2.4.4 Summary

Fig. 2.8 summarises the key properties of distributed approaches in holonic and agent research by linking them with the reconfigurability requirements in Fig. 2.7. As can be seen, the architectural properties can address the static requirements of product/process diversity and easy modifiability, while the operational properties can address the dynamic requirements of responsiveness and fault-tolerance and also help improve the diversity via dynamic integration of product information.

2.5 Reconfigurable Control Research in Other Domains

The concept of reconfigurable control based on distributed approaches has also been studied in domains other than manufacturing, particularly where it remains impossible to employ a centralised control structure. A brief review of this related research is presented in this section to gain insights on the nature of approaches used therein to attain reconfigurability.

2.5.1 Formation Control of Robots or Aircraft

Maintaining a formation of multiple robots or aircraft operating in a close proximity has gained interest recently in areas where unmanned operations are essential (Egerstedt & Hu 2001, Beard, Lawton & Hadaegh 2001, Giulietti, Pollini & Innocenti 2000). Typical of such applications include exploration of unknown environments, coordinated path following and pushing objects in a coordinated fashion. The formation may be time-varying and may be subjected to various hard or soft constraints, such as retain minimum distance between robots or aircraft.

The use of multi-agent control schemes based on coordination have become popular in this domain primarily because the environmental stimulations in which the distributed entities operate remain unknown *a priori*. Beard *et al.* (2001), for example, classified the coordination approaches used into three categories: (i) *leader-following*, where all agents (*i.e.,* robots or aircraft controllers) follow the path of a common leader agent; (ii) *behavioural*, where the group behaviour emerge from the localised behaviour of all agents and (iii) *virtual structure*, where the formation is treated and controlled as a single structure, which in turn directs the actions of the individual agents. See (Beard *et al.* 2001) for further details.

2.5.2 Congestion Control in Communication Networks

With ever increasing use of internet and communication technology, the control of traffic management in communication networks has become important. The problem is further complicated because of uncertainties in the time at which traffic may arise or the amount of network resources that it may demand (Kelly, Maulloo & Tan 1998). One problem in traffic management is *flow control* – for a given network configuration, adjust the incoming traffic such that the network utilisation is maximised. The other problem is *routing* – for a given network configuration and utilisation level, determine the routing of data packets across the network such that the priority constraints (*e.g.*, importance of certain data over others) are satisfied.

Two streams of solution strategies have evolved over the years for these two problems. One stream assumes that individual users are self-maximising agents and aim to maximise their utility for a given shared access of the network. The concept of non-cooperative game theory (Basar & Olsder 1995) is used to characterise the resulting equilibrium conditions for the solution. The properties such as fairness, efficiency of utilisation and quality of service are studied here (Korilis & Lazar 1995, Korilis, Lazar & Orda 1997, Altman, Başar & Srikant 2002, Orda, Rom & Shimkin 1993). The other stream takes a control-theoretic view where the aim of the study is the stability of the equilibrium in the presence of feedback delays arising between user/source pairs. The metrics such as convergence, capacity tracking and robustness to changing dynamics are studied to define the distributed control laws for traffic management (Kelly *et al.* 1998, Vinnicombe 2000, Johari & Tan 2001).

2.5.3 Power Systems and Electricity Markets

Increasing competition has led to many electricity markets being deregulated worldwide. Under new trading rules, individual generators and consumers submit their bids for supply or demand of electricity to a common regulator. The regulator evaluates the bids based on forecast demand and decides a market clearing price at which the electricity is traded. Since the generators and consumers operate as self-utility maximisers, the concept of non-cooperative game theory provides a right platform to study the equilibrium pricing and establish trading mechanisms to reach equilibrium for a given structure of grid and transmission capacity (Stothert & MacLeod 2000, Green & Newbery 1992, Kleindorfer, Wu & Fernando 2001, Hobbs, Metzler & Pang 2000).

Power networks also face the problem of responsiveness against faults and disturbances. Similar to process plants, the grid connections between individual generators or consumers introduce tight coupling between their local processes. A minor or small fault in one part of the network can, as a result, propagate to other parts or the whole network if not properly managed in time. A prompt diagnosis and isolation of fault thus remains ever so important, but the ability of remaining generators or consumers to compensate for this grid imbalance also is equally important to avoid blackouts. It is however impossible to manage this problem centrally due to large size of the networks in most cases. Instead, the concept of decentralised control has been used frequently as discussed earlier in the review of distributed coordination literature (see, for example, Guo *et al.* 2000).

2.5.4 Supply Chain Management

Research in supply chain management and control has flourished in recent years due to increased attention on customisation and diversification in global markets (Maloni & Benton 1997, Tayur *et al.* 1999, Strader, Lin & Shaw 1998). The supply chains nowadays are required to respond and adapt to constantly changing conditions. Their conventional monopolistic form cannot however realise this level of response due to fixed and rigid structure. Instead supply chains are now regarded as supply chain networks (SCN) – an integration of multiple supply chains that evolve and scale according to changing needs of the market (Fox, Barbuceanu & Teigen 2000).

Specifically, the concept of so-called *virtual enterprise* (Strader *et al.* 1998, Camarinha-Matos *et al.* 2003) has emerged. In a virtual enterprise multiple equal-interest companies come together to form a chain that can exploit the fast-changing market opportunities. Each alliance is formed and operated via distributed interactions between companies. Once the opportunity ceases, the alliance is dissolved and the companies move towards forming new partnerships. The effective operation of supply chains, in particular virtual enterprises, requires sharing information between partners and synchronising their local operating policies. The multi-agent technology in this sense has provided a platform for modelling the underlying distributed interactions. See (Chaib-draa & Müller 2006) for a collection of recent references.

2.5.5 Discussion

When considering reconfigurable process control, many lessons can be learnt from developments in the above and other domains. Similar to process plants, in all four domains described above the agents are subjected to hard or soft network constraints. In formation control, robots or aircrafts must maintain a fixed distance. In communication networks, the capacity of network links may limit the data that the users can put on the network. In supply chains, companies remain connected via transport routes and the operating policies they use also need to fit with those of immediate customers, suppliers and transporters. Similarly, in all four domains, the agents must also maintain a stable operation of the global system under time-varying conditions. In formation control or supply chains, the behaviour emerges via co-operation between distributed entities, while in communication or power networks this is enforced by the need for reaching a system-wide equilibrium. Note that in all four domains these static or dynamic properties of the global system emerges via direct, bottom-up interactions between distributed agents.

The research in supply chain networks is particularly relevant to this work. Supply chains exhibit a multi-stage character of commodity flow which – in a sense – is similar to the flow of materials in manufacturing systems comprising network constraints such as process plants. The notions such as 'product', 'product demand', 'customer order' as viewed in a manufacturing system also relate to supply chains in a similar manner. Interestingly, the supply chain paradigm also extends the market or contracting approach used in previous holonic or agent research by introducing the network interactions of 'supplier-to-supplier' type apart from 'customer-to-supplier' type in a market or contracting approach. As discussed in the next chapter, this extension provides the basis for our distributed approach to reconfigurable process control.

2.6 Summary

This chapter has built a foundation for understanding existing work in process control, distributed control and coordination and the role of reconfigurability in this domain. We next move onto the main body of the monograph which proposes a distributed approach to reconfigurable process control.

http://www.springer.com/978-1-84800-059-9

A Distributed Coordination Approach to Reconfigurable Process Control

Chokshi, N.; McFarlane, D.

2008, XI, 192 p., Hardcover
ISBN: 978-1-84800-059-9