PERGAMON

Robotics and Computer Integrated Manufacturing 16 (2000) 29–42

Robotics and Computer Integrated Manufacturing

www.elsevier.com/locate/rcim

Object-oriented and distributed approach for programming robotic manufacturing cells

J. Norberto Pires[a,*], J.M.G. Sá da Costa[b]

[a]*Mechanical Engineering Department, University of Coimbra, 3030 Coimbra, Portugal*

[b]*Mechanical Engineering Department, IST - Technical University of Lisbon, Av. Rovisco Pais, 1, 1096 Lisboa Codex, Portugal*

Received 15 December 1998; received in revised form 8 April 1999; accepted 23 April 1999

Abstract

Flexible manufacturing systems (FMS) are essential for small/medium batch and job shop manufacturing. These types of production systems are used to manufacture a considerable variety of products with medium/small production volumes. Therefore, the manufacturing platforms supporting these types of production must be flexible and organized in flexible manufacturing cells (FMC). Programming FMCs remains a difficult task and is an actual area of research and development. This paper reports an object-oriented approach developed for FMC programming. The work presented was first thought for application in industrial robot manipulators, and later extended to other FMC equipments just by putting the underlying ideas in a general framework. Initially, the motivation for this work was to develop means to add force control to a standard industrial robot manipulator. This problem requires remote access to the robot controller, remote programming and monitoring, as also is required to program and monitor any other FMC equipment. The proposed approach is distributed based on a client/server model and runs on Win32 platforms, i.e., Microsoft Windows and Windows NT. Implementation for the special case of industrial robot manipulators is presented, along with some application examples used for educational, research and industrial purposes. © 2000 Elsevier Science Ltd. All rights reserved.

Keywords: Robotics; Object-oriented programming; Flexible manufacturing systems; Flexible manufacturing cells and CIM

1. Introduction

Actual manufacturing systems are evolving rapidly to flexible systems. Hard Automation manufacturing systems, composed by highly productive and dedicated machines, are not suitable for today's manufacturing platforms. Today the enormous diversity of products along with the requirement for better quality at lower prices makes the product life cycle very short. This is incompatible with hard automation manufacturing systems.

One of the most recent developments in the area of industrial automation is the concept of flexible manufacturing systems (FMS). These are highly computerized systems composed of several types of equipments, usually connected through a local area network (local network using MAP[1] protocols [1]) under some hierarchical computer integrated manufacturing (CIM) structure [2–4]. The factory (*shop floor*) equipments are organized in flexible manufacturing cells (FMC) with transportation devices connecting the FMCs. In some cases, functionally related FMCs are organized in flexible manufacturing lines (FML) (Fig. 1). Each FML may include several FMCs with different or equal basic capabilities. The organization proposed in Fig. 1 is a hierarchical structure [3,5] where each FMC has its own controller (cell level). Therefore, if the manufacturing process is conveniently organized in FMLs we will have several controllers in the shop floor level, e.g., one controller for each FML (process level). With this setup, an intelligent and distributed job dispatching and awarding mechanism may be implemented taking advantage of the installed industrial network [3,6–8].

The essential factor of an FMC is its flexibility, i.e., its adaptability to new manufacturing requirements that can

* Corresponding author. Tel.: + 351-39-7000758; fax: + 351-39-7000701.

E-mail addresses: norberto@dem.uc.pt (J.N. Pires), msc@gcar.ist.utl.pt (J.M.G. Sá da Costa)

[1] Manufacturing automation protocol (MAP).

0736-5845/00/$ - see front matter
PII: S0736-5845(99)00039-3

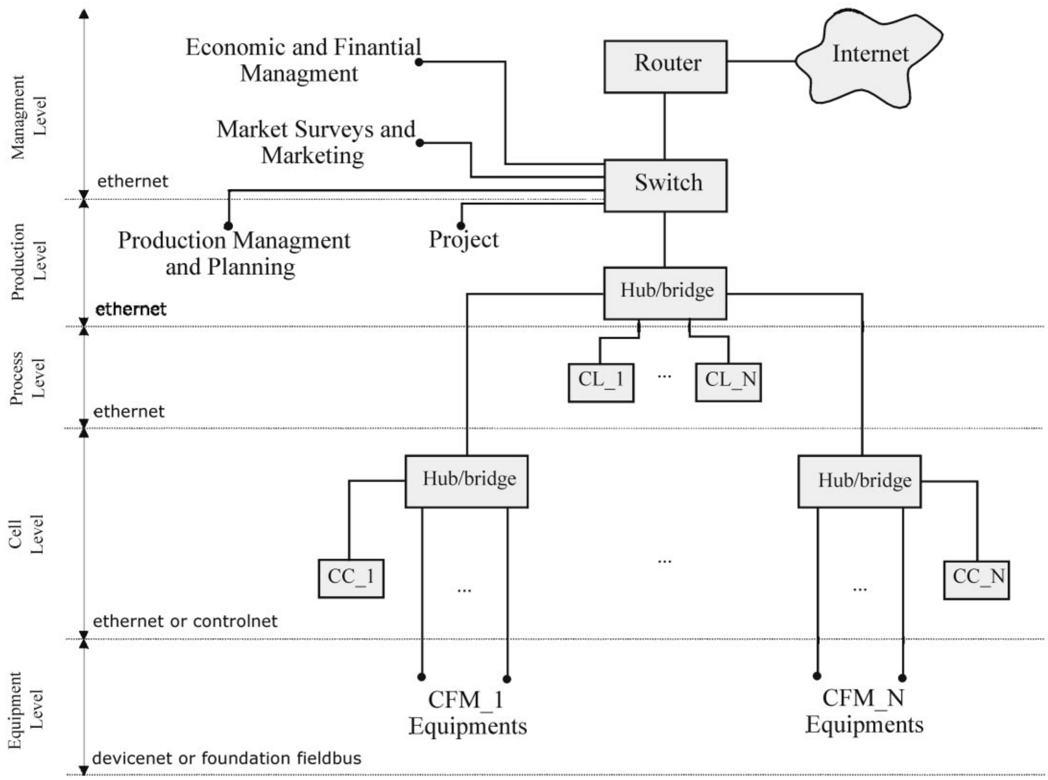

Fig. 1. Example of an industrial network organized in hierarchical levels (shop floor control structure is presented in detail). The industrial network presented is conveniently segmented (using bridge HUBs) maximizing communication throughput in each segment.

go from a modified product to a completely new product. The flexibility results from the fact that FMC equipments are programmable and easily reconfigured machines: this is the case of industrial robot manipulators, mobile robots for parts handling and transportation, programmable and logic controllers (PLC), CNC machines, vision systems, conveyors, etc.

Nevertheless, programming an FMC remains a difficult task mainly because usually FMC equipments come from different manufacturers, having their own programming language and environments. A specialist is then needed for cell programming, even if the required changes necessary to meet new manufacturing demands are only simple adaptations in the original program. This reduces significantly the FMC flexibility, i.e., its potential flexibility is only barely used. In this paper, we present an object-oriented approach for robot programming, monitoring and control. The proposed software architecture may be extended and applied to other FMC equipments, and is an alternative approach to the ones presented in the literature. These approaches usually are based on the development of a unique language for FMC programming, having specific interpreters to convert the code to each equipment native language [9]: this is complex and reduced to a limited number of manufacturers. Other object-oriented approaches showed that it is possible to have a flexible programming environment, and still program each equipment using its native language [10,11]. We consider here that for any individual equipment we can define a set of complex functions, which include all the tasks that would require the equipment when inserted into an FMC. We also consider that these functions can be requested remotely by the FMC controller, which provides also the proper parameterization.

The approach presented here is basically a collection of software controls and a communication mechanism between the host computer and the controllers of the FMC equipments. From among the several available technologies, either for software controls or communications, we selected the ones that could serve better our needs. The basic idea was to provide means for easy equipment programming and monitoring, hiding from the user all the less important details of how to address individual equipments, how to communicate, where to collect data from, etc. All of these are encapsulated into software controls (objects) that provide to the user the properties and methods necessary to do the job. For this we use actual standards, and did not tried to define a new distributed platform (as the European consortium OSACA [12], although the approach is similar), because we wanted to run the architecture from the usual PCs using standard tools.