Studies in Computational Intelligence 694

Theodor Borangiu
Damien Trentesaux
André Thomas
Paulo Leitão
José Barata Oliveira *Editors*

Service Orientation in Holonic and Multi-Agent Manufacturing

Proceedings of SOHOMA 2016

Springer

Studies in Computational Intelligence

Volume 694

Series editor

Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, Poland
e-mail: kacprzyk@ibspan.waw.pl

About this Series

The series "Studies in Computational Intelligence" (SCI) publishes new developments and advances in the various areas of computational intelligence—quickly and with a high quality. The intent is to cover the theory, applications, and design methods of computational intelligence, as embedded in the fields of engineering, computer science, physics and life sciences, as well as the methodologies behind them. The series contains monographs, lecture notes and edited volumes in computational intelligence spanning the areas of neural networks, connectionist systems, genetic algorithms, evolutionary computation, artificial intelligence, cellular automata, self-organizing systems, soft computing, fuzzy systems, and hybrid intelligent systems. Of particular value to both the contributors and the readership are the short publication timeframe and the worldwide distribution, which enable both wide and rapid dissemination of research output.

More information about this series at http://www.springer.com/series/7092

Theodor Borangiu · Damien Trentesaux
André Thomas · Paulo Leitão
José Barata Oliveira
Editors

Service Orientation in Holonic and Multi-Agent Manufacturing

Proceedings of SOHOMA 2016

Springer

Editors
Theodor Borangiu
Faculty of Automatic Control
and Computer Science
University Politehnica of Bucharest
Bucharest
Romania

Damien Trentesaux
LAMIH UMR CNRS 8201
University of Valenciennes
and Hainaut-Cambrésis
Valenciennes
France

André Thomas
École Nationale Supérieure des
Technologies et Industries du Bois
University of Lorraine
Nancy
France

Paulo Leitão
Campus StaApolónia
Polytechnic Institute of Bragança
Bragança
Portugal

José Barata Oliveira
Departamento de Engenharia Electrotécnica
Universidade Nova de Lisboa
Caparica
Portugal

ISSN 1860-949X ISSN 1860-9503 (electronic)
Studies in Computational Intelligence
ISBN 978-3-319-51099-6 ISBN 978-3-319-51100-9 (eBook)
DOI 10.1007/978-3-319-51100-9

Library of Congress Control Number: 2016960765

Printed on acid-free paper

This Springer imprint is published by Springer Nature
The registered company is Springer International Publishing AG
The registered company address is: Gewerbestrasse 11, 6330 Cham, Switzerland

Contents

Digital Transformation in Service and Computing Oriented Manufacturing

Theodor Borangiu, Damien Trentesaux, André Thomas, Paulo Leitão and José Barata Oliveira

This volume gathers the peer reviewed papers which were presented at the sixth edition of the International Workshop "Service Orientation in Holonic and Multi-agent Manufacturing—SOHOMA'16" organized on October 6–7, 2016 by the New University of Lisbon, Portugal in collaboration with the CIMR Research Centre in Computer Integrated Manufacturing and Robotics of the University Politehnica of Bucharest, Romania, the LAMIH Laboratory of Industrial and Human Automation Control, Mechanical Engineering and Computer Science of the University of Valenciennes and Hainaut-Cambrésis, France and the CRAN Research Centre for Automatic Control, Nancy of the University of Lorraine, France.

T. Borangiu (✉)
Department of Automation and Industrial Informatics,
University Politehnica of Bucharest, Bucharest, Romania
e-mail: theodor.borangiu@cimr.pub.ro

D. Trentesaux
LAMIH, CNRS UMR 8201, Université de Valenciennes et du
Hainaut-Cambrésis, Lille, France
e-mail: damien.trentesaux@univ-valenciennes.fr

A. Thomas
CRAN, CNRS UMR 7039, Université de Lorraine, Nancy, France
e-mail: andre.thomas@univ-lorraine.fr

P. Leitão
Artificial Intelligence and Computer Science Laboratory,
Polytechnic Institute of Bragança, Porto, Portugal
e-mail: pleitao@ipb.pt

J. Barata Oliveira
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa,
Caparica, Portugal
e-mail: jab@uninova.pt

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_1

The main objective of the SOHOMA workshops is to foster innovation in sustainable manufacturing and in this context to promote concepts, methods and solutions addressing trends in the service orientation of agent-based control technologies with distributed intelligence and the integration with enterprise management.

The book is structured in eight parts, each one grouping a number of chapters describing research in actual domains of the digital transformation in manufacturing and trends in future service and computing oriented manufacturing control: *Part 1*: Cloud and Cyber-Physical Systems for Smart Manufacturing, *Part 2*: Reconfigurable and Self-organized Multi-Agent Systems for Industry and Service, *Part 3*: Sustainability Issues in Intelligent Manufacturing Systems, *Part 4*: Holonic and Multi-agent System Design for Industry and Service, *Part 5*: Should Intelligent Manufacturing Systems be Dependable and Safe?, *Part 6*: Service-oriented Management and Control of Manufacturing Systems, *Part 7*: Engineering and Human Integration in Flexible and Reconfigurable Industrial Systems, *Part 8*: Virtualization and Simulation in Computing-oriented Industry and Service.

These eight evolution lines have in common concepts, methodologies and implementing frameworks for **Digital transformation in service and computing oriented manufacturing**.

By defining an Internet-scale platform for networked production encapsulating the right abstractions to link effectively and scalably the various stakeholders of the Manufacturing Value Chain (materials and component producers, manufacturing plants, technology providers, services, and integrators) the actual vision and initiatives about developing generic architectures and core technologies for the Digital Transformation of Manufacturing (DTM) are presented in the research reported in the workshop, and included in the present book.

An effect of this initiative is shifting from good-dominant logic to service-dominant logic (also expressed as Product-Service Extension] which enhances the utility of product delivered to customer, e.g., installing, configuring/tuning, training the customer, repairing, maintaining/upgrading) adding value in customer operations.

The global vision for DTM refers to: (i) Pervasive instrumenting manufacturing resources, materials flows and environments, (ii) Interconnecting orders, products and resources in a secured Industrial Internet of Things, (iii) Taking smart decisions in production management and control by distributing intelligence among multi-agent systems (MAS) acting as information counterparts of physical assets, and orchestrating production workflows as manufacturing services in Service Oriented Architectures (SOA).

New developments are related to the digital transformation of manufacturing; they are described in this book.

Cloud manufacturing (CMfg), one of these new lines, has the potential to move from production-oriented manufacturing processes to customer- and service-oriented manufacturing process networks, e.g. by modelling single manufacturing assets as services in a similar way as SaaS or PaaS software service solutions. In CMfg all manufacturing resources and abilities for the manufacturing life cycle can be provided in different service models. The Industrial IoT (IIoT) integrated in the cloud allows creating novel network architectures seamlessly integrating smart connected objects, and distinct cloud service providers. The IIoT represents a core enabler for product-centric control and increasing servitization: the product directly requests processing, assembly and materials handling from available providers while it is in execution and delivery. The product monitors its own status; notifies the user when something goes wrong; helps the user to find and access the necessary product-related models and information from manufacturers in the CMfg ecosystem, and eases the synchronization of product-related data and models.

To achieve high levels of productivity growth and agility to market changes, manufacturers will need to leverage Big Data sets to drive efficiency across the networked enterprise. A number of contributions formulate proposals for a framework allowing the development of Manufacturing Cyber Physical Systems (MCPS)—ICT systems (sensing, actuating, computing, communication) embedded in physical objects, interconnected through several networks including the Internet, and providing businesses with a wide range of innovative applications based on digitalized data, information and services. MCPS include capabilities for complex event processing and Big Data analytics, which are expected to move the manufacturing domain closer towards digital- and cloud manufacturing within the Contextual Enterprise.

Projects are described related to the German governmental vision Industry 4.0 to support processes connected with the industrial revolution based on CPS: enterprise integration, networking, and digital engineering. The general goals of this initiative are: efficient control of complex distributed systems as a society of autonomous units; integrating the virtual world (where each physical element including sensors, products, human operators, robots and machines is represented by a SW unit/information counterpart) with the physical world (CPS); optimize decision making and efficiency (such as cost effectiveness, high performance and energy efficiency); new business models and approaches to value creation (service orientation, product-service extension, Direct Digital Manufacturing, CMfg …).

A brief description of the book chapters follows.

Part 1 reports recent advances and on-going research in *Cloud and Cyber-Physical Systems for Smart Manufacturing*. The contributions point at: design of High Availability (HA) CMfg systems integrating distributed MES agents; an analysis framework for the classification of CPS development solutions; formal modelling of distributed automation CPS with CP-agnostic software; integrating advanced data analytics into industrial CPS; the gap analysis on research and innovation for MCPS; development of a redundant and decentralised directory facilitator for resilient Plug and Produce CPS.

Part 2 includes papers devoted to *Reconfigurable and Self-organized Multi-Agent Systems for Industry and Service*. Multi-agent Systems represent the backbone of the distributed intelligent control in manufacturing; however, additional techniques and tools are needed for enhancements: Big Data, ontologies, simulation tools, security modules; predicting the unexpected; fault tolerance and high availability. SOA—dual to MAS, is used more and more as implementation of MAS in different layers, providing interoperability between enterprise businesses, MES and shop floor layers. The included papers describe: a self-organising model for mobile robots in large assembly structures using MAS; specification of self-organising logistics systems in terms of openness, intelligence and decentralised control; generic reconfigurable and pluggable material handling system based on genetic algorithm; principles of smart condition-based maintenance for a fleet of mobile entities.

Part 3 introduces *Sustainability Issues in Intelligent Manufacturing Systems*, analysing the greenness dimension of products, manufacturing processes and systems to maintain equilibrium between economic, social and environmental requirements and constraints. The following subjects are developed: definition of emerging key requirements for future energy-aware production scheduling systems in the MAS and holonic perspective; MAS framework for manufacturing sustainability and optimization; solution for data mining of energy consumption in manufacturing environment; modelling cybersecurity and resilience for software-defined networks-based manufacturing applications.

Part 4 includes recent research in the area of *Holonic and multi-agent system design for industry and service*. Research activities reported in the field focus on holonic and MAS control frameworks that integrate predictive, proactive and reactive mechanisms to provide efficient global batch production performance, and to face increasing unpredicted events. The holonic approach is the main engine for distributing intelligence of MES middle control layer (mixed batch planning and product scheduling, resource allocation, product traceability, production history, predictive maintenance) among agents representing the physical assets, with the scope of rejecting disturbances and reconfiguring shop-floor teams in real-time. In order to cope with both optimality in batch execution and robustness to unforeseen events such as occurrence of rush orders and degradation of resource capabilities, some of the papers included in this section propose holarchies of semi-heterarchical type combining centralized behaviours with decentralized ones, such as the duality System Scheduler—Delegate MAS or the formal specification of a self-sustainable holonic system.

Part 5 groups papers dealing with *Dependability and safety of Intelligent Manufacturing Systems*. Reliability, availability, security, testability and maintainability of products, processes and systems as well as their greenness are evaluated through dependability and safety studies oriented towards: application of measurement-based AHP to product-driven system control; operational research models and approaches applied to product-driven systems facing unexpected perturbations; holonic facility environment monitoring and control for radiopharmaceutical agent-based production; MAS framework for autonomous real-time resource management of CPS when disruptions are frequent; exploring the design space for myopia-avoiding distributed control systems using classification models.

Part 6 introduces contributions for the *Service-oriented Management and Control of Manufacturing Systems*. The concepts of services are integrated into holonic manufacturing systems (HMS), leading to Service-oriented HMS. The service becomes the main element of negotiation and exchange among holons, by defining:

- How manufacturing services are presented to the system in terms of identification and richness of description (from the manufacturing ontology);
- The strategies and methods to compose complex process workflows by combining individual services to reproduce the desired results;
- The strategies for executing atomic or composite manufacturing services (how to access and invoke such services).

Some papers in this section deal with: dynamic service reconfiguration with MAS tasks; semantic model to perform pluggability of heterogeneous smart devices into the Smart City environment; active monitoring of the product to solve the "lack of information" issue in the use phase.

Part 7 gathers contributions in the field of *Engineering and human integration in flexible and reconfigurable industrial systems*. These papers analyse how the human operator is integrated in the control architecture of an Intelligent Manufacturing System (IMS). "Human-in-the-loop" concepts consider the intervention of humans (typically for information providing, decision making or direct action on physical components) during the intelligent control of any functions at the operational level of manufacturing, such as scheduling, maintenance, monitoring, supply, etc. The objectives of systems integration evolved in the last few years, shifting from a model where the system was intended to adapt automatically and be equipped with reasoning and decision making capabilities aiming at replacing human ones to a model where CPS are focusing on use scenarios fully involving the humans. With a human-centred design approach in IMS, human resources can be assisted by recently developed ICT tools helping them detecting in advance problems, proposing efficient solutions and taking smart decisions. Representative papers in this section discuss respectively about instantiating the PERFoRM system architecture for industrial case studies, and defining a highly flexible, distributed data analysis framework for Industry 4.0 Manufacturing Systems.

Part 8 is devoted to *Virtualization and simulation in computing-oriented industry and service*. Papers in this section present resource virtualization techniques and resource sharing in industrial environments. Virtualization of resources and their capabilities allow for manufacturing services encapsulation in the cloud. In CPS (Cyber Physical System) approach of manufacturing, a major challenge is to integrate the computational decisional components (i.e. cyber part) with the physical automation systems and devices (i.e. physical part) to create such network of smart cyber-physical components at MES and shop floor levels. The basic concept of MES and shop floor virtualization involves migration of all workloads that were traditionally executed on physical machines located on the shop floor to the private cloud infrastructure as virtual workloads. The included papers describe: a simulation platform for Virtual Manufacturing Systems (VMS); environments to simulate: distributed agent-based manufacturing systems, adaptable agent-based smart grid management and an object-oriented holonic controller for a modular conveyor system.

The studies included in this book show that control paradigms for the manufacturing domain have evolved over time from centralized to decentralized or semi-heterarchical and were mainly driven by the new trends in information and communication technology such as: mobility, connectivity, increase of the decisional capabilities, service orientation and more recently the usage of cloud infrastructures to host control applications, run intensive optimization procedures and store large amount of production and resource data. This is how the concept of cloud manufacturing arose; its scope is to handle manufacturing resources and processes according to the quality of services (QoS) they provide in response to user preferences, and thus efficiently operate and manage them in a unified manner, providing high availability and flexibility in realizing customer orders in both small batches and mass production.

In conventional production structures with multiple workstations which allow job-shop fabrication scenarios, there are several physical control and computing entities that are single points of failure (SPoF) which can be avoided only by hardware redundancy; among these entities which execute centralized tasks are the system scheduler (for high performance computing tasks), the product router (the PLC responsible for routing products throughout the cell towards assigned resources), and the central application for production tracking and resource monitoring.

Private cloud platforms integrated as IaaS in manufacturing solve the high availability problem through resource virtualization techniques and provisioning of computing resources (CPU, storage, I/O) and applications. Also, the system scheduler (SS) implemented in the cloud infrastructure must communicate with the distributed MES (dMES), in which manufacturing resources and product execution orders are agentified; these agents cooperate in a multi-agent framework, both with the SS to receive scheduled orders and report their execution, and between them at dMES level.

Secure communication protocols must be used in such CMfg systems in order to access real-time production data and resource status information from agentified devices: (i) mobile intelligent embedded devices containing both the order execution data (scheduled operations for product execution and assigned resource for each operation) and the WIP data over the product's execution lifecycle; (ii) stationary computing devices providing resource status data and receiving from the SS application programs for assigned operations on products.

The book offers a new integrated vision on *Cloud and HPC, Big Data, Analytics* and *virtualization* in *Computing-oriented Manufacturing*, combining emergent information and communication technologies, service-oriented control of MAS and holonic architectures as well as total enterprise integration solutions based on SOA principles. The MCPS philosophy adopts heterarchical and collaborative control as its information system architecture, based on MAS—SOA duality. In this approach, MAS and Data Analysis are the basic technologies proposed to address the requirements and features envisioned by Industrie 4.0, and also taking in consideration the MCPS principles. The first provides the conceptual framework to realize the underline system infrastructure that is required to achieve the desired flexibility and adaptability levels, while the second provides the proper tools capable to analyse and obtain the required information to fulfil the desired system functionalities, also taking advantage of the increased data availability.

These two levels present specific characteristics and requirements which are covered by CPS principles, where the operational level is mainly related with the physical world, characterized by the IoT and its smart devices, also demanding the processing and analysis of real time data streams in order to attend the rapid response monitoring and control requirements. On the other hand, the supervisory level is hosted in a virtual world defined by a cloud-based infrastructure where robust software applications are used to attend the requirements of complex system management and high level information for decision-making, supported by big data analytics.

All these aspects are treated in the present book, which we hope you will find useful reading.

Part I
Cloud and Cyber-Physical Systems for Smart Manufacturing

High Availability Cloud Manufacturing System Integrating Distributed MES Agents

Silviu Răileanu, Florin Anton and Theodor Borangiu

Abstract The paper describes a semi-heterarchical manufacturing control solution based on a private cloud infrastructure which collects data in real time from intelligent devices associated to shop-floor entities: resources and mobile devices embedding the work in process (WIP) on products during their execution cycle. The cloud platform acts as a centralized system scheduler (SS), planning jobs and allocating resources optimally at batch level, and integrates real-time data and status information from agentified shop floor devices. The cloud infrastructure is also used for storage of historic data, for manufacturing set up and control (configuring resource teams for production orders received, selecting the control strategy choice, dynamic rescheduling of order execution in case of resource failure or rush order occurrence) and for hosting the web interface for: remote cell monitoring, reception of client requests, cell configuration, raw material inventory and reports generation. Implementation and experimental results are reported.

Keywords Cloud manufacturing · Private cloud · MES · High availability · Distributing intelligence · Mobile/intelligent devices · Agentification

1 Introduction

Control paradigms for the manufacturing domain have evolved over time from centralized to decentralized or semi-heterarchical [1] and were mainly driven by the new trends in information and communication technology (ICT) such as: mobility,

---

S. Răileanu (✉) · F. Anton · T. Borangiu
Department of Automatic Control and Applied Informatics, University Politehnica of Bucharest, Bucharest, Romania
e-mail: silviu.raileanu@cimr.pub.ro

F. Anton
e-mail: florin.anton@cimr.pub.ro

T. Borangiu
e-mail: theodor.borangiu@cimr.pub.ro

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_2

connectivity, increase of the decisional capabilities, service orientation and more recently the usage of cloud infrastructures to host control applications, run intensive optimization procedures and store large amount of production and resource data [2]. This is how the concept of cloud manufacturing (CMfg) arose [3]; its scope is to handle manufacturing resources and processes according to the quality of services (QoS) they provide, and thus efficiently operate and manage them in a unified manner, providing high availability and flexibility in realizing customer orders in both small batches and mass production [2].

At the manufacturing execution system (MES) level, cloud computing refers mainly to virtualization of applications such as mixed batch planning, job scheduling and resource allocation, or product traceability and production tracking [4]. While MES implementations are different and dependent of the physical infrastructure, the generic MES functions are standardized by ISA-95.03 specifications [5]. ISA-95 defines 5 levels for the hierarchical organization of a manufacturing enterprise, as follows: (i) layers 0, 1 and 2 represent the process control level together with its associated intelligent devices and control system, (ii) layer 3 represents the manufacturing operating level and consists of a series of activities such as scheduling, quality management, maintenance, production tracking, a.o. and (iii) layer 4 is in charge with managing the business-related activities of the manufacturing operation.

In conventional production structures with multiple workstations which allow job-shop fabrication scenarios, there are several physical control and computing entities that are single points of failure (SPoF) which can be avoided only by hardware redundancy; among these entities which execute centralized tasks are the system scheduler (for high performance computing (HPC) tasks), the product router (the controller (PLC) responsible for routing products throughout the cell towards assigned resources), and the central application for production tracking and resource monitoring.

Private cloud platforms integrated as IaaS (Infrastructure as a Service) in manufacturing solve the high availability (HA) problem through resource virtualization techniques and provisioning of computing resources (CPU, storage, I/O) and applications. Also, the system scheduler (SS) implemented in the cloud infrastructure must communicate with the distributed MES (dMES), in which manufacturing resources and product execution orders are agentified; these agents cooperate in a multi-agent framework (MAS), both with the SS to receive scheduled orders and report their execution, and between them at dMES level [6].

Secure communication protocols must be used in such CMfg systems in order to access real-time production data and resource status information from agentified devices: (i) mobile intelligent embedded devices containing both the order execution data (scheduled operations for product execution and assigned resource for each operation) and the WIP data over the product's execution lifecycle; (ii) stationary computing devices providing resource status data and receiving from the SS application programs for assigned operations on products [7, 8].

The paper is structured in six sections: Sect. 2 describes the structure of the dual SS-dMES control architecture, Sect. 3 describes the dynamic model of the control architecture, Sect. 4 describes the cloud infrastructure which will host the centralized system scheduler and integrating architecture, Sect. 5 presents an experimental case study and the last section is devoted to conclusions.

2 The Control Architecture

The proposed semi-heterarchical control architecture performs manufacturing tasks as follows: (i) resource team configuring, batch planning, product scheduling, resource allocation, cell and production monitoring will be done on the upper SS level running in cloud infrastructure for the received batch orders, and (ii) execution and rescheduling of orders in execution will be done on the lower dMES level; information will be communicated to the upper level for recording and decision making at special event occurrence Resource breakdown/recovery, rush order, local part storage depletion. The choice of having centralized batch optimization and decentralized production control (which may override the centralized optimization) is justified by the HPC availability to run complex optimization programs on far (batch) horizon while having the ability to quickly react at unexpected events possible through intelligence distribution, agentification and decentralization of the control structure. The advantage of this semi-heterarchical control architecture is that it can deal rapidly and locally with orders in execution while computing in parallel at the upper level with high availability an optimized schedule for the orders waiting to be executed. This reduces the myopia of the system at global batch level and preserves the system's agility.

The semi-heterarchical control solution is designed around a common database containing four types of information distributed in 10 tables: (i) shop-floor level information (1—available resources, 2—available operations, 3—products that can be executed, 4—distribution of operations on resources, 5—operations needed to execute products), (ii) clients information (6—clients identification and 7—placed orders), (iii) information resulted as a consequence of the planning and scheduling (8—sequence of orders, sequence of operations, the resources processing the operations and the specific time duration), and (iv) information needed to communicate between the web interface and the shop-floor level (9—operation execution reports and 10—synchronization data). The composing entities are instrumented and the categories of information above defined are updated and processed as depicted in Fig. 1.

(A) Orders, represented by WIP data, are instrumented with intelligent embedded devices capable to run a multi-agent system and communicating over wireless networks. From an informational point of view the WIP is represented by an order agent. Each order agent sends to the database located on cloud the order and operations status. It receives from cloud the product type, the operations to execute according to a certain control strategy (hierarchical/heterarchical).

3 The dynamic

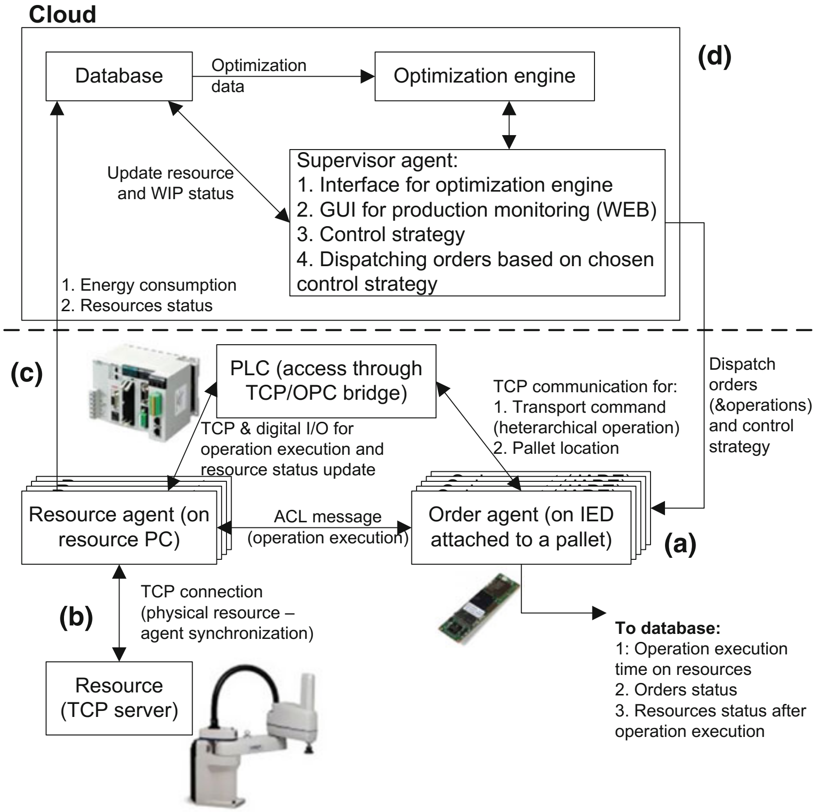


Fig. 1 The 2-layer generic model for manufacturing control

(B) Processing resources (robots, CNC machines) have computing terminals (PCs) attached to their controllers. The PCs run a multi-agent system used to integrate the resources with the mobile embedded devices (located on pallet carriers) embedding WIP, and with the cloud system. From an informational point of view the resources are represented by resource agents who run on the attached PC and send the periodically updated resource status to the database located on cloud.
Resource and order agents are located on the same network and exchange execution information (search operations on resources, request execution of an operation, receive acknowledges from resources).

(C) Transporting resources are represented by the devices of the conveyor (track motors, lifts, diverting elements, etc.) controlled by a PLC which is accessed through a TCP/OPC bridge. This type of resource communicates *only* with the mobile order agents uploaded on the embedded devices of each pallet where a product will be progressively executed by the visited resources, and fulfils transport services (ex.: order agent requests transport to a specified resource, transport resource realizes the transport operation and an acknowledgement is issued upon completion).

(D) The cloud infrastructure hosts a set of machines running the database, the optimization engine, the web server and the agent used to synchronize the data from the database with the optimization engine and with the shop-floor level.

3 The Dynamic Model

In order to provide coupling between applications running on the cloud infrastructure and shop-floor devices/resources composing the dMES [9], as well as a flexible message structure which helps the integration of composing entities, the Java Agent Development Framework (JADE) [10] has been used. JADE is specially designed for the development of decentralized systems; control systems also fall into this category, where software agents are loosely coupled and communicate through messages standardized by the Foundation for Intelligent Physical Agents (FIPA) (http://www.fipa.org).

Thus, each of the composing entities of the control architecture presented in Fig. 1 has an associated agent: WIP is represented by an **order agent**, the resources (both processing and transporting) are represented by a **resource agent** and a **supervisory agent** runs on the cloud infrastructure. The result is an event-driven architecture (EDA) [11] composed of individual agents used in a MAS framework for monitoring and control; EAD's main advantages are: easy integration of new entities, smart data utilization and an asynchronous and non-blocking operating mode.

The operating process (Fig. 2) of the control system is divided into several sub-processes as follows:

- *Continuous update of the GUI* with information from shop-floor entities (resources and WIP): resource confirm their online status periodically (each minute) and order agents update the state of operations associated to resources after their execution (Dynamic data for interface creation; System state and WIP, Update WIP and visited resources state, Confirm the realization of a product, Update resource data).
- *Receipt of orders* (orders are inserted through the CMfg GUI on the cloud platform and stored in the associate database segment).
- *Production optimization* (orders are planned and scheduled taking into account the resource availability); an optimization algorithm is selected and configured by the user through the cloud GUI according to the enterprise strategy.
- *Set up of the team of resources* that will be involved in the execution of the batch and updating their availability status.

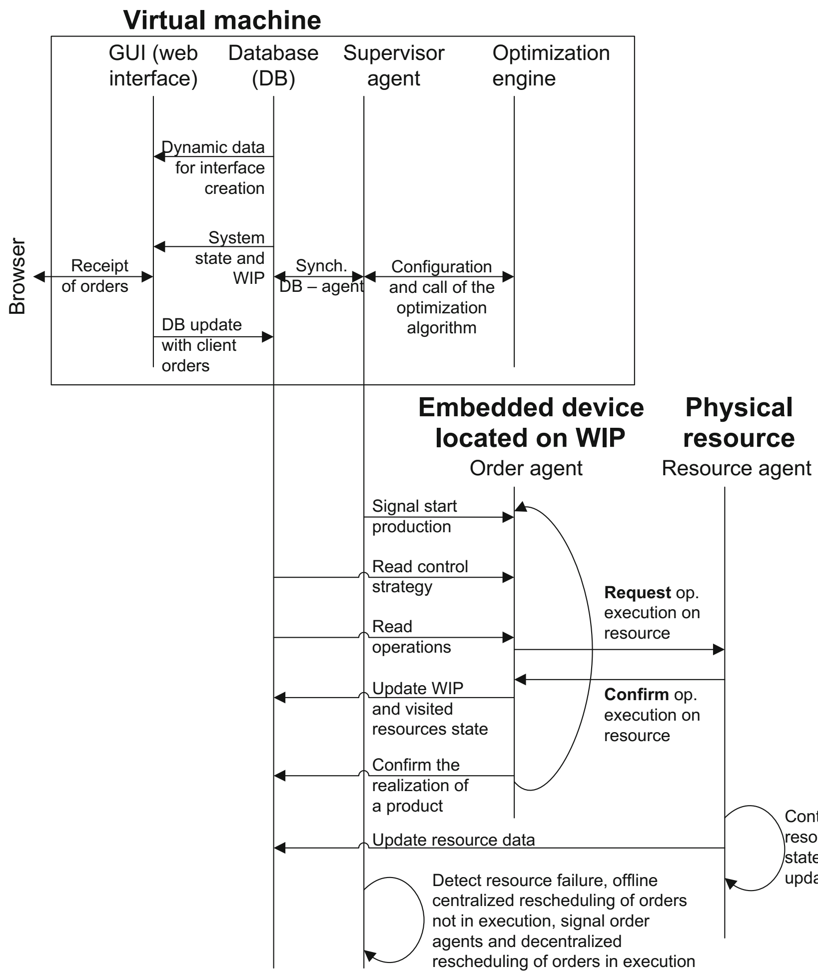


Fig. 2 The operating process of the control architecture

- *Transmission of planned and scheduled orders* to order agents embedded on mobile devices entering the shop floor for product execution. The supervisor agent is in charge with the individual transmission of each planned and scheduled order to a dedicated order agent running on the embedded devices entering the cell. (Signal start production, Read control strategy, Read operations).
- *Execution*: each order agent will be in charge with the execution of its associated product; when the product is finished, the pallet carrying the final product exists the cell and the order holon is deleted, allowing thus the cloud SS to transfer a new order agent for the execution of the next planned product.

- *Detection of resource failure and storage depletion*. These events are discovered due to continuous resource state update and monitoring of production execution; they generate a rescheduling of orders on the newly available resources. Resource update is done by resource agents in order to confirm they are online; they are sent periodically (each minute) to the cloud infrastructure. Execution monitoring is done by order agents, which are asynchronously tracking the progressive realization of operations on resources.

4 HA Design of the Cloud Infrastructure

The upper control layer (Fig. 1) has been implemented using an IaaS (Infrastructure as a Service) cloud system running the supervisor agent, database and optimization engine. The cloud system used is an ISDM (IBM Service Delivery Manager) based on Tivoli Service Automation Manager (TSAM) version 7.2 which uses as a hypervisor VMware ESXi 4.1 [12]. Because the applications are running in cloud one could consider that the applications benefit from all the cloud facilities, such as High Availability (HA) offered by the hypervisor and also Distributed Resource Sharing (DRS) offered by the hypervisor too. In reality the situation is not as simple as that; using a cloud system does not mean that the CMfg applications will just inherit the same facilities like the platform they are using.

Because the cloud system is an IaaS, it offers virtual machines as a service; these virtual machines can be used to set up the environment in order to run the applications: in our case a complex database updated with differentiated timing and the supervisor agent. The services in cloud are offered with the facilities that the cloud system supports like HA, meaning that the virtual machines (VM) used will be always available, but that does not guarantee that the applications inside the VM will be offered with HA. The reason for this is that the software could have bugs (the applications or the operating system) which will render the CMfg service unavailable, even if the VM is properly running from hardware point of view.

In order to solve this high availability problem extended to real time CMfg monitoring and control applications, an HA cluster was foreseen for database and application availability. The solution we propose is depicted in Fig. 3 and is composed by the following VMs:

1. Two load balancers (VMs) which are running in a cluster; the *Load Balancer VM* is publicly available and can be accessed from Internet. The function of the Load Balancer cluster is to get requests from internet for two types of services: secure https requests (port 443) and JAVA agent communication, and to forward these requests to the HA cluster in the internal network. The Load Balancer acts also as a gateway for the HA Nodes. For the JAVA agent communication requests a VPN tunnel implemented with IPSec (Internet Protocol Security) [13] has been created between the Load Balancer and the manufacturing infrastructure. In this way the https requests are coming from Internet and the JAVA

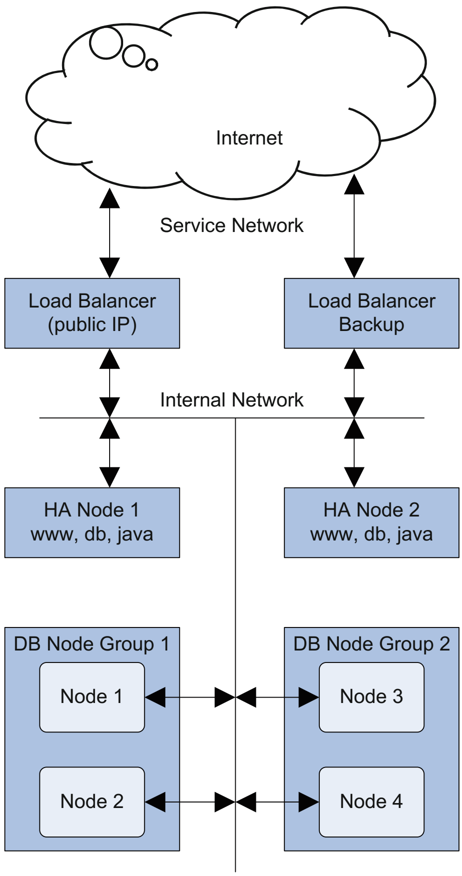


Fig. 3 The VM infrastructure in cloud

agent communication requests are coming from an internal network over an encrypted communication channel using VPN. The requests are processed by the Load Balancer and distributed to the HA Nodes using a Round Robin algorithm in order to load evenly the nodes. The HA Nodes are checked for service availability by many agents (agents used to verify if the Node is up and running and the service is accessible) from the Load Balancer. If a service is unavailable on a HA Node the session will be commuted to the other node and the other incoming requests are sent to the available node until the failed one recovers. The *Load Balancer Backup* is used only to check the status of the main Load Balancer; if the main Load Balancer fails, the backup one will take over by assigning the public IP and starting the load balancing function—in this way it becomes the main load balancer and the failed node becomes the backup one. The load balancer is implemented using the piranha [14] load balancer from RedHat on the Linux operating system.

2. The HA Cluster is composed by two nodes (VMs) and offers availability for three services: (i) *web interface access* using the https protocol on port 443; the service uses apache as a web server and the connection is encrypted using 1024 bit SSL (Secure Socket Layer) certificates [15]; (ii) *JAVA agent communication*, the connections being made between the agent which is running on the HA node and the agents on the manufacturing infrastructure; the connection is encrypted in Internet (an IPSec tunnel has been created between the Load Balancer and the gateway of the manufacturing infrastructure), the connection is unencrypted in the internal network; (iii) *database access*: the database management system used is a MySQL database implemented using MySQL Cluster for High Availability.
   The HA Cluster is configured to execute the services on both nodes in parallel which is quite unusual for the HA solution which was chosen, namely a RedHat High Availability Cluster. This configuration has been designed in order to offer a plus of performance by using load balancing. The behaviour of the cluster if a service fails is to try to restart that service on the node where the service failed; if this action has no effect and the service is already running on the other node the service will be stopped on the current node. The last special situation is when the service is not running on both nodes and the restart has failed; in this situation the service will be migrated on the other node. The fencing function (a function which isolates the failed node in order to avoid data corruption) was inhibited on the cluster because the two nodes can execute the services in parallel without restrictions.
3. The MySQL cluster is using four VMs grouped in two Node Groups. The Database is distributed between the two Node Groups, for example if we consider a table on the database which has four fields (F1, F2, F3 and F4), Node 1 will store F1 and F3, Node 2 will store F3 and F1, Node 3 will store F2 and F4, and Node 4 will store F4 and F2. Between the nodes the cluster offers data replication and data consistency. In the example above the database will be available and consistent even if two nodes from two node groups will fail. The MySQL Cluster protects against outages offering the following facilities:

   - *Synchronous Replication*—Data within each data node is synchronously replicated to another data node.
   - *Automatic Failover*—the heart beating mechanism detects in real time any failures and automatically fails over to other nodes in the cluster, without service interruption.
   - *Self-Healing*—Failed nodes are able to self-heal by automatically restarting and resynchronizing with other nodes before re-joining the cluster, with complete application transparency.
   - *Shared Nothing Architecture*, *No Single Point of Failure*—each node has its own disk and memory, so the risk of a failure caused by shared components such as storage, is eliminated.

If an error event is detected on any level of the infrastructure (Load Balancer, HA Cluster or MySQL Cluster) a notification to the system administrator will be sent in order to inform about the problem and to offer the opportunity to debug and solve the malfunctions.

5 Case Study: Experimental Results

Experiments have been done in order to validate data collection from shop-floor entities (consisting of mobile devices associated to WIP and resources) and the integration of this data into a cloud infrastructure. The results are shown in Fig. 4.

Thus, a batch of orders has been defined and launched in execution using the control application running on cloud. The steps taken are: update resource status, define product recipes using the available operations, define batch of orders from the available products, plan and schedule orders, validate the delivery date based on the computed makespan, select semi-heterarchical as control strategy, launch in execution planned and scheduled orders.

During the execution period a resource has changed its state from online to offline, no longer being available for production (resource failure). Since it no longer updates its status in the cloud database it will be considered offline and the remaining batch of orders was rescheduled using the available resources while the orders in execution (WIP) are negotiating in real-time the allocation of remaining operations on the available resources.

Since the data synchronization between the two levels (cloud and shop-floor devices) is practically instantaneous, the efficiency of using the cloud infrastructure

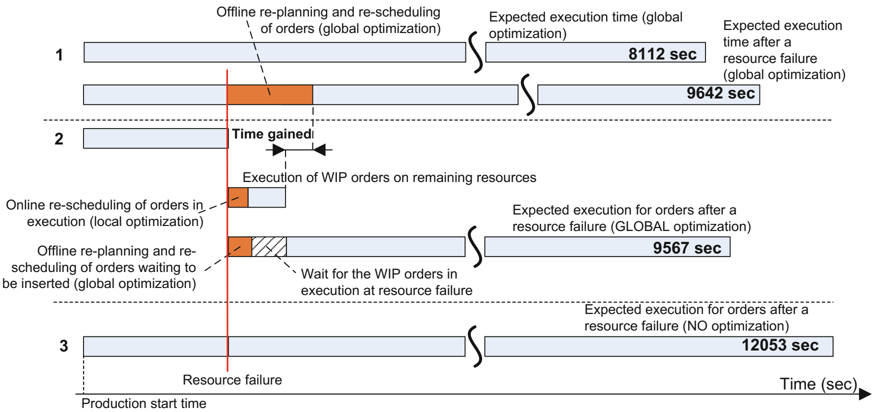


Fig. 4 Comparison of the execution times for the 3 different execution strategies: *1* centralized planning and scheduling (done in cloud) for all orders, *2* centralized planning and scheduling for orders waiting execution and decentralized re-scheduling for WIP orders, *3* decentralized planning and scheduling for all orders

to realize the offline planning and scheduling consists in the fact that production is still running with local, order agents optimization at shop-floor level until the Cloud re-planning and rescheduling of remaining products are performed and the hierarchical mode is resumed. Thus, time is gained by realizing the two activities in parallel (Fig. 4), since the Cloud SS optimization of scheduling a large batch of orders takes less time (due to its speed HPC characteristics) in comparison with classical SS implementing, thus guaranteeing the termination of rescheduling until the execution of WIP completes.

6 Conclusions

The paper describes a semi-heterarchical manufacturing control solution based on a private cloud infrastructure which collects data in real time from intelligent devices associated to shop-floor entities: resources and mobile devices embedding the WIP on products during their execution cycle. The cloud platform acts as a centralized SS, planning jobs and allocating resources optimally at batch level, and integrates real-time data and status information from agentified shop floor devices. The advantages brought by this control architecture are: easy integration of manufacturing data into a cloud infrastructure using a multi-agent framework distributed on the field entities and on the cloud (Fig. 1), fault tolerance and high availability of the applications centralized at MES level, running on cloud (Fig. 3) and the possibility to run in real manufacturing time intensive applications (such as optimization engines used for re-planning and re-scheduling in case of shop floor disturbances) in the cloud (Fig. 3).

A Public Key Infrastructure (PKI) solution with SSL authentication and encryption for intelligent products travelling on pallets in the shop floor and embedding WIP data to be transferred to the Cloud has been designed. From an implementation perspective, SOA alignment at shop floor level involves TCP/IP-based communication over Wi-Fi supporting higher level protocols. Our contribution in the security area CMfg implementation of the communication between the Cloud SS and the intelligent devices in the dMES (embedded devices for WIP, resource orders discovering the cell status) considered security challenges associated with data and information flow protection as well as authentication and authorization of the actors involved (order agents and resource agents).

Another advantage of using the Cloud IaaS for manufacturing management is the service oriented connectivity with the upper, business layer of the enterprise, offering direct access of clients to order acceptance, resource availability, and estimation of delivery of products via a cloud GUI. SOA principles can be used for the standardization of the data and information flow from the shop floor (production execution) level to the management (customer order tracking) level of the enterprise.

Regarding the infrastructure and the tools used to implement the CMfg solution, they offer High Availability on two levels: the cloud system offers HA at the infrastructure level and the Load Balancing/HA Clustering solutions offers HA at the application level.

From the point of view of security, the communication between clients and the web interface is encrypted using self-signed SSL certificates that offer a good level of security by using the Diffie-Helman algorithm [16]. The communication between JAVA agents is also encrypted in Internet by using an IPSec VPN tunnel between the gateways of the manufacturing infrastructure and the load balancer in the cloud infrastructure. The other connections are made in the local network which is a private network implemented with virtual LAN's (VLAN). In the cloud system the HA Cluster and the MySQL cluster are in the same VLAN, but if a higher level of security is required there is the possibility to isolate the MySQL Cluster from the Load Balancer by using a VLAN which will connect the HA Cluster with MySQL Cluster.

Acknowledgements This research work has been partially supported by the IBM FA 2016 Project: Big Data, Analytics and Cloud for Digital Transformation on Manufacturing—DTM.

References

1. Borangiu, Th., Răileanu, S., Berger, T., Trentesaux, D.: Switching mode control strategy in manufacturing execution systems. Int. J. Prod. Res. **53**(7), 1950–1963 (2015). doi:10.1080/00207543.2014.935825, ISSN: 0020-7543 (Print), 1366-588X (Online)
2. Kubler, S., Holmstrom, J., Främling, K., Turkama, P.: Technological Theory of Cloud Manufacturing, Service Orientation in Holonic and Multi-Agent Manufacturing. Studies in Computational Intelligence, vol. 640, pp. 267–276. Springer International Publication (2016). doi:10.1007/978-3-319-30337-6_24
3. Wu, D., Greer, M.J., Rosen, D., Schaefer, D.: Cloud manufacturing: strategic vision and state-of-the-art. J. Manuf. Syst. **32**, 564–579 (2013). doi:10.1016/j.jmsy.2013.04.008
4. Morariu, O., Morariu, C., Borangiu, Th.: Shop-floor resource virtualization layer with private cloud support. J. Intell. Manuf. **30**, 1–16 (2014). doi:10.1007/s10845-014-0878-7
5. Meyer, H., Fuchs, F., Thiel, K.: Manufacturing Execution Systems. McGraw-Hill (2009). ISBN: 978-0-07-162602-6
6. Novas, J.M., Bahtiar, R., Van Belle, J, Valckenaers, P.: An approach for the integration of a scheduling system and a multi-agent manufacturing execution system. towards a collaborative framework. In: Proceedings of the 14th IFAC Symposium INCOM'12, Bucharest, IFAC PapersOnLine, pp. 728–733 (2012)
7. Morariu, O., Morariu, C., Borangiu, Th.: Security issues in service oriented manufacturing architectures with distributed intelligence. In: Borangiu, T., Trentesaux, D., Thomas, A., McFarlane, D. (eds.) Service Orientation in Holonic and Multi-Agent Manufacturing. Springer Series SCI, vol. 640, Chapter 5.4, pp. 205–224. Springer International Publication (2016). doi:10.1007/978-3-319-30337-6_23, ISBN: 978-3-319-30335-2
8. Wang, L., Shen, W., Lang, S.: Wise-ShopFloor: a web-based and sensor-driven shop floor environment. In: The 7th International Conference on IEEE Computer Supported Cooperative Work in Design, pp. 413–418 (2002)
9. Främling, K., Kubler, S., Buda, A.: Universal messaging standards for the IoT from a lifecycle management perspective. IEEE Internet Things J. **1**(4), 319–327 (2014)
10. Bellifemine, F., Carie, G., Greenwood, D.: Developing Multi-agent Systems with JADE. Wiley (2007). ISBN: 978-0-470-05747-6

11. Theorin, A., Bengtsson, K., Provost, J., Lieder, M., Johnsson, C., Lundholm, T., Lennartson, B.: An event-driven manufacturing information system architecture for Industry 4.0. Int. J. Prod. Res. (2016). doi:10.1080/00207543.2016.1201604
12. Buyya, R., Vecchiola, C., Selvi, T.: Mastering Cloud Computing Chapter 3—Virtualization, pp. 71–109 (2013). doi:10.1016/B978-0-12-411454-8.00003-6
13. Rowan, T.: VPN technology: IPSEC vs SSL. Netw. Secur. **2007**(12), 13–17 (2007)
14. Red Hat Enterprise Linux 5: Virtual Server Administration, Red Hat (2009)
15. Meyer, H.: Securing intranet data with SSL client certificates. Comput. Secur. **16**(4), 334 (1997)
16. Harn, L., Lin, C.: Efficient group Diffie-Hellman key agreement protocols. Comput. Electr. Eng. **40**(6), 1972–1980 (2014)

Classification of Cyber-Physical Systems Developments: Proposition of an Analysis Framework

Olivier Cardin

Abstract Cyber-physical systems have encountered a huge success in the past decade in several scientific communities. The main attraction of the concept relies in the fact that it encompasses many scientific topics that were distinct before. The downside is the lack of readability of the current developments about cyber-physical systems. This work intends to introduce an analysis framework able to classify the developments. An extensive study of literature enabled to extract four major criteria that are to be used in the framework: cognitive capabilities, application domain, interaction with humans and network technologies. Several examples of literature are used to illustrate the use of the framework.

Keywords Cyber-physical systems • Cognition • Framework • Human factor • Applications

1 Introduction

The first definition that can be found about cyber-physical systems (CPS) dates from 2006 [1], during a workshop with the American National Science Foundation[1] (NSF). The extension of cybernetic systems towards CPS is therefore explicitly dealt with in literature since 2006–2007 and is constantly growing in popularity. A short analysis of the main scientific journals' editors about the evolution of the number of publications in journals directly dealing with CPS shows a fast and

[1] http://www.nsf.gov/.

O. Cardin (✉)
LUNAM Université, IUT de Nantes – Université de Nantes,
IRCCyN UMR CNRS 6597, Nantes, France
e-mail: olivier.cardin@univ-nantes.fr

O. Cardin
Institut de Recherche en Communications et Cybernétique de Nantes,
2 avenue du Prof. Jean Rouxel, 44475 Carquefou, France

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_3

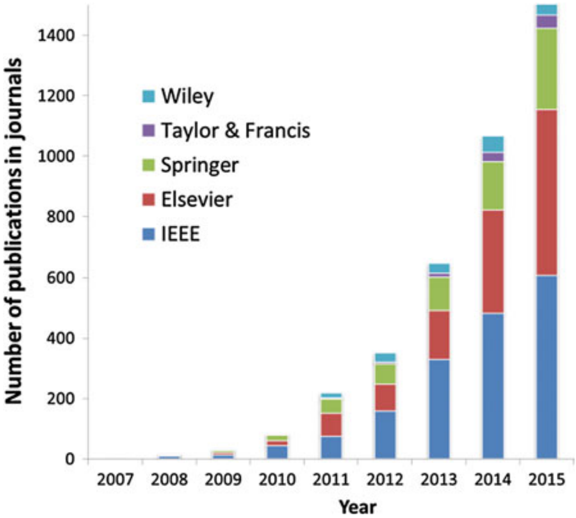


Fig. 1 CPS concept dissemination in literature

global evolution and a constant raise of the number of papers all along the last ten years (Fig. 1). All along their development, more synthetic definitions were suggested, such as those of [2] or [3]. The recent one, suggested by [4], allows a clear synthesis of the various aspects of this large concept, coupling in addition the notion of services with CPS: "Cyber-Physical Systems (CPS) are systems of collaborating computational entities which are in intensive connection with the surrounding physical world and its on-going processes, providing and using, at the same time, data-accessing and data-processing services available on the internet".

To do so, embedded software in CPS uses sensors and actuators, connect with each other and with humans communicating via standard interfaces, and have abilities of storage and processing of data coming from sensors or from the network [5]. The main characteristics that can be expressed for general purposes CPS are [6]: a high level of physical/cyber integration; data processing in each physical components, because of generally limited resources of communication and/or processing; high level of connectivity via cabled or wireless networks, Bluetooth, GSM, GPS, Wi-Fi, etc.; adapted to multiple scales in time and space; able to dynamically reconfigure or reorganize themselves; highly automated, in closed loops; reliable or certified in some cases.

The notion of CPS is very wide and encompasses an extremely large class of systems. As a matter of fact, numerous fields of research are relevant of this keyword. This is probably a huge opportunity as it gives the possibility to create a consistent ecosystem in numerous fields of applications, such as intelligent manufacturing for example. However, there is a risk of scattering of research efforts inside this wide notion. To overcome this eventuality, this article intends to introduce an analysis framework aiming at classifying the various developments found in literature and positioning them relatively in each application field.

First, the framework will be presented and the items defining the axes of the framework are described. The last section introduces the application of the framework to various examples of developments found in literature in order to illustrate the use of the framework.

2 Classification

Analyzing the general functionalities that are to be expected from a CPS [7], interesting aspects that could be used in the framework are expressed:

1. Integrated in its environment: record and evaluate physical data from sensors, act or react to specific events on physical environment with actuators;
2. Efficient in its environment: specific to an application domain and enhancing the capabilities and possibilities of the overall environment;
3. Connected to its environment: connect to a global network and to the other CPS with digital tools in order to use available data and services in order to interact with this digital environment;
4. Cooperating with its environment: equipped with multimodal human-machine interfaces.

Based on this decomposition, the analysis framework will be detailed in the next sections following these four main functionalities of CPS.

2.1 Integration Level

A classification of CPS in five categories (5C) according to the level of integration it offers was presented by [8]:

- **C1**. At Connection level, CPS operate on a Plug and Play network and use data provided by sensors on the network;
- **C2**. At Conversion level, CPS process data and aggregate them in a higher semantic level;
- **C3**. At Cyber level, CPS can apprehend other CPS and their environment and can interact with them in order to enrich their own data processing;
- **C4**. At Cognition level, CPS are able to process data in order to diagnose their own state, based on simulations and a differential analysis of sensors data;
- **C5**. At Configuration level, CPS shall adapt on their own facing disturbances, reconfigure or adjust their parameters in an autonomous way in order to get back to a nominal behaviour as soon as possible.

This classification was initially introduced in order to provide a step by step CPS deployment tutorial, from sensing functionalities to functions creating more added value. The total integration of these 5 levels in a CPS is currently extremely rare and might not be pertinent in any situations. We suggest in this article to extend the use of this functional decomposition and use it in order to evaluate the level of autonomy and intelligence embedded in a given CPS. This constitutes the first criterion of the framework that we suggest to define (C1 to C5).

2.2 Application Domain

A second classification criterion is defined as the application domain of the CPS. This feature was already dealt with in numerous works in literature. Therefore, we suggest using in this framework the list of potential fields established in a synthesis[2] made by the NIST and the universities of Virginia and Berkeley:

- **Communication**: this field deals with every CPS aiming at designing innovative communication means;
- **Consumer**: integrates all the CPS having an objective of modification of consuming habits of the customers;
- **Energy**: deals with all the applications in production or distribution of energy;
- **Infrastructure**: includes both works about buildings and civil engineering;
- **Health Care**: includes all the aspects of health care management;
- **Manufacturing**: includes all the aspects of industrial engineering;
- **Military**: focuses on the application in the field of security and Defence;
- **Robotics**: includes all the applications oriented towards robotics, whether humanoid, production, service or mobile;
- **Transportation**: includes all the modes of transport, individual or collective, of people or merchandises, urban traffic management systems, transportation logistics and supply chains.

This criterion might be expressed as one or as the conjunction of two application domains.

2.3 CPS and Humans

The objectives of systems integration evolved in the last few years, shifting from a model where the system was intended to adapt automatically and be equipped with reasoning and decision making capabilities aiming at replacing human ones to a

[2] http://cyberphysicalsystems.org.

model where CPS are focusing on use scenarios fully involving the humans. Two possible scenarios [9] were specifically designed.

In the first one (called Automation scenario), the human is guided by the CPS, i.e. the global decision making process is performed by the CPS and the human executes the operation himself. It is also a human that is responsible of the implementation and maintenance of the CPS. In a manufacturing context, his scenario corresponds roughly to an online and pulled flow oriented transposition of the classical functions of planning and scheduling that can be encountered currently. This scenario is intended to fit well to the workshops implying a heavy manual duty in a manufacturing environment targeting a high flexibility.

The second scenario (Tool scenario) emphasizes the human in the core of the decision loop. The idea is to make the CPS being guided by a human initiated to the cooperation with the CPS, but still actively assisting the human in the decision making process. This scenario intends to fit the activities in which operations are partially or fully automated, but where the operator's expertise brings a significant contribution notably in terms of agility and quality enhancement.

In both first and second scenario, it is the combination between calculation abilities of CPS and communication with human capacities that enable the enhancement of the performance of the cooperation system. Frameworks such as HilCP^{2}SC (Human-in-the-Loop Cyber-Physical Production Systems Control) are being developed [10], in order to offer the possibility to integrate the preferences of the human in a multi-objective decision making context led by the CPS. More than a framework, it is probably necessary to modify the (manufacturing) distributed systems design paradigm, using CPS reference models that need to be anthropocentric, such as those suggested by [11] or [12] for example. Whatever the scenario, a significant evolution of the tasks, qualifications and skills of the human operators in charge of the cooperation with the CPS [13] is foreseen. Therefore, basic and professional trainings of those operators needs to adapt in order to give them the keys to efficiency in a problem resolution process, to have a more accurate conscience of process interdependencies of which they are a link and to be able to take regulated initiatives for self-organization in case of disruptions occurring in the nominal working of the system.

Considering this criterion, a larger list was induced from the levels suggested by [9]:

- **Full**: the human only has a role of supervision of the CPS, which is able to take all the necessary decisions without any intervention of the human;
- **Automation**: the CPS guides the human during its task by taking most of the decisions and leaves the functions of adaptation to the human;
- **Tool**: the human guides the CPS and is in charge of most of the decisions;
- **Manual**: CPS only provides data to the human, who is in charge of all the decisions.

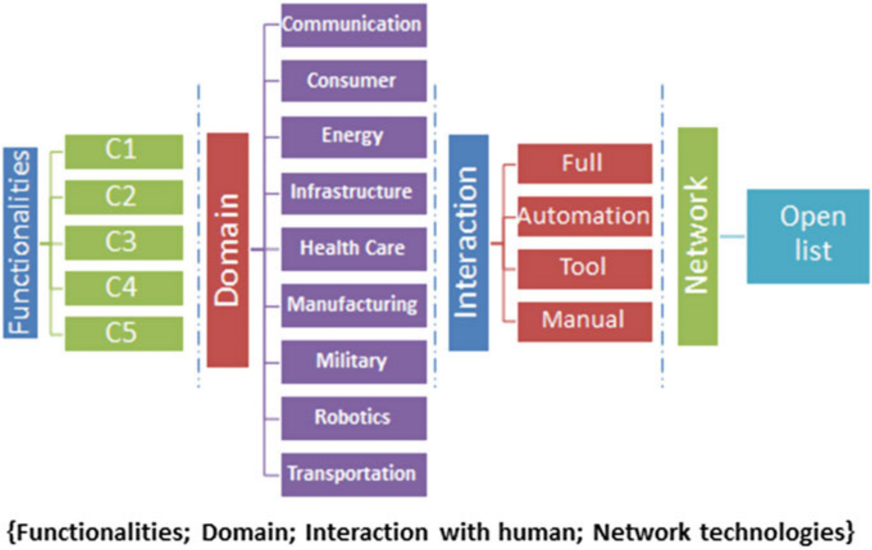


Fig. 2 Framework graphic view

2.4 Last Criterion and Framework Expression

Finally, the last criterion that is suggested here covers a technological aspect of the CPS relative to the communication technologies used to plug on the network. The levels of this criterion are open, because of the constant evolution of available technologies. The following levels can be cited as current examples of what can be used: Ethernet, Wi-Fi, Zigbee, WSN (Wireless Sensors Network), GSM, GPS, etc.

With this last criterion, a 4-criteria-framework enabling to specify the characteristics of a study relative to CPS is defined, and might be written in the following way: {Functionalities; Domain; Interaction with human; Network technologies} (Fig. 2).

3 Framework Application to Current Literature

This section introduces examples retrieved from literature and illustrates the use of the framework through an analysis grid. From this grid, an analysis is presented that highlights the position of current research in the framework in several application domains.

Table 1 CPS developments examples classification

	Full	Automation	Tool	Manual
C1				[14] *{C1; Health; Manual; WSN}*
C2	[15] *{C2; Manufacturing; Full; Ethernet}*		[16] *{C2; Health Care; Tool; GSM}* [17] *{C2; Manufacturing; Tool; Wifi}*	
C3	[18] *{C3; Transportation; Full; WSN}*		[19] *{C3; Transportation/Robotics; Tool; Radio}* [20] *{C3; Manufacturing; Tool; Ethernet}* [21] *{C3; Manufacturing; Tool; Ethernet}* [22] *{C3; Manufacturing; Tool; Ethernet}* [23] *{C3; Manufacturing; Tool; Zigbee}*	
C4	[24] *{C4; Infrastructure; Full; WSN}*		[25] *{C4; Transportation; Tool; Ethernet}*	
C5				

3.1 Analysis Grid of Current Literature

The articles exposed in this section are representative of the current developments currently available in literature and specifically addressing the concept of CPS in addition to some industrial applications in Manufacturing. It is obvious that a lot more applications can be encountered without addressing CPS, but these are not included in the limits of this analysis. Table 1 positions a dozen articles relatively to their level of integration (C1 to C5) and their connection with humans (Full to Manual). For each reference, the whole classification is also given (below in italic, Table 1) in order to analyse the current advances relatively to the application domains.

3.2 Conclusions Drawn from the Analysis Framework

The first conclusion that can be drawn from this table is the high density of applications between C2 and C4. Globally, it was expected to encounter relatively few applications with C5 abilities, as these cover at the time being more perspectives than actual possibilities. However, the low interest for C1 level is more surprising. In parallel, the application domain of the sole C1 application is interesting to notice. Health Care domain is a sensible field where autonomous CPS might take time to be accepted. This is correlated with the fact that both the Health Care applications found are in Manual and Tool modes, i.e. with a high interaction with human operators who still has the last word during decision making.

At the other side, the Transportation field, aiming at providing autonomous vehicles as CPS, is developing CPS with a high level of autonomy and relatively few interactions with humans (Tool or Full), which is coherent with the objectives of the end products. Same conclusions for Infrastructures, where the nature of the product dealt with enables relatively rapidly the setup of a high level of autonomy.

In Manufacturing, a very high density of works is located around {C3; Manufacturing; Tool}, which seems relevant to the state of the art of such a domain. What can be finally noticed is the lack of references in Automation mode. One possibility is that such a mode requires the human to trust the CPS, which might not be the case at the time being. However, it does not presume of the future applications which might implement massively this mode.

4 Conclusion

This article introduces a new analysis framework for classifying CPS applications relatively to their cognitive abilities, their application field, the interaction with human operators and finally the network technologies that are used.

This framework was described and applied to several examples retrieved from literature. From the grid extracted from this analysis, several conclusions are drawn for each encountered application domain, noticing that this framework globally fits the main trends that can be spotted in each specific field.

The perspective of this framework is to extend the analysis to a wider class of applications in order to refine the grid and provide a clearer and sharper outlook on the research efforts in each application domains.

References

1. Lee, E.A.: Cyber-physical systems—are computing foundations adequate? In: Position Paper for NSF Workshop On Cyber-Physical Systems, Austin, Texas, (2006)
2. Rajkumar, R., Lee, I., Sha, L., Stankovic, J.: Cyber-physical systems: the next computing revolution. In: 2010 47th ACM/IEEE Design Automation Conference (DAC), pp. 731-736 (2010)
3. Baheti, R., Gill, H.: Cyber-physical systems. Impact Control Technol. **12**, 161–166 (2011)
4. Monostori, L.: Cyber-physical Production Systems: Roots, Expectations and R&D Challenges. Procedia CIRP **17**, 9–13 (2014)
5. Strang, D., Anderl, R.: Assembly process driven component data model in cyber-physical production systems. In: Proceedings of the World Congress on Engineering and Computer Science, vol. 2 (2014)
6. Shi, J., Wan, J., Yan, H., Suo, H.: A survey of cyber-physical systems. In: 2011 International Conference on Wireless Communications and Signal Processing (WCSP), pp. 1–6 (2011)
7. Geisberger, E., Broy, M.: Agenda CPS: Integrierte Forschungsagenda Cyber-Physical Systems, vol. 1. Springer (2012)

8. Lee, J., Bagheri, B., Kao, H.-A.: A cyber-physical systems architecture for Industry 4.0-based manufacturing systems. Manuf. Lett. **3**, 18–23 (2015)
9. Dworschak, B., Zaiser, H.: Competences for cyber-physical systems in manufacturing—first findings and scenarios. Procedia CIRP **25**, 345–350 (2014)
10. Gaham, M., Bouzouia, B., Achour, N.: Human-in-the-Loop Cyber-Physical Production Systems Control (HiLCP2sC): a multi-objective interactive framework proposal. In: Service Orientation in Holonic and Multi-agent Manufacturing, pp. 315–325. Springer International Publishing (2015)
11. Zamfirescu, C.-B., Pirvu, B.-C., Gorecky, D., Chakravarthy, H.: Human-centred assembly: a case study for an anthropocentric cyber-physical system. Procedia Technol. **15**, 90–98 (2014)
12. Pirvu, B.-C., Zamfirescu, C.-B., Gorecky, D.: Engineering insights from an anthropocentric cyber-physical system: a case study for an assembly station. Mechatronics (2015)
13. Dombrowski, U., Wagner, T.: Mental strain as field of action in the 4th industrial revolution. Procedia CIRP **17**, 100–105 (2014)
14. Santos, D.F., Almeida, H.O., Perkusich, A.: A personal connected health system for the internet of things based on the constrained application protocol. Comput. Electr. Eng. (2015)
15. Al-Hammouri, A.T.: A comprehensive co-simulation platform for cyber-physical systems. Comput. Commun. **36**(1), 8–19 (2012)
16. Håkansson, A., Hartung, R.: An infrastructure for individualised and intelligent decision-making and negotiation in cyber-physical systems. Procedia Comput. Sci. **35**, 822–831 (2014)
17. Bergweiler, S.: Intelligent manufacturing based on self-monitoring cyber-physical systems. In: UBICOMM 2015, pp. 121–126, Nice, France (2015)
18. Wan, J., Suo, H., Yan, H., Liu, J.: A general test platform for cyber-physical systems: unmanned vehicle with wireless sensor network navigation. Procedia Eng. **24**, 123–127 (2011)
19. Fink, J., Ribeiro, A., Kumar, V.: Robust control for mobility and wireless communication in cyber–physical systems with application to robot teams. Proc. IEEE **100**(1), 164–178 (2012)
20. Blanc, P., Demongodin, I., Castagna, P.: A holonic approach for manufacturing execution system design: an industrial application. Eng. Appl. Artif. Intell. **21**(3), 315–330 (2008)
21. Aissani, N., Beldjilali, B., Trentesaux, D.: Dynamic scheduling of maintenance tasks in the petroleum industry: a reinforcement approach. Eng. Appl. Artif. Intell. **22**(7), 1089–1103 (2009)
22. Chové, E., Castagna, P., Abbou, R.: Hoist scheduling problem: coupling reactive and predictive approaches. In: INCOM 2009 Proceedings, pp. 2077–2082 (2009)
23. Gamboa Quintanilla, F., Cardin, O., Castagna, P.: Evolution of a flexible manufacturing system: from communicating to autonomous product. In: Service Orientation in Holonic and Multi Agent Manufacturing and Robotics, vol. 472, p. 167–180. Springer (2013)
24. Wang, S., Zhang, G., Shen, B., Xie, X.: An integrated scheme for cyber-physical building energy management system. Procedia Eng. **15**, 3616–3620 (2011)
25. Le Mortellec, A., Clarhaut, J., Sallez, Y., Berger, T., Trentesaux, D.: Embedded holonic fault diagnosis of complex transportation systems. Eng. Appl. Artif. Intell. **26**(1), 227–240 (2013)

Formal Modelling of Distributed Automation CPS with CP-Agnostic Software

Dmitrii Drozdov, Sandeep Patil and Valeriy Vyatkin

Abstract This paper introduces the application of formal modelling to cyber-physical systems for the purpose of formal verification of cyber-physical agnosticism properties of decentralised automation software. Service-oriented cyber-physical systems are modelled using the IEC 61499 function block architecture. Then, the IEC 61499 model is represented in terms of SMV formal language, which is verified using the nuXmv model checker. The timestamped events mechanism is implemented in IEC 61499 using the existing syntax of the standard. The impact of jitter on the behaviour of physical system is demonstrated in both simulation within IEC 61499 execution environment and in counter-examples extracted from nuXmv. The methodology presented in this paper creates the premise for automatic verification of the cyber-physical agnosticism of systems.

Keywords Formal verification · Distributed automation systems · Cyber physical systems · IEC 61499 · nuXmv · SOA

1 Introduction

Wired and wireless networking is becoming a necessary part of industrial control systems design and is one reason for which new cyber-physical approach to modelling is required. As illustrated in Fig. 1, the phenomenon observed in cyber-

D. Drozdov · S. Patil (✉) · V. Vyatkin
Luleå University of Technology, Luleå, Sweden
e-mail: sandeep.patil@ltu.se

D. Drozdov
e-mail: dmitrii.drozdov@ltu.se

V. Vyatkin
e-mail: vyatkin@ieee.org

D. Drozdov
Penza State University, Penza, Russia

V. Vyatkin
Aalto University, Helsinki, Finland

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_4

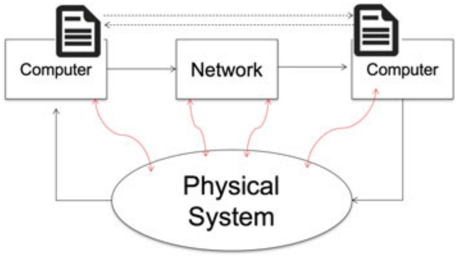


Fig. 1 Cross-dependencies and influences in cyber-physical systems

physical systems is the variety of cross-dependencies and influences between physical processes, computational devices and communications, which needs to be taken into account in the design and verification stage. Since most of the systems' functionality nowadays is implemented via software, the ability of software to be agnostic to the mentioned cross-influences is of high importance. An industrial motivation for achieving invariance of cyber-physical systems' (CPS) behaviour was presented, for example, in the SmartGrid domain in [21, 28].

It has been demonstrated in a number of recent works, e.g. [4–7] that distributed CPS with service-oriented distributed architecture can be efficiently modelled using IEC 61499 function blocks, while [9] demonstrates the use of web-services in IEC 61499 applications and [10] shows a case study combining IEC 61499 and Arrowhead SOA platform. In [27] the work started on addressing the cyber-physical agnosticism (CPA) challenge of software design by finding an appropriate design architecture. By modelling in PtolemyII environment and by applying the time-stamped events pTides semantics to the operation of control system, it was demonstrated that disturbances on the physical system behaviour introduced by jitter in communication can be efficiently mitigated. The demonstration was achieved by simulation of systems with jitter, by comparing performances of a conventional event-driven model and by the time-stamped events semantics. This paper attempts to extend the range of disturbances for which the stress-test of systems is performed by applying more exhaustive model-checking instead of simulation. For that, formal model of the cyber-physical system needs to be developed.

The paper is structured as follows. In Sect. 2 a brief overview of related works is presented. Section 3 discusses implementation of timestamps without ground change of the IEC 61499 syntax and then continues with addressing methods of modelling timestamps in SMV modelling language. Section 4 presents details of the case study. Section 5 presents modelling of the cylinder CPS case study in SMV, and Sect. 6 presents elements of timestamps modelling in SMV. The paper is concluded with a summary, acknowledgements and references.

2 Related Work

2.1 *Invariance in Cyber-Physical System Design*

The CPA demonstration in [27] extends the intentions of latency insensitive design of (only digital) systems presented in [3]. These works are related to the large body of works on networked event-driven controls, such as [1, 2], whose mathematical apparatus can be used, although their motivation is completely different (e.g. saving batteries of wireless sensor nodes).

2.2 *IEC 61499*

The IEC 61499 standard introduces a system-level reference software architecture for distributed automation systems. The most essential claim of the IEC 61499 architecture is about minimizing developers' efforts in deploying automation software to different distributed architectures of hardware [22]. The event-driven activation mechanism of FBs helps to preserve causality in distributed systems, which is an important enabler of this distributed deployment transparency. This can be considered as a provision of cyber-physical agnosticism w.r.t. re-deployment of software to different networking hardware configurations.

2.3 *IEC 61499 Formal Modelling and Verification*

The IEC 61499 architecture has been the target of numerous formal modelling and verification approaches, started from works [17, 18, 24–26] which demonstrate the potential of automatic check of system properties when distributed system reconfiguration occurs. In particular, these works focus on physical mechatronic system reconfiguration, which entails partial software evolution and may impact on the system's compliance with functional or non-functional requirements. For that purpose, [10] emphasises the use of closed-loop modelling pattern where behavioural model of plant is taken into consideration in order to verify properties relevant to the physical system's behaviour. The early works on formal modelling of IEC 61499 were summarised in the survey [14]. Subsequent works attempted to extend the modelling framework by handling richer data types [13], by proposing comprehensive automatic model-generation methods and tools [16]. Timing aspects of IEC 61499 modelling were also in the focus, for example in [23]. In [19, 20] IEC 61499 was modelled with one of the most powerful model-checking language SMV.

2.4 Time-Stamped Semantics

While cyber-physical systems can be modelled in the PtolemyII environment using heterogeneous combination of continuous and discrete state domains (with pTides semantics), implementation of such systems would require the corresponding modification of the implementation language, the IEC 61499, by extending its event concept with timestamps. The corresponding attempts were made in [8, 15, 27]. While these attempts require further harmonization we will be mainly relying on the version of [27].

3 Timestamped Semantics for IEC 61499 SMV Models

For the objectives of this paper a simplified implementation of timestamps in IEC 61499 is used. For each event input or output of a function block an additional data input/output of type TIME is introduced for transferring the timestamp between function blocks. In this way simulation can be conducted without changing the underlying execution environment and tools of IEC 61499. In our case the nxtStudio environment is used.

3.1 Global Time in SMV Model

To control timestamp events, the incrementing global timer is introduced to the SMV model. To bound the model state space, only the time interval from 0 to T_max is considered during verification. The global time variable is determined by the rule:

$$T_{Global} < T_{max} \vee \gamma \vee (\wedge(Do_i)) \Rightarrow T_{Global} = T_{Global} + D_{min} \tag{1}$$

Also, the rules for delay blocks' timestamp changing were modified to stop any execution after Tmax is reached. The listing in Fig. 2 shows the SMV code of time scheduler with global time for two delay blocks.

4 Case Study

As a case study we continue the one cylinder example from [27]. It can be easily scaled up and put in the context of many real-life applications. It is schematically presented in Fig. 3. The function block application implements the control of

```
MODULE TimeScheduler (D1o, D1i, D2o, D2i, beta, gamma)
VAR V1 : integer;
VAR V2 : integer;
VAR DMin : integer;
VAR TGlobal : integer;     global timestamp
ASSIGN
init(TGlobal):=0;
V1:=case
        D1o >= 0 : D1o;
        TRUE : Tmax;
esac;
V2:=case
        D2o >= 0 : D2o;
        TRUE : Tmax;
esac;

DMin:=case
        V1 <= V2 : V1;
        V1 > V2  : V2;
        TRUE : 0;
esac;
D1i:=case
        (gamma & D1o > 0) & TGlobal < Tmax : D1o - DMin;
        TRUE: D1o;
esac;
D2i:=case
        (gamma & D2o > 0) & TGlobal < Tmax : D2o - DMin;
        TRUE: D2o;
esac;
next(TGlobal):=case
        TGlobal<Tmax & (gamma & (D1o > 0 | D2o > 0)):
                TGlobal + DMin;
        TRUE: TGlobal;
esac;
DEFINE Tmax:= 3000;
```

Fig. 2 Time scheduler for two delay blocks

a "pneumatic cylinder with a retracting spring", using a position sensor and proportional valve actuator. The control tracks a certain desired position of the cylinder as provided by the setpoint input *SP* of the FB *Error*. The sensor reading is obtained and initially processed in the FB *Sensor*. Then it is passed to the FB *Error* by emitting event. The FB *Error* calculates the difference between *SP* and current position and passes it to the FB *Controller* that recalculates the control signal and passes it to the *Actuator* FB.

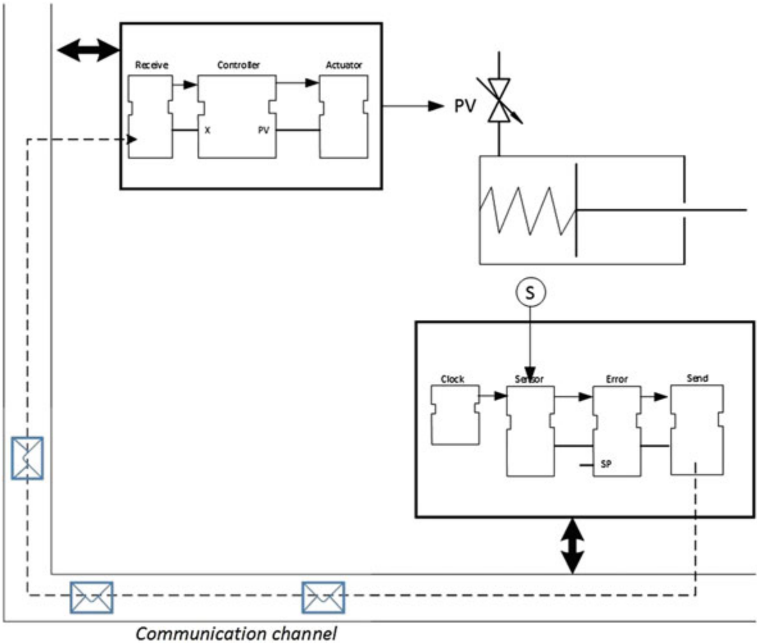


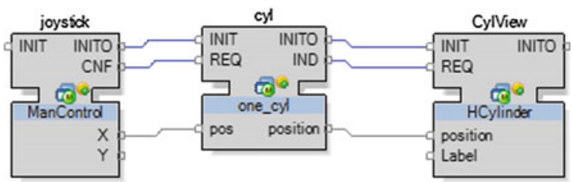
Fig. 3 Model of pneumatic cylinder with a network-connected actuator

5 Modelling of the Cylinder with PID Control Application in SMV

Figure 4 shows the top-level model in nxtStudio. The *one_cyl* function block type is the combined model of cylinder and controller which is to be converted to SMV model for verification. Joystick (FB type *ManControl*) and *CylView* (FB type *HCylinder*) are HMI modules used for real-time simulation in nxtStudio and are not used for formal verification. However, to emulate input setpoint change (*pos* input of the model), some additional SMV code was added manually to the model. Figure 5 shows the FB network of the verified model—combined cylinder and control.

The SMV model for the presented function blocks is generated automatically using the fb2smv [11] converter, but as it does not fully support SMV models of IEC

Fig. 4 Function block model of the cylinder CPS



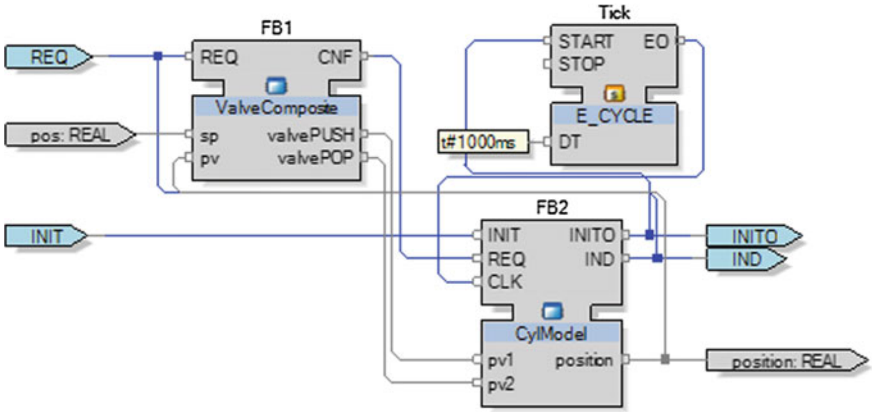


Fig. 5 Closed-loop model

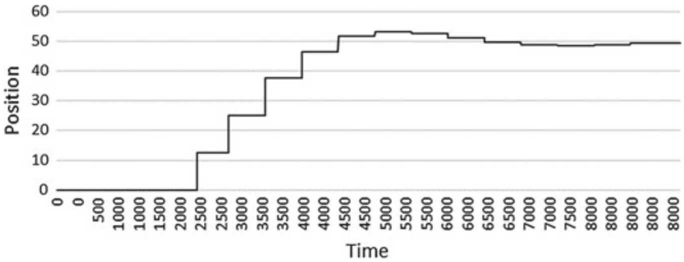


Fig. 6 Cylinder position plot built from the counterexample data

61499 function blocks with time delay, additional code implementing time scheduler [12] was added manually as shown in Fig. 2. The input set point initially has value 0 and at the time moment when *Tglobal = 500* it is changed to *50*.

Figure 6 shows the cylinder position plot built from counterexample data for the SMV model with parameters as follows: time bound *Tmax = 8000*, sampling time *E_CYCLE.DT = 500*. All time values are given in discrete model time units, which can be considered as milliseconds on real time scale. The counterexample, generated by nuXmv model checker is converted to table format in .csv file and MS Excel is used to draw the plot.

6 Modelling Random Time Delay in IEC 61499

Following the approach of [27] we introduce the time-stamped events concept to the application and compare its performance with the reference cases. In [27] the comparison was done using the Ptolemy II modelling environment (Fig. 7). In this

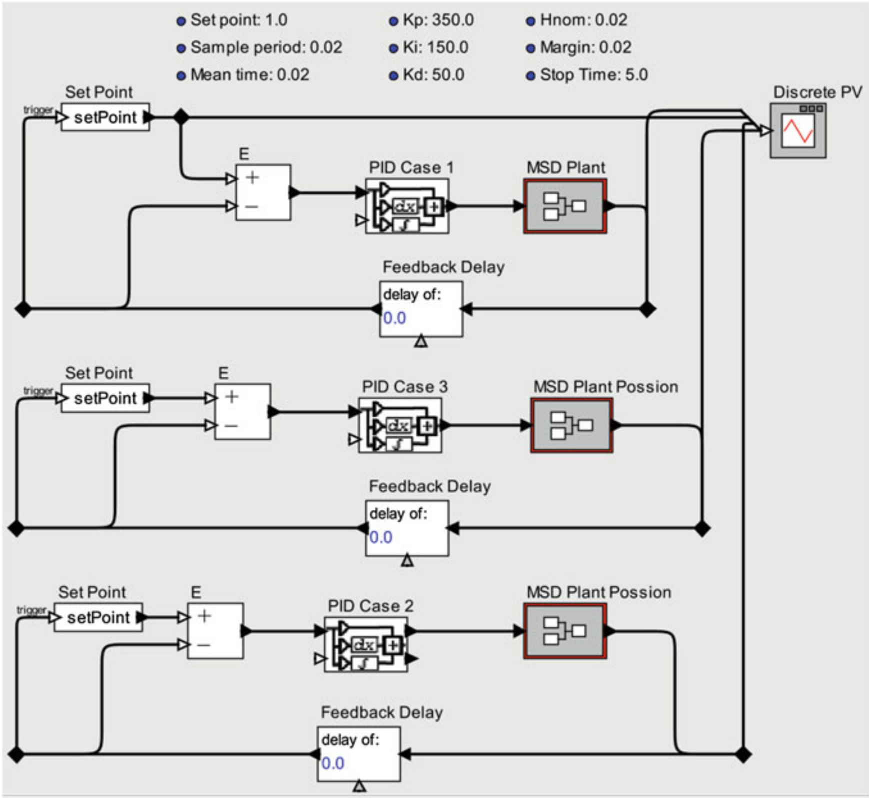


Fig. 7 PtolemyII model of three communication delay cases for comparison [27]

paper we make a step towards implementation and implement the same setup in IEC 61499. As IEC 61499 does not support timestamp and random event delay emulation, the application was modified to implement timestamps through standard input-output data relations. Two CAT blocks: *GetTimestamp* and *TimeDiff* are introduced and added to the cylinder-control model together with the *E_DELAY* function block which emulates communication delay between sensor and controller (see Fig. 8). *GetTimestamp* CAT block outputs the time when *REQ* event was triggered. *TimeDiff* block calculates the difference between input timestamp and current system time when its *REQ* event is triggered and outputs this time difference to the controller. Figure 9 shows the nxtStudio model of three cases combined, the ideal case without any communication delay, the case with communication jitter observed for the controlled with fixed sampling rate, and the case with delay and timestamps, where the controller is using the DT parameter based on the actual communication time calculated using timestamps.

The delay randomizer CAT block (*randomizer1*) is added to the model to generate random communication delay value synchronously for two cases: cylinder model

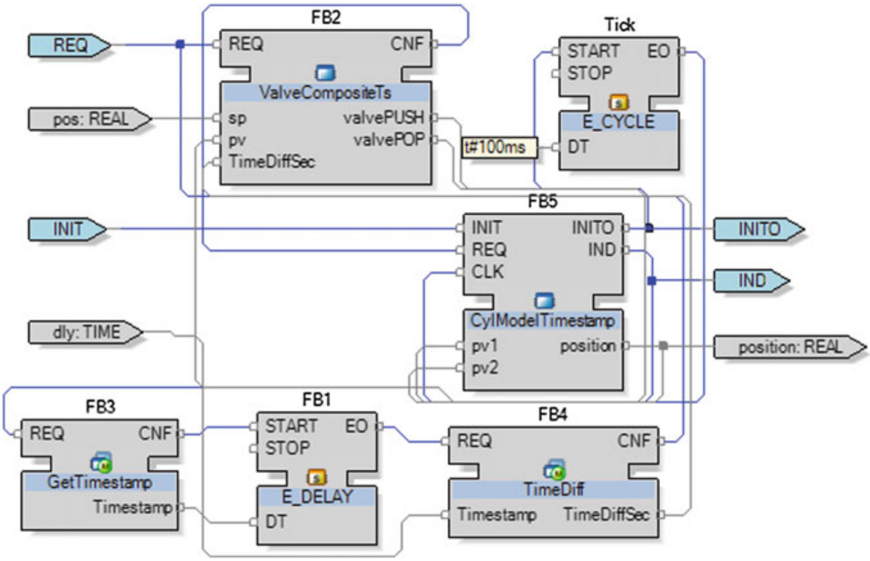


Fig. 8 IEC 61499 cylinder model with PID control and communication delay

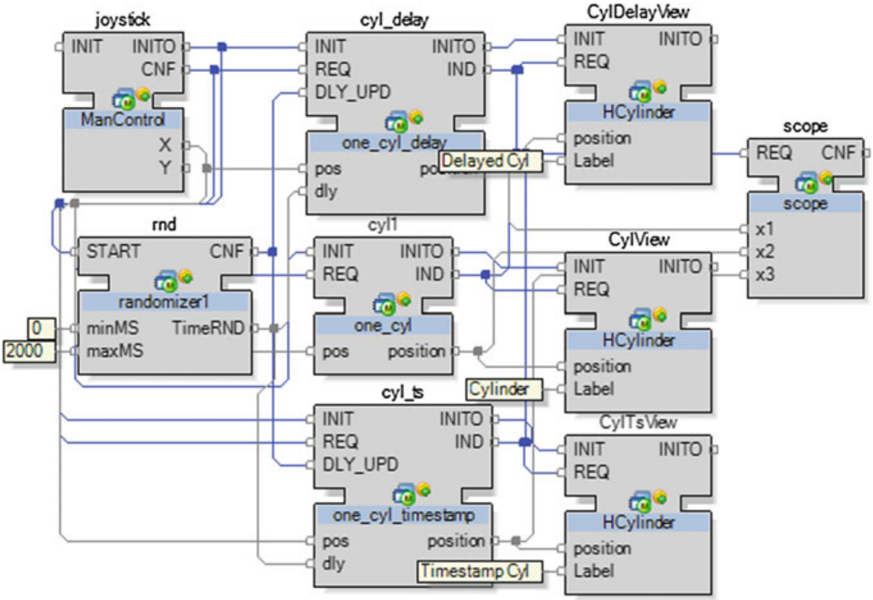


Fig. 9 nxtStudio model of three cases

with delay and model with delay and timestamps. Outputs from all three cases are combined into one plot (drawn by the *scope* block) to make the comparison. The *randomizer1* block, after START event is triggered, generates a new random delay value within the range [0;2000] milliseconds every 100 ms and passes this value to delayed cases. This assures that the modelled delays are always similar for both delayed cases and plots comparison is correct.

In SMV, the random delay is modelled by leaving the *E_DELAY* input *DT* unassigned; in this case, the verifier will consider all possible delay values. However, to reduce the model state space when the delay range is relatively wide, the *DT* input can be assigned only with some key values instead of all possible values within the range.

7 Conclusion

Preliminary results on modelling of cyber-physical systems with jitter in the SMV modelling language are presented. The paper exemplifies the methodology using a simple cylinder example. The initial results are promising both in terms of modelling the jitter in CPS by means of SMV, and modelling the mitigation measures such as use of the time-stamped semantics in IEC 61499 and subsequent modelling of it in SMV. Future work will include definition of metrics of cyber-physical agnosticism and their representation in specification languages supported by the nuXmv model-checker. These steps will be followed by comprehensive verification trials of the CPA properties of CPS controlled by the IEC 61499 extended by the time-stamped events semantics.

Acknowledgements This work was supported, in part, by the Vetenskapetsråd (Sweden) project DIACPA D.nr. 2015-04675.

References

1. Årzén, K.E.: A simple event-based pid controller. In: 14th IFAC World Congress (1999)
2. Åström, K.J.: Event based control. In: Analysis and Design of Nonlinear Control Systems, pp. 127–147 (2008)
3. Carloni, L.P., McMillan, K.L., Sangiovanni-Vincentelli, A.L.: Theory of latency-insensitive design. IEEE Trans. Comput. Aided Des. Integr. Circuits Syst. 1059–1076 (2001)
4. Dai, W., Christensen, J.H., Vyatkin, V., Dubinin, V.: Function block implementation of service oriented architecture: case study. In: 2014 12th IEEE International Conference on Industrial Informatics (INDIN), pp. 112–117. IEEE (2014)
5. Dai, W., Peltola, J., Vyatkin, V., Pang, C.: Service-oriented distributed control software design for process automation systems. In: 2014 IEEE International Conference on Systems, Man and Cybernetics (SMC), pp. 3637–3642. IEEE (2014)

6. Dai, W., Vyatkin, V., Christensen, J.H.: The application of service-oriented architectures in distributed automation systems. In: 2014 IEEE International Conference on Robotics and Automation (ICRA), pp. 252–257. IEEE (2014)
7. Dai, W., Vyatkin, V., Christensen, J.H., Dubinin, V.N.: Bridging service-oriented architecture and IEC 61499 for flexibility and interoperability. IEEE Trans. Industr. Inf. 771–781 (2015)
8. Dai, W., Vyatkin, V., Pang, C., Christensen, J.H.: Time-stamped event based execution semantics for industrial cyber-physical systems. In: 2015 IEEE 13th International Conference on Industrial Informatics (INDIN), pp. 1263–1268. IEEE (2015)
9. Demin, E., Dubinin, V., Patil, S., Vyatkin, V.: Automation services orchestration with function blocks: Web-service implementation and performance evaluation. In: Service Orientation in Holonic and Multi-Agent Manufacturing, pp. 213–221. Springer (2016)
10. Derhamy, H., Drozdov, D., Patil, S., van Deventer, J., Eliasson, J., Vyatkin, V.: Orchestration of arrowhead services using IEC 61499: Distributed automation case study. In: Proceedings of 2016 IEEE 20th International Conference on Emerging Technologies & Factory Automation (ETFA 2016). IEEE International Conference on Emerging Technologies and Factory Automation. Proceedings, IEEE (2016)
11. Drozdov, D.: fb2smv tool. https://github.com/dmitrydrozdov/fb2smv
12. Drozdov, D., Patil, S., Dubinin, V., Vyatkin, V.: Formal verification of cyber-physical automation systems modelled with timed block diagrams. In: 2016 IEEE 25th International Symposium on Industrial Electronics (ISIE). IEEE (2016)
13. Gerber, C., Ivanova-Vasileva, I., Hanisch, H.M.: Formal modelling of IEC 61499 function blocks with integer-valued data types. Control Cybern. 197–231 (2010)
14. Hanisch, H.M., Hirsch, M., Missal, D., Preuße, S., Gerber, C.: One decade of IEC 61499 modeling and verification–results and open issues. In: IFAC Proceedings Volumes, pp. 211–216 (2009)
15. Pang, C., Dai, W., Vyatkin, V.: Towards IEC 61499 models of computation in ptolemy ii. In: Industrial Electronics Society, IECON 2015-41st Annual Conference of the IEEE, pp. 001988–001993. IEEE (2015)
16. Pang, C., Vyatkin, V.: Automatic model generation of IEC 61499 function block using net condition/event systems. In: 2008 6th IEEE International Conference on Industrial Informatics, pp. 1133–1138. IEEE (2008)
17. Patil, S., Vyatkin, V., Pang, C.: Counterexample-guided simulation framework for formal verification of flexible automation systems. In: 2015 IEEE 13th International Conference on Industrial Informatics (INDIN), pp. 1192–1197, July 2015
18. Patil, S., Drozdov, D., Dubinin, V., Vyatkin, V.: Cloud-Based Framework for Practical Model-Checking of Industrial Automation Applications, pp. 73–81. Springer International Publishing, Cham (2015)
19. Patil, S., Dubinin, V., Vyatkin, V.: Formal verification of IEC 61499 function blocks with abstract state machines and smv–modelling. In: 2015 IEEE Trustcom/BigDataSE/ISPA, vol. 3, pp. 313–320. IEEE (2015)
20. Patil, S., Dubinin, V., Vyatkin, V.: Formal modelling and verification of IEC 61499 function blocks with abstract state machines and smv–execution semantics. In: International Symposium on Dependable Software Engineering: Theories, Tools, and Applications, pp. 300–315. Springer (2015)
21. Patil, S., Zhabelova, G., Vyatkin, V., McMillin, B.: Towards formal verification of smart grid distributed intelligence: Freedm case. In: IECON 2015-41st Annual Conference of the IEEE Industrial Electronics Society, pp. 003974–003979. IEEE (2015)
22. Sorouri, M., Patil, S., Salcic, Z., Vyatkin, V.: Software composition and distributed operation scheduling in modular automated machines. IEEE Trans. Industr. Inf. 865–878 (2015)
23. Stanica, M., Guéguen, H.: Using timed automata for the verification of IEC 61499 applications. In: Discrete Event Systems 2004 (WODES'04): A Proceedings Volume from the 7th IFAC Workshop, Reims, France, 22–24 Sept 2004, p. 375. Elsevier (2005)
24. Vyatkin, V., Hanisch, H.M.: Development of adequate formalisms for verification of IEC 1499 distributed applications. In: SICE 2000. Proceedings of the 39th SICE Annual Conference. International Session Papers, pp. 73–78. IEEE (2000)

25. Vyatkin, V., Hanisch, H.M.: A modeling approach for verification of IEC1499 function blocks using net condition/event systems. In: 1999 7th IEEE International Conference on Emerging Technologies and Factory Automation, 1999. Proceedings. ETFA'99, vol. 1, pp. 261–270. IEEE (1999)
26. Vyatkin, V., Hanisch, H.M.: Verification of distributed control systems in intelligent manufacturing. J. Intell. Manuf. 123–136 (2003)
27. Vyatkin, V., Pang, C., Tripakis, S.: Towards cyber-physical agnosticism by enhancing IEC 61499 with ptides model of computations. In: Industrial Electronics Society, IECON 2015-41st Annual Conference of the IEEE, pp. 001970–001975. IEEE (2015)
28. Zhabelova, G., Patil, S., Yang, C.W., Vyatkin, V.: Smart grid applications with IEC 61499 reference architecture. In: 2013 11th IEEE International Conference on Industrial Informatics (INDIN), pp. 458–463, July 2013

Industrial Cyber Physical Systems Supported by Distributed Advanced Data Analytics

Jonas Queiroz, Paulo Leitão and Eugénio Oliveira

Abstract The industry digitization is transforming its business models, organizational structures and operations, mainly promoted by the advances and the mass utilization of smart methods, devices and products, being leveraged by initiatives like Industrie 4.0. In this context, the data is a valuable asset that can support the smart factory features through the use of Big Data and advanced analytics approaches. In order to address such requirements and related challenges, Cyber Physical Systems (CPS) promote the development of more intelligent, adaptable and responsiveness supervisory and control systems capable to overcome the inherent complexity and dynamics of industrial environments. In this context, this work presents an agent-based industrial CPS, where agents are endowed with data analysis capabilities for distributed, collaborative and adaptive process supervision and control. Additionally, to address the different industrial levels' requirements, this work combines two main data analysis scopes: at operational level, applying distributed data stream analysis for rapid response monitoring and control, and at supervisory level, applying big data analysis for decision-making, planning and optimization. Some experiments have been performed in the context of an electric micro grid where agents were able to perform distributed data analysis to predict the renewable energy production.

Keywords Cyber physical systems · Data analytics · Multi-agent systems

---

J. Queiroz (✉) · E. Oliveira
Faculty of Engineering, University of Porto, 4200-465 Porto, Portugal
e-mail: jonas.queiroz@fe.up.pt

E. Oliveira
e-mail: eco@fe.up.pt

J. Queiroz · P. Leitão · E. Oliveira
LIACC - Artificial Intelligence and Computer Science Laboratory, 4169-007 Porto, Portugal
e-mail: pleitao@ipb.pt

P. Leitão
Polytechnic Institute of Bragança, Campus Sta Apolónia, 5300-253 Bragança, Portugal

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_5

1 Introduction

The technological advances and reduced costs for collecting, storing, sharing and analyzing data, as well as the mass adoption of digital and internet based services by consumers, enterprises and governments have led to a digital transformation of the economy [1]. This process, called digitization, is drastically affecting companies and industries, requiring the transformation and reshaping of their business models, organizational structures, as well as their operations, where the traditional relationship they have with their products and customers has been changed to become continuous and unbounded, throughout the products' life-cycle [2]. In this domain, the digitization has been powered and promoted by the advances and the mass utilization of embedded and networked sensors and cloud computing technologies. It has also contributed to a considerable increase in the access and availability of diverse types of data and information, as well as with the emergence of the smart devices and products, which are mainly researched within the context of the Internet of Things (IoT) and Services [3].

Furthermore, these concepts go beyond the industrial domain (smart factories, machines and products), covering buildings and homes (smart buildings) and even complete cities and their infrastructures (smart cities, transportation and grids). In the digitized industry, data represents the main factor responsible for the changes and transformation in both its operations and businesses and because of this it should be understood and actively managed as a valuable asset that can leverage the generated value [4]. In this sense, the data from smart machines and products has a great value, which can be further increased when integrated and combined with other data, e.g., from historical and others external, contextual and enterprise information [2]. Additionally, the high volume of data, which some years ago was underused mainly because of the lack of tools and expertise to process and analyse it, nowadays can have its fully value extracted through the use of Big Data and advanced analytics approaches [5, 6].

In this context, data analysis has been widely applied in industrial domain, e.g., at operational level for process monitoring, diagnosis, optimization and control, and at business level for customer relationship management, supply chain, sales and others [7, 8]. For example, the continuous analysis of the available data in industrial environment can enhance the operation management, as well as online products reviews can be analysed to improve the way companies treat their products and customers, optimizing and enhancing their profits. According to [4] the use of Big Data and advanced analytics can result in 20 to 25 percentage increase in production volume and up to a 45 percentage reduction in downtimes.

These new concepts of smart machines and products require industries to build and support several new technologies and infrastructures. Among them, data management (e.g., retrieval, integration and analysis) comprises an essential requirement to support the smart capabilities and features, which pose several challenges, regarding mechanisms to integrate distributed, heterogeneous, dynamic, and stream data sources [6]. Also, a cloud-based infrastructure can support all the remote and local software features, such as the monitoring of health and performance, diagnostic, remote control, optimization and reconfiguration strategies and algorithms, as well as to enable the autonomy of smart products and machines, enabling them to learn and adapt to their environment, user preferences and operate on their own, which are essential requirements to achieve Industrie 4.0 goals [9, 10]. All these aspects should be combined to develop supervisory and control systems capable to support the management of the large amount of heterogeneous, distributed and dynamic components, as well as the processes and operations, which are constantly subject to disturbances or evolving to attend new requirements, optimize and ensure the quality of outcomes and reduce the downtimes. In this context, Cyber Physical Systems (CPS) [10] propose the integration of physical and virtual worlds to support all these requirements and capabilities, by embedding computational elements in physical entities and connecting such entities in a cloud-based infrastructure, aiming to provide a more effective management of the physical environment and their processes.

In order to realize the CPS features, namely self-adaptation, reconfiguration, responsiveness, fault tolerance, automated diagnosis and proactive maintenance [10, 11], Multi-agent systems (MAS) [12] have being pointed as a suitable approach. In MAS, several autonomous, collaborative and self-organizing decision-making entities called agents interact and exchange knowledge to achieve their goals [12]. Additionally, MAS employ standardized communication interfaces and protocols which ensure more flexibility, modularity and openness for such systems. The application of agent-based technology in the industrial domain to solve problems related to production automation and control, supervision and diagnosis, production planning, and supply chain and logistics, is surveyed in [13, 14], and has been covered by the field of Industrial Agents [15].

While most of existing works focus on the design of CPS control approaches, this work intends to contribute with the issues and challenges related to the supervisory aspects, considering the advanced data analysis as a key enabler for these systems' smart features. The main objective is to provide a conceptual model to support more intelligent and adaptive supervisory and control industrial CPS. For this purpose, this work addresses two industrial levels, combining two main data analysis scopes: (1) *at operational level*, applying distributed data stream analysis for rapid response monitoring and control, and (2) *at supervisory level*, applying more robust and big data analysis for decision-making, planning and optimization. This advanced distributed and collaborative data analysis approach proposes to endow agents with data analysis capabilities and cooperation strategies, enabling them to perform local and collaborative data analysis, continuously improve and dynamically adapt their capabilities, based on the aggregation of knowledge. Some experiments in the context of an electrical micro grid have being performed where the preliminary results show that agents are able to perform distributed predictive data analysis of renewable energy production.

The rest of the paper is organized as follows. Section 2 presents the main concepts of the proposed approach for the distributed advanced data analysis. Section 3 overviews and discusses the preliminary experiments and results by considering an electrical micro grid case study. Finally, Sect. 4 rounds up the paper with the conclusions and points out the future work.

2 Combining Data Analysis and MAS to Realize Industrial CPS

Considering the issues previously discussed, this work intends to design and develop an agent-based data analysis approach towards a flexible and adaptive industrial supervisory and control CPS, capable to cope with the requirements of Industrie 4.0. Additionally, this approach is more concerned with the supervisory and monitoring aspects than with the system and process control, covering the models and mechanisms to obtain and provide the required information for process management and decision-making, based on the data of different industrial automation levels.

2.1 Agent-Based Data Analysis Powering Industrial CPS

The design of the proposed approach requires the consideration of some essential requirements and features related to ongoing and upcoming challenges and issues raised by the Industrie 4.0 vision, as illustrated in the left side of Fig. 1. Such features cover different industry automation levels that can be generalized in the operational level characterized by the monitoring and control of processes, and the supervisory and business level, characterized by the supervision of the whole plant and its operation, integrated with the business management. These two levels present specific characteristics and requirements which are covered by CPS principles (Fig. 1—centre), where the operational level is mainly related with the physical world, characterized by the IoT and their smart devices, also demanding the processing and analysis of real time data streams [16] in order to attend the rapid response monitoring and control requirements. On the other hand, the supervisory level is hosted in a virtual world defined by a cloud-based infrastructure where robust software applications are used to attend the requirements of complex system management and high level information for decision-making, supported by big data analytics.

In this approach, *MAS* and *Data Analysis* (*DA*) (Fig. 1—right) are the basis technologies proposed to address the requirements and features envisioned by Industrie 4.0, and also taking in consideration the CPS principals. The first provides the conceptual framework to realize the underline system infrastructure that is required to achieve the desired flexibility and adaptability levels, while the second, provides the proper tools capable to analyse and obtain the required information to fulfil the desired system functionalities, also taking advantage of the increased data

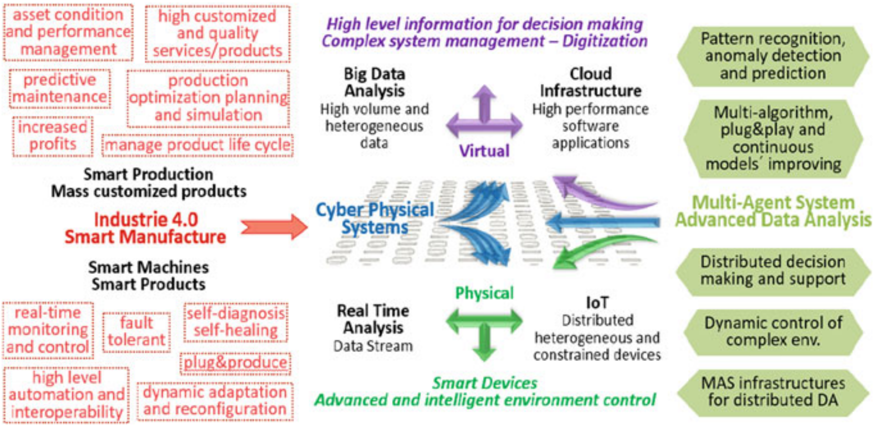


Fig. 1 Essential requirements and features of the proposed approach

availability. In general, MAS and data analysis have been used successfully but separately to address several issues in industrial domain. In particular, MAS is used to develop adaptive and intelligent control systems, while data analysis provides effectively data-driven decision-making algorithms. In this sense, several works leverage and discuss the potentials and how the integration of them can provide better solutions [17].

Besides combining MAS with DA, this approach also intends to address two data analysis scopes: Big Data and Data Stream analysis. The first is related with the analysis of great volumes of heterogeneous data to extract valuable information for supporting the decision making, optimization and planning, while second is related with the analysis of the continuous incoming operational data, at real or near real-time, providing simpler information, but addressing the process' monitoring and control rapid response requirements.

In this context, there are already some works that cover the combination of these data analysis levels. For example, the *lambda architecture* that is proposed in [18] to address Big Data batch processing and real time data stream processing was adapted by [19] for an architecture of recommender systems, and also by [20] for an architecture to a data analysis system for a vehicular network. Other works discuss similar approaches, e.g., in [21] a distributed highway traffic stream mining system is presented that considers a central server to perform more robust DA tasks and support the monitoring components at lower level.

In this context, other desired features that can be achieved by the combination of MAS and data analysis include (see Fig. 1—right): *MAS infrastructures for distributed DA*, and *multi-algorithm, plug&play and continuous models' improving*. The first focuses on the use of MAS architectures and organizations to support and enhance the various data analysis phases, providing a flexible and scalable data analysis infrastructure. For example, agents can be employed to perform, in a distributed and cooperative way, the data retrieval, pre-processing, integration and analysis. The second comprises three related aspects which focus on the use of MAS to provide a dynamic and adaptive infrastructure to perform DA. For instance, the use of multiple DA algorithms and models, e.g., one per agent, which can perform the same task over the data, and at the end the results could be combined to obtain more accurate information. Also, the use of MAS to provide an open and dynamic infrastructure that enables the seamless addition (plug&play) of new algorithms and data sources to the system, as well as mechanisms and algorithms that enable DA models to be continuously updated to fit the environment's dynamics.

While the previous features are more related to infrastructural aspects, there are also others related to industrial supervisory and control aspects. In Fig. 1, the *distributed decision-making and support* element comprises coordination and negotiation mechanisms and algorithms for agents based on distributed and local DA, and takes collaborative decisions for the monitoring and diagnosis system's conditions. In this sense, agents could use different DA algorithms and model to analyse environment variables, interacting with other agents to enhance or rectify the outputs. The *pattern recognition, anomaly detection and prediction* element represents the common application of DA for monitoring, prediction, diagnosis and to obtain additional information about the environment and the conditions of its elements, usually aiming at identifying problems and possible system improvements, while the *dynamic control of complex environments* element comprises the support of dynamic adaptation and optimization of operations and processes in face of changes in the operating conditions.

2.2 Agent-Based Model

Considering the features and requirements analysed in the previous section, an agent-based model is proposed (Fig. 2—left), comprising two layers of agents and a set of components that define the agents' capabilities (Fig. 2—right). In the left side of Fig. 2, at the lower layer, agents are in charge of stream data analysis providing simple information about the processes (e.g., operation status, triggers and events), but attending rapid response constraints. In this layer, each agent is responsible to retrieve and analyse the data from process devices, in order to support monitoring and control actions. These agents could be embedded into devices to perform distributed data analysis and intelligent monitoring, cooperating to identify problems or provide information about the system. At the upper layer, agents are responsible to process and analyse great amounts of historical and incoming data from plant operations, business and also contextual or external data, in order to provide information for high level decision-making, systems optimization or activities planning (e.g., performance, quality or degradation indicators, event diagnosis, trends and forecasts). These agents could be deployed in a cloud-based computing environment, taking advantage of such kind of infrastructure and other tools to perform their tasks and also to manage the lower level agents.

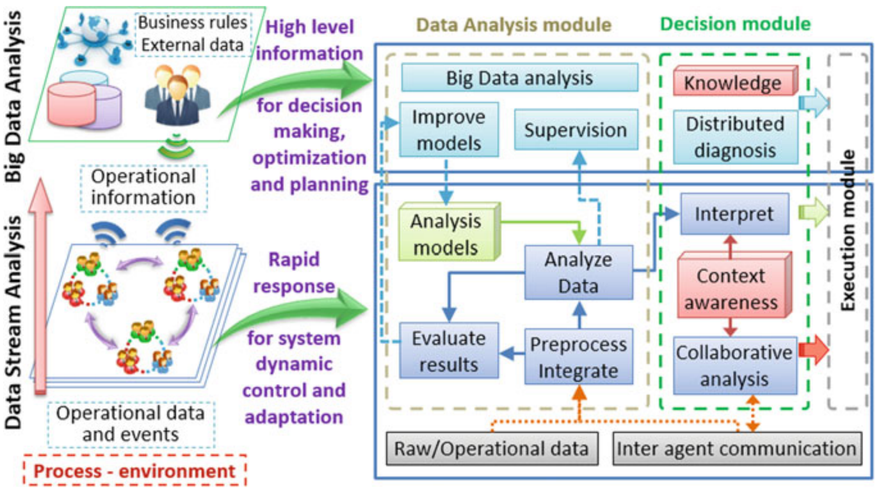


Fig. 2 Agent-based data analysis approach for adaptive industrial supervisory control systems

In this approach, the agents of each layer comprise three modules (Fig. 2—right) that group a set of specific components, which define the agent behaviours and capabilities. The *Data Analysis module* defines the components that perform DA tasks, the *Decision module* defines the components that process, organize and consolidate the analysis outputs, and the *Execution module* defines the components that use the consolidated information to act in the environment. Agents from both layers have two common components, *Raw/Operational data* and *Inter agent communication*, responsible to retrieve external data from the environment and manage the agent interaction, respectively.

The proposed model comprehends only two types of agents, the lower and upper layer agents, with their monitoring and supervisory capabilities, respectively. In this sense, this model doesn't suggest any specialized role for the agents, which is application dependent, i.e., given an application other capabilities can be added to each agent in order to address its role in the application. For example, in a manufacturing system, the agents responsible to monitor the machines should present additional components that are different from the agents that monitor the products. In the model presented in Fig. 2, the *Execution module* generalizes where these specific components should be defined. This module is not detailed here, since the components of this module are application dependent and also the focus of this project is in the data analysis aspects.

The components of lower layer agents were designed to provide desired features like collaborative and scalable data analysis for a real-time monitoring and control, as well as a local autonomy and intelligence, required for the system's dynamic adaptation. In this sense, the components of lower layer agents *Data Analysis* module comprise:

- *Preprocess Integrate*, which prepares the raw data to be analysed through the application of algorithms and methods for filtering, features extraction, data transformation and integration, as well as other pre-processing tasks that should be implemented by this component;
- *Analyze Data*, which performs several types of data analysis, such as classification, prediction, clustering. This component implements the methods required to continually apply the analysis models using the available data (provided by previous component), in order to obtain information that will be used by the agent to take actions and to monitor the related components. This component can be simple, using a single DA model, or more complex using and combining several data analysis models;
- *Analysis models*, which comprises all the data analysis models used by the agent *Analyze Data* component. One agent can have several models that can be used simultaneously (e.g., an agent could apply multiple models and combine the information for better results) or for specific context or environment condition, which should be defined by the *Context aware* component;
- *Evaluate results*, which assess the analysis model accuracy to improve the agent data analysis capabilities (e.g., by comparing the model output with a system feedback). The results of this process could be used by the own agent or sent to others, in order to improve its analysis models, if their accuracy were not good enough as before;

The *decision module* defines the components that process, organize and consolidate the analysis result, comprising:

- *Interpret*, which contextualizes and makes assumptions over the analysis outputs. It should implement the required decision making mechanism to enable the agent to handle with multiple, complementary, contradictory or incomplete information;
- *Collaborative analysis*, which realizes if the agent needs any kind of information that can be provided by other agents, taking the required actions by interacting with other agents, to support its analysis tasks, decisions and actions;
- *Context aware*, which provides a local knowledge to support the tasks of the other components. It provides all kinds of contextual information, such as the current state or condition of the environment, its components and operations, including system and user requirements and business rules, in a passive or active way.

Like in the lower layer, the agents' components of upper layer were designed to attend some desired features regarding the analysis of aggregated data from different sources in order to obtain high level information for assisting the decision-making, system optimization and planning tasks. In this sense, the *data analysis* module comprehends the following components:

- *Supervision*, which receives data analysis outputs and monitoring information from lower layer agents and uses it to obtain the status of a component, an operation or the whole system. This higher level information is used by these agents to support their tasks;

- *Improve models*, which (re)builds or (re)trains data analysis models used by lower layer agents, based on their feedback. This component can provide information about the confidence of the analysis outputs provided by lower level agents;
- *Big Data analysis*, which considers data from different sources, including contextual, external and historical data, as well as current data and events coming from lower layer agents (that can be firstly aggregated by the Supervision component), in order to extract information for a broader context;

The *decision module* of upper layer agents is responsible to monitor the system components (e.g., agents or devices) to support the system dynamic adaptation and comprises the following components:

- *Knowledge*, which keeps the knowledge related to operational and technical characteristics and constraints associated to some parts or the whole system;
- *Distributed diagnose*, which compiles and provides information about the conditions of some parts or whole system, and suggests actions and their possible consequences in the system (what-if information). It also considers the information provided by the other components to interact with other upper layer agents to collaboratively identify and diagnose the whole system conditions.

3 Preliminary Experiments and Results

The described approach is being validated on a case study in the context of an electric micro grid comprising some wind turbines and photovoltaic panels. In micro grids, distributed and flexible approaches capable to manage the local requirements and autonomy of each micro grid node and at the same time assure the global micro grid energy efficiency and self-sustainability are needed. In this context, one challenge is the integration and analysis of multiple distributed data streams produced by energy resources (micro grid nodes and their energy elements). Another challenge regards the prediction of renewable energy production and consumption that are directly dependent on the intermittent weather conditions. It means that the accuracy of the predictions is lower for future time periods, requiring the dynamic adaptation of the system for the current conditions.

In this sense, the proposed approach can address the global and local requirements and features. The use of agent-based approach can provide the required distribution and flexibility to manage the autonomy of each grid node and their energy elements. Additionally, the use of advanced data analysis approaches can provide the appropriate tools that can take advantage of the large amount of data streams produced by the energy elements to obtain the necessary information to

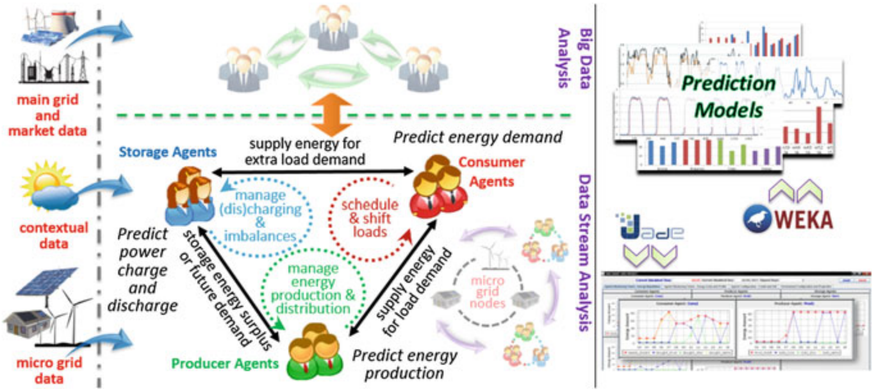


Fig. 3 Micro grid agent-based data analysis developed experiment

support the management and control of the energy production, distribution, storage and consumption.

The performed experiments only covered the lower layer aspects of the proposed approach. In this layer, three types of agents were developed (see Fig. 3): Storage (SA), Consumer (CA) and Producer (PA) agents, responsible to manage each kind of electric elements. In this sense, each kind of agent can present different data analysis capabilities according with its roles. For instance, CAs can perform the prediction of the energy demand based on the user profile, while PAs can predict the renewable energy production based on the weather conditions [22]. This anticipated information could be used to support the energy distribution along elements, as well as optimize the schedule/shift of loads, minimize peak demand and even negotiate in an energy market or provide information to the main grid operator manage the high scale energy generation. In the current experiments the upper layer agents were not developed, however such information can be send to these agents that will be in charge to perform such tasks.

In order to test the proposed approach, a simple simulation environment was developed, where agents were implemented using JADE framework and data analysis models using WEKA Java API. In this environment SAs and CAs implement simple and static battery and consumption models, while PAs use real data from micro grid photovoltaic (PV) panels and wind turbines to simulate the energy production [22]. Also for the PAs different data analysis models were developed for the prediction of renewable energy production using historical data, as well as external data from weather forecasts. The historical data used in the experiments comprehends three years of monitoring the renewable energy production operations, which record one sample at each 5 min. The development of prediction models was performed using four different algorithms provided by WEKA (M5P, M5Rules, Linear Regression, and Multilayer Perceptron) [23].

The experimental results showed that PAs are capable to perform distributed data analysis for different time scopes and goals. For instance, PAs performed

Table 1 The prediction performance for renewable energy production

	PV system			
	Current step	1 step ahead	3 steps ahead	Next hour
Corr.Coef.	0.990	0.991	0.977	0.951
RAE (%)	4.635	10.899	15.723	25.997
	Wind system			
Corr.Coef.	0.892	0.839	0.779	–
RAE (%)	38.157	40.252	46.128	–

short-term prediction of the energy production for the current 5 min period, 1 step ahead (next 5 min period) and 3 steps ahead. The first aims at monitoring the operational conditions of the production units to identify abnormality in energy production, while the second and third aim at providing information to correct energy imbalances. Also, through a mid-term prediction of energy production, performed by integrating external weather forecasting data, PAs were able to provide information about the amount of energy expected to be produced for the next hour interval, which could be used by engineers, grid operators and other systems to enhance and optimize the energy distribution [22]. During the energy predictions, agents were able to continuously evaluate and improve their analysis models.

Table 1 summarizes the results of the developed experiments, illustrating the accuracy of the renewable energy production prediction, based on the models built by M5P algorithm, which was the one that in general presented better results. The high correlation coefficients (*Corr.Coef.*) illustrate the applicability of DA for renewable energy production, while the relative absolute error (*RAE*) illustrates that the prediction models presented better results for the short-term and the PV system, which can be justified by the high instability of weather conditions which is higher for the wind system. Even presenting a higher error rate for the long-term prediction, using short-term prediction, the system can dynamically adapt its operation to correct the imbalances. Most of the prediction deviation that was responsible for the higher RAE in the PV system was observed in cloudy days. It means that the prediction models should be tuned for these kinds of days, or even specific models can be built for such days. In general, the wind system presented bad results, when the days with high variable wind speed were the main responsible. Because of such bad results the experiments were not performed for the next hour period.

4 Conclusions and Future Work

The increased industrial digitization and market demands for high customized products and services have posed new requirements and challenges for industries and companies. The Industrie 4.0 initiative is promoting the related concepts and their benefits in order to improve the management of manufacturing processes and operations, optimizing and ensuring the quality of outcomes and reducing the downtimes. Among these technologies, the advanced DA and MAS can be used to attend the Industrie 4.0 vision, contributing for the realization of industrial CPS features, namely self-adaptation, reconfiguration, responsiveness, fault tolerance, automated diagnosis and proactive maintenance. In industrial domains, MAS have been used as a suitable approach to design and develop flexible and adaptable industrial control systems, while data analysis is being used to provide effective algorithms to support data-driven decision-making. In this context, this work proposes the combination of these two technologies to design an agent-based data analysis approach for intelligent and adaptive industrial supervisory control systems, where agents are endowed with data analysis capabilities to enable the dynamic distributed and collaborative process supervision and control. Additionally, to address the different industrial automation levels' requirements, the proposed approach covers two data analysis scopes, applying data stream analysis at operational level, attending the monitoring and control rapid response requirements combined with a more robust Big Data analysis at supervisory and business levels to support decision making, planning and optimization tasks.

The preliminary experiments showed promising results, where several data analysis capabilities were successfully embedded in agents, including the self-improving of data analysis model. The agents were able to perform in distributed and collaborative mode the prediction of renewable energy production for different time scopes, and use such information for the management of micro grid nodes.

Future work encompasses the detailed specification of the mechanisms and strategies to cover the more advanced aspects and features of the proposed approach. Regarding the case study, it can be further explored, e.g., by developing predictive capabilities for CAs and SAs in order to manage energy consumption and power storage of micro grids nodes, also extending the preliminary experiments in order to validate and assess other aspects. Moreover, it is intended to explore another case studies scenario in the manufacturing domain.

References

1. Sabbagh, K., Friedrich, R., El-Darwiche, B., Singh, M., Koster, A.: Digitization for economic growth and job creation: regional and industry perspective. In: The global information technology report 2013—Growth and Jobs in a Hyperconnected World, Geneva, World Economic Forum, pp. 35–42 (2013)
2. Porter, M.E., Heppelmann, J.E.: How smart, connected products are transforming companies. Harvard Bus. Rev. 96–112 (2015)
3. Aggarwal, C.C., Ashish, N., Sheth, A.P.: The internet of things: a survey from the data-centric perspective. In: Managing and Mining Sensor Data, pp. 383–428. Springer (2013)
4. Wee, D., Kelly, R., Cattel, J., Breunig, M.: Industry 4.0—How to Navigate Digitization of the Manufacturing sector. McKinsey & Company (2015)

5. Sagiroglu, S., Sinanc, D.: Big data—a review. In: International Conference on Collaboration Technologies and Systems (CTS) (2013)
6. Obitko, M., Jirkovský, V., Bezdíček, J.: Big data challenges in industrial automation. In: Industrial Applications of Holonic and Multi-Agent Systems, pp. 305–316. Springer (2013)
7. Qin, S.J.: Survey on data-driven industrial process monitoring and diagnosis. Annu. Rev. Control **36**(2), 220–234 (2012)
8. Harding, J.A., Shahbaz, M., Kusiak, A.: Data mining in manufacturing: a review. J. Manuf. Sci. Eng. **128**(4), 969–976 (2006)
9. Drath, R., Horch, A.: Industrie 4.0: hit or hype? [Industry Forum]. IEEE Ind. Electron. Mag. **8** (2), 56–58 (2014)
10. Lee, E.: Cyber physical systems: design challenges. In: 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing, pp. 363–369 (2008)
11. Lee, J., Lapira, E., Bagheri, B., Kao, H.A.: Recent advances and trends in predictive manufacturing systems in big data environments. Manuf. Lett. **1**(1), 38–41 (2013)
12. Wooldridge, M.: An Introduction to Multiagent Systems. Wiley (2009)
13. Leitão, P.: Agent-based distributed manufacturing control: a state-of-the-art survey. Eng. Appl. Artif. Intell. **22**(7), 979–991 (2009)
14. Metzger, M., Polakow, G.: A survey on applications of agent technology in industrial process control. IEEE Trans. Industr. Inf. **7**(4), 570–581 (2011)
15. Leitão, P., Karnouskos, S.: Industrial Agents: Emerging Applications of Software Agents in Industry. Morgan Kaufmann (2015)
16. Gaber, M.M., Zaslavsky, A., Krishnaswamy, S.: Mining data streams: a review. ACM SIGMOD Rec. **34**(2), 18–26 (2005)
17. Cao, L., Gorodetsky, V., Mitkas, P.: Agent mining: the synergy of agents and data mining. IEEE Intell. Syst. **24**(3), 64–72 (2009)
18. Marz, N., Warren, J.: Big Data: Principles and Best Practices of Scalable Realtime Data Systems. Chapter 1, 1st edn., p. 328. Manning Publications (2014)
19. Twardowski, B., Ryzko, D.: Multi-agent architecture for real-time big data processing. In: International Joint Conference on Web Intelligence and Intelligent Agent Technology, pp. 333–337 (2014)
20. Daniel, A., Paul, A., Ahmad, A.: Near real-time big data analysis on vehicular networks. In: International Conference on Soft-Computing and Networks Security (ICSNS-2015) (2015)
21. Liu, Y., Choudhary, A., Zhou, J., Khokhar, A.: A scalable distributed stream mining system for highway traffic data. In: Proceedings of the 10th European Conference on Principle and Practice of Knowledge Discovery in Databases (PKDD’06) (2006)
22. Queiroz, J., Leitão, P., Dias, A.: Predictive data analysis driven multi-agent system approach for electrical micro grids management. In: Proceedings of the IEEE ISIE’16, pp. 738–743 (2016)
23. Witten, I.H., Frank, E., Hall, M.A.: Data Mining: Practical machine learning tools and techniques, 3rd edn. Morgan Kaufmann (2011)

Gap Analysis on Research and Innovation for Cyber-Physical Systems in Manufacturing

Anna De Carolis, Giacomo Tavola and Marco Taisch

Abstract In defining the Roadmap of the Research Priorities for adoption of CPS in manufacturing industry, it is crucial to identify the key elements preventing a fast and smooth transition from the current status to the desired one. In such complex environments characterized by many industrial sectors and processes, external factors and social/economical influences, it is important to address only the main issues to achieve the result. This paper is aiming to illustrate the results of the Gap Analysis activities carried out in the sCorPiuS (http://scorpius-project.eu/) project.

Keywords Cyber-physical systems · Gap analysis · Research roadmap · Manufacturing industry

1 Introduction

Manufacturing is a key asset for Europe: it is a backbone for research, innovation, productivity, job and GDP creation and exports. De-Industrialization hit Europe, but EU initiatives decided since the beginning of the crisis are aiming to reverse the trend. Cyber-Physical Systems (CPS) are one of the key and most promising enabling technologies for the transformation of manufacturing companies towards the fourth industrial revolution [1, 9]. In fact, their intrinsic potentialities allow them to obtain a relevant role for the digitalization of manufacturing environment, providing higher value essentially in three main dimensions: smart product, smart manufacturing and changes in business models [6]. Such opportunities are guaranteed by CPS' ability to integrate physical objects with computing, memorizing

A. De Carolis (✉) · G. Tavola · M. Taisch
Politecnico di Milano, Via Lambruschini 4/B, 20156 Milan, Italy
e-mail: anna.decarolis@polimi.it

G. Tavola
e-mail: giacomo.tavola@polimi.it

M. Taisch
e-mail: marco.taisch@polimi.it

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_6

and communicating elements supported by actuators and connected each other via web. Several challenges need to be addressed in order to spread the adoption of CPS allowing achieving major breakthroughs, both from the technical (i.e. security, human interaction, robust connectivity) and business perspectives [2]. Additionally, specific types of Obstacles related with macro-dynamics (social, economic or technological) or with legal compliance have been found. For this reason, these breakthroughs and obstacles have been grouped in clusters defined in order to better scope and identify the potential existing gaps which limit CPS implementation. These are the six main clusters identified in sCorPiuS State of the Art on Cyber-Physical Systems, 2016 [7]:

(1) *New data-based services and business models*, which refers to the CPS ability to open new business opportunities, allowing companies to be closer to customers' needs by offering them high customized and value added services;
(2) *Data-based improved products*, which allows companies to obtain feedback in real time from product usage and thus to allow them to create ad hoc services and customized products thanks to the higher visibility level obtained;
(3) *Closed-loop manufacturing*, which affects also other stakeholders in the value network, such as suppliers and customers;
(4) *Cyberized plant/"Plug & Produce"*, which allows companies to obtain a full flexibility and reconfigurability of the production environment and to facilitate operations management optimization;
(5) *Next step production efficiency*, which enables a better utilization of factory assets and so a more efficient production system;
(6) *Digital ergonomics*, which entails different consequences for workers, such as a faster knowledge transferring, the improvement of worker experience and of support at work and the operative complexity reduction.

This paper is aiming at illustrating the results of the gap analysis carried out in the sCorPiuS project in order to describe and to link breakthroughs coming from CPS implementation with the related obstacles that limit their adoption in manufacturing.

2 Methodology

This chapter describes the approach for analysis and identification of existing gaps. These gaps can be considered as the missing links between the strategic breakthroughs CPS adoption may bring into Manufacturing industry and the major obstacles to their achievement into the overall frame of the sCorPiuS vision [8]. The identification of the gaps is the result of the sequence of steps listed below carried out in the early sCorPiuS project phases:

1. Consolidation and validation of output from the state of-the-art analysis, knowledge capture events and experts' interviews;
2. Identification and definition of clusters collecting the main characteristics of breakthroughs and obstacles;
3. Identification, for each cluster, of some sub-clusters gathering breakthroughs and obstacles elements. The idea is to have a set of homogeneous groups of specific identified breakthroughs to match with specific subset of obstacles. In the crossing, the gap to overcome in order to remove the obstacle and achieve the envisaged result is identified

Crossing breakthrough and obstacle sub-clusters and selecting the most significant intersections will point out the researched gaps.

3 Gap Analysis Results

Cluster 1: New data-based services and business models

Crossing the identified breakthrough sub-clusters with each obstacle sub-cluster, one main gap arises. CPS could generate new business opportunities driven by product sense systems and service-oriented business models, but their costs and benefits are currently not well evaluated and therefore the return on the investment (ROI) is still unknown and difficult to predict (Table 1).

This obstacle mainly limits both the achievement of technical and technological innovative systems and potential results from new business opportunities.

Table 1 New data-based services and business models—Gap

New data-based services and business models		Obstacles		
		CPS embedded complexity	Need of guidelines and risk of disillusion	Uncertain ROI—benefits and obstacles not clear
Breakthrough	Active information and big data generation			
	Customer driven responsive business models			
	Digital business ecosystems for sensing products/services			1
	Distributed collaboration—horizontal and vertical integration			
	Predictive management			
	Socio-economic changes			

Cluster 2: Data-based improved products

From the analysis carried out, it is possible to identify two main gaps that need to be further explored. The first one arises from the intersection between "Data-driven products use/maintenance (MoL)" and "Communication problems—smart products versus old factories". Embedding CPS in final products could simplify communication and information exchange among products and between factory and products. In this way, companies could have visibility on the way their products are used, give customers guidance and help in case of malfunctioning and also inform customers when preventive maintenance is needed. However, the main problem is that, although embedded products are relatively easy to produce, the old factories are not so easy to modernize. In fact, letting a product communicate with others and with the factory is not enough; factories that are able to understand are the real missing part [3, 4].

The second gap comes from the match between "Improve quality and added value of the product" and "Complexity versus usability". In this case, CPS ability to improve quality and value added of the products is limited by their intrinsic increasing complexity (Table 2).

Cluster 3: Closed-loop manufacturing

Through the crossed analysis of breakthrough and obstacle sub-clusters, two main gaps emerge (see Table 3). CPS could facilitate communication within and beyond the factory, allowing people to take decisions in a decentralized way. However, data security, privacy and safety problems have a high impact on the achievement of these benefits. In fact, they have been selected as primary cause of the missing implementation of decentralized decision systems. The second gap shows that integration and interoperability issues affect potential benefits of horizontal and vertical integration. In other words, rapid integration enabled by ubiquitous communication and integration infrastructure enabling closed-loop manufacturing cannot be obtained mainly due to difficulty in realizing interoperable systems. Other problems related to this topic are the value chain technological fragmentation, data security, privacy and safety.

Table 2 Data-based improved products—Gaps

Data-based improved products		Obstacles	
		Complexity versus usability	Communication problems—smart products versus old factories
Breakthrough	Data-driven product use/maintenance (MoL)		2
	Data-driving processes		
	Facts and Feedback based design and production		
	Improved quality and added value of the product	3	

Table 3 Closed-loop manufacturing—Gaps

Closed-loop manufacturing		Obstacles	
		Integration and interoperability issues	Security, privacy and safety
Breakthrough	Autonomous decentralized decisions		4
	Customer in the loop		
	Flexible, resilient and agile value chain		
	Horizontal and vertical integration	5	
	Real time control from data availability		

Cluster 4: Cyberized Plant/"Plug and Produce"
From the analysis of the breakthroughs and obstacles included in this Cluster, a key topic needing further studies is related to the benefits from the plant flexibility such as the ability to enable the information sharing between devices and the self-reconfiguration of the stations. The Cyber-Physical flexibility enables the rapid intervention at shop floor level by means of augmented reality solutions. These opportunities are hard to be reached mainly due to difficulties in terms of time, effort and costs of transforming the current production systems in a cyberized environment as well as limits related to the uncertain performance reliability and availability of the systems. The same obstacles also affect the prediction and forecasting of the plant behaviour and its related benefits without being a specific gap. Another area requiring further research is related to the system and process reconfigurability. In fact, to obtain a fully reconfiguration of the production environment, some characteristics (layout, working conditions, operators activity) of the environment have to be changed, and this could create potential risks for workers and system security (Table 4).

Cluster 5: Next step production efficiency
Regarding the 5th cluster identified, the "Legacy and old technologies CPS migration" obstacle represents the main limit to the CPS ability to improve energy efficiency and to optimize plant operations. This is because it makes difficult to find solutions for an efficient and cheap integration of CPS in brownfield plants (Table 5).

Cluster 6: Digital ergonomics
The identified gaps concerning this Cluster are reported in Table 6. The first one considers the advantages for workers, which are facilitated in doing their job and take decisions thanks to the human automation co-working, limited by problems related to their safety and by the lack of new regulations in terms of security, while the second one concerns the system over-functionalities that could limit the reduction of management complexity.

Table 4 Cyberized plant/"Plug & Produce"—Gaps

Cyberized plant/"Plug & produce"		Obstacles			
		Difficulties in transforming the current plant into CPS	Legacy, integration and Extreme Performance Requirement	Safety and security limits	Uncertain performance reliability
Breakthrough	Cyber-physical flexibility	6			7
	Full transparency				
	Monitoring & control				
	Prediction and forecasting				
	Real time information at right place				
	Reconfigurability			8	
	Self-X (recovery, learning, analysis, ….)				
	Traceability				

Table 5 Next step production efficiency—Gaps

Next step production efficiency		Obstacle
		Legacy and old technologies CPS migration
Breakthrough	Allow small lot sizes	
	Cyber-physical production efficiency	
	Improve energy efficiency	9
	Plant operations optimization	
	Faster engineering process	
	Flexible equipment, agile processes	
	Improve logistics processes (reduce failure, inventory)	
	Predictive problem solving	
	Real and digital fusion	
	Self-learning zero defect manufacturing	

3.1 Additional Findings

Carrying out the analysis of Obstacles associated to Breakthrough executing the Gap Analysis Exercise, some other types of Obstacles have been identified as related with macro-dynamics (social, economic or technological) that this study

Table 6 Digital ergonomics—Gaps

Digital ergonomics		Obstacles				
		Complexity (over functionalities)	Data security	Human safety and related aged regulations	Lack of skilled end educated workforce	Legacy systems
Breakthrough	Enhanced humans sensing and intelligence			10		
	Factory in my pocket					
	Reduced management complexity	11				

could not expect to be addressed in a technology oriented research roadmap [5]. These categories are mentioned below.

Cultural, Educational and Perception: This issue can be addressed involving schools and universities in a significant effort to spread awareness of CPS adoption concepts in manufacturing and to bring CPS concepts outside the niche boundaries of people of the specific sector.

Overestimation of costs: CPS technologies are considered "expensive" both in terms of infrastructure and implementation, independently from the expected advantages. In this respect (especially for SME), the opportunity for supporting modernization investments should be considered.

Law, Regulations, Technology Enablers and EU Macro Economic Factors: this part concerns the aspects that are beyond the technical and process perspective, like legislation of CPS adoption, and liability of their utilization, both as provider and as user.

Standards and certifications: as far as standardization is concerned, the definition or fostering existing standards is definitely a point that needs to be addressed as the CPS technologies are not mature enough to have defined a clear de facto standard.

Moreover, the increasing complexity of cyberized products and the commercialization of products in many countries bring up the issue of product certifications for each possible market. That could represent a major effort to undertake (especially for SME with limited production volumes).

4 Conclusions

The output of the analysis described in this paper can be summarized in Table 7.

Table 7 Summary of Gap analysis exercise

Cluster	Breakthrough	Gaps
CL 1: New data-driven services and business models	Digital business ecosystems for sensing product/services	1. Uncertain ROI Benefits and costs not clear Missing KPIs
CL 2: Data-based improved products	Data-driven products use/maintenance (MoL)	2. Communication problems - smart products versus old factories
	Improve quality and added value of product	3. Complexity versus usability
CL 3: Closed-loop manufacturing	Autonomous decentralized decisions	4. Data security and privacy
	Horizontal and vertical integration	5. Integration and interoperability
CL 4: Cyberized plant/ "Plug & Produce"	Cyber-physical flexibility	6. Difficulties in transforming current plant into CPS based systems
	Cyber-physical flexibility	7. Uncertain performance reliability
	Reconfigurability	8. Safety and security limits
CL 5: Next step production efficiency	Improve energy efficiency	9. Legacy and old technologies CPS migration
	Plant operations optimization	
CL 6: Digital ergonomics	Enhanced humans sensing and intelligence	10. Workers role and related aged regulations
	Reduce management complexity	11. Management complexity (over functionalities)

A short description of the 11 identified Gaps is listed below.

Uncertain ROI—Benefits and costs not clear—Missing KPI: In this new context, it is crucial to provide decision makers of tools to forecast and monitor benefits and risks. The challenge is the definition of simple and reactive tools and methodologies, supported by a strong technology able to provide information about the value of the investments.

Communication and Coexistence—smart products versus old factories: The problem arises from the fact that these products are manufactured in "classical" environments, not designed for supporting "smart" products. The question is how to evolve the production environment to support the fast evolution of the complex and poly morphed (hard components, software, services, eco-sustainability, etc.) products.

Complexity versus usability: CPS enabling technologies and CPS related process add a new level of complexity, not always perceived as part of the "core business". From that comes the need to a seamless, easy-to-use CPS environment to be deployed in transparent way in the factory and along the value chain.

Data Security and Privacy: CPS is dealing with big amount of data. A complete new scale of magnitude of complexity arises from data management and sharing to ensure that data are available, protected and reserved.

Integration and interoperability: It is necessary to interpret the integration and interoperability along a much broader range in term of geographical spread, but also as time horizon as CPS diffusion is challenging a new approach that involve a great number of players, stakeholders and components.

Difficulties in transforming current plant into CPS based systems: Smart products are going to be produced in existing facilities. What are the criteria to define an evolutionary path for these infrastructures without impacting the process performances and efficiency? There are required both a technological and a methodological approach to such challenge.

Uncertain performance reliability: Although CPS approach promises a huge increase in potential performance of the systems, it raises the question about the reliability of the overall systems and the implication of a failure or a degradation of their functions.

Safety and security limits: It is crucial to clearly identify collaboration patterns between the human operator and automated systems in order to avoid possible problems of safety on the shop floor, as more functions and operations are going to be delegated to such autonomous systems.

Legacy and old technologies migration to CPS: At the level of the basic monitoring, actuation and cooperation a dramatic change of the technologies or at least their encapsulation is required to support CPS approach.

Workers role and related aged regulations: It is clear that the human factor can contribute to make more flexible the CPS system or support it in case of problems, but it is not clear how they can interact, with which interfaces. This is a social megatrend that is challenging the profile of the classical "blue collar", both from the age perspective and for the knowledge of the workers.

Management Complexity (over functionalities): There exists a risk that the complexity of the viewable context depletes the quality of the available information to support the right choice. In fact, it is important that the huge amount of information provided by CPS is made available when needed to who is actually concerned and with the appropriate context to avoid wrong decisions.

The identification of the Gaps to overcome in order to achieve the envisaged Breakthroughs in CPS adoption in manufacturing contexts is a key step for the definition and the validation of the research priorities constituting the research roadmap for adoption of CPS in manufacturing, as carried out in sCorPiuS project.

Acknowledgements sCorPiuS is supported by the European Union's H2020 Framework Program, under grant agreement no. 636906.

References

1. Gunes, V., Peter, S., Givargis, T., Vahid, F.: A survey on concepts, applications, and challenges in Cyber-Physical Systems. KSII Trans. Internet Inf. Syst. **8**(12), 4242–4268 (2014)
2. Hu, F., Lu, Y., Vasilakos, A.V, Hao, Q., Ma, R., Patil, Y., Xiong, N.: Robust Cyber–Physical Systems: concept, models, and implementation. Futur. Gener. Comput. Syst. (2015)
3. Lee, E.: Cyber-Physical Systems—are computing foundations adequate?, p. 1–9 (2006)
4. Lee, E.: Cyber physical systems : design challenges. In: Proceedings of the 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing (ISORC'08), p. 363–369 (2008)
5. Lee, J., Bagheri, B., Kao, H.A.: A Cyber-Physical Systems architecture for Industry 4.0-based manufacturing systems. Manuf. Lett. **3**, 18–23 (2015)
6. Oettinger, G.: Europe's future is digital. Hannover (2015). https://ec.europa.eu/commission/2014-2019/oettinger/announcements/
7. sCorPiuS State of the art on Cyber-Physical Systems (March 2016)
8. sCorPiuS Vision white paper (March 2016)
9. Wang, L., Törngren, M., Onori, M.: Current status and advancement of cyber-physical systems in manufacturing. J. Manuf. Syst. (2015)

Redundant and Decentralised Directory Facilitator for Resilient Plug and Produce Cyber Physical Production Systems

Flávio Páscoa, Ivo Pereira, Pedro Ferreira and Niels Lohse

Abstract Cyber Physical Production Systems (CPSS) built on the concept of "Plug-and-Produce" aim at delivering truly agile production systems. These systems are modular by nature and can be adapted based on changing requirements. One of the challenges in this domain is how to store and propagate information within CPPS. In this paper, a Redundant and Decentralised Directory Facilitator will be introduced to provide the capability to store and broadcast the existing system assembly capabilities. Additionally, this solution will provide redundancy and delocalization of the assembly capabilities information. The model used is described, as well as interactions, behaviours and deployment strategies. Finally a validation scenario is presented and conclusions are discussed.

Keywords Cyber physical production systems • Decentralised directory facilitator • Capability dissemination agent

1 Introduction

The need for agile production systems has resulted in creation of the Cyber Physical Production Systems (CPPS) concept. This concept builds on the "Plug-and-Produce" paradigm which uses the modular concept for production systems making

F. Páscoa
Everis, Consulting IT & Outsourcing Professional Services, Lisboa, Portugal
e-mail: flavio.pascoa@everis.com

I. Pereira (✉) · P. Ferreira · N. Lohse
Wolfson School of Mechanical, Electrical and Manufacturing Engineering,
EPSRC Centre for Innovative Manufacturing in Intelligent Automation,
Loughborough University, Loughborough, UK
e-mail: i.pereira@lboro.ac.uk

P. Ferreira
e-mail: p.ferreira@lboro.ac.uk

N. Lohse
e-mail: n.lohse@lboro.ac.uk

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_7

them easy to configure and reconfigure [1, 2]. The equipment modules are cyber physical units that are able to independently operate and interact with other units. One of the most promising implementations for CPPS is the use of agent technology as a means to embed the intended intelligence to each individual module and allow distributed control in a networked system [1]. This technology provides the needed dynamic control capabilities as well as the necessary decoupling and communication between modules. This means one can clearly represent each assembly capability inside the control agent architecture as shown in [3].

A method to keep track, organize and disseminate the existing assembly capabilities through the system is then essential to achieve optimal system functionality and performance. This method will provide the information about all the existing system assembly capabilities and available modules to all the agents in the environment, which is critical for the emergent behaviour of the system.

One solution to accomplish this is a dedicated repository, where the information about all the available assembly capabilities and modules can be stored, grouped and organized, such as the JADE platform Directory Facilitator (DF) [4]. JADE is the most common agent platform and has been widely used in the modular assembly system domain [1]. The use of DF as part of the JADE platform is a centralized service. The service has some limitations regarding the exchanged message content, since it was designed to be quite generic, and therefore may not provide optimal solutions for CPPS. Furthermore, having this centralized service may cause issues in performance when system is scaled up as the DF has a limit of 100 as a response to a query [4]. More critically the use of a centralized approach can cause bottlenecks when a considerable amount of requests are being made. This will constitute a severe problem to the overall MAS performance, as delays on the repository response propagate to the assembly process execution.

Another important consideration is that a central service repository has a single point of failure, this means upon problems or malfunctions the CPPS will stop working. This means that the system would not be resilient to problems, as it would require some intervention with high potential of the system having to be stopped. In fact the use of a central approach would be contra intuitive to the plug-and-produce concept which is the basis for CPPS.

This paper reports on a Redundant and Decentralised Directory Facilitator for Resilient as an extension of the available DF. This agent is called Capability Dissemination Agent (CDA) and will provide agents the necessary information to CPPS agents aiming at delivering resilient behaviour.

This paper is organised as follows: Sect. 2 specifies the requirements for the CDA; Sect. 3 describes the CDA model, with details about interactions, behaviours and deployment strategies; Sect. 4 presents the validation study; and Sect. 5 draws some conclusions.

2 Capability Dissemination Agent (CDA) Requirements

An effective way of providing and organizing all the assembly capabilities available in a system is critical for enabling CPPS. These assembly capabilities are representations that are used within the control environment as triggers for the assembly processes. The use of the Skill concept [3] provides an established formalization for this representation. This is the first step for the creation of a working CPPS, therefore each CDA needs to contain this information and be able to provide it upon request. However, availability of information for the modules does not guarantee a resilient behaviour. Thus, it is proposed that multiple CDAs be present in any given system, making the capability dissemination a distributed process. Therefore, a method for the CDA to store and broadcast these Skills to other CDAs is then required.

Due to the large amount of information that is expected to be stored, quick mechanisms to store the information and access it are needed. This will ensure performance targets can be achieved, since a slow CDA response to requests will cause delays on the system functionality.

The CDA must provide a method for other CDA's and other agents in the system to know when a CDA is introduced or removed. This ensures the "life beat" of all elements in the system, which is critical to deliver highly dynamic and agile systems. Additionally, this ability should to be independent of the platform where the CDA is launched, as one expects multiple platforms to exist in a CPS environment. This poses another implicit requirement which is the dissemination of when modules are added or removed across all platforms. Only with those discovery mechanisms can updated Skill information be maintained and distributed across the system to deliver a resilient behaviour.

Due to the usage of agent technology, interactions between all the CDA and other agents must be defined. This would allow them to communicate with one another, allowing them to cooperate to achieve their individual objectives.

3 Capability Dissemination Agent Model

In a CPPS using agent technology, the assembly capabilities are executed by the hardware modules present in the system. These modules are independent having no knowledge about other modules and their capabilities. If each individual module capabilities are propagated through the system to all its existing members, an enormous number of messages would be exchanged, which could cause undesirable system underperformance.

A place to gather and organize the distributed assembly capabilities available in the system is then required. The proposed CDA is then used as an assembly capability gathering point, organizing and structuring the information, while propagating it through the system. A CDA will then have the information about the registered local capabilities. However, several CDAs can then be present. Each CDA must also know the local capabilities registered in all the other CDAs. This is obtained by regular information exchange between CDAs. Each CDA will then have the knowledge about all the assembly capabilities available throughout the system.

During the CDA life cycle, it is needed to always maintain updated information about the entire system assembly capabilities. Only then can each CDA maintain an accurate view of the assembly capabilities present in the system.

To establish the CDA model, one needs to know what interactions are required. The need to clearly enumerate each of the available relations and interactions arises so that its behaviours can be defined. Also, for this model, the Skill concept will be used to represent the assembly capabilities [5]. The interactions that may occur with the CDA can be seen in Fig. 1.

Any agent in the environment can register or deregister Skills within the CDA. Queries about existing Skills providers can also be made. The CDA will broadcast a message to the system informing it is born when it is first launched and a death message when he leaves the system. Periodic messages regarding the internal information status of the CDA will also be broadcasted.

These messages define the possible interactions with the CDA. Its analysis will provide insight into all the required protocols. These protocols are the basis for creating an agent behaviour model which uses the protocols to achieve the agent objectives. Only with a clear and structured agent behaviour model will it be possible to create, implement and run the intended CDA. The first step to obtain the wanted model is to define the internal information each CDA needs to hold.

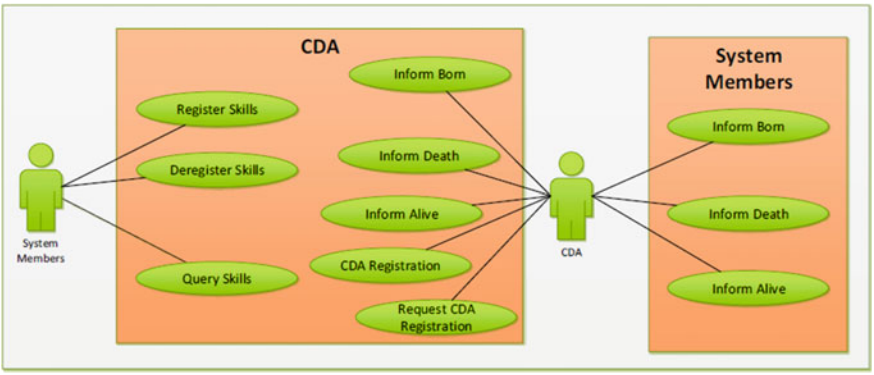


Fig. 1 CDA use cases overview

3.1 Capability Dissemination Agent (CDA) Interactions and Behaviours

All the interactions that may occur with the CDA need to be defined so that the proposed solution is stable and scalable. These interactions are mainly created using agent communication mechanisms which will be defined based on FIPA [6]. FIPA covers generic agent interactions which will significantly facilitate the development of the communication methods. For each interaction to be properly made, protocols need to be established. These protocols will provide the means for each member in the environment to interact with the CDA. FIPA establishes in its message contents the necessary information for a reply to be sent to the specific agent. The remaining message content is specific to each message and is defined accordingly.

- **CDA Born Protocol**—Consists in a broadcasted message to the network, specifying that the agent is alive;
- **CDA Registration Protocol**—Consists in a registration message to the existing CDAs, informing that it is in their neighbour;
- **CDA Request Registration Protocol**—Due to the dynamic environment, to request another CDA registration, another message is sent. After this message, the CDA Registration Protocol is used to finish this protocol;
- **CDA Status Protocol**—During a CDA life cycle periodic messages will be exchanged between all the CDA in the system. This will allow them to know about each other internal information status as well as their existence;
- **CDA Request Status Protocol**—When for some reason a CDA needs to know other CDA internal information, it will use this protocol to obtain it;
- **CDA Termination Protocol**—When a CDA has a controlled exit of the system, it will broadcast a message informing it. A message with the same characteristics as the CDA Born message will be broadcast to the system;
- **Skill Registration Protocol**—When an agent knows how to execute some Skills and wants other agents know about it, it will register these Skills in the local CDA;
- **Skill Deregistration Protocol**—Same as above but with the objective to deregister skills in the local CDA;
- **Skill Query Protocol**—When a member needs to know which agent can perform a Skill, it will query the local CDA using this protocol.

Having the internal information structure defined, and also how each CDA will interact with other CDAs and other members in the system, it is now possible to create the behaviours' models to obtain the wanted CDA objectives.

The CDA requires a mechanism to broadcast messages to the system. Its behaviours can then take advantage of this functionality. Each CDA is wanted to be functioning independently from each other. Each CDA will be associated with a different main container so that if one fails, the others can still keep functioning. A CDA registered in a JADE main container [4] cannot send messages to agents registered in others without previously knowing their individual address and names.

A mechanism so that a CDA can broadcast messages to agents in other containers is defined. This will be obtained using the Ethernet UDP Protocol [7]. It will provide a good performance approach as it does not requires connections to be established and maintained through the system. While the UDP protocol is not 100% reliable, its usage within a controlled environment will prove itself to be robust enough, as messages very rarely are lost. For this UDP broadcast to be functional, all the system members will be part of the same network as well as known the Group Port that the messages are sent and received through. All the agents that want to receive these messages must be listening to this defined port. Only with those prerequisites will allow that each message sent this way can be correctly received. This will allow the wanted message to reach every agent in the system, present in remote containers or in the local container, so long as agents are listening.

3.2 *Capability Dissemination Agent Deployment Strategy*

The CDA deployment is defined and adjusted considering the analysed system. In an extreme situation, a CDA for each module can be created. However, this implies a massive communication load. On the other side if only one CDA is present, then the system would not provide the necessary robustness. The deployment in the system needs to balance between the message load and the level of redundancy in the system. An overview of the CDA deployment strategies is shown in Fig. 2.

In this figure three strategies are considered. The left strategy has one CDA for the entire system. This will represent the central repository approach that will bring the MAS to a complete stop in the case of failure. The middle strategy has several clusters formed having a CDA representing them; this will reduce the number of exchanged messages across the system but at the same time providing some system

Fig. 2 CDA deployment strategies overview

redundancy. The right strategy has a considerable amount of CDAs along with other agents. This will guarantee the module level redundancy but will again increase the number of messages exchanged between the existing CDA and make meaningless the use of CDA.

4 Validation of Capability Dissemination Agent's Adaptability

The initial validation for the CDA functionality was obtained in the previous experimental scenarios, as the CDA was used to store the system existing Skills, and diffusing then through the system. To validate the re-adaptation capabilities of the CDA, two stations will be used along with an external computer terminal.

Two CDAs will be initially launched, one in each of the two workstations. The equipment modules will be represented as Resource Agents (RA), which are also launched for each one of the stations. Afterwards, another CDA will be launched in the external computer without any RA associated with it. Its sole objective will be to display the information stored in its internal defined tables, and with that, observe the CDA functionality. To be able to visualise the necessary functionalities, an interface to display these tables was created, as seen in Fig. 3.

After the initial synchronization and information exchange, all the assembly capabilities registered in each individual station CDA are known by the external CDA. Also, their registered local skill providers are known.

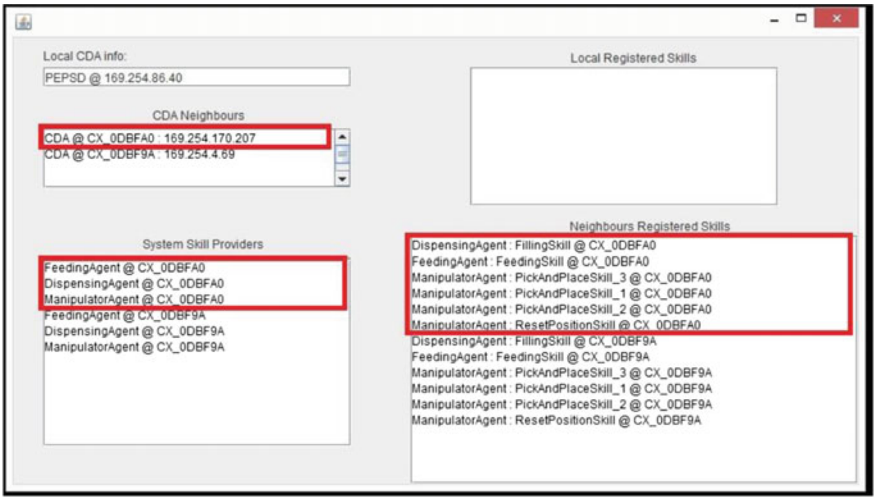


Fig. 3 CDA tables information display before neighbour CDA removal

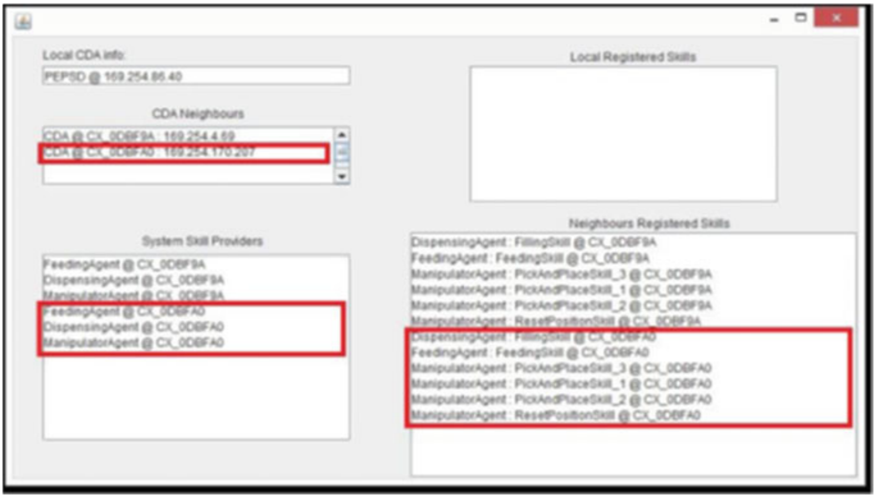


Fig. 4 CDA tables information display after neighbour CDA reconnection

All the system existing capabilities are known by the CDA. Next, one station will be abruptly shut down. This will provide confirmation on the CDA capability to readapt to partial system breakdowns. All the information regarding that specific CDA is no longer available once the CDAs synchronized. The information that is expected to disappear is shown highlighted by a red rectangle in Fig. 3.

The capabilities and skill providers associated with the removed station were removed from the CDA tables as expected, and can be confirmed in Fig. 4, since the top of the table has changed. The reintroduction highlighted in red in Fig. 4, is achieved after the station is reintroduced in the system.

As observed in Fig. 4, highlighted by the red rectangle, the capabilities from the new introduced system were correctly propagated and their existence acknowledged by the external CDA.

This experimental scenario validates the proposed CDA model providing correct propagation and re-adaptation to changes of the assembly capabilities. The discovery mechanism fulfilled its task, as neighbours know when a new CDA was introduced.

5 Conclusion

The use of a centralized approach for the storage of assembly capabilities in the context of CPPS is a drawback for the realization of this concept. However it is clear that the information needs to be stored and readily available for this concept to work. This paper proposes multiple delocalized CDA to maintain the information of all the system capabilities while being capable to readapt to changes. Each CDA can communicate across different platforms and keep track of other CDAs, along with their registered Skills. Redundancy and persistence emerge from the interaction of this collaborative behaviour. It increases the system robustness by not having the central capability repository, which allows the system to keep functioning even if part of it goes down. This approach proves very resilient to changes in the system and addresses the challenges of scalability as there are no limits for the number of modules in the system. Moreover, because these agents are delocalised they could be distributed to cater for computational resource limitations, without impact on the system's operation.

The deployment strategies for this approach are discussed, however more work needs to be done in order to establish which is the ideal distribution of CDAs in a system taking into account the system scale and topology, as well as performance requirements. Nevertheless, this paper demonstrates the first steps to achieve a redundant and decentralised directory facilitator for CPPS which is able to maintain, broadcast and secure all available assembly capabilities in the system.

Acknowledgements The authors acknowledge funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 680735 and support from the EPSRC Centre for Innovative Manufacturing in Intelligent Automation, in undertaking this research work under grant reference number EP/IO33467/1.

References

1. Arai, T., Aiyama, Y., Maeda, Y., Sugi, M., Ota, J.: Agile assembly system by "Plug and Produce". CIRP Ann. Manuf. Technol. **49**(1), 1–4 (2000)
2. Ferreira, P.: An agent-based self-configuration methodology for modular assembly systems, PhD thesis, University of Nottingham (2011)
3. Ferreira, P., Lohse, N., Razgon, M., Larizza, P., Triggiani, G.: Skill based configuration methodology for evolvable mechatronic systems. In: IECON 2012-38th Annual Conference on IEEE Industrial Electronics Society, pp. 4366–4371 (2012)
4. Bellifemine, F., Caire, G., Trucco, T., Rimassa, G., Mungenast, R.: Jade administrator's guide, JADE 4.0 edn. Boston, MA (2010)
5. Ferreira, P., Lohse, N.: Configuration model for evolvable assembly systems. In: 4th CIRP Conference on Assembly Technologies and Systems. Ann Arbor, USA, (21–22 May 2012)
6. Fipa: FIPA ACL Message Structure Specification, Foundation for Intelligent Physical Agents (2002)
7. Tanenbaum, A.S., Wetherall, D.: Computer Networks. Prentice Hall (2011)

Part II
Reconfigurable and Self-organized Multi-agent Systems for Industry and Service

A Self-organisation Model for Mobile Robots in Large Structure Assembly Using Multi-agent Systems

Spartak Ljasenko, Niels Lohse, Laura Justham, Ivo Pereira and Michael Jackson

Abstract Mobile, self-organising robots are seen to be a possible solution to overcome the current limitations of fixed, dedicated automation systems particularly in the area of large structure assembly. Two of the key challenges for traditional dedicated automation systems in large structure assembly are considered to be the transportation of products and the adaptation of manufacturing processes to changes in requirements. In order to make dynamic, self-organising systems a reality, several challenges in the process dynamics and logistical control need to be solved. In this paper, we propose a Multi-Agent System (MAS) approach to self-organise mobile robots in large structure assembly. The model is based on fixed-priority pre-emptive scheduling and uses a blackboard agent as a central information source and to facilitate more common goal directed distributed negotiation and decision making between agents representing the different needs of products and available mobile resources (robots).

Keywords Multi-agent systems · Fixed-priority pre-emptive scheduling · Large structure assembly

---

S. Ljasenko (✉) · N. Lohse · L. Justham
Wolfson School of Engineering, Loughborough University, Loughborough, UK
e-mail: S.Ljasenko@lboro.ac.uk

N. Lohse
e-mail: N.Lohse@lboro.ac.uk

L. Justham
e-mail: L.Justham@lboro.ac.uk

I. Pereira · M. Jackson
EPSRC Centre for Innovative Manufacturing in Intelligent Automation, Loughborough, UK
e-mail: I.Pereira@lboro.ac.uk

M. Jackson
e-mail: M.R.Jackson@lboro.ac.uk

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_8

1 Introduction

The modern manufacturing industry is facing a number of challenges due to the global market's frequently fluctuating demands [1] Traditional manufacturing systems are required to shift from mass production to mass customization [2]. The most common method of transporting products between manufacturing resources is by using conveyor belts [3]. This approach is not practical when the products are too large, heavy or awkward to handle. In large structure assembly, products are commonly transported between manufacturing resources via cranes. This approach is very slow and expensive [4]. Moreover, fixed automation systems like the ElectroImpact E6000 [5] and HAWDE [6] have a fixed infrastructure which makes it difficult to change and adapt manufacturing processes. In our previous work [7], we showed that a mobile system is able to control product delivery times and adapt to fluctuations in demand better than a fixed automation system. By making manufacturing resources move to products as opposed to products to resources, the need to transport products is greatly reduced. Also, if the products are large enough, then the production rate may be scaled up by placing several mobile robots around one product. To our knowledge, no control model has been developed to facilitate scalable processes with the objective to minimise the Total Weighted Tardiness (TWT) of such a system.

Multi-Agent System (MAS) technology [8] is seen as a great tool for controlling various systems in real-time and dynamic environments. Because of the local problem solving, these systems should be able to deal with a high level of complexity, require less information exchange and respond quickly to unexpected events [9].

In this paper, we present a model to control the product flow in large structure assembly. This model can be considered somewhat centralised due to using a central blackboard agent as an information source. The blackboard agent does not control other agents. Instead, it helps them to exchange information. In comparison to a more decentralised system, on one hand such an agent increases the load on information exchange, but on the other it increases each agent's knowledge of the environment. As a result of that, the agents are able to make more informed decisions at the cost of requiring more communication between these entities [10].

The paper is organised as follows: in Sect. 2, the fundamental structure of the mobile robot factory model underpinning this work is described. The development of a priority aging policy for our model is described in Sect. 3. The architecture of the proposed MAS approach is described in Sect. 4. A simulation model and some initial results are shown in Sect. 5. Finally, Sect. 6 draws conclusions from the initial testing of this approach and points out some directions for our further work.

2 Problem Description

The principle layout of the shop floor model underpinning this work is illustrated in Fig. 1. The fundamental assumption is that a number of products are fixed on static workstations and the mobile robots can freely travel on the floor between them. Each product therefore has a fixed location, capacity required work load, a due time and an associated tardiness cost. In the event of failing to meet the due time, the system is penalised with the tardiness cost of the tardy product multiplied by the tardiness time. The objective for our problem, minimising TWT, is shown in Eq. (1), where w_j is the tardiness cost, T_C is the completion time and T_d is the due time. $w_j = 0$ if the respective job j is completed on time.

$$\text{Min } \sum_{j \in J} w_j \left(T_c - \mathrm{T_{d)}}\right) \quad (1)$$

Many resources are allowed to work on the same product in order to achieve the required work rate. This is permitted, because the products are assumed to be large enough to enable a number of mobile robots to work simultaneously on them. The tasks are pre-emptive, meaning that mobile robots may temporarily pause one job in order to work on another one. Hence, the central challenge this shop floor model is to find best schedule for both products and mobile robots that minimise TWT for a given work load.

In this paper, a MAS-based approach for the self-organisation of mobile robots in large structure assembly is proposed. The motivation behind this approach is to help mobile robots better cope with the complexity of deciding how to distribute themselves among products in fixed locations. Finding the optimal schedule is a NP hard problem even for relatively static environments as is the case in large structure assembly situations [11]. Following well-established architectural patterns for MAS in manufacturing [12], the objectives of each product instance have been

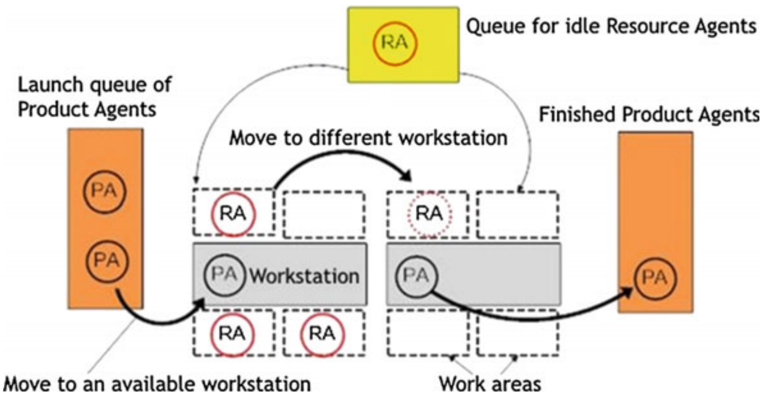


Fig. 1 Illustration of the principle shop floor organisation for mobile robot based assembly

represented through Product Agents (PA) and the capabilities of each mobile agent as Resource Agents (RA).

In order to achieve effective self-organisation between the agents, the decision making policies for the agent types as well as their communication protocols need to be defined. The scheduling policy that most closely matches the presented problem is fixed-priority pre-emptive scheduling that is commonly used in task scheduling for operating systems [13]. There, tasks are allocated to resources based on their priorities. A common issue with this is the starvation of low priority products when high priority products are constantly launched. In order to not starve low priority products, priority aging has been introduced. In our problem, the products must also meet set due times and therefore an appropriate priority aging policy is of high importance. The most commonly used priority aging policies are very basic and do not consider any due times or tardiness costs. Therefore, a more sophisticated priority aging policy was required for this model.

The objective of this paper was, therefore, to modify the fixed-priority pre-emptive scheduling model to incorporate scalability and minimise TWT.

3 Development of the Priority Aging Policy

The integration of the priority aging policy into the task scheduling approach is the first key component of this paper. It is required to ensure that no low-priority products are starved. This is a common problem in fixed-priority pre-emptive scheduling where low priority tasks never get processed due to constantly arriving higher priority tasks. The challenge in this problem is not simply to avoid starving a product, but also to meet its due time. On some occasions, the arriving product work load requirements can be greater than the manufacturing capacity of the mobile system and it is therefore inevitable that some products will be tardy. To our knowledge, no suitable priority aging policy for our model exists, therefore we propose and analyse our own ones.

The first analysed priority aging policy was the linear policy (2) and the second one was the exponential one (3). In the two shown equations, *P(t)* is priority in time, C_t is tardiness cost, t is the current time and d_t is the due time. If the due time has passed, the priority equals to the tardiness cost.

$$P(t) = Ct * \left(\frac{t}{dt}\right) \tag{2}$$

$$P(t) = Ct^{\frac{t}{dt}} \tag{3}$$

4 Model Structure

The model uses three types of agents: product (PA), resource (RA) and blackboard (BA).

The BA is the first agent to be launched in the simulations. It is followed shortly by the PAs and RAs. The RAs send a message to the BA that includes their locations. The PAs send a message to the BA that includes their due times, location, tardiness costs and capacity. From this information, the BA is able to compile an initial schedule based on the priority aging of each product. The main purpose of the BA is to identify conflicts in schedules and notify the potentially tardy PAs about it. It is recognised that without the BA, it would be difficult, if not impossible, for the PAs to have sufficient knowledge to solve scheduling conflicts efficiently.

After the notifications are sent to the PAs, the BA listens to further messages from PAs. The further messages can be from newly launched PAs or changes to the schedule from existing ones.

The PAs send their location, due time, tardiness cost and capacity to the BA as soon as they are launched. They then listen for messages from the BA or other PAs. A message from the BA means that this particular product will be tardy unless the schedule is altered. This triggers the PA to negotiate a better schedule with other PAs. The PAs have a master-slave relationship with RAs. This means that the PAs do not need to negotiate with the resources. The negotiation only occurs among PAs once the initial schedule has been completed and tardy products identified.

Based on the priority ranking at any moment in time, the highest ranking products have the right to order just enough resources as is necessary to meet the due time. As a result of this, the lowest ranked products will always be starved if there are insufficient resources.

The RAs have a straightforward behaviour. Firstly, they notify the BA of their location and secondly they listen for orders from PAs. As the slave in the master-slave relationship with PAs, the RAs follow orders from products.

So far, the structure of the model and each agent's purpose has been described. Below, we describe all the interaction protocols that are used in the model.

Product—Blackboard: Product agents have two reasons to communicate with the blackboard agent. Firstly, all product agents send their locations, due times, tardiness costs and capacities to the blackboard. Secondly, when the blackboard agent identifies that a product agent is tardy; it sends it a notification about it.

Product—Resource: This interaction is straightforward because product agents have a master-slave relationship with resource agents. Once product agents have agreed which resource agents each one will be occupying, they send an order to their resource agents to move to products and start working.

Product—Product: This interaction is the second key component in this model. As no priority aging policy can be expected to achieve an optimal result (due to the vast range of possible scenarios) this interaction serves as a corrective measure. This interaction is triggered by a notification from the BA. The intention of the protocol is to change the schedule in a way that reduces total weighted tardiness in the whole system.

When a product agent is notified about expected tardiness, is sends a message to all product agents that have due times after it. The message contains the product's due time, location, ID, expected tardiness cost and resource shortage. The shortage is the capacity that is not met by the due time in the initial schedule. The responding product agent compares the tardiness cost of the requesting agent to its own if the transaction is to be accepted. If the total weighted tardiness is lower as a result of accepting, then the responding agent sends an accepting message with the numerical value of how much the interaction will reduce the total weighted tardiness.

When several responses are received by the requesting agent, it means there is more than one favourable transaction available. In such a situation, the requesting agent accepts the first most beneficial response.

The reason why the request is not sent to product agents with earlier due times is because they only request as much as is required and any loss of resources would cause the whole product to be tardy for a much greater time than is reasonable.

5 Simulation and Initial Results

In order to analyse how various priority aging policies respond in different circumstances, a shop floor was modelled in NetLogo (version 5.2.1 [14]). As inputs to this model, there were four products with due times, capacities and tardiness costs. The simulations then followed the flowchart shown in Fig. 2 and compared the results with different priority aging policies. Before launch, a product's priority was set to zero. The priority remained equal to the tardiness cost of a product if it was not completed by its due time.

The linear priority aging policy was found to produce poor results, as shown in Fig. 3. The basic nature of this aging policy resulted in frequent conflicts even during solving of some relatively simple problems. Products with low tardiness costs were often denied sufficient resources near their due times by products with higher tardiness costs and later due times. This problem occurred much more rarely with the exponential priority aging policy. In this policy, the priority of each product is increasing exponentially until its due time. This gives an advantage to products with low tardiness costs and earlier due times.

The exponential priority aging policy meets the intention of prioritizing products closer to their due times. The sharp increase in priority close to the due time caused products with low tardiness costs to reach high priority rankings and therefore increase the chances of receiving the necessary resources as seen in Fig. 4a. Further priority aging policies were considered, but they produced either poor or inconsistent results. The only difficulties arise when multiple products have close due

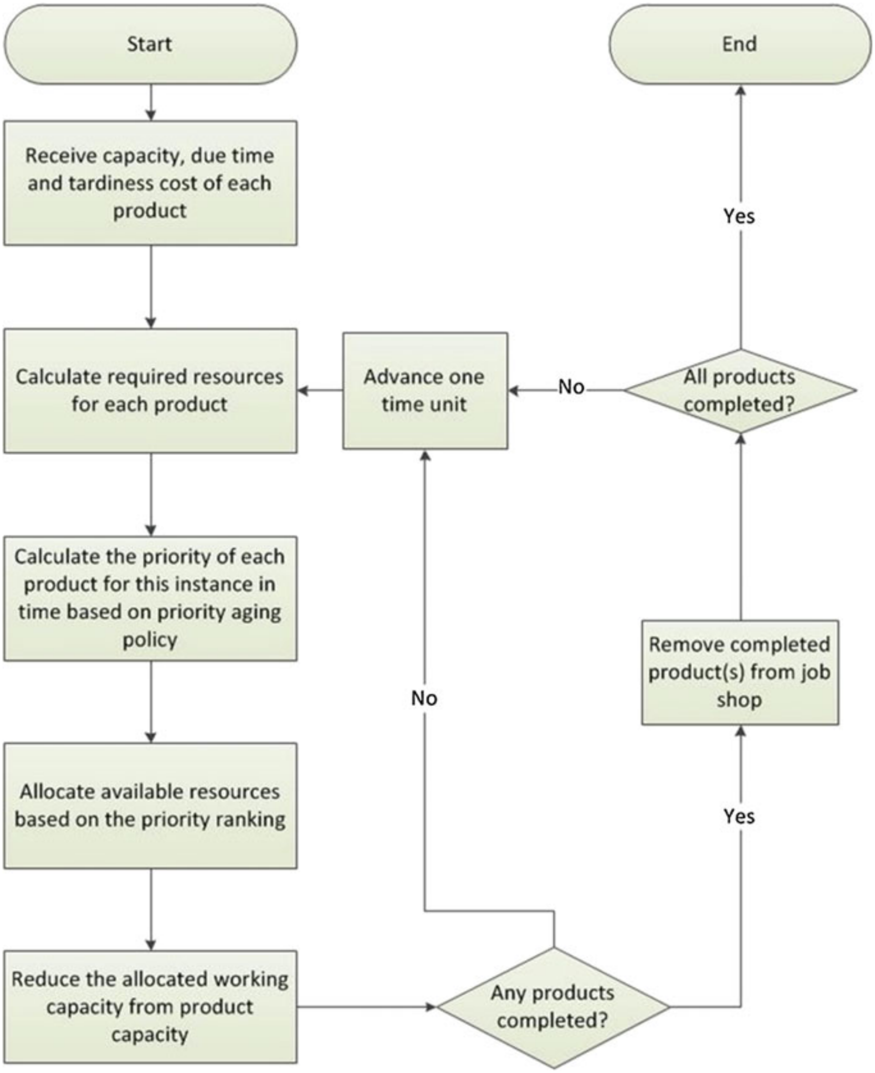


Fig. 2 The flowchart for the priority aging simulation

times, as seen in Fig. 4b. In such cases, the product with the higher tardiness cost and later due time may cause the earlier product to starve. Alternatively, occasionally products with earlier due times and lower tardiness costs caused products with later due times and higher tardiness costs to be tardy.

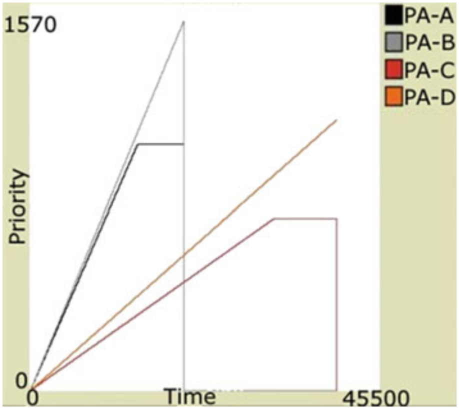


Fig. 3 The priority in time plot for the exponential priority aging policy

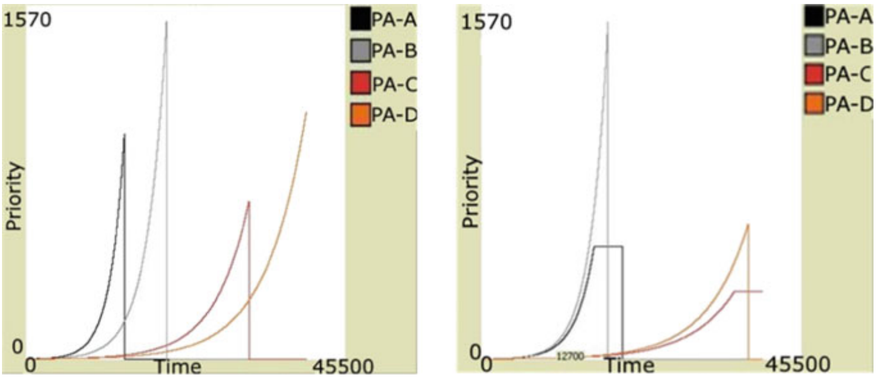


Fig. 4 The priority in time plot for the exponential policy in **a** easy and **b** difficult environments

6 Conclusions and Further Work

In this paper a novel heterarchical scheduling approach using MAS was proposed. The model is intended for mobile robots to minimize TWT in large structure assembly. The novel contribution of the proposed approach is the scheduling for products that may have several resources processing them. The mobile robots are treated as resources that can be dynamically allocated to any of the existing products in order to complete them on time. Where completion is not possible, the lowest possible TWT is achieved. Although MAS were used, the proposed agent organisation is only partially distributed due to the use of a central blackboard agent (BA). The response of this model to disruptions and increasing the number of entities will be assessed after further testing. In our further work, we will compare this model to other MAS models.

Acknowledgements The authors acknowledge support from the EPSRC Centre for Innovative Manufacturing in Intelligent Automation, in undertaking this research work under grant reference number EP/IO33467/1.

References

1. Barbosa, J., Leitão, P., Adam, E., Trentesaux, D.: Dynamic self-organization in holonic multi-agent manufacturing systems: the ADACOR evolution. Comput. Ind. **66**, 99–111 (2015)
2. Leitão, P., Restivo, F.: ADACOR: a holonic architecture for agile and adaptive manufacturing control. Comput. Ind. **57**, 121–130 (2006)
3. Zhang, S., Xia, X.: Optimal control of operation efficiency of belt conveyor systems. Appl. Energy **87**(6), 1929–1937 (2010)
4. Yuguang, Z., Kai, X., Dongyan, S.: Computer-Aided Design Assembly unit partitioning for hull structure in shipbuilding. Comput. Des. **45**(12), 1630–1638 (2013)
5. Remley, D., Rediger, J., Haworth, P., Holden, R.: Slug Rivet Machine Installs 16 Rivets Per Minute Drill-Rivet-Shave (2009)
6. Smith, S., Moore, I., Jackson, T.: 2004-01-2806 HAWDE Five Axis Wing Surface Drilling Machine, Assembly, no. 724 (2004)
7. Ljasenko, S., Lohse, N., Justham, L.: A comparison of the manufacturing resilience between fixed automation systems and mobile robots in large structure assembly. In: 49th CIRP Conference on Manufacturing Systems, 2016. Submitted to CIRP and accepted (2016)
8. Wooldridge, M.: An Introduction to Multiagent Systems (2002)
9. Mes, M., van der Heijden, M., van Hillegersberg, J.: Design choices for agent-based control of AGVs in the dough making process. Decis. Support Syst. **44**(4), 983–999 (2008)
10. Corkill, D.D.: Blackboard and Multi-agent Systems & the Future. Collab. Softw. (2003)
11. Wan, L., Yuan, J.: Single-machine scheduling to minimize the total earliness and tardiness is strongly NP-hard. Oper. Res. Letter **41**(4), 363–365 (2013)
12. Leitão, P.: Agent-based distributed manufacturing control: a state-of-the-art survey. Eng. Appl. Artif. Intell. **22**(7), 979–991 (2009)
13. Leet, C., Hahn, J., Seot, Y., C.Y. Parkq, M. Lee, and Kim, C.S.: Enhanced analysis of cache-related preemption delay in fixed-priority preemptive scheduling *. IEEE Trans. Comput. p. 187–198 (1997)
14. Wilensky, U.: NetLogo Home Page (2016). https://ccl.northwestern.edu/netlogo/

Specifying Self-organising Logistics System: Openness, Intelligence, and Decentralised Control

Shenle Pan, Damien Trentesaux and Yves Sallez

Abstract This paper contributes to the development of a novel paradigm in logistics, i.e. self-organising logistics system (SoLS). A SoLS can be considered as a logistics system that can function without significant intervention of human and without central control by software. It is functioning based on contextual local interactions. By such definition, SoLS could be a powerful solution to manage nowadays logistics that is much larger and complex than ever before. The goal of this paper is to provide a comprehensive and in-depth discussion to specify expected advantages and functionalities of SoLS, supported by some recent emerging concepts and technologies. The paper also attempts to provide a theoretical framework for the future work.

Keywords Logistics · Self-organisation · Openness · Intelligence · Decentralised control · Holonic and Multi-agents system

1 Introduction

Nowadays logistics system is much larger and complex than ever before, and, therefore, difficult to manage as a whole. One may have heard about many emerging technologies, concepts and paradigms that are proposed as solutions to cope with this difficulty, e.g. intelligent product or intelligent logistics [1], Holonic

S. Pan (✉)
MINES ParisTech, PSL Research University CGS—Centre de Gestion Scientifique, i3 UMR CNRS 9217, 60 Bd St Michel, 75006 Paris, France
e-mail: shenle.pan@mines-paristech.fr

D. Trentesaux · Y. Sallez
LAMIH, UMR CNRS 8201, University of Valenciennes and Hainaut-Cambrésis, 59313 Valenciennes, France
e-mail: damien.trentesaux@univ-valenciennes.fr

Y. Sallez
e-mail: yves.sallez@univ-valenciennes.fr

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_9

and Multi-Agents System [2], Cyber-physical System [3], or Physical Internet [4]. One may be interested in the following question: what could be the next era of logistics if all these solutions have a joint effort to develop the logistics? This paper discusses one of the possible eras, namely self-organising logistics system (SoLS hereinafter).

Self-organisation is a popular term in many research fields, for example in computer science, human society and behaviour, and biology [5–7]. However, in the field of logistics, it is still a developing paradigm. As suggested in [8], a SoLS is a logistics system that “can function without significant intervention of managers, engineers, or software control”. The suggested definition implies some potential advantages of such systems to today’s large-scale and complex logistics. Based on these works, we attempt to provide a wider and deeper vision to specify SoLS, by defining its main desired functionalities and analysing its advantages in the context of logistics organisation. The advantages are discussed qualitatively from perspectives of logistics performance and on sustainability, to illustrate that SoLS could make logistics more autonomous, efficient and effective, i.e. a more sustainable system.

For the sake of providing an overview and insightful discussion of SoLS, this paper proposes to discuss several main functionalities of future SoLS: openness, intelligence, and decentralised control. Openness means that a SoLS should allow actors (e.g. shippers, service providers, or customers) or assets (e.g. trucks, containers, or warehouses) to easily join and leave the system, for the sake of fast reconfiguration. Intelligence emphasises the object-based capability of local real-time communication and activeness [9]. Decentralised control focuses on collaborative rules and communication protocols designing, i.e. environment, for SoLS. It should be noticed that decentralised control for SoLS aims to avoid unexpected or disastrous outcomes of the system, but not to optimally plan activities or control objects. Further, we will also discuss expected advantages and performance of SoLS with regard to current challenges in logistics.

2 Relevant Literature

2.1 Self-organisation in Logistics

There is limited literature investigating self-organisation in the field of logistics. References [10–12] propose to consider supply network as complex adaptive system where self-organisation is considered as an internal mechanism between agents. Differently, Ref. [8] considers self-organisation as an organisational paradigm to cope with complex assembly lines, by using “bucket brigade’’ assembly lines as illustration. Other studies from the same standpoint can be found in the field of manufacturing, e.g. [2, 13].

Table 1 Advantages and expected performance of SoLS functions

	Description	SoLS expected advantages
Effectiveness	How well a goal is adequately met? [15]	Individuals should be able to make autonomous decision to meet a given goal
Efficiency	How well the resources expended are utilised? [15]	Individuals should intelligently use the on-hand resources
Agility	Ability of a system to rapidly reconfigure [16]	Individuals should have good connectivity to each other and to environment for rapid reconfiguration
Flexibility	Ability of a system to change status within an existing configuration [16]	Individuals should be able to connect and adapt to each other to provide flexible solution
Resilience	Ability of a system to return to its original state or move to a new, more desirable state after being disturbed [16]	SoLS should be able to dynamically self-reconfigure to adapt to unpredictable and disruptions
Sustainability	Ability of a system to protect, sustain and enhance the human and natural resources that will be needed in the future while meeting a desired goal [17]	SoLS should function on the basis of common goal of sustainability, by protecting human and natural resources

This paper is in line with [8] and extended to investigate SoLS functionalities for effectiveness and efficiency, which are particular important to cope with current and future challenges in logistics, see Table 1. From our point of view, a SoLS is an open, intelligent and holonic logistics system that aims to harmonise and lead individuals within the system towards a system-wide common goal, without significant human intervention from outside. An individual within SoLS can be an object (e.g. a truck), a manufacturer, a service provider (e.g. 3PL), a receiver (e.g. a customer), even a supply chain or a supply network as a whole. Assuming that individuals may have different constraints and objectives, e.g. service rate improvement or cost reduction, we argue that SoLS should have the ability to respect individual's constraints and objectives, meanwhile leading them towards a common goal—the sustainability for example. To this end, individuals should be coordinated by system-wide well-designed rules, i.e. the system environment [14].

2.2 *Expected Advantages and Performance*

Based on the suggested definition, some advantages of SoLS can be expected. We here discuss qualitatively expected advantages according to six main indicators of performance measurement, as described in Table 1.

2.3 Related Concepts and Paradigms

Before illustrating the features and functions of SoLS, it is necessary to provide consensual definition to the emerging concepts and paradigms used in this paper. The related concepts and paradigms are classified here according to their objective: conceptual criteria, technical paradigm, technology and application. Table 2 gives a short description from the literature adapted for the purpose of this paper.

After this short literature review, the next section introduces specifications of the SoLS.

Table 2 Concepts and paradigms related to SoLS

Classes	Short description
Conceptual criteria	**Openness**: system boundary is unset but variable and flexible [4] **Adaptation**: objects adapt to each other for coordination of activities and leading to more efficient processes [18] **Reconfiguration**: add, remove and modify logistics activities and functions [19] **Self-organisation**: a system can function without significant intervention of managers, engineers, or software control [8]
Technical paradigm	**Cyber-Physical System**: integrations of computation with physical processes [3] **Intelligent and active products**: a product able to self-identify its state, and able to send information once certain pre-conditions are met [20] **Multi-agents System**: system to coordinate intelligent behaviour among a group of agents which are autonomous and flexible computational systems [21] **Holonic System**: a system composed of holons. Holons are autonomous, cooperating and potentially recursive decisional entities, which can simultaneously be part or sub-whole of the system. A holon is most of the time composed of a physical part along with a digital one [22] **Complex Adaptive System**: a system that emerges over time into a coherent form, and adapts and organises itself without any singular entity deliberately managing or controlling it [11]
Technology	**IoT**: Internet of Things is the network of physical objects that are connected via Internet and technologies providing unique addressing schemes, such as RFID tags, sensors etc. [23] **ICT**: Information and communications technology for the communication between objects or systems [24] **Embedded systems**: combination of computer hardware and software, and perhaps additional mechanical or other parts, designed to perform a dedicated function inside a single identified system [25]
Application	**Intelligent logistics**: a will to plan, manage or control logistics activities in a more intelligent way [1]. It may rely on intelligent products, data techniques etc. **Physical Internet**: the network of logistics networks using the internet analogy [4]

3 Specifying Self-organising Logistics System

To construct an effective and efficient SoLS, and to avoid some unexpected or undesirable outcomes, we propose several functionalities for the system. This part discusses three that are identified as crucial factors: openness, intelligence and decentralised control.

3.1 Openness

Openness in SoLS means the boundary of the system is unset and open so that individual (actors, assets, supply chain, etc.) can easily join and leave the system. An illustrative example can refer to connecting a computer to the Internet from all over the world for different purpose, like searching information or communicating with others. Individuals join the SoLS to provide, procure or share logistics assets (truck, facilities etc.) and logistics service (delivering, planning etc.). By that, SoLS becomes fast extendable and reducible to answer highly flexible logistics requirements, as shown in Fig. 1. Here we discuss several functions that are essential to this end.

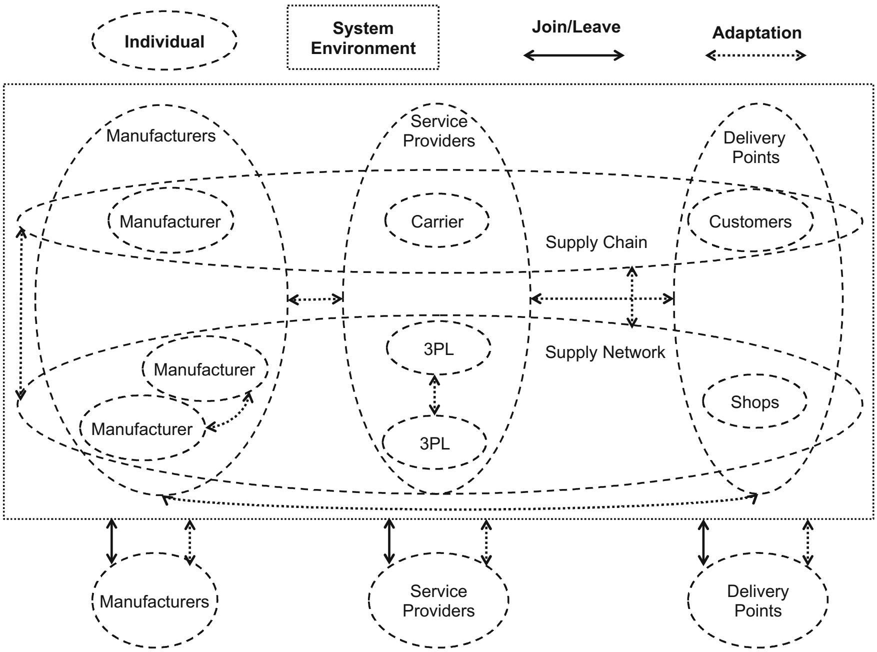


Fig. 1 Open logistics system

Function 1.1 Connectivity: this function means that individuals can easily connect with others and/or with the environment. For example in Fig. 1, individuals outside can easily connect to the system or other individuals within, if rules and standards are respected. System-wide modularisation and standardisation of physical assets (e.g. Physical Internet Container [20]), information systems (e.g. ICT) and organisation models are crucial to enable good connectivity. Examples can refer to Physical Internet [4, 26]. Notions of interoperability must equally be considered to allow an adequate communication between the different actors.
Function 1.2 Reconfiguration: Once the system receives new logistics requests (transport, stocking etc.), it should be able to add or change some current functions inside to fulfil the requests. The reconfiguration function is particularly important to cope with disruptions, i.e. the system can rapidly self-reconfigure [27].
Function 1.3 Adaptation: individuals should adapt to each other and to environment, in order to rapidly build up effective and efficient coordination between them [11].

3.2 Intelligence

Intelligence in SoLS means that every individual within the system is able to make and execute autonomous decision, and to interact with other individuals and with the environment. For that, they should have the ability to collect, store and process information from other individuals (their state and decision) and from environment (rule modification). We propose that IoT technology is a fundamental support to such requirements. Further, two important functions are identified for such a functionality.

Function 2.1 Activeness: individuals should have the ability to collect information from other individuals (their state and decision) and from the environment (rules and any modifications), then to store and process the information for autonomous decision-making. Further, they should also be able to send information to others, to advertise the changes of their state and to take decision. Each event can be seen as a change to the system, e.g. a warehouse is disrupted, a truck is full, or an order is cancelled. Activeness is therefore an important functionality for SoLS.
Function 2.2 Autonomous acting: once a decision has been made, systems should also be able to apply and execute the decision. This function could be relying on embedded systems of autonomous operating assets, autonomous supply chain or autonomous network, including sensors and actuators inside decisional loops.

Figure 2 illustrates some existing and emerging examples of intelligence applied in logistics. It is obvious that some of the three functions are questionable under current organisation and technologies in logistics. However, emerging innovative technologies may help us to project the future of intelligence in SoLS, e.g. drone, autonomous operating truck, or Kiva.

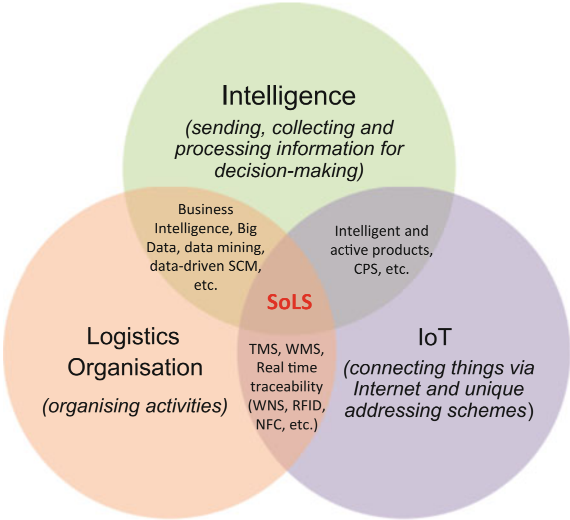


Fig. 2 Technologies and techniques of intelligence applied in logistics

3.3 Decentralised Control

As stated in [11], "imposing too much control in a complex system detracts from innovation and flexibility; conversely, allowing too much emergence can undermine managerial predictability and work routines". In this paper we propose to specify SoLS as rule-based decentralised control system, for the sake of effectiveness and efficiency [28]. More precisely, rule-based decentralised control in SoLS aims to help the autonomous decision-making of individuals within the system. It has a twofold objective: to avoid unpredictable and undesirable outcomes, and to lead individuals towards a system-wide common goal. We propose that decentralised control should rely on system-wide well-designed rules and individual-wide protocols, which should be respected when local autonomous decisions occur [28]. In such a context, we propose that the decentralised control in SoLS should be modelled as a holonic and multi-agent system [7, 21], with the following functions.

Function 3.1 Holonic multi-agent system: we propose to model SoLS as holonic multi-agents system (HMAS), since controlling in such a system should be at individual level, i.e. at holon level (whole, part, or sub-whole). Each holon is composed of a physical part mirrored by an agent in the model. Each agent is self-controlling so that no controlling agent is necessary in the HMAS.

Function 3.2 Rule- and protocol-based: rules here represent the common regulations of the system that every individual should respect; protocols stand for communication protocols that allow individuals communicate to each other, i.e. interoperability. Take digital internet as an example, in which rules are laws and protocol is internet protocol like TCP/IP. In other words, rules are designed for meeting a common goal, and protocols for effective and efficient communication.

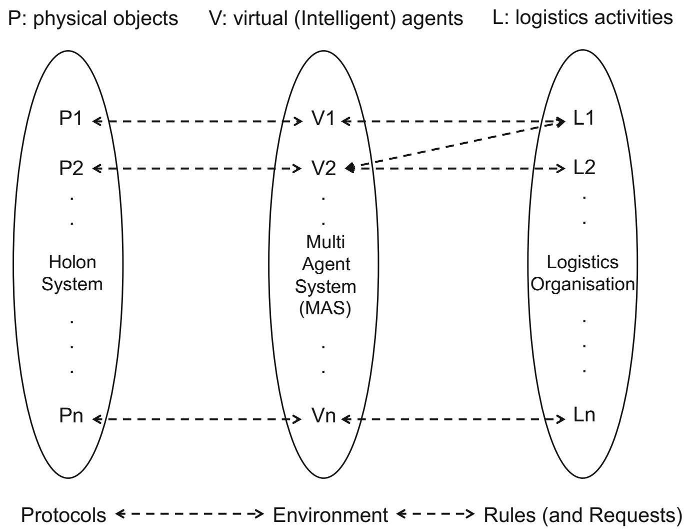


Fig. 3 Modelling SoLS as rule- and protocol-based holonic multi-agent system

Figure 3 presents the system architecture of rule- and protocol-based HMAS. We consider here a transport request as an example of logistics activities. *As per* current organisation, driver (P1) and transportation means (P2) are essential for it. In SoLS scenario, a HMAS model is employed to help P1 and P2 to make decisions. More precisely, each of P1 and P2 is mirrored by an agent in the MAS model. Rules of logistics organisation (e.g. speed limit, preconizing multi-modal transport, etc.) and protocols between objects (e.g. driving instruction, autonomous driving control etc.) are translated as the environment of the MAS model. By that, P1 (or V1) and P2 (or V2) should respect the environment when they make autonomous decisions. The MAS model here thus aims at coordinating and putting in collaboration the agents—physical objects, but not to control.

4 Conclusion

This paper aims at contributing to specify advantages and functionalities of self-organising logistics system (SoLS), which is a novel paradigm in logistics. Openness, intelligence and decentralised control are the three main functionalities proposed and discussed in this paper as vital functionalities of SoLS to cope with the current and future challenges in logistics. Real world examples can refer to the recent DHL research project—Parcelcopter SkyPort.[1] Basically, the idea is to

[1] http://www.dpdhl.com/en/media_relations/specials/parcelcopter.html.

conjointly use autonomous drones (Parcelcopter) and automated parcel stations (Packstations) to cope with demanding logistics challenges such as "same day delivery", especially in rural zones. Drones and Packstations together can be seen as a SoLS. Once a package to be delivered is arrived at Packstation with known constraints (volume, delivery location and time, etc.) the system will autonomously plan and optimise the delivery. Only monitoring is necessary from outside. We may assume that, supported by the emerging technologies and techniques, more and more examples of SoLS will be found and realised.

The paper provides also an insightful overview and theoretical framework for the future work. For example, organisational and technical issues related to SoLS, such as protocol design, coordination or security issues can be investigated through mechanism design theory for example. Quantitative researches through simulation and optimisation approaches are also necessary to investigate the performance and viability of SoLS in real-life cases.

Acknowledgements The authors like to thank the French National Research Agency (ANR) for the support to Project PI-NUTS (ANR-14-CE27-0015).

References

1. McFarlane, D., Giannikas, V., Lu, W.: Intelligent logistics: involving the customer. Comput. Ind. (2016)
2. Barbosa, J., et al.: Dynamic self-organization in holonic multi-agent manufacturing systems: the ADACOR evolution. Comput. Ind. **66**, 99–111 (2015)
3. Lee, E.A.: Cyber physical systems: design challenges. In: 2008 11th IEEE International Symposium on Object Oriented Real-Time Distributed Computing (ISORC). IEEE (2008)
4. Ballot, E., Montreuil, B., Meller, R.: The Physical Internet: The Network of Logistics Networks, La documentation Française (2014)
5. Serugendo, G.D.M., et al.: Self-organisation: Paradigms and Applications. Springer (2003)
6. Aliu, O.G., et al.: A survey of self organisation in future cellular networks. Commun. Surv. Tutor. IEEE **15**(1), 336–361 (2013)
7. Serugendo, G.D.M., Irit, M.-P., Karageorgos, A.: Self-organisation and emergence in MAS: an overview. Informatica **30**(1) (2006)
8. Bartholdi, I.J.J., Eisenstein, D.D., Lim, Y.F.: Self-organizing logistics systems. Ann. Rev. Control **34**(1), 111–117 (2010)
9. Sallez, Y.: The augmentation concept: how to make a product "Active" during Its Life Cycle. In: Borangiu, T., Thomas, A., Trentesaux, D. (eds.) Service orientation in holonic and multi-agent manufacturing control, pp. 35–48. Springer, Berlin Heidelberg (2012)
10. Wycisk, C., McKelvey, B., Hülsmann, M.: "Smart parts" supply networks as complex adaptive systems: analysis and implications. Int. J. Phys. Distrib. Logist. Manag. **38**(2), 108–125 (2008)
11. Choi, T.Y., Dooley, K.J., Rungtusanatham, M.: Supply networks and complex adaptive systems: control versus emergence. J. Oper. Manag. **19**(3), 351–366 (2001)
12. Hülsmann, M., et al.: Autonomous cooperation–a way to cope with critical incidents in International Supply Networks (ISN)?. In: An Analysis of Complex Adaptive Logistic Systems (CALS) and their Robustness. in 24th EGOS Colloquium, Upsetting Organizations, web-publication (2008)

13. Bousbia, S., Trentesaux, D.: Self-organization in distributed manufacturing control: state-of-the-art and future trends. In: 2002 IEEE International Conference on Systems, Man and Cybernetics. IEEE (2002)
14. Narahari, Y., et al.: Game theoretic problems in network economics and mechanism design solutions. Springer, London (2009)
15. Mentzer, J.T., Konrad, B.P.: An efficiency/effectiveness approach to logistics performance analysis. J. Bus. Logist. **12**(1), 33–62 (1991)
16. Bernardes, E.S., Hanna, M.D.: A theoretical review of flexibility, agility and responsiveness in the operations management literature: toward a conceptual definition of customer responsiveness. Int. J. Oper. Prod. Manag. **29**(1), 30–53 (2009)
17. Labuschagne, C., Brent, A.C., Van Erck, R.P.: Assessing the sustainability performances of industries. J. Clean. Prod. **13**(4), 373–385 (2005)
18. Jahre, M., Fabbe-Costes, N.: Adaptation and adaptability in logistics networks. Int. J. Logist. Res. Appl. **8**(2), 143–157 (2005)
19. Koren, Y., et al.: Reconfigurable manufacturing systems. CIRP Ann. Manuf. Technol. **48**(2), 527–540 (1999)
20. Sallez, Y., et al.: On the activeness of intelligent Physical Internet containers. Comput. Ind. **81**, 96–104 (2016)
21. Botti, V., Boggino, A.G.: ANEMONA: a multi-agent methodology for Holonic Manufacturing Systems. Springer Science & Business Media (2008)
22. Giret, A., Botti, V.: Holons and agents. J. Intell. Manuf. **15**(5), 645–659 (2004)
23. Atzori, L., Iera, A., Morabito, G.: The Internet of Things: a survey. Comput. Netw. **54**(15), 2787–2805 (2010)
24. Shen, W., et al.: Applications of agent-based systems in intelligent manufacturing: an updated review. Adv. Eng. Inf. **20**(4), 415–431 (2006)
25. Barr, M.: Embedded Systems Glossary. Neutrino Technical Library (2007). Accessed 21 April 2007
26. Montreuil, B.: Toward a Physical Internet: meeting the global logistics sustainability grand challenge. Logist. Res. **3**(2–3), 71–87 (2011)
27. Kim, Y., Chen, Y.-S., Linderman, K.: Supply network disruption and resilience: a network structural perspective. J. Oper. Manag. **33–34**, 43–59 (2015)
28. Bakule, L.: Decentralized control: an overview. Ann. Rev. Control **32**(1), 87–98 (2008)

A Generic Reconfigurable and Pluggable Material Handling System Based on Genetic Algorithm

Andre Dionisio Rocha, Pedro Caetano and Jose Barata Oliveira

Abstract With the increasing customization of products there was a need to create new manufacturing systems that were able to satisfy these needs of the market. The Reconfigurable Manufacturing System (RMS) emerges as a more recent approach allowing the reconfiguration of the line and all the manufacturing systems. The balancing line is a common problem in the system reconfiguration but may not be enough, being also important to reconfigure the material handling system itself. Genetic Algorithms (GA) are one of the most known and used alternatives in optimization problems being widely used in shortest path problems like the travelling salesman. In this paper a solution that allows reconfiguring an agent based material handling system using a genetic algorithm is presented.

Keywords Reconfigurable manufacturing system · Material handling system · Genetic algorithm

1 Introduction

The Assembly Line Balancing Problem (ALBP) is a common case of study to reconfigure manufacturing lines. In this problem the workstations of the assembly line are configured with the necessary tasks to minimize the production time. However, with the need to reconfigure the line multiple times and due to the

A.D. Rocha (✉) · P. Caetano · J. Barata Oliveira
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Caparica, Portugal
e-mail: andre.rocha@uninova.pt

P. Caetano
e-mail: p.caetano@campus.fct.unl.pt

J. Barata Oliveira
e-mail: jab@uninova.pt

A.D. Rocha · J. Barata Oliveira
UNINOVA institute, Centre for Technology and Systems, Caparica, Portugal

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_10

difficulty to change the stations and other workspaces, the balancing problem becomes inefficient [1, 2]. So, to solve that problem, a genetic algorithm was proposed to reconfigure a material handling system on an agent based approach. The genetic algorithm will not reconfigure the line but instead, and given an existent assembly line, will create the best way for every product flow in the line. The system has agent based architecture for the material handling structure. There is a main agent called optimizer that is responsible to check every product agent and run the genetic algorithm to reconfigure every single one and *n*-product agents that have the information of the products to be manufactured in the assembly line.

This paper is structured as follow: first, a brief state of art will be presented, in which is an introduction to the reconfigurable systems and some related work about his reconfiguration and about genetic algorithms will be provided; next the system's architecture will be analysed, in which we describe how the system is designed, followed by a more detailed explanation of its implementation. Finally, results of some tests will be presented and the final conclusions will be formulated, as well as the used references.

2 Related Work

Since the beginning of mass customization the manufacturing lines are constantly changing in order to fulfil better and more quickly the needs of the market. The manufacturing line changed from a more dedicated production to a more agile approach and with that, different types of manufacturing systems have been developed. The Flexible Manufacturing System is as a good concept that opposes the dedicated lines and the mass production [3]. It can produce a variety of products but it is usually composed by expensive tools such as CNC machines and has a low throughput, making the cost per unit higher. On the other hand, Reconfigurable Manufacturing Systems (RMS), a more recent approach, are systems that are designed around part families and focus on scalability and adaptability of new products [4]. RMS was proposed to solve the excess of demand and variety, as a model that tries to have both the characteristics of mass production offered by dedicated lines, and mass customization offer by the FMS.

The system must be adjustable to allow both flexibility and adaptability for the creation of new products. These characteristics can be achieved by adding new machines in the system and also by configuring the same machines both in hardware or software [4]. The RMS proposes a system where machine components, machines, cells or a material handling system can be added, removed or modified, in order to quickly respond to the changes in the production [3].

With the development of the reconfigurable manufacturing systems new paradigms emerge. Many authors began to study new optimization methods to configure these lines, both as in machine layout and in material handling configuration.

The assembly line balancing problem (ALBP) is one of the most frequent case study and optimal algorithms, such as particle swarm (PSO) or ant colony optimization are often used to configure them. In [5] a PSO is used to find an optimal solution for the Robotic Assembly Line Balancing problem by assigning robots to stations in a balanced manner to perform activities while minimizing cycling times and maximizing production rates. In [6] an ant colony optimization is used to balance a U-type line. A graph in which the nodes and arcs respectively represent the tasks and their own priorities was used. The weights of the nodes represent the execution time of each task. A precedence graph is also used in [5].

However, it isn't always easy to change the multiple time cycles of production lines. We cannot always rearrange the line again, being necessary to reconfigure the material handling system itself.

Genetic algorithms are often used to solve this kind of optimization problems, being one of the main algorithms to solve the travelling salesman problem. This problem is based on finding the shortest possible route that passes between certain cities, similar to the problem treated in this paper. However, our problem is not so simple, since it is not easy to codify the individual products due to existing precedencies. Gen and Cheng created a commonly used coding called PRIGA (Priority Genetic Algorithm) [7]. In [8] a genetic algorithm was proposed to route an automatically guided vehicle in a FMS. The authors use Petri Nets to model the system and use the PRIGA algorithm. Also in [9] a PRIGA solution is used to solve a multi-model assembly line balancing problem.

The fitness function is also an important step of the algorithm, which evaluates how good a solution is. In [9] a function with different weights like cycle time and the processing time of each task is used. Also this algorithm uses a Partial Mapped Crossover operator.

[10] proposes an approach to arrange the material handling systems using a GA and lean methods. A graph was used to model the material handling system. This work can be a first approach to the problem exposed but ends up being based on a similar method to the ALBP.

In [11] several algorithms were proposed, including a modified GA, to solve a Problem of Searching for Optimal Path (PSOP) for an Automated Material Handling System. The system was again modelled using a graph. The conclusions are promising, but the work features some restrictions, such as non-repetition of nodes/stations.

3 System Architecture

The system's architecture is a multi-agent based architecture as depicted in Fig. 1.

A multi-agent system is a computer system composed of a number of intelligent computational entities (agents) capable of interacting with each other and perceiving the environment to solve a common goal with the best results. This approach allows the design of a more distributed system based on these agents in

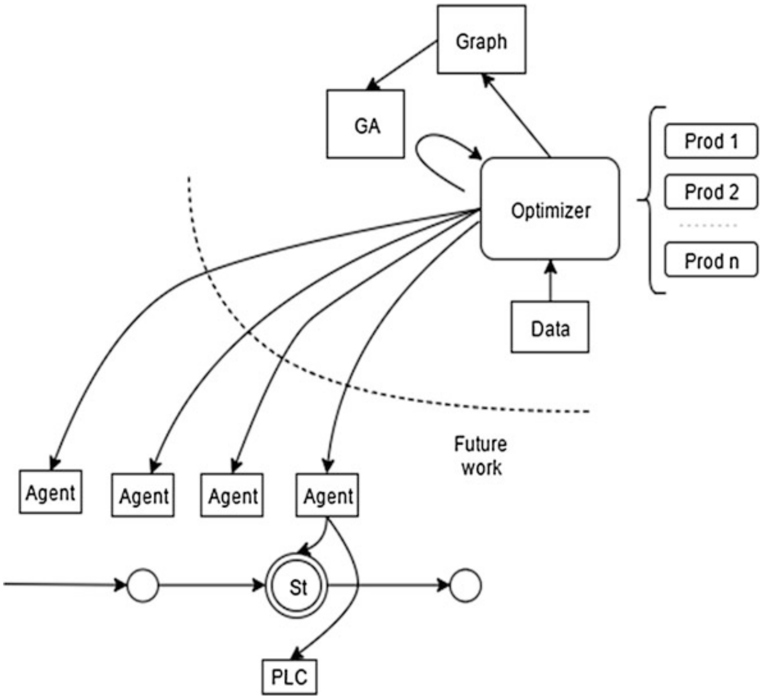


Fig. 1 System architecture

contrast with a more centralized approach. More about multi-agent systems and their application in material handling systems can be found in [1, 12, 13]. In this architecture there are two types of agents, the optimizer and the product agent.

The *optimizer* is the main agent of the architecture, being responsible for reading external data to the system, create the graph and run the GA The data consists of a first file containing the skills that exist in the manufacturing line with the information of each associated cost, and a second file in which all the necessary information about the manufacturing line are contained, such as their key points and all the links between them. These points are subdivided in decisions points or stations, where "station" is a particular set of decision points that contains skills. By help of this information, the optimizer will model the system in the form of a graph.

The graph is the representation chosen of the transportation system that consists of nodes and arcs. Nodes are the decision points and stations and there is an associated weight or cost, which may have different interpretations such as the processing time of each skill, decision time, power consumption, etc.

Arcs are the connections between nodes, basically any type of transportation that connects each decision node and stations, like a conveyor or an AGV that transports a product from a point to another. The main goal at the end will be to set the nodes with the configuration of the forward arcs in order to ensure an optimal solution for the assembly line. Every arc has also associated a cost.

Both costs will be considered by the genetic algorithm.in the search for the best solution. These costs are defined by the user to correspond to a correct graph interpretation. Nodes with higher costs may represent stations that require more work/time to perform a skill. On the other hand, an arc with a higher cost can be representative for a longer conveyor. The main goal of this work is to set an optimal path to different products, trying not to focus very much on allocating resources and stations, like in the ALBP, or plan the entry sequence of products. Naturally, products that may have a different throughput as compared to others can be set with a different cost in order to ensure a better configuration.

The *product agent* is the agent responsible for containing the information of all the skills that will be performed in the system on the respective product. With the creation of a new product the optimizer agent will gather all of them and run the genetic algorithm. There has been also created in this architecture the concept of skill as an independent entity, in which the information about its cost, type, and eventual sub-skills is contained.

The genetic algorithm is a search algorithm based on the theories of evolution, genetics and natural selection and is composed by a population, a set of several different individuals, each one being a solution of the problem. In this particularly problem, an individual is a representation of the products' path that will result in an assembly process in the production line. If there is more than one type of product in the assembly line, it is necessary to set multiple paths, leading to a multiple configuration. For example: if there is a product named XPTO and another named OPTX, the individual's chromosome has a representation of both paths. Another approach would be to create a genetic algorithm for each product; however, this will not relate them, leading to a non-optimal solution.

Associate to every individual there is a value indicating how good the solution is. This value is calculated by a fitness function. Through some operators, such as crossover and mutation, the genetic algorithm will search over several generations a best solution, using some stop conditions. Those operators are independent on how many products exist since they will perform for each path of each individual independently.

The optimizer agent will use that found solution to reconfigure the line. In future work is intended to create another type of agent for each PLC, or any responsible device, so that the optimizer can send them the proper configuration.

The architecture represented in Fig. 1 was structured and developed in the reported work. Both the optimizer and product agent and their interactions were created. The graph's representation of the manufacturing line was modelled and developed and the complete genetic algorithm was designed, from the representation of individuals to the development of the genetic operators and stopping conditions.

4 Implementation

The system was implemented with an agent based architecture through a framework allowing the development of agents in Java, using JADE [14]. Figure 2 shows how the algorithm works, and describes the procedure to run the GA. It has to be noticed

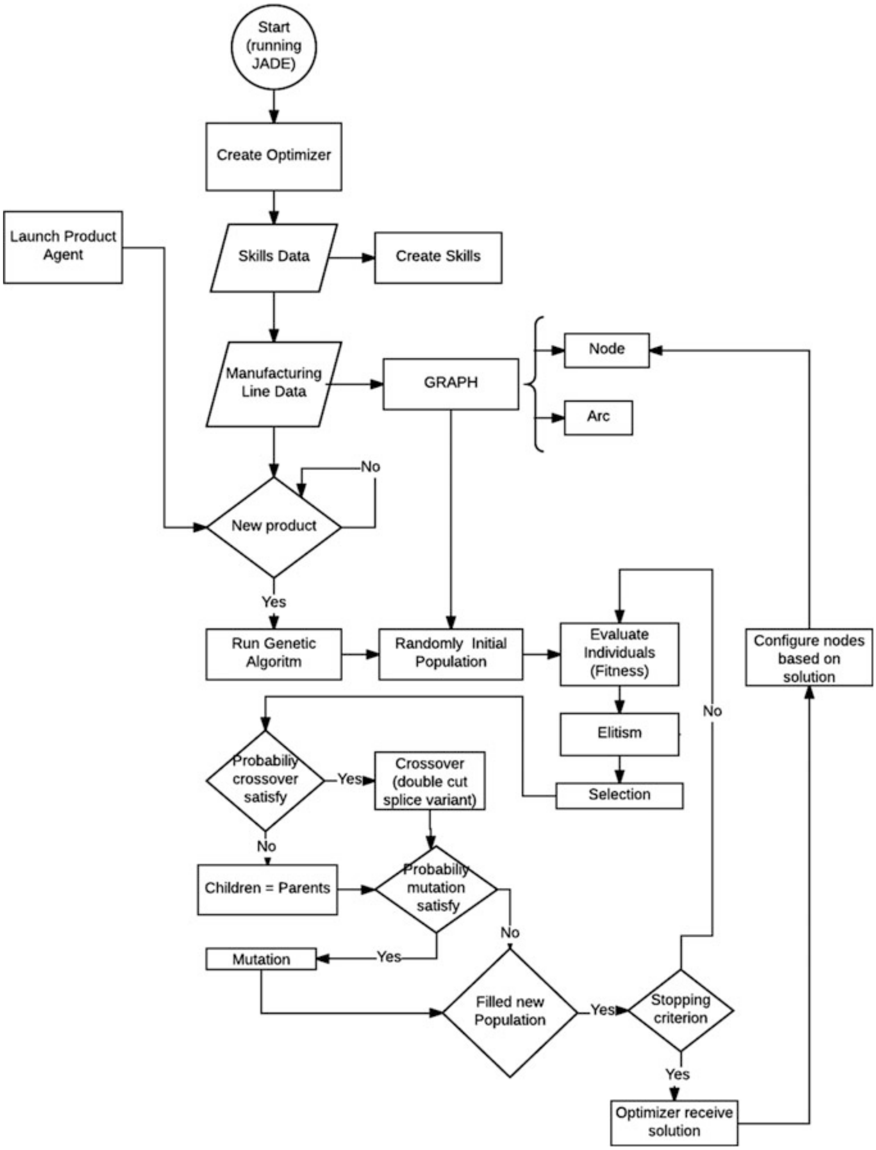


Fig. 2 Brief description of the system's implementation

that the extra functions and points in the algorithm are not included in this description.

Basically, the optimizer reads the data from two .txt files, from which it will create a graph with all the nodes and arcs. The nodes hold the information of their parent and their own children, allowing creating a path from the start to the end node. There is also a parameter that represents the depth of the node, allowing identifying a retroaction on the line, so that the products can return to the beginning of the manufacturing line and execute pending skills.

With the graph created, the optimizer agent will wait for products to be launched and runs the genetic algorithm. The algorithm will first create a population of random individuals and evaluates them with a fitness function. This particular fitness function for now will only add all the costs of the arcs and nodes that exist in the individual. In the case of more than one product, the function will add the fitness of all the products but, since not always the best solution for multiple products is the best solution for each of them, in the future the fitness function can be easily reconfigured to relate the products and find a more optimal solution.

A first approach has been developed to allow some interoperability between products by evaluating how many times the products move toward each station. If there are parallel stations in the manufacturing line, the algorithm will have that ability to distribute the different types of products to them.

The algorithm will then pass through the several genetic operators. First, to a method called elitism, in which a certain number of the best individuals are selected to prevail unchanged for the new generation, ensuring that the best solution will be chosen at the end. Next, the algorithm will select two individuals for the crossover and mutation methods, through a method based on the rank technique, where the individuals are probabilistically selected based on their ranking, which is influenced by their fitness.

The crossover was implemented in a double cut and splice variant. The objective is to find two common points in both parents and the data between them but, since the chromosome can be repeated multiple times, this technique can result in some conflicts. To solve that, it was decided to find only a common point and to swap the data from that point to the end, performing this operation twice. It should be noted that the common points must be decision nodes, in order to perform more and better changes in the individuals.

In mutation, a random decision point from the chromosome is selected and a new implementation is generated from that point to the next decision point.

These procedures, from selection to mutation, will be performed several times until the new population is completed; they will be performed over several generations, creating new populations based on previous ones. A stopping criterion was defined based on the convergence of the solutions for the same fitness; the minimum and maximum numbers of generations were also configured.

After finding the best solution, the optimizer agent will set the nodes with the configuration in order to ensure the correct path for each product.

5 Results

The algorithm was exposed, during his development, to several tests using different simulation environments with different complexities. In the end, the algorithm was then tested in the layout presented in Fig. 3.

This model is a known model of manufacturing line. It has a one entry point and two exit points, identified as *S*. In this line, there are five stations to perform four operations, with the existence of parallel stations. These stations can be identified in the figure as SK_X with X being the skill to be executed. The different skills were called A, B, C and D.

The tests are intended not only to understand the behaviour of the algorithm in these lines when exposed to different products but also to analyse some parameters such as population length and temporal complexity. Four different products were considered for the specific manufacturing line; they were tested 30 times, each one individually in order to evaluate their own complexity. There was also made a fifth test, in the same conditions, with two products simultaneously. The following results were obtained (Table 1).

It was expected that, with the increase of fitness, the processing time also increases; however this was not always confirmed. The reason is that the algorithm stops in different conditions, running in some tests a few unnecessary generations.

Finally, the impact of the population size in processing time was also tested. For that, the algorithm was submitted to the same conditions of the fifth previous test, the one with two products, and tested 30 times for each variation in population. The following results were obtained (see Fig. 4).

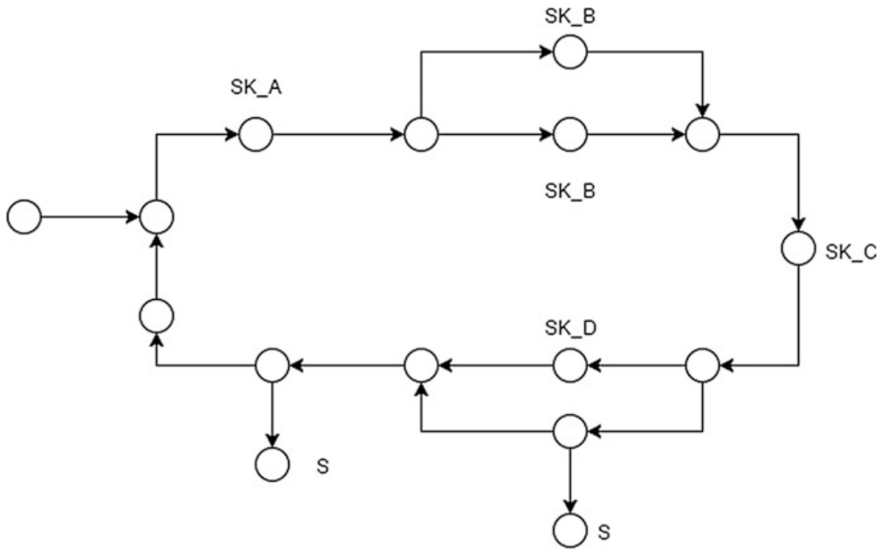


Fig. 3 Model of the manufacturing line tested

Table 1 Processing time comparing to average fitness for each product

Products (Skills executed)	Average processing time (ms)	Standard deviation	Average fitness
A; B; C	155.3	54.75	59
C; A; B	233.06	45.3	102.9
A; B; C; D	229	31.67	115
A; B; C; D; A; B; C; D; A; B; C; D	451.56	64.50	251
A; B; C; D; A; B; C; D; A; B; C; D; A; B; C; D	757.83	68.72	357

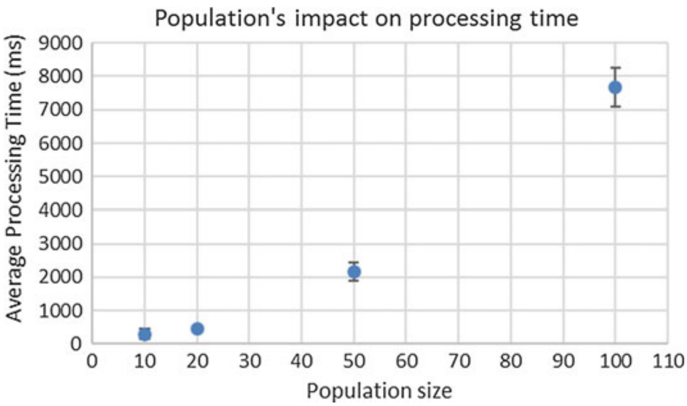


Fig. 4 Population's impact on processing time

As one can see the processing time increases a lot with the increase in population. The number of individuals is also related to the complexity of solutions. For simple products that don't have many skills to operate, it is useless to work with a higher population. However, with the increase of their complexity, a higher number of individuals might be necessary.

For all the previously described tests a population of individuals based on the following formula was used:

$$\text{Population} = 20 + \text{no\_Products} * \text{no\_Skills} / 10 \qquad (1)$$

This formula produces a good outcome, the value 20 being considered as the base of population size. The formula intends to create more individuals with the increase in the product's complexity. Despite the fact that the complexity of the problem is part of the layout of the manufacturing line, it is impossible to evaluate and translate complexity into a mathematical expression, with parameters dependent only on the type of products.

6 Conclusions

In this paper was proposed a genetic algorithm to solve a problem of reconfiguring a material handling system. The results demonstrate that this algorithm can be seen as a possible solution for the material handling system optimization but it is also possible to conclude that its complexity increases a lot with the increase of individuals in the population.

It was noticed in these tests that the solutions quickly converge to the final solution, raising the question of whether the algorithm would not be a too complex solution for a simple problem or simple manufacturing lines. However, with the increase of products' complexity, as in the 4th and 5th test, the algorithm's behaviour justifies its choice.

These results are encouraging in the sense that the approach seems to be a possible solution to the problem yet, more tests need to be done and compared to different approaches. For further research, the algorithm should be improved in terms of space complexity. Also in the future it is intended to implement new agents that can configure immediately the PLCs and other hardware in the assembly line, in order to allow the optimizer agent to send the respective configuration.

References

1. Rocha, A.: An agent based architecture for material handling systems (2013)
2. Setchi, R.M., Lagos, N.: Reconfigurability and reconfigurable manufacturing systems: state-of-the-art review, Ind. In: INDIN '04 2nd IEEE International Conference, Informatics, pp. 529–535 (2004)
3. El Maraghy, H.A.: Flexible and reconfigurable manufacturing systems paradigms. Flex. Serv. Manuf. J. **17**(4), 261–276 (2006)
4. Koren, Y., Heisel, U., Jovane, F., Moriwaki, T., Pritschow, G., Ulsoy, G., Van Brussel, H.: Reconfigurable manufacturing systems. CIRP Ann. Manuf. Technol. **48**(2), 527–540 (1999)
5. Mukund Nilakantan, J., Ponnambalam, S.G.: An efficient PSO for type II robotic assembly line balancing problem. In: IEEE International Conference on Automation Science and Engineering (CASE12), vol. 56, no. 3, pp. 600–605 (2012)
6. Sabuncuoglu, I., Erel, E., Alp, A.: Ant colony optimization for the single model U-type assembly line balancing problem. Int. J. Prod. Econ. **120**(2), 287–300 (2009)
7. Gen, M., Cheng, R.: Genetic Algorithms and Engineering Design, pp. 1–65 (1997)
8. Umar, U.A., Ariffin, M.K.A., Ismail, N., Tang, S.H.: Priority-based genetic algorithm for conflict-free automated guided vehicle routing. Procedia Eng. **50**, 732–739 (2012)
9. Yu, H., Shi, W.: A genetic algorithm for multi-model assembly line balancing problem. In: IEEE International Symposium on Assembly and Manufacturing (ISAM13), vol. 7, no. 1, pp. 369–371 (2013)
10. Xiaohong, L., Zhenyuan, J., Fuji, W., Wei, L.: Workshop material handling system lean planning based on genetic algorithm. Int. Symp. Comput. Intell. Design **1**, 261–264 (2008)
11. Luo, M., Zhang, J.B., Wong, M.M., Zhuang, L.Q., Ng, K.C.: Computational intelligence approaches for improving serviceability of material handling systems, vol. 10, pp. 218–223 (2007)

12. De Oliveira, J.A.B., De Matos, P.L.C.: Coalition based approach for shop floor agility—a multiagent approach, Electrical Engineering and Robotics Integration Manufacturing, Universidade Nov. Lisboa, Fac. Ciências e Tecnol., vol. Ph. D., p. 329 (2003)
13. Beyer, T., Yousefifar, R., Abele, S., Bordasch, M., Wehking, K.H.: Flexible agent-based planning and adaptation of material handling systems. IEEE Int. Conf. Autom. Sci. Eng. 1060–1065 (2015)
14. Bellifemine, F., Caire, G., Greenwood, D.: Developing Multi-agent with JADE Systems (2007)

Smart Condition Based Maintenance (S-CBM) for a Fleet of Mobile Entities

Adoum Fadil, Joffrey Clarhaut, Guillaume Branger and Damien Trentesaux

Abstract In a highly competitive market, the dynamic maintenance of mobile entities in the field of transportation systems is a major research field, in which maintainability, availability, maintenance charging and optimization are vital. Based on the ISO 13374 standard and previous research works, this article proposes a hybrid cooperative architecture for the monitoring of mobile entities. This architecture increases on-board diagnostic capabilities and optimizes the overall maintenance of a fleet of mobile entities.

Keywords Mobile entities, railway · Smart maintenance · Hybrid architecture · CPS · Monitoring equipment · CBM · Big data · Manufacturing

1 Introduction

1.1 Industrial Context and Needs

The increasing mobility of people and goods across the world implies new needs in terms of maintenance of complex transportation systems, typically trains, trucks, ships and planes. To answer these needs, transportation system manufacturers and mobile system owners have to increase the quality of service and the availability of

A. Fadil (✉) · J. Clarhaut · D. Trentesaux
UVHC, LAMIH UMR CNRS 8201, 59313 Valenciennes, France
e-mail: AhmatFadil.Adoum@etu.univ-valenciennes.fr; adam.adham_fadil@rail.bombardier.com

J. Clarhaut
e-mail: Joffrey.Clarhaut@univ-valenciennes.fr

D. Trentesaux
e-mail: Damien.Trentesaux@univ-valenciennes.fr

A. Fadil · G. Branger
Bombardier Transport, 1 Place des Ateliers, 59154 Crespin, France
e-mail: guillaume.branger@rail.bombardier.com

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_11

mobile entities coupled with a reduction of maintenance operating costs [1]. In a context of progressive market liberalization and with the entry of emerging countries thus promoting more competition, the battle is fierce between transportation systems manufacturers, making intelligent preventive maintenance a real strategic and political issue [2]. The challenges of intelligent preventive maintenance are huge and offer the following opportunities:

- **For mobile system manufacturers**: To ensure a good reliability of manufactured entities through returns on experience; to improve the image and the company's competitiveness and to create tomorrow's jobs and services.
- **For mobile system owners**: To provide more availability and a better planning of mobile entities; to optimize maintenance from a financial point of view (Life Cycle Cost (LCC)); to reduce the number of immobilized mobile entities and to improve the image and the company's competitiveness for its customers.
- **For mobile system users**: To maintain satisfaction, trust, and security and to reduce the loss of time.

1.2 Research Opportunities and Proposal

Recent sciences and technology innovations in the field of connected objects offer the following opportunities for mobile system manufacturers:

- Information and communication technology through connected objects, embedded systems [3] and Cyber-Physical Systems (CPS) [4] allows the implementation, the information sharing and the monitoring of distributed solutions based on intelligent agents in order to satisfy preventive maintenance requirements.
- Communication standards in terms of reliability, mobility support and fast data exchanges favour the emergence of new applications and services for maintenance that were until now unattainable [5].

Based on these innovations, this paper focuses on the definition of a generic method of diagnosis and maintenance across a fleet of mobile entities (fleet of trucks, trains, airplanes, ships…). Each mobile entity is composed of a set of hierarchically organized systems and it is assumed that each system has one or more transmitters and can communicate with remote control centres using wayside communication systems and infrastructure. In this context, the main objective is:

- To monitor distributed system components with specific hierarchical cooperative and embedded systems. These embedded systems interact with the wayside system located on the infrastructure.
- To increase the level of availability of each entity while reducing the cost of a maintenance level fleet (Fleet level LCC).

2 Dynamic Monitoring and Maintenance of Mobile Entities

2.1 Needs and Expectations

To maintain innovation and competitiveness, train manufacturers, like Bombardier Transport (BT), has launched major research and development projects. The SURFER project (SURveillance active FERroviaire, translated as "active train monitoring") was initiated by BT to provide a more advanced solution for the on-line diagnosis of incipient failures and faults than can occur during the train service [6]. Based on this project, the scientific work presented in this article proposes a generic diagnosis and monitoring method based on a holonic architecture composed of agents embedded on a fleet of mobile entities. Embedded Holons will have decision-making and communication skills to determine health status of monitored system, opening the door to health status prediction and thus, smart condition-based maintenance.

2.2 Relevant Issues

This objective is not obvious to reach because train manufacturers face some issues when trying to reach an effective condition-based maintenance at fleet level. These difficulties are as follows:

- **Logistics**: Today, during its exploitation phase, data acquisition from sensor networks of one mobile system is not an easy task for manufacturers. This acquisition often requires specific embedded systems with communication devices and infrastructure licenses. This exploitation difficulty often impacts the durability and/or increases implementation delays of embedded solutions.
- **Heterogeneous environment**: the fleet monitoring architecture has to take into account other existing heterogeneous monitoring applications. This architecture has to negotiate, to cooperate on task executions and to share results with other previously integrated applications.
- **Big Data**: in addition to the known problems of Big Data as acquisition mode (real-time, semi real-time and/or batch), performance and data volume management must be considered [7]; a fleet dynamic monitoring implies additional features in terms of time constraint, lack of generic methods (monitoring methods by homogeneous class of systems) and recursive monitoring (data can be analysed at several levels as needed [6]).
- **Security**: Data and results of the fleet maintenance should inspire authenticity, confidentiality, integrity and availability [8].

- **Organizational impact**: The objective of LCC fleet level cannot be ever achieved without a profound reorganization and an adaptation of maintenance teams to the behaviours of intelligent service agents.

In the remaining of this paper, we focus on the train transportation. Before introducing our proposal, a short literature review is suggested in the domain of condition-based maintenance standards.

3 Condition-Based Maintenance of a Railway Fleet

3.1 State of the Art

The ISO13374 standard [9] for monitoring and diagnosis presented by Le bold [10, 11] and Thurston [11] establishes general guidelines of specifications, processing, communication and presentation data for monitoring and diagnostics of machines. Figure 1 illustrates the layers of this guideline. This standard was originally developed to monitor machines or isolated industrial equipment but not for the monitoring of mobile systems. For mobile systems, Le Mortellec in [6] proposed a decomposition of these layers in two axes as follows:

- **A vertical distribution of layers**: processes associated with the functional layers of ISO 13374 can be made between mobile entities (layers #1, #2, #3, and #4) and the maintenance centre (layers #5 and #6);
- **A horizontal distribution of layers**: the monitoring of a mobile entity is made thanks to the monitoring of its own systems and its corresponding sub-systems. Therefore, processes associated with the functional layers of the ISO 13374 must be made according to these subsystems, in which technology differs from one subsystem to another.

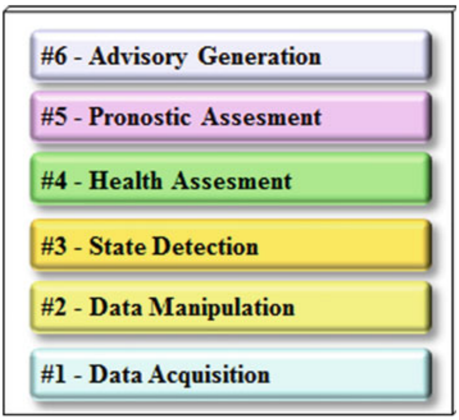


Fig. 1 ISO architecture model 13374 [9]

This decomposition takes into account the constraints of mobility, subsystems and environment of mobile entities and the requirements of a monitoring system in terms of accuracy, comprehension, dependability, responsiveness and confidence. However, despite this decomposition evolution, difficulties arise when incorporating specific requirements and constraints such as:

- Monitoring a homogeneous class of systems, independently of the mobile entities within a fleet. It aims to facilitate comparative diagnostics systems for discriminating tasks needs for Dynamic maintenance;
- Providing support for default risk and maintenance charge to optimize continuously the performance level fleet (F-LCC-D);
- Allowing exchanges with others existing heterogeneous applications of the monitoring system.

3.2 Generic Diagnosis Method and Architecture Proposal

Based on the holonic monitoring architecture, the proposed hybrid architecture extends the services of EMH architecture (Embedded Monitoring Holarchy), presented in [6]. The EMH architecture allows developing a robust board diagnostic decentralized and cooperative on board mobile target systems [12, 4]. The EMH architecture offers the following specifications:

- Choosing of a diagnosis Embedded Decentralized and Cooperative Architecture enables a comprehensive diagnosis of the target system from the information provided by different diagnostic entities located aboard the target system and can interact with each other. It also offers the accuracy, adaptability, responsiveness and trust.
- The holonic approach was preferred over multi-agent approach because it allows considering both the physical part of a monitored target system and the "active" monitoring entity associated with it. This approach also allows for recursively describing the target system and reflects its tree monitoring structure obtained by systemic breakdown [13], and is based on autonomous, recursive and cooperative monitoring entities named holons.
- At each level of the decomposition of the target system, a monitoring holon is formed by a tangible system under surveillance and a monitoring function in charge to diagnose this system. It has behavioural inhibition, cooperation (extended inhibition) and fusion events.

Our contributions are as follows:

- Implementing a new layer called "dynamic maintenance layer"; this layer uses the results of monitoring of the Health and/or Prognostic layers and adds an optimization and management dimension for better planning of resources and increasing the availability level of the mobile entities while reducing the overall cost (F-LCC-D);

- The monitoring system is carried out in a homogeneous class application, independently of mobile entities within a fleet as showed in Fig. 2. These systems have their own response and cycle times. Therefore, this type of homogeneous analysis facilitates monitoring, discrimination suspicious behaviour and comparative diagnostics systems to discriminate the needs of condition-based maintenance.
- Increasing security exchanges between mobile entities and the layers of the Wayside Maintenance Center (WMC) [8] against the various threats of cyber-crime [14]. Each mobile entity has a certificate signed by the wayside maintenance centre; this certificate allows it to communicate with other entities and secures its communications exchanged according to the standard Public Key Infrastructure (PKI) [15]. This ensures authenticity, confidentiality, integrity and availability of data.
- A hybrid implementation on both mobile entities and on the wayside maintenance centre, such as follows:
  - **On mobile entities (Trains)**; it facilitates a dynamic implementation of Data Acquisition, Data Manipulation, State Detection and Health Assessment layers according to the requirements of embedded systems calculators;
  - **On the wayside maintenance centre (WMC)**; it provides for the implementation of all the layers the choice of monitoring in redundant layers made through negotiations and the pooling of monitoring with the layers of the mobile entities; implementation takes into account also the existence of heterogeneous applications.

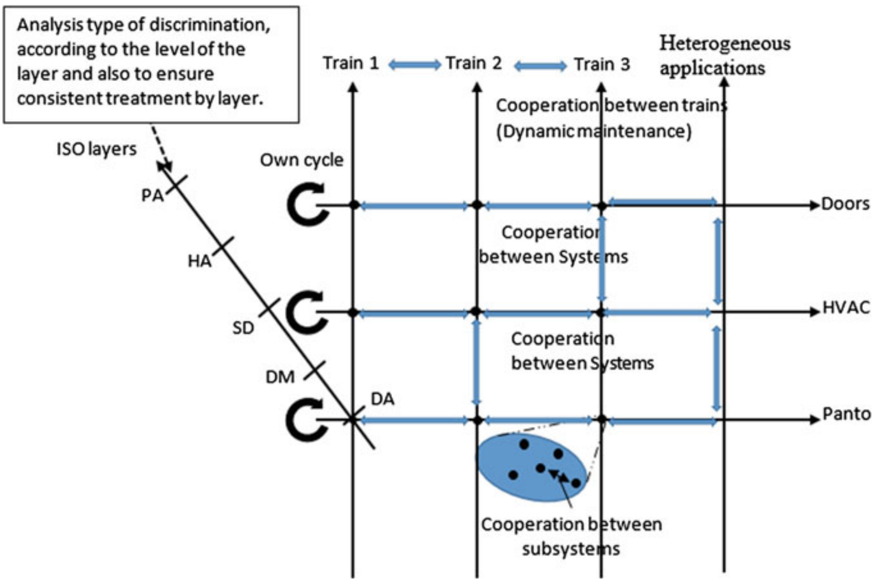


Fig. 2 Monitoring homogeneous class systems

Table 1 Missions of holons layer by layer

Layer	Mission of holons in mobile entities (Trains)	Mission of holons in the wayside maintenance centre (WMC)
Data acquisition (DA)	Converting continuous signals delivered by sensors networks or transducers into a digital parameter	Acquisition and integration of signals delivered by acquisition agents of mobile entities, where these have not yet implemented the upper layers (MD, SD, etc.)
Data manipulation (MD)	• Analyses signals and creating inductors • Exchanges with heterogeneous application	Signals analysis, exchange with heterogeneous application and creation of inductors for systems of mobile entities that have not yet embedded manipulation layer
State detection (SD)	• Searches for anomalies on signals; generate alerts/alarms • Exchanges with heterogeneous application	Exchanges with heterogeneous application and generate alerts or alarms for mobile entities that have not yet embedded detection layer
Health assessment (HA)	• Determines health status (reports) • Exchanges with heterogeneous application	• Acquisition of reports and/or determines health status for systems of mobile entities that have not yet embedded HA layer • Exchanges with heterogeneous application
Prognostic assessment (PA)		• Determines a future failure mode for systems of mobile entities, fleet level • Exchanges with heterogeneous application
Smart condition-based maintenance		• Planning and optimization of maintenance tasks (F-LCC-D) level float • Exchanges with heterogeneous applications

- In addition to classical monitoring agents, each layer owns new agents for generating advises and maintenance actions (Advisory generation).

Table 1 illustrates, for each layer of the proposed hybrid monitoring architecture, the missions of the Holons.

3.3 Expectations and Current Developments

The hybrid monitoring architecture presented in this paper is a redundant, flexible, dynamic architecture that offers the following advantages:

- It allows a progressive implementation thanks to the dynamic maintenance layer without disturbing the overall implementation schedule;

- It accelerates learning time and return on experience of monitoring holons;
- It allows cooperation with its heterogeneous environment applications;
- It enables a dynamic system monitoring support with the deployment of embedded monitoring agents;
- It integrates management, cost and optimization through the dynamic maintenance layer;
- It increases the level of system monitoring through dynamic load balancing between mobile entities and the WMC;

Actually, ten trains are already equipped with computers dedicated to preventive maintenance. In the same way, in the WMC, the implementation of the heterogeneous architecture of surveillance is currently in progress.

4 Conclusion and Future Research Works

In this article, a holonic architecture for the smart condition-based maintenance of a fleet of complex mobile systems is proposed. Some holon missions have been detailed. First applications are being made and developments are being embedded into some trains by Bombardier Transportation. Short term perspectives are on the analysis of this architecture in order to have some return on experience. Medium and long term perspectives concern the gradual implementation of monitoring systems for one fleet of mobile entities and the implementation of several mobile fleets within the same hybrid architecture.

References

1. Yokoyama, A.: Innovative changes for maintenance of railway by using ict–to achieve “smart maintenance”. Procedia CIRP. **38**, 24–29 (2015)
2. Muzalevsky, R.: The Implications of China’s High-Speed Eurasian Railway Strategy for Central Asia. http://www.jamestown.org/single/?no_cache=1&tx_ttnews%5Btt_news%5D=36234
3. Paolucci, P.S., et al.: Dynamic many-process applications on many-tile embedded systems and HPC clusters: the EURETILE programming environment and execution platforms. J. Syst. Archit. **69**, 29–53 (2016)
4. Rho, S., Vasilakos, A.V., Chen, W.: Cyber–physical systems technologies and application—part II. Future Gener. Comput. Syst. **61**, 83–84 (2016)
5. Mitra, R.N., Agrawal, D.P.: 5G mobile technology: a survey. ICT Express. **1**, 132–137 (2015)
6. Le Mortellec, A., Clarhaut, J., Sallez, Y., Berger, T., Trentesaux, D.: Embedded holonic fault diagnosis of complex transportation systems. Eng. Appl. Artif. Intell. **26**, 227–240 (2013)
7. Wang, G., Gunasekaran, A., Ngai, E.W.T., Papadopoulos, T.: Big data analytics in logistics and supply chain management: certain investigations for research and applications. Int. J. Prod. Econ. **176**, 98–110 (2016)
8. Moldovan, L., Gergely, A.M., Crainicu, B.: The concept of a distributed repository for validating X.509 attribute certificates in a privilege management infrastructure. In: 9th

International Conference Interdisciplinarity in Engineering, INTER-ENG 2015, 8–9 October 2015, Târgu Mureş, Romania. Procedia Technol. 22, pp. 926–930 (2016)
9. BSI - BS ISO 13374-1—Condition monitoring and diagnostics of machines Data processing, communication and presentation Part 1: General guidelines, Engineering 360. http://standards.globalspec.com/std/248624/bsi-bs-iso-13374-1
10. Lebold, M., Reichard, K., Boylan, D.: Utilizing dcom in an open system architecture framework for machinery monitoring and diagnostics. In: Proceedings IEEE Aerospace Conference, 2003. pp. 3_1227–3_1236 (2003)
11. Thurston, M., Lebold, M.: Open standards for condition-based maintenance and prognostic systems. ResearchGate (2015)
12. Le Mortellec, A., Clarhaut, J., Sallez, Y., Berger, T., Trentesaux, D.: An embedded cooperative holarchy for diagnosing complex moving systems. In: Borangiu, T., Dolgui, A. (eds.) 14th IFAC Symposium on Information and Control Problems in Manufacturing (INCOM). IFAC, Bucharest, Romania (2012)
13. Belhadaoui, H.: Safe design of intelligent mechatronic systems critical applications. PhD thesis at the National Institute Polytechnique de Lorraine (2011)
14. Markelj, B., Bernik, I.: Safe use of mobile devices arises from knowing the threats. J. Inf. Secur. Appl. **20**, 84–89 (2015)
15. Blanch-Torné, S., Cores, F., Chiral, R.M.: Agent-based PKI for distributed control system. In: 2015 World Congress on Industrial Control Systems Security (WCICSS), pp. 28–35 (2015)

Part III
Sustainability Issues in Intelligent Manufacturing Systems

Emerging Key Requirements for Future Energy-Aware Production Scheduling Systems: A Multi-agent and Holonic Perspective

Damien Trentesaux, Adriana Giret, Flavio Tonelli and Petr Skobelev

Abstract The aim of this paper is to study a set of emerging key-enabling requirements for the design of multi-agent or holonic manufacturing systems dealing with the energy aware scheduling of future production systems. These requirements are organized according to three different views, namely informational, organizational and lifecycle views. It is shown that these emerging key-enabling requirements are not sufficiently addressed by the research literature. An illustrative futuristic example of a system complying with these requirements is provided. From this example, new research opportunities and issues can be easily found.

Keywords Energy aware scheduling • Intelligent manufacturing systems • Multi-agent systems • Holonic systems • Requirements

D. Trentesaux (✉)
LAMIH, UMR CNRS 8201, University of Valenciennes and Hainaut-Cambrésis, UVHC, Le Mont Houy, 59313 Valenciennes Cedex, France
e-mail: damien.trentesaux@univ-valenciennes.fr

A. Giret
Dpto. Sistemas Informáticos y Computación, Universidad Politécnica de Valencia, Valencia, Spain
e-mail: agiret@dsic.upv.es

F. Tonelli
Department of Mechanical Engineering, Energetics, Management and Transportation (DIME), University of Genoa, Via all'Opera Pia 15, 16145 Genoa, Italy
e-mail: flavio.tonelli@unige.it

P. Skobelev
Smart Solutions Ltd. and Samara University, Samara, Russia
e-mail: petr.skobelev@gmail.com

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_12

1 Introduction

This paper focuses on the challenging issue of energy aware production scheduling. Energy aware production activities are nowadays and will remain in the future a major industrial issue because of the increasing short-term volatility in cost and availability of energy coupled to an increasing average (long-term) cost of energy.

This paper lies within the field of Intelligent Manufacturing Systems (IMS). In that field, one can face an important set of contributions dealing typically with the design of multi-agent systems (MAS) and holonic manufacturing systems (HMS) to control in a predictive or reactive way the scheduling of manufacturing activities. Among this abundant literature, energy starts to be considered in the development of energy-aware MAS and HMS [1] in response to the challenging issue introduced. One main advantage of using IMS based approaches such as MAS and HMS in energy-aware production comes from their potential ability to cope with the unexpected and the complexity induced by the energy-dimension [2].

Meanwhile, designing energy-aware production scheduling systems ($EAPS^2$) requires a set of profound modifications not only in MAS/HMS scheduling control models but also in informational and organizational processes in industry to exploit or permit the use of these MAS/HMS based $EAPS^2$. In other words, focusing only on the core activity of scheduling with no attention paid to key-enabling energy-compliant requirements is a too restrictive view: a more global view is required, where the $EAPS^2$ is immerged in a more global environment enabling its functioning [3]. In that sense, this paper is aligned with the conclusion of an interesting study [4] pointing out the gap between research and industry in energy management in production and fostering researchers to work on: the design of energy-aware decision support systems to visualize and handle trade-offs between energy and other KPIs; realizing full benchmarking activities; developing improved monitoring and control systems of energy; and, extending MES (Manufacturing Execution Systems) and ERP (Enterprise Resource Planning) to support the energy as a new clear dimension.

In this context, this paper intends to study key-enabling emerging energy-requirements of MAS/HMS based $EAPS^2$ that enable a viable integration of the designed MAS/HMS based $EAPS^2$ into next generation production environments. We assume then that the $EAPS^2$ is designed using IMS principles, the introduced expected advantages of MAS/HMS paradigm in the context of energy management being considered. Meanwhile, it is important to note that this paper does not focus on energy generation issues and does not deal with the internal design of the $EAPS^2$ itself either, for which several literature reviews and a large amount of work are available [5].

2 Key-Enabling Emerging Requirements in MAS/HMS Based Energy-Aware Production Scheduling

The impact of the design of energy aware processes in future factories has been studied in recent years [6]. Inspired from [7] and aligned with the conclusions presented in [4], we study here several key-enabling emerging requirements that we have classified in three categories, namely informational, organizational, and life cycle points of view, as summarized in Table 1.

As introduced, these requirements are emerging key-enablers. In other words, they are pre-requisites to design viable MAS/HMS based $EAPS^2$ in a near future context where the availability of energy as well as its cost will be more variable and unpredictable. It is important to note that this context will not evolve in a favourable way: the long-term average energy cost, still increasing, coupled with the reduction of fossil-based energy availability will force industrialists to consider alternative integrated energy sources (wind, solar energy). These sources will contribute to increase further the unpredictability in availability and cost of energy purchased and produced.

It is clear that these key-enabling emerging requirements are not proved to be exhaustive, but from our point of view these are among the most critical ones. In the remaining of the paper, these key-enabling emerging requirements are detailed and discussed. When available, illustrative examples from the literature are provided.

Table 1 Classification and presentation of the proposed requirements

Point of view	Requirement reference	Description of the main features of the requirement
Informational	REQ#I1	A MAS/HMS based $EAPS^2$ must be interoperable with other information systems of the enterprise or beyond
	REQ#I2	A MAS/HMS based $EAPS^2$ must be able to observe and control energy variables of energy consuming resources
Organizational	REQ#O1	A MAS/HMS based $EAPS^2$ must be able to control the different types of energy wastes that have been identified and categorized
	REQ#O2	A MAS/HMS based $EAPS^2$ must be able to support reconfigurations of the manufacturing systems, according to the cost and the availability of energy in the mid-long term as well as manufacturing resources availability
Life-cycle	REQ#L1	Like any system designed according to sustainability principles, the lifecycle assessment of a MAS/HMS based $EAPS^2$ is recommended

3 Integration of the Key-Enabling Emerging Requirements in MAS/HMS Based Energy-Aware Production Scheduling

3.1 REQ#I1: Interoperability with Information Systems

The MAS/HMS based $EAPS^2$ must be at least interoperable with the enterprise's information system application layers dealing with production, mainly ERP, APS (Advanced Planning and Scheduling), MES and SCADA (Supervisory Control And Data Acquisition) systems. The interaction with these systems occur at a transactional level (ERP, MES, SCADA) as well as at a simulated off-line level (APS); this last interaction improves convergence between ERP and MES by allowing "what-if" scenario generation and analysis and where $EAPS^2$ can be an important component. Often neglected, the impact of paying attention to energy in a manufacturing system is profound. It is causing shifts in the way manufacturing companies operate, interact with their customers (especially other businesses), and offer services and products. In fact, almost every aspect of a business will be affected by a comprehensive global sustainability approach. In order to provide solutions to this situation, enterprise level management modules are starting to be conceived as tools in which energy and sustainability needs are intertwined (an example is ERP Green from Sustainable Dynamics). This requirement drives attention to the impact of the right interoperability between information systems and a MAS/HMS based $EAPS^2$ in order to implement a holistic sustainable approach, unless parts of the "global chains" are broken. It is crucial that the information flowing among the different levels of the global environment in which the $EAPS^2$ is immersed is properly taken into account.

The upper management levels (typically, ERP/APS) are key for the right functioning of $EAPS^2$, being MAS/HMS based or not. The information that comes from, for example an ERP, defines the activating events for an $EAPS^2$ in order to start a new schedule or to dynamically react, modifying a running one through for example the interaction with a supervisory or staff holon (PROSA-like approach) [8]. Not only the incoming information is key, but also the outgoing information that must flow timely and with the right format to the upper management levels in order to assure the optimized treatment of the means by the global manufacturing system and to provide energy-oriented (and more, globally, sustainability oriented) relevant information to decision makers. Furthermore, such integration will pave the way to "information platforms" that visualize the means of the factory on different levels taking into account new kinds of Key Performance Indicators (KPI). These KPIs should also be used for benchmarking or even for efficiency labels for factories, processes, machines and products.

Research works that explicitly tackle this requirement are still lacking, despite the fact that it was already identified in the Strategic Research Agenda ranked list from the REViSITE project ICT4EE roadmap [9]. From an industrial point of view, some companies in the ERP-sector have already launched initiatives to include analyses on resource and energy efficiency as well as carbon emissions in their software, but this is not sufficient [4].

At lower management levels, the link with MES and SCADA systems is also to be studied in a holistic way. Holonic Manufacturing Execution Systems (HMES) could be for example extended in the context of $EAPS^2$ using for example the concept of go-green holon [10] to enable the monitoring of energy-based KPI (cf. REQ#I2).

Meanwhile, this "vertical" interoperability is not sufficient and must be completed by a "horizontal" interoperability, the one that deals with product and systems life cycle management (e.g., Product Lifecycle Management PLM software). From our point of view, the concept of "intelligent products" [11] can be adapted, for example, in the context of energy-aware lifecycle management.

More, even this widening is not sufficient. Indeed, some innovative interoperability links can be identified when considering that the MAS/HMS based $EAPS^2$ should also be interoperable with systems beyond the enterprise itself, and among them, energy provider and other energy consumer systems. This interoperability will concern the gathering of *reliable* information about energy peaks, availability, and costs, but will also concern possible negotiation with these systems to find a compromise between production needs and energy offers, as it will be illustrated in the example provided at the end of the paper.

3.2 REQ#I2: Ability to Observe and Control Energy Variables

From a software point of view, the capability to collect and to manage energy consumption information at the desired granularity (from sensors/actuators level to business management level) is needed to identify demand patterns and to develop appropriate strategies and policies to address energy management and saving opportunities. Also, the capability to measure detailed energy utilization insights into sub-system interactions can lead to further efficiency improvements as much as the interactions are comprised and modelled. From a hardware point of view, energy-aware sensors and actuators are required to allow such a pervasive monitoring and energy control. Moreover, standards for the design and deployment of optimal sensor networks as well as "smart" energy devices are required for the integration with higher-level energy management systems (see for instance IEC/TR 62837 Energy efficiency through automation systems). This requirement bridges then the digital world with the physical one. Within the context of MAS/HMS based $EAPS^2$, controllability and observability logically lead to the integration of the concept of "cyber-physical systems" (CPS) when defining the MAS/HMS based $EAPS^2$.

A MAS/HMS based $EAPS^2$ requires such a CPS approach to enable it to "close the loop", from energy sensors to actuators through control decisions. This requirement refers to the well-known concepts of "controllability" and "observability" of systems from the control theory:

- *Energy controllability* ensures that the decisions of the MAS/HMS based $EAPS^2$ *actually* influence energy-related KPIs. This is often the case but some basic attention must be paid to avoid uncontrollability (e.g., if a holon controls the speed of a robot for which the energy used per time unit is inversely proportional to this speed, then the total energy consumption for this robot is unfortunately uncontrollable by the holon).
- *Energy observability* requires that energy-consuming resources (in a broad sense) must be equipped with technological solutions enabling the real-time measurement and the forecasting of:
  - manufacturing energy supply or needs;
  - resources energy consumption;
  - peak overall power consumption during manufacturing;
  - energy prices and energy costs for manufacturing;
  - re-used/recycled energy wastes (cf. REQ#O1);
  - energy efficiency measures and related performance indicators.

The energy consumption remains hard to measure and to model, then hard to predict. Typically, consumption is function of various parameters depending on low-level physical processes, operations, product characteristics (weights), age of resources, maintenance, etc. In [12], authors propose a standard-based infrastructure to collect and monitor energy data in real time for manufacturing and production systems, along with a manufacturing energy management system (MEMS). In case of difficulties to measure energy directly, control theory and modern identification technics (observers and state re-constructors) can be used and embedded into holons [13]. Other techniques, such as function blocks approaches, discrete event modelling and simulation, axiomatic or feature-based design may also help to measure indirectly energy and to predict its evolution [14].

Meanwhile, in the context of IMS and from our review, existing energy profile models embedded into holons remain very (too) simple. For example, in [10] a MAS/HMS based $EAPS^2$ was applied to a flexible manufacturing system for which direct resource consumption was not measurable in real-time. To solve this issue, a highly simplified energy consumption model has been designed from offline single tests with global measurements at cell level. From these essays, a 3-constant levels consumption model has been chosen for each of the robot controlled by a holon (robot off mode: consumption 0; sleeping mode where power was not supplied to the robot: constant consumption C_1; and working mode: constant consumption C_2). This kind of model is clearly not sufficient since the consumption depends in fact on the load of robots and peak/Dirac consumptions at starting moments of movements were always neglected.

More elaborated energy models have been developed in [15]. In this work, each production resource is associated with an intelligent agent providing the interface with information systems (e.g., APS, MES…). The intelligent agent provides the resource's state, the resource's total energy consumption, the resource's operations, and the resource's performances for an operation (speed, timeliness, power consumption, quality of service, etc.). This agent observes the instantaneous energy consumption of the resource and calculates KPI such as energy used per operation type, total consumption, etc. The global architecture of this MAS/HMS based $EAPS^2$ allows supporting dynamically and in an easier manner changes in the physical world providing an additional improvement of the decision-support tool for performance measurement and management in the industrial sustainability and sustainable supply chains domain [16]. But, as for the previous example, observers consider the observed system as a black box to be identified according to classical control theory approaches.

In fact, controllability and observability difficulties come from the situation when resources have not been designed to support energy measurement and control. Nowadays, recent machines and robots are being equipped with "energy modules". Meanwhile, a great majority of existing resources are still not equipped while for some of them, cost issues or contextual constraints (corrosive environment, etc.) forbid and will still forbid such equipment.

Finally, it is important to note that all the previous discussion deals with energy expressed in term of electric power. This holds also true for the literature. Meanwhile, other kinds of energy sources can be found, for example energy from air compressors. Even if the primary energy is electricity to generate compressed air, the actual use of compressed air is complicated to express in terms of energy usage. In the previously cited example [10], the conveying system uses both electrical energy (for shuttle moves) and air (for switching gates). The measurement of the conveying system energy consumption was then so hard to realize that the authors of this MAS/HMS based $EAPS^2$ neglected this consumption.

3.3 REQ#O1: Ability to Control the Different Types of Energy Wastes

This requirement deals with organizational issues since it requires the revision of industrial processes and production methods themselves to integrate lost energy harvesting and re-injection systems. It is important to note that energy wastes make the energy use efficiency decrease (and for some industries, the amount of waste is of the same order of magnitude than the energy used to add value to products). The idea is to recycle energy wastes as a possible "free" energy input in the production system and to offer the MAS/HMS based $EAPS^2$ opportunities to reuse these wastes (e.g., by defining holons managing these wastes). Moreover, this provides a new decisional criterion for agents or holons when controlling production since they must also consider the minimization of the part of the wasted energy that cannot be harvested and that will be definitely lost. This kind of requirement concerns for example chemical/process industries or high energy consuming manufacturing systems like plastic production by injection moulding or manufacturing systems with heat treatments. In such industries, energy waste generated from heat losses during machining can be considered as a potential non-negligible "free" input energy to be handled.

Meanwhile, despite the fact that this original requirement can be seen as a source of important savings, from our knowledge there is no MAS/HMS based $EAPS^2$ paying attention to it. One reason comes from the fact that only few industrialists have mature solution to gather energy losses. From our point of view, the coupling of the global Monozukuri design principle from the Japanese automotive industry [17] with HMS/MAS based $EAPS^2$ could be an interesting evolution as well. Also, inspiration may come from other domains: building engineering, embedded systems and transportation of energy in the grid.

3.4 REQ#O2: Ability to Support Reconfigurations of the Manufacturing Systems

MAS/HMS based $EAPS^2$ must be able to support reconfigurations of the manufacturing system. "Reconfiguration" is here understood in its broad sense, according to different conditions (long-, mid-, and short-term) aligned with the management of energy at the same long-, mid-, and short-terms.

The long-term reconfiguration is related to the factory life-cycle and involves lifecycle engineering and product/process revisions in order to reduce energy consumption in terms of material, transformation, processing of raw materials, components, and final products or alternative ways to supply energy to the factory using integrated renewable resources or acquiring energy somewhere else. The long-term reconfiguration can also include new designed or purchased equipments with energy efficiency increased capability. In this case the MAS/HMS based $EAPS^2$ should transform existing energy requirements into the new reconfigured resources and constraints networked layout. As a consequence, and to support long-term reconfiguration, a agent/holonic control architecture must be reconfigurable (i.e., adaptable, evolvable) as well, following the different configurations of resources. This kind of reconfiguration is closely related to the REQ#L1 detailed hereinafter.

The mid-term reconfiguration, typically addressed at a planning level, tries to recombine existing resources in what-if planned scenarios (the typical domain of APS systems) by using aggregate data on resources availability and production constraints (one of these being energy) with respect to some cost or target function. In this case the MAS/HMS based $EAPS^2$ should be capable to elaborate feasible solutions, accordingly to contextual decision variables and decision maker's preferences, and showing their impact in terms of energy and cross-functional KPIs [18]. This can be done typically through cooperation or negotiation mechanisms among holons or agents. The obtained results can then be deployed to local areas, plants and lines subject to the interoperability requirement (see REQ#I1).

The short-term reconfiguration involves scheduling (and hence optimization where possible) of improved sequences flattening energy peaks, reducing total energy demand in a given time frame (day, shift, hour), selecting alternative resources in order to maximize energy saving, up to real-time monitoring and control of equipment and machines, providing switching on/off, stand-by and other sensor-actuator policies [19]. This kind of reconfiguration is the most addressed in the literature since it is the closest to the scheduling issue [20].

3.5 REQ#L1: Lifecycle Assessment of MAS/HMS Based EAPS^2

In conjunction with the previously introduced long-term reconfiguration ability and in addition to the improvement of current operations, consideration must also be given to the performance of future systems and equipment. For this, Life Cycle Assessment (LCA), which uses historical data, must be communicated to system and equipment designers alike. This data will permit designers to develop systems that can be more easily adjusted to reduce peak energy requirements and to provide overall gains in average consumption from past experiences. Furthermore, designers will not only have a better knowledge of expected energy efficiency, but will be also able to better model and design the system to achieve even greater savings. Effective Life Cycle Engineering (LCE) analysis, for instance, requires a holistic analysis of product design as well as process design in order to allow energy awareness in the manufacturing phase, providing key features not only at product level but at equipment and machines level, specifying requirements in terms of energy metering, saving, and controlling phases.

Accordingly to this extended view of factory life-cycle, a MAS/HMS based EAPS^2 should support the capability to integrate advances in technology allowing for example long-term product/process reconfiguration (REQ#O2); this opportunity may only occur occasionally so when it arises, efforts must be made to incorporate all existing knowledge into the design of higher efficiency systems and equipment. Typically, a holonic architecture complying with this requirement must be designed to support different technology modelling in a generic way (e.g., libraries).

A MAS/HMS based EAPS^2 is also a software product, which is designed to operate in an environment in order to achieve a set of goals. In order to assure its correct and efficient execution, performance-monitoring activities must be performed. The key to any system's effectiveness is whether its operations and outputs are achieving the performance targets. For a MAS/HMS based EAPS^2 this is directly linked to its environment and its set of goals, considering at the same time the changes that these elements may experience during the life cycle of the global manufacturing system. The environment in which the MAS/HMS based EAPS^2

exists is outlined by the definition of the different levels of the manufacturing company in which it is running, and the manufacturing processes that are scheduled in order to fabricate products. On the other hand, the set of goals that must be achieved by the MAS/HMS based $EAPS^2$ is normally defined as a set of multi-objective criteria that can include: classical time-based (completion times, flow times, tardiness/earliness…), or mixed time/quantity-based (throughput…) production objectives, together with sustainability means such as energy, CO_2 emission, waste, scrap, pollution, etc. [5, 21]. As a consequence, any change or new requirement on any of these elements will affect the performance of the MAS/HMS based $EAPS^2$. The majority of works on designing MAS/HMS based $EAPS^2$ take into account the elements described before but only their static view (assuming that no changes will appear during the entire life cycle of the manufacturing system). Few works pay attention to the dynamic nature of these elements. Such works focus solely on the dynamic (at run-time) changes of the sustainability goals in terms of targets changes but not on new sustainability goals. A literature review on dynamic approaches of $EAPS^2$ can be found in [21]. It is stated that an urgent attention to reconfigurable MAS/HMS based $EAPS^2$ must be paid by the specialized research community in order to equip these tools with powerful and fault tolerance features demanded by todays and future industrial scenarios.

Finally, it is also important to note that the MAS/HMS based $EAPS^2$ is a system itself, for which a LCA can thus be led. This will force designers to adopt a broader view on the different steps of the whole life of their MAS/HMS based $EAPS^2$. A LCA for MAS/HMS based $EAPS^2$ should arrange the performance-monitoring activities that will trigger/start the reconfiguration of the $EAPS^2$ in order to adapt to a change/reconfiguration of the products and manufacturing processes. With such an approach, a MAS/HMS based $EAPS^2$ may have different versions/configurations depending on the concrete definition of its environment and goals, and may evolve to other versions/configurations when any of these elements changes or when the performance targets are not met. According to this, for example, some holons may appear while other may disappear in the $EAPS^2$, the holonic architecture can also evolve to be more efficient. Applying the four main phases for LCA defined by the ISO 14040 [22] and 14044 [23] standards, the steps will be:

- Goal and scope definition, with a detailed specification of the environment in which the MAS/HMS based $EAPS^2$ is currently running, its goals and the performance targets,
- Life cycle inventory analysis for the input and outputs of the $EAPS^2$ taking special attention to the sustainability means,
- Life cycle impact assessment in order to evaluate the significance of the impacts of the Life cycle inventory results, and
- Interpretation for identifying, quantifying, checking and evaluating the need to evolve to other version/configuration depending on the performance targets.

4 An Illustrative Example

In this part, we provide an illustrative example of the application of previously introduced key-enabling requirements in a MAS/HMS based EAPS^2, focusing only on the informational and organizational requirements. This example is theoretical and futuristic, but is inspired from a real industrial need and opens some new interesting perspectives for researchers. In this example, a holonic system (consisting of three subsystems) for the coordinated scheduling of manufacturing production and energy generation in a region is considered, see Fig. 1:

- A Smart Plant #1, which is a factory energy consumer (or a regional supply chain), including a first MAS/HMS based EAPS^2 for manufacturing scheduling and control, denoted in short "**smart factory scheduler**",
- A Smart Plant #2, which is an energy provider/supplier—including a second MAS/HMS based EAPS^2 for energy generation scheduling and control, from different regional energy sources, denoted "**smart energy scheduler**",
- A regional MAS for conflict resolution and overall monitoring.

This open and distributed architecture can be seen as a MAS/HMS "system of systems" that gives the possibility to make coordinated decisions on manufacturing activities and energy consumption that should gain from being based on a service-oriented architecture and an enterprise service bus supporting P2P interactions ("peer to peer") [24]. It also comprises a knowledge base system consisting of two major components: an **ontology** that defines a model of domain knowledge and a **world scene** memorizing past events and decisions mirroring the reality and the evolution in the environment and reflecting a model of the situation with the actual state of all participants in real time. This architecture can be reconfigured (REQ#O2), and connections can be made using plug and play technologies to adapt or open this network at any time.

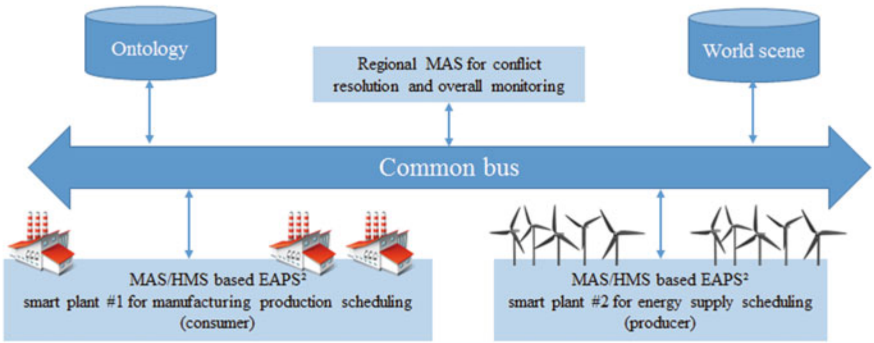


Fig. 1 The MAS/HMS "System of Systems" view

In this example, the top-level regional MAS for conflict resolution and overall monitoring contains agents representing both types of schedulers which receive data about their statuses and plans for the calendar period and time of the day. Smart factory and smart energy schedulers interact with their own ERP and MES systems (REQ#I1) to get production plans, batches of operations and their status. Each of the considered schedulers makes its own resource scheduling: the smart factory scheduler deals with manufacturing orders, processes, equipment and workers in the factory, and the smart energy scheduler deals with energy generation plans, for example, turning "on" or "off" turbines or launching CHP (combined heat and power plant) on coal, taking into account characteristics of orders and processes, supply chains, resource requirements, equipment condition, etc.

The schedule created for each of the systems (producer and consumer) is transferred to the general regional multi-agent system for conflict resolutions, which can ask any one of the systems to reschedule if necessary. If there is no conflict, the role of agent-representatives is the elaboration of fast coordination of plans and the sending of approvals to the schedulers. Meanwhile, in case of important events that violate the balance of interests, special vertical and horizontal protocols for negotiation and conflict resolution can be used.

Let us consider, for illustration purpose, a scenario example:

1. The factory receives at a date t_0 an unexpected new large order.
2. The smart factory scheduler adapts the production schedule, but notes then that it will imply the use of highly power-consuming equipment.
3. The smart energy scheduler is informed about the forecast in the increase of the energy consumption of the factory. The rise of the power needed is precisely evaluated.
4. Aside this process, the smart energy scheduler is also informed of unexpected variations to come in the availability of the renewable energy part, because of weather conditions. The conjunction of these two processes enables the smart energy scheduler to detect two overload peaks, see Fig. 2.
5. The smart factory scheduler receives precise information about such peak time intervals.

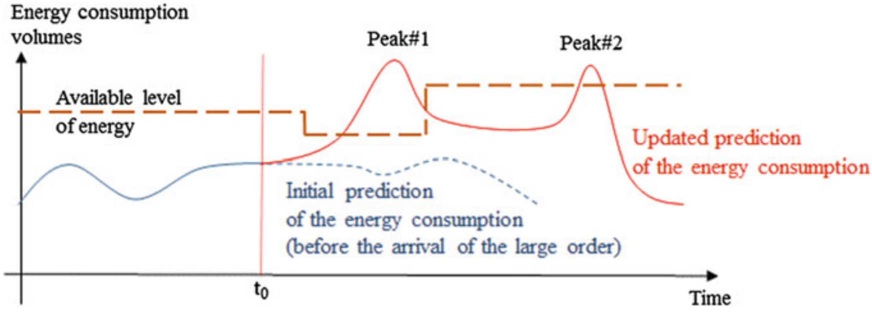


Fig. 2 Graph of energy plan coordination

6. The smart factory scheduler tries to short-term reconfigure (REQ#O2) and to rebuild a new production schedule avoiding these peaks, for example, by smoothing production, scheduling factory workers on a night shift, plugging in new production resources (facilitated using the holonic paradigm) or outsourcing part of the production (REQ#O2).
7. When finding a solution through iterative negotiation, both schedulers sign an updated agreement, which reflects the new balance of supply and demand.
8. If necessary, and as an illustration of the reconfiguration process (REQ#O2), the top-level regional MAS for conflict resolution can connect in the negotiation process additional smart energy schedulers of other energy providers to improve the level of generated energy on demand, in particular from alternative energy sources (e.g., solar and wind) for time intervals of favourable weather forecast (REQ#O1).

Obviously, the energy availability and consumption depend on which other smart factories and smart energy providers are active and connected to this network at that time. There may be not only two as in the given example, but much more. As a result, the situation changes dynamically, depending on many factors. Thus, the global levels of available and required energy are dynamic as well. Similarly, negotiation protocols can be built in case of sudden decrease in energy consumption or introduction of new sources of cheap energy, etc. Moreover, this architecture can be expanded not only by increasing the number of providers and consumers, but also recursively by increasing the number of implemented levels using holonic principles of self-similarities, providing the requirements for openness and flexibility, performance, scalability, reliability and viability.

This example architecture illustrates how the introduced key-enabling requirements force designers of MAS/HMS based $EAPS^2$ to have a broader, more global view of the initial energy-aware scheduling design issue as well as help them to open their mind to find new global and original solutions.

Currently the discussed smart factory and smart energy schedulers are under development and are planned to be applied for manufacturing in aviation industry and modelling of regional energy consumption [25, 26].

5 Conclusion

The aim of this paper was to foster researchers to pay attention to key-enabling requirements when designing MAS/HMS based $EAPS^2$ beyond requirements directly relevant to the $EAPS^2$ system itself. These requirements force widening the attention paid by researchers during design in order to ensure the most global view of the implications of the principles of sustainability at different levels of the manufacturing system and their proper treatment by the concrete MAS/HMS based $EAPS^2$. It has been discussed the idea that these key-enabling requirements are not sufficiently addressed by the literature.

An illustrative example of a system complying with some of these requirements was provided to illustrate the potential benefits of such a more global design view. From our work, it seems that some of the introduced requirements remain consistent even if they are not specifically addressed within the context of HMS/MAS and we intend to pursue our work in that direction.

Acknowledgements This research was partially supported by research grant TIN2015-65515-C4-1-R of the Spanish government. This paper was also prepared with the financial support of the Ministry of Education and Science of the Russian Federation - contract № 14.579.21.0084, project unique ID is RFMEFI57914X0084.

References

1. Tonelli, F., Bruzzone, A.A.G., Paolucci, M., Carpanzano, E., Nicolò, G., Giret, A., Salido, M. A., Trentesaux, D.: Assessment of mathematical programming and agent-based modelling for off-line scheduling: application to energy aware manufacturing. CIRP Ann. Manuf. Technol. **65**, 405–408 (2016)
2. Trentesaux, D., Borangiu, T., Thomas, A.: Emerging ICT concepts for smart, safe and sustainable industrial systems. Comput. Ind. **81**, 1–10 (2016)
3. May, G., Barletta, I., Stahl, B., Taisch, M.: Energy management in production: a novel method to develop key performance indicators for improving energy efficiency. Appl. Energy **149**, 46–61 (2015)
4. Bunse, K., Vodicka, M., Schönsleben, P., Brülhart, M., Ernst, F.O.: Integrating energy efficiency performance in production management—gap analysis between industrial needs and scientific literature. J. Clean. Prod. **19**, 667–679 (2011)
5. Gahm, C., Denz, F., Dirr, M., Tuma, A.: Energy-efficient scheduling in manufacturing companies: a review and research framework. Eur. J. Oper. Res. **248**, 744–757 (2016)
6. Karnouskos, S., Colombo, A.W., Lastra, J.L.M., Popescu, C.: Towards the energy efficient future factory. In: 7th IEEE International Conference on Industrial Informatics June 23 INDIN 2009 (2009)
7. Prabhu, V.V., Trentesaux, D., Taisch, M.: Energy-aware manufacturing operations. Int. J. Prod. Res. **53**, 6994–7004 (2015)
8. Verstraete, P., Saint Germain, B., Valckenaers, P., Van Brussel, H., Belle, J., Hadeli, H.: Engineering manufacturing control systems using PROSA and delegate MAS. Int. J. Agent-Oriented Softw. Eng. **2:**62–89 (2008)
9. Kuhn, D., Ellis, K., Fouchal, F.: ICT enabled energy efficiency in manufacturing. In: Seliger, G. (ed.) Sustainable Manufacturing, pp. 157–162. Springer, Berlin Heidelberg (2012)
10. Trentesaux, D., Giret, A.: Go-green manufacturing holons: a step towards sustainable manufacturing operations control. Manuf. Lett. **5**, 29–33 (2015)
11. Främling, K., Holmström, J., Loukkola, J., Nyman, J., Kaustell, A.: Sustainable PLM through intelligent products. Eng. Appl. Artif. Intell. **26**, 789–799 (2013)
12. Delgado-Gomes, V., Oliveira-Lima, J.A., Martins, J.F.: Energy consumption awareness in manufacturing and production systems. Int. J. Comput. Integr. Manuf. **30**, 84–95 (2017)
13. Lee, S., Prabhu, V.V.: Energy-aware feedback control for production scheduling and capacity control. Int. J. Prod. Res. **53**, 7158–7170 (2015)
14. Wilson, J., Arokiam, A., Belaidi, H., Ladbrook, J.: A simple energy usage toolkit from manufacturing simulation data. J. Clean. Prod. **122**, 266–276 (2016)
15. Raileanu, S., Anton, F., Iatan, A., Borangiu, T., Anton, S., Morariu, O.: Resource scheduling based on energy consumption for sustainable manufacturing. J. Intell. Manuf. 1–12 (2015)

16. Taticchi, P., Garengo, P., Nudurupati, S.S., Tonelli, F., Pasqualino, R.: A review of decision-support tools and performance measurement and sustainable supply chain management. Int. J. Prod. Res. **53**, 6473–6494 (2015)
17. Aoki, K., Staeblein, T., Tomino, T.: Monozukuri capability to address product variety: a comparison between Japanese and German automotive makers. Int. J. Prod. Econ. Part B **147:**373–384 (2014)
18. Paolucci, M., Anghinolfi, D., Tonelli, F.: Facing energy-aware scheduling: a multi-objective extension of a scheduling support system for improving energy efficiency in a moulding industry. Soft Comput. 1–12 (2015)
19. Bruzzone, A.A.G., Anghinolfi, D., Paolucci, M., Tonelli, F.: Energy-aware scheduling for improving manufacturing process sustainability: a mathematical model for flexible flow shops. CIRP Ann. Manuf. Technol. **61**, 459–462 (2012)
20. Leitão, P., Barbosa, J., Trentesaux, D.: Bio-inspired multi-agent systems for reconfigurable manufacturing systems. Eng. Appl. Artif. Intell. **25**, 934–944 (2012)
21. Giret, A., Trentesaux, D., Prabhu, V.: Sustainability in manufacturing operations scheduling: a state of the art review. J. Manuf. Syst. Part 1 **37**, 126–140 (2015)
22. ISO 14040:2006—Environmental management—Life cycle assessment—Principles and framework. http://www.iso.org/iso/catalogue_detail?csnumber=37456
23. ISO 14044:2006—Environmental management-Life cycle assessment—Requirements and guidelines. http://www.iso.org/iso/catalogue_detail?csnumber=38498
24. Rzevski, G., Skobelev, P.: Managing Complexity. WIT Press (2014)
25. Leitão, P., Barbosa, J., Vrba, P., Skobelev, P., Tsarev, A., Kazanskaia, D.: Multi-agent system approach for the strategic planning in ramp-up production of small lots. In: 2013 IEEE SMC Conference, pp. 4743–4748 (2013)
26. Tsarev, A., Skobelev, P.: Multi-agent Supply Scheduling System Prototype for Energy Production and Distribution. Advances in Practical Applications of Scalable Multi-agent Systems, The PAAMS Collection, pp. 290–293. Springer International Publishing (2016)

Multi-agent Framework for Manufacturing Sustainability Analysis and Optimization

Flavio Tonelli, Massimo Paolucci, Melissa Demartini and Davide Anghinolfi

Abstract Manufacturing sustainability is related to complex adaptive systems crossing different layers; sustainability issues, and their potential resolutions, are transversal to supply and shop-floor levels, often including interaction between them; in addition, multiple actors across the supply chain interact displaying non-linear and non-rational behaviours characterized by feedbacks and time lags. This complexity is nowadays addressed with classical simulation approaches such as System Dynamics (SD), Dynamic Systems (DS) at very high and aggregate level, or Discrete Event Simulation (DES) at very low and disaggregated level. Agent Based Modelling (ABM) approach may potentially address such issues in a more effective way because it exploits the strengths of both aggregated and disaggregated models, while minimizing the drawbacks. The purpose of this paper is to present a self-developed multi-agent based framework as a basis for sustainable manufacturing modelling, simulation and analysis. A literature review on agent based modelling applied to sustainable manufacturing issues is reported to justify choices adopted in the developed tool. Then an illustrative case of a semi-automated food production line is described with the aim of evaluating the self-developed framework through some sustainability and productivity KPIs, and showing easiness and adaptability. The complete agent based simulator and the interaction sequence are reported and validated by an experimental campaign.

F. Tonelli (✉) · M. Demartini
Department of Mechanical Engineering Energetics Management and Transportation (DIME), University of Genoa, Via all'Opera Pia 15, 16145 Genoa, Italy
e-mail: flavio.tonelli@unige.it

M. Paolucci
Department of Informatics Bioengineering Robotics and System Engineering, University of Genoa, Via all'Opera Pia 15, 16145 Genoa, Italy
e-mail: paolucci@dist.unige.it

D. Anghinolfi
Novigo Technology s.r.l. and Genoa University, Genoa, Italy
e-mail: anghinolfi@novigotechnology.com

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_13

Keywords Manufacturing sustainability • Complex adaptive systems • Agent based modelling • Industrial case

1 Introduction

The modern global industrial system, even if delivering major benefits in wealth creation, technological advancement and enhanced well-being, is estimated to be responsible for about 30% of the greenhouse gases and is a major consumer of primary resources [1]. Although some progress towards sustainability has been achieved at micro-level (eco-efficiency, cleaner production, recycling initiatives, and extended producer responsibility directives), overall industrial sustainability and its macro- and micro level interactions have not yet improved [2], also because of scarcity of integrated tools capable to catch this complexity. Modelling and simulation can play an important role toward this integration but existing tool still are too much sophisticated, being developed mainly for micro level studies and analysis. Similarly, despite impressive improvements in material productivity and energy efficiency in many industry sectors, overall energy consumption and material throughput continue to rise. It is apparent that to deliver long-term goals on this scale requires fundamental changes in the global industrial system, and this necessitates an integrated approach that goes well beyond current efficiency and productivity initiatives where modelling and simulation can play an important role.

The purpose of the paper is to describe the features of a self-developed agent based simulation framework for manufacturing, including special features for sustainability issues. In Sect. 2 a literature review on Sustainable Agent Based Modelling is presented and discussed. Section 3 proposes the self-developed agent based simulation framework, and Sect. 4 describes an illustrative case study of a semi-automated food production line. Finally, conclusions and future development are presented in the last section.

2 Literature Review on Agent Based Modelling Application to Sustainable Manufacturing

The literature review was performed by analyzing the Scopus dataset, in particular searching the publications appeared between 2000 and 2015 that include "Sustainable Agent Based Modelling" or "Sustainability Agent Based Modelling" in the title, abstract and keywords. The result was a set of 234 papers, the earliest of which published in 2001 and the most recent in 2015.

The distribution of publications over years displays an interesting sharp growing trend of articles on this subject over the past decade. The trend line illustrates the average number of articles published until the previous year. It reveals a significant increase from 2012. In terms of geographical distribution, the publications contributing to this research area originate from many different countries around the world, based on the main author's university affiliation, with a predominant position of USA.

The most relevant authors are Leitão P., Trentesaux D., Giret A., whereas the list of the first journals includes Journal of Cleaner Production, International Journal of Production Economics, Journal of Manufacturing Systems, International Journal of Production Research, Renewable and Sustainable Energy Reviews.

From a qualitative review of the collected paper, Agent Based Modelling (ABM) approaches are particularly interesting because of their characteristics: (i) the heterogeneity of the agents, (ii) their adaptive behaviour and (iii) the generative approach they follow. Agents have (i) a bounded rationality that is subject to stress, time pressure and emotive forces, (ii) a memory and a set of properties defined by the modeller that helps them to implement different types of behaviour, and (iii) they can learn from their mistakes and they can adapt and react to different scenarios (i.e. a different set of properties or a modified environment). In this way, macro-behaviour is not simulated but emerges from the micro-decisions of individual agents. This flexibility makes ABMs a powerful interdisciplinary tool and thanks to these characteristics, these models are potentially able to represent any type of situation or system encountered in Industrial Sustainability (IS) domain. The key advantage offered in the specific IS domain, with respect to other possible simulative approaches, is the ability to take into account heterogeneity and behavioural interactions, which can lead to emergent behaviour that would not be obvious or might be very difficult to foresee in an aggregate model as it could occur in the current manufacturing networks. Agent-Based (AB) models in the industrial sustainability field are emerging and various authors have identified the potential value and effectiveness and advocated such simulation approaches [3–5]. The findings of this study established the potential and capability of ABM for investigating decision options for optimal eco-industrial systems in both 'open-loop' and 'closed-loop' structures/systems. More recently, [6] explore the use of ABM for eco-innovation policy assessment through the example of Extender Producer Responsibility (EPR). [7] assessed bio-inspired multi-agent systems for reconfigurable manufacturing systems. [8] assessed and compared mathematical programming and ABM for off-line scheduling with an application to energy aware manufacturing systems. The analysed papers, qualitatively reviewed, prove that Agent-Based (AB) models are a very promising approach for including the sustainability dimension [9]) in the industrial context studies, and in order to improve industrial sustainability. AB models can be considered, in this context, a new generation decision-making tool [10], capable to integrate sustainability at macro- and micro- levels including sustainable supply chains and sustainable manufacturing areas. Various applications towards manufacturing sustainability, based on AB, can be found in recent literature [8, 11, 12]. In summary, ABM is an interesting emerging field that seems to have significant potential for exploring the complex issues of sustainable industrial systems. The ABM approach also presents some limits even if advantages outweigh drawbacks; for a general evaluation of these pros and cons see [13]. In particular, it is complex to manage and properly model the heterogeneity of agents and their emergent properties without having a proper tool [14–17]. Commercial tools are not specifically designed to address industrial sustainability or sustainable manufacturing problems (see, for example AnyLogic or NetLogo). Nevertheless, the researcher will face a trade-off between complete information coverage, and need to manage such complexity, leaving out from the analysis several factors. The reported evidences confirm the appropriateness of the ABM approach and suggest the importance to identify or develop Multi Agent Simulation (MAS) framework for manufacturing sustainability as the one proposed in this paper.

3 MagisQuam: A Multi-agent Simulation Framework for Manufacturing Sustainability

This section introduces the main features of a new ABM framework developed through the collaboration of academic research and industry by IROI s.r.l., a spinoff of the University of Genova (Italy). MagisQuam (MQ) Simulator has been designed to be a highly versatile tool that can be adopted to analyse through simulation systems and processes belonging to different types of industrial contexts, in particular to effectively support in understanding complex systems dynamics in a very intuitive and friendly way. MQ, in fact, is based on a visual design environment through which the modellers can define system models in a rather simple way, starting from either a set of base building blocks or exploiting the available vertical industry block libraries. Sustainability issues have been taken into account in such libraries through the implementation of specific behaviours and key performance indicators (KPIs).

A MQ model can be defined as a classic Multi-agent System (MAS) that includes the following main elements:

- *Entities*: they represent the overall classes of agents. These entities are characterized by:
  - Attributes, defining the properties of the entity;
  - Behaviours, specifying for each entity the set of the possible states and the transition procedures among them;
  - Message traps, specifying the behaviour of a class of agents as a consequence of the reception of a message from another agent class.
- *Agents*: they represent the instances of the entities and are characterized by specific values for their attributes.
- *Messages*: they are used to enable the communication among the agents during the simulation and are characterized by their field structure and subject.
- *Behaviours*: associated with each entity/agent state there is a behaviour that is defined as a workflow.

In addition, in order to represent the state evolution of each agent, local and global variables are defined. The local variables are used to allow each agent to represent its state as well as to provide memory for the actions performed and their outcome. On the other hand, the global variables represent common dynamic features of the environment where the agents operate and are available to all the agents needing them, and are modified as a consequence of the agents' actions. Beside these general purpose variables, MagisQuam Simulator permits to introduce a set of performance indicators (KPI) that consist in specialized global variables allowing evaluating the performance of the considered process.

The agent-based simulator corresponds to a synchronous discrete-time model: hence, the simulation evolves as a sequence of simulation time steps: at each step, for each agent, the behaviour associated to the state in which the agent is at the beginning of the step is performed. Beside these general purpose variables, MQ permits to introduce a set of performance indicators that consist in specialized global variable that allow evaluating the performance of the considered process; a set of specific sustainability KPIs has been implemented into the framework.

Typical complex systems behaviour such as the ones addressed in industrial sustainability domain can be obtained by focusing on the actual operating entities and letting the overall picture to emerge from their interaction. Entity/agent view allows also to represent the desired model level, high aggregate or detailed disaggregate, as well as to make different levels to interact; the user can then impose the desired organization structure (e.g., hierarchy, coalition, federation). How to model such aspects depends on the specific context under concern, the desired level of detail and complexity; for example, the generation of waste can be represented as emerging from the interaction of the behaviours of different agents operating on the same material, or by associating as an active behaviour the material itself, i.e., modelling pharmaceutical production lots as agents in charge of monitoring their state. In the following section a simple illustrative case is introduced to show how the product waste due to different production schedules can be easily analysed.

Regarding sustainability in manufacturing processes, MQ models appear an effective option since they can provide specialized agents, behaviour and KPI to account for resource waste, energy consumption, emission, environmental or economical dangerous processes.

4 An Illustrative Example

4.1 Description

The manufacturing system proposed in this paper takes inspiration from a real plant producing food for animals as depicted in Fig. 1 in order to test and validate the MQ framework and simulation tool in a semi-real industrial instance with hybrid manufacturing behavior (continuous and discrete processing steps, deterministic

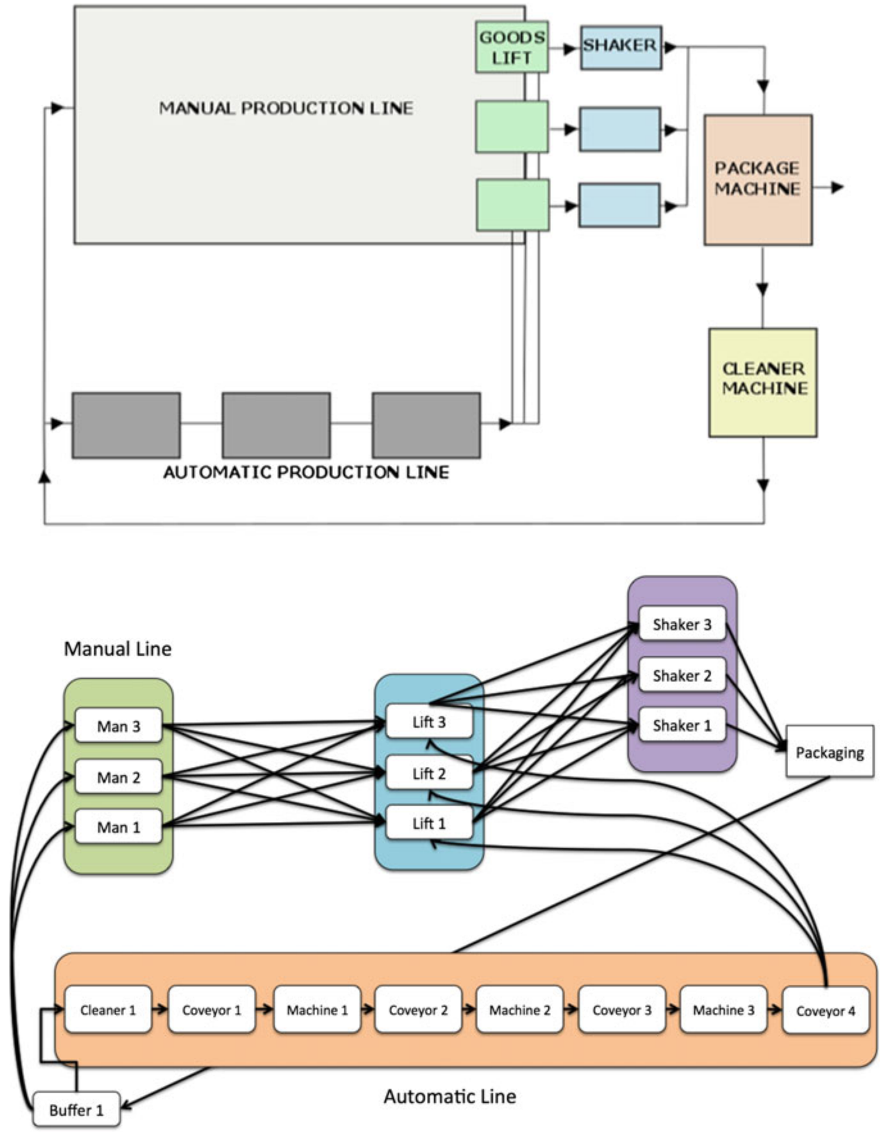


Fig. 1 The process scheme of the illustrative example

and stochastic variables, and automatic/human interaction with interrelated constraints). The plant is characterized by two convergent production lines, one fully automated, whereas the other requiring manual working phases.

Due to these different types of processing, the manual and the automatic lines operate with different manufacturing times, in particular, the automatic line showing almost constant processing times, while the manual high stochastic ones that may depend on the specific operator skill as well on the team composition and shift. The two lines operate in independent manner producing separately two parts of the same recipe for a specific food type. The plant in fact produces several recipes through the processing and the combination of different raw materials. The manufacturing of each of these recipes is performed in a distinct container (bin) that is moved in the automatic line where at different stations the main ingredients are added. A similar processing is performed in the manual line where very small quantities of critical ingredients are prepared for each recipe. In general, the processing times required by a recipe in the two lines are different. At the end of each line the two semi-finished products (the bin and the critical ingredients) for each recipe are combined: as the plant is organized on two floors, the semi-finished products from the automatic line on the ground floor are sent by means of some freight elevators to the first floor where human operators work on the manual line. Then the two recipe components are joined under the operators control and mixed in a shaker. Finally, the finished products are forwarded to the packaging machinery where they are packed and then delivered to their destinations.

The manufacturing process ends with the washing of the bins used during the realization of the recipes.

The main typical issue that production planning must take into account is the demand satisfaction, in particular, producing with a make-to-stock policy according to the demand forecast. Usually, the definition of the detailed bin schedule was not considered a critical aspect as production capacities are sufficient to satisfy the demand. However, the production process has a certain criticality as for several recipes the overall production time should not exceed a given time limit, since in this case the product deteriorates and the whole bin is discarded with a waste of raw material and capacity. Such aspect was not explicitly considered by plant planners, who simply increase the demand level for the involved recipes in order to account for wastes.

However, it is apparent that one of the most critical aspects for reducing the wastes for this production process is the synchronization of the activities on the two lines. Therefore, a more effective way to determine the detailed production schedules has been devised, which is based on two interacting steps. In the first step a recipe sequence is generated by some heuristics, and in the second step such sequence is evaluated by a MQ simulation in order to take into account the possible delays and synchronization problems due to the highly stochastic behaviour of the manual line. The example in this paper focuses on the MAS simulation model at the second step.

The above introduced animal food manufacturing process was modelled with MQ Simulator, with the aim of simulating several possible schedules of recipes in order to jointly improve the overall productivity of the plant and to reduce the processing waste. This case study then represents a typical combined manufacturing and sustainability problem not so easy to be implemented and solved with manufacturing specific DES tools.

The following entities were introduced in the model, whose behaviours were represented by means of workflows associated with the entities' states:

- **Bin**: this entity models the receptacles used during the production phases; it has attributes to take into account the type of recipe and the time at which the different production phases start.
- **Buffer**: it was introduced to represent the supervisor of the production process; specifically, the buffer represents the point of departure and arrival of the bins, reading the scheduling of the recipes, associating each bin with a recipe, providing the elements of the production chain with the information about the association recipe-bin and the quantity of the ingredients required for each recipe.
- **Filler Machine**: the entity filler machine was introduced to model the automatic production line. The agents derived from this entity represent the process of filling the bins associated with the recipes: in particular, each ingredient is associated to a filler agent and filling is modelled as a delay. The dwell time of the container under the filler machine depends on the quantity of ingredient required by the recipe. The filler machine agents can communicate with each other with the buffer agent, through messages.
- **Conveyor Belt**: the entity conveyor belt models the transportation of bins within the automatic line. These agents allow the transport of one bin at a time between a pair of filling machines and are represented as a system delay. The conveyor agents are able to communicate with the filler agents, with the buffer agents and with the elevator agents. Note that the automatic production line is then composed by a sequence of filling machine and conveyor agents and it operates in sequence following a "first-in-first-out" policy.
- **Operator**: the operator is the entity which identifies the workers composing the manual production line. Differently from the automatic line, such agents work in parallel on different bins. Also for the manual line, the processing of the recipe is modelled as a delay, but, in order to take into account the uncertainty characterizing the manual work stages, stochastic production times have been considered for the realization of the different recipes by different workers.
- **Good Lift**: the role of this entity is to connect the two production lines. Specifically, whenever the automatic line finishes to process parts of a recipe, it puts the bin in the first lift that is idle; whenever the manual production line finishes the production of its part of a recipe, it checks whether in one of the elevators the other part of the same recipe is already present and, in the positive case, the two parts are merged and sent to mixing.
- **Shaker**: this entity models the mixing machines. The agents derived from this entity are placed downstream of each elevator and their processing is represented as a delay.
- **Packager**: this entity is associated with the packaging machine in which the final products are packed. The packaging machine is also responsible for checking the quality of the product: specifically, it is verified if the production process for the critical recipes lasted too long and, whenever such situation happens the recipe in the bin is marked as deteriorated and the product is discarded.

- **Cleaner**: is the entity devoted to the cleaning of the bins. Before starting a new production cycle, the empty bins are controlled: if the current recipe of a bin is the same as the recipe that is assigned next by the buffer, the bin is re-introduced in the production cycle; otherwise the bin is washed by means of specific machinery and then is introduced again into production process.

All the listed and described agents have been created without writing code, directly by the user using the MQ framework.

Figure 2 depicts the interaction sequence among the agents. The list of exchanged messages and their sequence are overviewed in this figure. The vertical line (from each agent) in the diagram represents time and the execution thread of each agent. An arrow between two agents represents a message passing from one agent to the other (i.e. the sending agent is requesting the execution of a given function from the receiving agent, or the sending agent is informing the receiving agent a given data is ready).

The solving sequence is initiated by the master agent (*Buffer*) which executes the function *Item request* in order to fulfil a production order. Then, the request is added to the list of orders, waiting to be processed (*Add to input list*).

If the *Cleaner* is available the *Item* can be send from the *Buffer* to the *Cleaner* (*Item send*); by contrast if the *Cleaner* is occupied, the *Item* isn't sent and it queues up at the *Buffer* (*Item Failed*). In both cases there is an update of the component state (*Update Component*). Then considering the type of production which the *Item* needs to be done (manual or automated), the *Cleaner* sends it to a *Man* in case of a manual process or to a *Conveyor* for an automated production.

Analysing more precisely the automated process, it is evident that it has more constrains with regard to the manual line. In fact the automated line is composed by

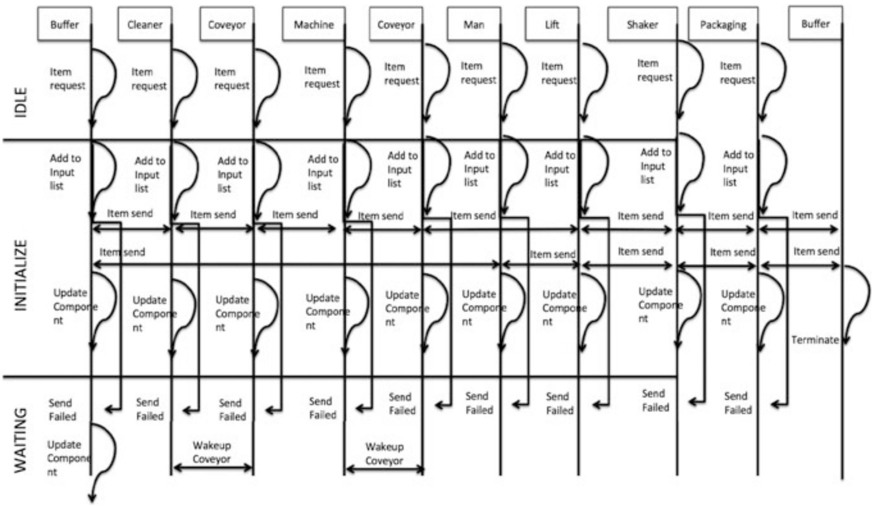


Fig. 2 Agents' interaction sequence and messages

three different processing techniques in series, and for each machine there is a check condition that confirms the availability of the machine itself. Differently, the manual line is operated by three men in parallel. It is important to observe that in this model the automatic production line is composed by a sequence of three machine and conveyor agents; in order to simplify the agents' interaction sequence and because of the logical process of the machines is the same only one machine is considered in the scheme.

After the manual or automated production stages, *Items* are dispatched to the Lift Station, waiting for the Shaker process; also in this case there is a check that verifies the availability of the station. The role of this agent is to connect the two *Items*.

Subsequently to the *Shaker* process in which *Items* are mixed based upon the recipe, *Items* are prepared to be packaged. The process ends with the *Packaging* procedure, and empty bins are sent to the *Buffer* (*Terminate*).

4.2 Simulation and Results

In this section the results obtained from the simulation runs performed for a set of schedules (recipe sequences) are reported and discussed. For the case study here analysed, the automatic production line is composed by three Filler Machines and four Conveyor Belts, whereas for the manual line three agents of type Operator are considered. Only one agent derives from the entities Buffer, Packager and Cleaner, while three agents are instantiated from the entities Good Lift and Shaker (see Fig. 2).

Moreover, in this study, some scenarios were tested in which three different recipes and five possible schedules were considered. For all the simulation runs, thirty recipes were scheduled and the simulation time step was fixed equal to 60 s. In order to take into account the deterioration of the recipes produced in the plant, for each of these a maximum time of permanence in the process was imposed. In particular, for recipe 1 the maximum production time was set equal to 300 s, for recipe 2 to 350 s and for recipe 3 to 450 s. In addition, in order to show the goodness of the proposed approach some KPIs were considered and compared which correspond to: the Average Processing Time (APT) for recipe type expressed in seconds, the Rate of Discarded Bins (RDB) for recipe, i.e., the ratio between the bin associated with a successful production and the total number of bins for a given recipe, and the Average Exceeded Time (AET) for recipe, expressed in seconds, with indicates at which extent in average the processing time of a recipe exceeded the maximum allowed one.

These KPIs are shown in Tables 1, 2, 3, 4 and 5, referring to the single schedule and in Table 6 where the results aggregated over all the recipes are given. In particular, from the KPIs in Table 6, it is possible to observe that the best results were obtained for the schedules 4 and 5. Furthermore, the KPIs in Table 4 highlight

Table 1 KPIs obtained for the schedule 1

	Recipe 1	Recipe 2	Recipe 3
APT (s)	298	320	315
RDB (%)	30	40	50
AET (s)	82	117	62

Table 2 KPIs obtained for the schedule 2

	Recipe 1	Recipe 2	Recipe 3
APT (s)	251	304	330
RDB (%)	60	50	30
AET (s)	86	99	76

Table 3 KPIs obtained for the schedule 3

	Recipe 1	Recipe 2	Recipe 3
APT (s)	318	349	269
RDB (%)	60	30	20
AET (s)	77	65	98

Table 4 KPIs obtained for the schedule 4

	Recipe 1	Recipe 2	Recipe 3
APT (s)	310	342	294
RDB (%)	20	30	30
AET (s)	42	47	18

Table 5 KPIs obtained for the schedule 5

	Recipe 1	Recipe 2	Recipe 3
APT (s)	261	329	294
RDB (%)	30	50	10
AET (s)	76	70	88

Table 6 Global KPIs for all the schedules considered

Schedule	Total process time (s)	Total RDB (%)
1	11148	
2	10969	47
3	10465	37
4	10211	27
6	9879	30

that adopting the schedule 4 it is possible to achieve a more balanced solution, since the percentage of waste for each recipe is rather equivalent. Finally, Table 6 also shows that lower total process times correspond to lower rate of discarded products.

5 Conclusion

The aim of this paper is to outline the potential of ABM and MAS in industrial and manufacturing sustainability analysis, design, and evaluation, focusing on a not trivial industrial case featuring typical complexity pattern of sustainable manufacturing domain. Since one of the major issues in approaching manufacturing sustainability is the modelling phase, a specific ABM platform, MagisQuam Simulator, has been presented at its preliminary development stage. The simple but significant application to a specific real industrial case shows the greater easiness in developing such a kind of models, including sustainability decision (specifically material productivity). This has to be considered as a first step in developing a more complete application framework for this domain; the positive results obtained are encouraging and confirm the assumptions and concepts presented in the first part of the paper.

Acknowledgements This research was partially supported by research grant National Technological Cluster on Intelligent Factory, MIUR 2012 and IROI s.r.l. spin-off University of Genoa, Italy.

References

1. Evans, S. et al.: Towards a Sustainable Industrial System (2009). Accessed http://www.ifm.eng.cam.ac.uk/sis/
2. Tesfatsion, L.: Agent-based computational economics: growing economies from the bottom up. Artif. Life **8**(1), 55–82 (2002)
3. Bousquet, F., Le Page, C.: Multi-agent simulations and ecosystem management: a review. Ecol. Model. **176**(3–4), 313–332 (2004)
4. Heckbert, S., Baynes, T., Reeson, A.: Agent-based modeling in ecological economics. Ann. N.Y. Acad. Sci. **1185**, 39–53 (2010)
5. Kraines, S., Wallace, D.: Appl. Agent-based Ecol. **10**(1), 15–18 (2006)
6. Arfaoui, N., Brouillat, E., Saint-Jean, M.: Policy Design, Eco-Innovation and Industrial Dynamics in an Agent-Based Model : An illustration with the REACH Regulation, GREDEG WP No. 2013–22 (2013). Accesssed http://www.gredeg.cnrs.fr/working-papers.html
7. Leitão, P., Barbosa, J., Trentesaux, D.: Bio-inspired multi-agent systems for reconfigurable manufacturing systems. Eng. Appl. Artif. Intell. **25**(5), 934–944 (2012)
8. Tonelli, F., Evans, S., Taticchi, P.: Industrial sustainability: challenges, perspectives, actions. Int. J. Bus. Innov. Res. **7**(2), 143–163 (2013)
9. Monostori, L., Váncza, J., Kumara, S.R.T.: Agent-based systems for manufacturing. CIRP Ann. Manuf. Technol. **55**(2), 697–720 (2006)
10. Pahl-Wostl, C.: Towards sustainability in the water sector—the importance of human actors and processes of social learning. Aquat. Sci. **64**(4), 394–411 (2002)
11. Borangiu, T., Thomas, A., Trentesaux, D.: Service Orientation in Holonic and Multi-agent Manufacturing. Springer Publishing Company, Inc (2015)
12. Trentesaux, D., Giret, A.: Go-green manufacturing holons: a step towards sustainable manufacturing operations control. Manuf. Lett. **5**, 29–33 (2015)

13. Taticchi, P., Garengo, P., Nudurupati, S.S., Tonelli, F., Pasqualino, R.: A review of decision-support tools and performance measurement and sustainable supply chain management. Int. J. Prod. Res. **53**(21), 6473–6494 (2015)
14. Bonabeau, E.: Agent-based modeling: Methods and techniques for simulating human systems. In: Proceedings of the National Academy of Sciences, **99** (supl. 3), pp. 7280–7287 (2002)
15. Giret, A., Trentesaux, D.: Artefacts and guidelines for designing sustainable manufacturing systems. Stud. Comput. Intell. **640**, 93–101 (2016)
16. Tonelli, F., Bruzzone, A.A.G., Paolucci, M., Carpanzano, E., Nicolò, G., Giret, A., Salido, M. A., Trentesaux, D.: Assessment of mathematical programming and agent-based modelling for off-line scheduling: application to energy aware manufacturing. CIRP Ann. Manuf. Technol. (2016)
17. Trentesaux, D., Borangiu, T., Thomas, A.: Emerging ICT concepts for smart, safe and sustainable industrial systems. Comput. Ind. **81**, 1–10 (2016)

Data Mining of Energy Consumption in Manufacturing Environment

Andre Dionisio Rocha, João Aires Tapadinhas, Luis Flores and Jose Barata Oliveira

Abstract Markets and consequently manufacturing companies are facing an unprecedented challenge. The constant markets demand of more and more customized and personalized products combined with the recent evolution of information technologies, brought to the manufacturing world the integration of new solutions previously unimaginable in a production environment. Hence, in the last years manufacturing systems were changing and nowadays each component present in the shop floor generates a huge amount of data that is usually not used. In this paper the authors present a framework capable to deal with all this data generated from a production cell in the automotive industry and reduce the energy consumption. Firstly, it is described how the information is extracted and how the data clustering is done, then the data mining process and management are presented, together with the obtained results.

Keywords Data mining · Manufacturing · Energy consumption

A.D. Rocha (✉) · J.A. Tapadinhas · J. Barata Oliveira
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa,
Caparica, Portugal
e-mail: andre.rocha@uninova.pt

J.A. Tapadinhas
e-mail: j.tapadinhas@campus.fct.unl.pt

J. Barata Oliveira
e-mail: jab@uninova.pt

A.D. Rocha · J. Barata Oliveira
Centre for Technology and Systems, UNINOVA Institute, Caparica, Portugal

L. Flores
Introsys, SA, Moita, Portugal
e-mail: luis.flores@introsys.eu

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_14

1 Introduction

Nowadays the companies are being forced to make a huge paradigm shift from the old manufacturing paradigm with centralized approaches and without sharing any kind of information retrieved from the components that constitute the shop floor to a fully integrated solution where all the resources, workstations, people, other devices present on the production structure should share their own information and feedback to higher level systems. The concept of Industry 4.0 introduces this framework of fully distributed and interconnected world for the manufacturing companies. With the recent evolution of the IT solutions, such as Cloud Computing, Internet of Things, and more powerful processing technologies and devices, more data is being extracted from the production environments and consequently more data is available to be handled and processed. This huge amount of available data can be used to understand the behaviour of the systems in order to increase the volume of useful data and in this sense increase the overall performance of the industrial systems [1].

This study is focusing on data mining workflow value. As an input, there are data from the robot operation work cycle, such as tasks duration, speed and consumption. The process of data analyzing creates more valuable information about the system. By analyzing data, it is possible to understand some details about the robots work cycles—which was not possible before, and some margins for improvement come into perspective. This case study verifies the improvement possibilities for the robot's work cycle, concerning the restrictions established before.

2 Related Work

Big Data has today a strong presence in the work of technology; it has been considered one of the most valuables results of technologies of the 21st century. It can be characterized by its large amount of data, the speed at which data is produced, the reliability and variety of data and most important its value. Nowadays every type of technology such as social media, smartphones and internet websites, sensors and machinery produces daily millions of information about its way of function, consumption patterns of and utilities [2].

A great volume of information generated by many sources brings with it the necessity to analyse this raw data, in search of new valuable information. This is where data mining comes to action. Data mining is an iterative process defined by the discovery of new information, in manual or automatic processes. It can be highly valuable in a context where the standard of results of interest is not known, or discovering new valuable potentials from the data gathered occurs [3].

It can also be considered as improvement of information. This viewpoint is present in many areas of industry such as the: automotive industry, health care industry, business and financial sector, pharmaceutical and biotechnology analyses, manufacturing and machinery, all of them visible in scientific research [4].

Today there are many entities focused on energy saving studies. It is important for any factory or enterprise to save costs coming from energy consumption. Data analysis is one of the most important fields when it comes to smart factories [5], and because this trend is still growing, there are no standards to apply on data analyses.

This architecture was tested in a factory environment by monitoring its electrical consumption. Besides the energy network that powered the whole factory, there was another network to monitor the energy consumption of the energy network. Having a dedicated network to monitor the energy consumption has provided efficiency and productivity to the architecture. As a distributed architecture, it is possible to have independent devices and sensors all connected in a network working self-sufficiently. The network itself has a central process unit on the web which is responsible for gathering all the data together and processing them in real-time. This procedure uses a solution identifying which costs are relevant when related with the factory routines and workflows [6].

Another study investigates the best way to cut off energy consumption costs within the factory environment [7]. A multi-layer abstraction divides the whole company into sections. The first layer—the machinery level includes the entire assembly line: every machine, robot and conveyor system. The second layer contains the refrigerating system: every fan, liquid refrigerator system, air conditioning and so on. The third layer, a complex one, is responsible with analysing all the data collected on the other layers and providing different consumption profiles. Creating different profiles for energy consumption is logic when the energy consumption is directly related with the execution status of the product. This approach makes possible a clearer idea of the whole factory. Comparing the energy used in similar or different production cycles makes possible to identify which areas of production are more or less efficient [7].

A university in Ireland studied one of its own buildings, namely temperature performance. The sensors used were capable to measure temperature, carbon dioxide in the air, humidity and luminosity. The information collected during 3 years had the size of about twenty million logs, and was related to energy consumption inside the building. Because the sensors needed to be calibrated frequently, some values in the data collection were outside the evaluation range; therefore many entries in the logs were left out on the pre-processing data stage. Once again, this study tried to create different profiles with the indoor temperature. Classifiers to identify *cold*, *moderate*, *comfortable*, or *hot* were created. These categories were used to relate with outside temperature by decision algorithms. This study detected energy consumption patterns and made accurate evaluation of comfort requirements inside the building. It also allowed predicting energy necessities in similar buildings [8].

Another similar study demonstrates the possibility to improve the performance of process execution in a manufacturing environment by using data mining. The scope of this study is to analyse data in order to build a dynamic model of the production system. This requires an advanced information shared network to monitor manufacturing data in real-time, to understand and prevent errors and improve the system's performance. The system was built with three different layers of abstraction, (1) manufacturing environment's sensibility configuration, (2) acquisition and management of data from the shop floor machinery, (3) data analysis and improvement of a dynamic Bayesian network to represent the production system as a model [9].

There is a trend in smart factories solutions to perform integrated network frameworks based on with multi-agent systems [10]. This type of architecture aims at controlling and coordinating factory components to work and communicate autonomously. Besides that, it uses the multi-agent architecture to collect and analyse data from the factory environment components.

3 System Architecture

The following system architecture was developed to study the energy consumption of a robotic arm during its operations. Inside the shop floor environment it is planned to monitor every task of the robot and the energy costs associated. The main goal is to find a more efficient way to make the robot perform the same operations.

However, there are some restrictions that make this task challenging. It is very expensive for the factory to put the machinery to stop. The whole factory has every detail programmed to make a product on time, and every second counts. This leads to a second problem; it adds also many costs to reconfigure the machinery to make it more efficient. However, it is not economic to stop and reconfigure the machinery. With these restrictions in mind, there are a few variables that are allowed to change, such as the robot's subtasks without changing the timing of the main tasks.

The monitoring system in this project is used in two different stages: the data acquisition from the robot and the data-mining of the acquired information.

3.1 Data Acquisition

The data acquisition process starts in the robot cell. The computer that is connected directly to the robot's PLC (programmable logic controller) reads variables such as: date and time of the entry log, function mode manual or automatic, ready state for operating, step to execute, error and error type, cycle duration in seconds, position inside the cycle, main tasks to execute, executing speed, and power consumption of every motor of the robot arm. The computer is responsible for buffering the data acquired from the robot and sending it in real time to an online data base.

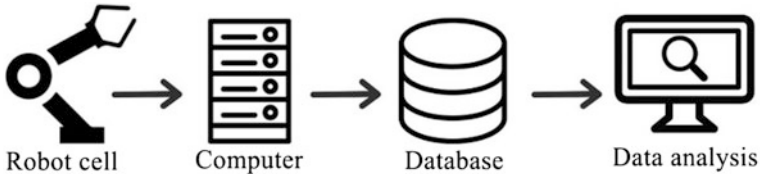


Fig. 1 Data workflow of the system from the robot cell to the finally data analysis station

3.2 Data Base

A data based stored online in the factory network collects all the data read from the robot cell. Its capacity is enough to store a large number of logs of monitoring the robot's power consumption. The data base provides all the information related to the robot operation cycles, the energy consumption and the operation modes to another entity responsible for analysing this big data (Fig. 1).

3.3 Data Processing

In this section, the data analysis will be focused on the energy consumption during the robot arm's operations. The robot operates in different modes with different speeds, the data collected being processed accordingly.

The main goal of this process is to match different subtask speeds in order to find the perfect combination of consumptions in tasks. In this way we can find a complete work cycle with standard duration. It is possible to have different speeds of subtasks operations with the aim of getting the most efficient combination possible.

4 Use Case

4.1 Data Processing Framework

For this process a software framework named Splunk is used, capable to connect to the database and powerful enough to analyse patterns and big amounts of values throughout the database. This software is used also in other areas such as data mining in networks, business and finance. With Splunk it is possible to apply mathematic functions and relate variables in time; it is also very flexible to the point that there is no need to reconfigure the software when something changes. It is easier to calculate consumptions accordingly with the work cycles associated, to analyse the data and find solutions [11].

Fig. 2 Splunk platform dashboard with consumed energy and its work cycle

Figure 2 illustrates the consumed energy throughout four work cycles and its search query. As shown in Fig. 2, the Splunk main search bar reads and consumes queries with different math operations and variable relations, and further, it can save many different results as charts and combine in an info graphic informative panel. Although it can perform those queries in different time ranges, the last 24 h or real-time patterns are of more interest.

4.2 *Demonstration*

The resource that the study relies upon is a Kuka robot arm that operates in a six phase work cycle.

The data available for the robot arm has a particular format. It is always known in which task the arm is operating. Looking at Fig. 3, the chart shows the consumption of the motor number one, and also indicators of the task that it is performing during four work cycles.

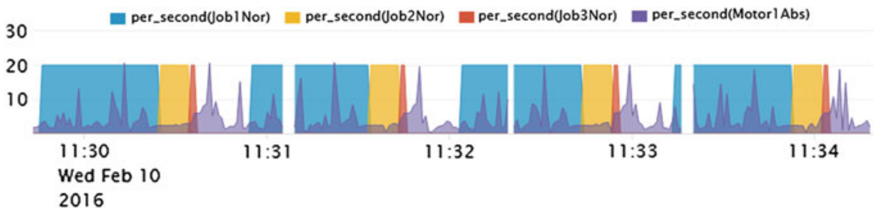


Fig. 3 Data chart with consumption of one motor and tasks indicators in 4 work cycles

One can also notice that in the chart there is missing data. The solution provided for this issue was to insert values in those spots to cover the error of the data acquisition process.

This issue becomes relevant when the consumption of the robot's work cycle is calculated, because some work cycles don't have the full information of their consumption. So the solution performed was to fill up the missing data with the average consumption of the rest of the existing samples.

The following expressions have been used as mathematic formulae:

$$Consumption\ available = \sum_{t_0}^{t_1} Consumption\ samples \tag{1}$$

$$Consumption\ average = \frac{\sum_{t_0}^{t_1} Consumption\ samples}{Number\ of\ samples} \tag{2}$$

$$Consumption\ available = Consumption\ average * Number\ of\ samples \tag{3}$$

To this point it is possible to calculate the available consumption with expression (3). However, to calculate the approximate value of the real consumption despite the missing data it is necessary to replace the number of samples in (3) with the number of seconds from t_0 to t_1. This is possible because the sample rate of data acquisition is the second. So every second has its sample, otherwise there is an error, see Fig. 3.

$$seconds_{t_0}^{t_1} = Samples_{t_0}^{t_1} \tag{4}$$

$$Consumption\ approximate = Comsumption\ average * Samples_{t_0}^{t_1} \tag{5}$$

This formula allows filling the empty spots with the average values of the consumption of the work cycle.

One can see from Fig. 4 that filling the missing data with average values gives a more accurate image of the consumption of the whole process.

Looking forward to this study, the next stage was to isolate the consumption values of the energy in every task of the process which can be identified as *Job1*, *Job2*, *Job3*, *Home*, *Nojob*. The database saved some variables with these names that

Fig. 4 Data chart of the consumption with approximate values filled

Fig. 5 Data chart of the consumption in Job1 only

are similar to Heaviside function: value 1 when active, value 0 when inactive. Therefore, to isolate consumptions in different states, it is only required to use these variables as Heaviside functions accordingly: *Job1* for task named *Job1* and so forth.

$$Consumption\,Job1 = \sum_{t_0}^{t_1}\left(\frac{Consumption\ samples}{Number\ of\ samples} * Samples_{Job1} * Job1\right) \quad (6)$$

Figure 5 illustrates the isolation of the consumption from the *Job1* task. Following this method, it is possible to isolate the consumption from every task. The next stage is to change the speed of the operation's tasks. Once it is possible to extract the consumption values from every task isolated, it is now interesting to extract the consumption values at different robot speeds during operation tasks.

The PLC used to extract the information measures the robot speed during operation tasks, expressed in the range 0–100% of its maximum speed. The power consumption is also measured in the range 0–100% of the robot's maximum current consumption. For the purposes of this study, it is enough to deal with these percentage scales.

4.3 Results

Table 1 shows the energy consumption for 5 speed steps in five different subtasks and the energy consumption throughout the entire work cycle. Table 2 shows the time of execution of every subtask, for different speeds and also for the normal one.

Table 1 Energy consumptions with the corresponding speed of work cycles

Speed	Job1	Job2	Job3	Home	Nojob	Whole cycle
100	669.25	20.674	0.3717	37.203	29.863	757.36
75	654.04	39.040	1.3849	41.854	30.237	766.55
50	688.57	36.168	2.1117	34.492	22.998	784.34
30	974.66	67.049	4.5011	36.864	29.347	1112.4
10	2139.1	188.00	19.953	30.579	36.939	2414.5

Table 2 Task duration of the work cycle in seconds

Speed	Job1	Job2	Job3	Home	Nojob	Whole cycle
100	30	5	1	8	7	51
75	32	8	2	9	8	59
50	38	9	3	9	8	67
30	49	14	4	10	10	87
10	107	37	9	11	17	181
Normal	38	10	2	10	9	69

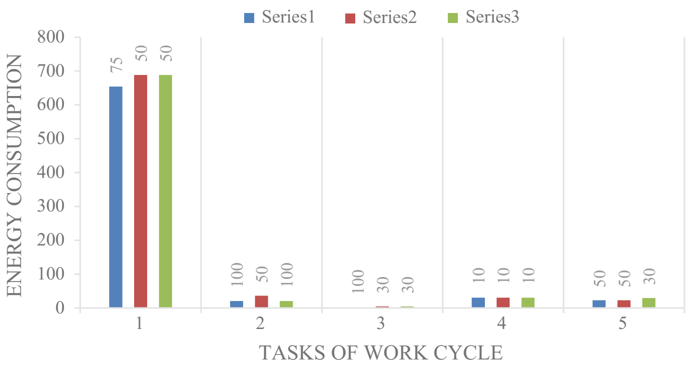


Fig. 6 Chart of different chosen speeds for different subtasks

Figure 6 illustrates the combinations chosen to increase the work cycle efficiency when it comes to energy consumption. Each series represent a combination of task speeds. For example, for the *Series1* the combination is: speed at 75% for the task1, 100% for the task 2, 100% for the task 3, 10% for the task 4, 50% for the task 5, etc.

Three series are the most efficient combination for energy consumption, but the timing of the whole new work cycle must be the same as the normal work cycle. As shown in the Table 3, Series3 is the most efficient new work cycle that has the closest timing of the normal operation work cycle to the standard one.

Table 3 New work cycles times and energy consumption

	Time	Energy
Series1	57	728.6586
Series2	70	782.8139
Series3	68	773.6686
Normal	69	784.3382

5 Conclusions

It is possible to discover new methods to improve the energy efficiency of a robot operation work cycle regardless restrictions. It is unusual for a factory to stop producing to improve its production efficiency, when it comes to energy consumption or work routines. Once the production is started, the factory has every millisecond count to make the whole system produce on time. However, it was proved that there are ways to make any process more efficient using data mining, collecting and analyzing data from any system, which is very important for evolution of any company.

References

1. Trentesaux, D., Borangiu, T., Thomas, A.: Emerging ICT concepts for smart, safe and sustainable industrial systems. Comput. Ind. **81**, 1–10 (2016)
2. Kaisler, S., Armour, F., Espinosa, J.A., Money, W.: Big data: issues and challenges moving forward. In: 46th Hawaii International Conference on Systems Science, pp. 995–1004 (2013)
3. Kantardzic, M.: Data Mining. IEEE (2009)
4. Cupek, R., Drewniak, M., Zonenberg, D.: Online energy efficiency assessment in serial production—statistical and data mining approaches. In: 2014 IEEE 23rd International Symposium on Industrial Electronics, pp. 189–194 (2014)
5. Gamarra, C., Guerrero, J.M., Montero, E.: A knowledge discovery in databases approach for industrial microgrid planning. Renew. Sustain. Energy Rev **60**, 615–630 (2016)
6. Delgado-Gomes, V., Oliveira-Lima, J.A., Lima, C., Martins, J.F., Jardim-Gonçalves, R., Pires, V.F.: Energy consumption evaluation to reduce manufacturing costs. In: International Conference Power Engineering, Energy and Electrical Drives, vol. 5, pp. 1012–1016 (2013)
7. Feng, L., Ulutan, D.: L, pp. 224–228. Mears, Energy consumption modeling and analyses in automotive manufacturing final assembly process (2015)
8. Gao, Y., Tumwesigye, E., Cahill, B., Menzel, K.: Using Data Mining in Optimisation of Building Energy Consumption and Thermal Comfort Management, pp. 434–439 (2010)
9. Zhangl, Y., Liul, S., Sil, S., Yang, H.: Production System Performance Prediction Model Based on Manufacturing Big Data, pp. 277–280 (2015)
10. Wang, S., Wan, J., Zhang, D., Li, D., Zhang, C.: Towards smart factory for Industry 4.0: a self-organized multi-agent system with big data based feedback and coordination. Comput. Netw. **101**, 158–168 (2015)
11. Operational Intelligence, Log Management, Application Management, Enterprise Security and Compliance, Splunk. https://www.splunk.com/

Cybersecurity and Resilience Modelling for Software-Defined Networks-Based Manufacturing Applications

Radu F. Babiceanu and Remzi Seker

Abstract In addition to productivity and quality output, for many years, manufacturing systems were also designed with reliability and safety requirements in mind. In the recent decade or so, that approach seems not to be adequate anymore. The current manufacturing global operations ask for more stringent requirements than ever before, which include privacy and security of transactions, among others. Manufacturing control is not new, but the use of cloud environments to integrate distributed manufacturing facilities and entirely control the production processes across those facilities is an active research area denoted in terms such as: virtual factory, cloud manufacturing, Industry 4.0, and more recently, software-defined networking-based manufacturing. Software-defined networking is a relatively new networking architecture that decouples the network data and control mechanisms and assigns the entire data control to a logically centralized control plane that can be software-programmed based on specific application needs. From the security point of view, this translates in the fact that anyone with access to the controllers that run the network control software could potentially control the entire network. This paper proposes an integrated modelling environment that addresses the manufacturing system assurance through cybersecurity and resilience mechanisms for software-defined networks-based (SDN-based) manufacturing applications. First, the paper presents the proposed combined cybersecurity-resilience ontology to be used in the requirements capture of the manufacturing network design stages. Then, the paper presents the framework for SDN-based cybersecurity-resilience mechanisms for manufacturing applications, and ends with a future research section concerning the proposed cybersecurity-resilience modelling environment.

Keywords Software-defined networking · Cybersecurity-resilience mechanisms · Manufacturing logical control

R.F. Babiceanu (✉) · R. Seker
Department of Electrical, Computer, Software, and Systems Engineering,
Embry-Riddle Aeronautical University, Daytona Beach, FL 32114, USA
e-mail: babicear@erau.edu

R. Seker
e-mail: sekerr@erau.edu

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_15

1 Introduction

The modern society relies on its engineered systems for delivering products and services essential for its well-being and maintaining the quality of life. Disruptions in the services provided can cause severe harm to the safety and security of the society and can skew how the society views its public servants and service providers. Manufacturing is no exception to this rule. Our engineered systems have evolved over time in such a way that they have harvested the power of and benefits coming from ubiquitously available computers. By using computing, we can reduce the cost of operations and increase the quality of the services being provided. To provide just a few examples, it is no wonder we have computers controlling our cars today. Using computers, we can have cars that are safer and more fuel efficient, not to mention cheaper to build. Also, our power grid which now we call the smart grid is controlled by computers. The power can be routed where it is needed automatically, in order to keep maximum utilization and efficiency of the network. Manufacturing control is not new, but the use of cloud environments to integrate distributed manufacturing facilities and entirely control the production processes across those facilities is an active research area denoted in terms such as: virtual factory, cloud manufacturing, Industry 4.0, and more recently, software-defined networking-based manufacturing [1, 2]. As the use of computers provides us with more convenient features, we have to also admit and be prepared to address the inherent associated risks. We must face the reality that our engineered systems can be attacked remotely, without the need to be physically present next to them. All the more recent developments in the cyber world give us no expectation that cyber attackers could ever be eliminated, thus we need to proactively prepare to protect our engineered systems.

Cybersecurity is a rapidly growing field devoted to safeguard the privacy, confidentiality, and integrity of digital information (data and control packets) stored and/or transmitted in any format over internal networks and/or over the Internet. With daily attacks becoming more and more sophisticated, cybersecurity protection through firewalls, intrusion detection, and other systems is becoming of utmost importance for individuals, businesses, and the government alike. Resilience is a measure used in natural, organizational, social, and engineered systems that evaluates the ability of the system to "prepare, absorb, recover from, or more potentially adapt to actual or potential adverse events" [3]. Another definition views resilience as the "recovery of levels of service … after a disturbance," where the service level recovery "may be accompanied by changes in system components and structure that are temporary and permanent" [3]. Some 25 different definitions of resilience were identified in a comprehensive study on resilience [4]. The study also proposes a framework for resilience assessment that includes the following aspects: system identification, resilience objective setting, vulnerability analysis, and stakeholder engagement. The overarching theme of the definitions, regardless their domain of origin, is that resilience is the ability to reduce the magnitude and/or duration of disruptive events and includes system's ability to absorb, adapt, and recover from disturbances [4].

This work proposes an integrated modelling environment that addresses the manufacturing system assurance through cybersecurity and resilience mechanisms for software-defined networks-based (SDN-based) manufacturing applications. Industrial control systems, which are part of distributed manufacturing networks are fundamental examples of assured systems expected to be both resilient and secure in order to provide dependable services. This work follows the framework defined in [5] and brings together the resilience and security characteristics of a system by viewing resilience as the preservation of a required state of security. Security is breached once the required state is no longer true, while resilience is the ability of the system to restore itself to the required state of security. Security and resilience have different timing for the application of detection, containment, and resolution measures. In the case of the security, all measures are applied after the attack (i.e., after the security is breached), whereas in the case of resilience, attempts are made to apply all the measures before the attack (i.e., to make the possible breach in security trivial with easily employed solutions for detection, containment, and resolution).

From this point forward the paper is structured as follows: Sect. 2 provides a review of the most important aspects of the SDN-based manufacturing control and general resilience aspects, and after that, Sect. 3 presents the proposed combined cybersecurity-resilience ontology to be used in the requirements capture of the network design stages, together with the framework for SDN-based cybersecurity-resilience mechanisms for manufacturing applications. Finally, the future research concerning the proposed cybersecurity-resilience modelling framework is outlined in the conclusions section.

2 Software-Defined Networking and Cybersecurity-Resilience

2.1 Brief Review on Software-Defined Networking for Manufacturing Applications

Three recent surveys and comprehensive reviews on software-defined networking were identified in the literature [6–8], with the cybersecurity-resilience aspects introduced as ongoing research efforts and challenges in SDN-based systems specifically by the first listed reference [6]. Other works also identify SDN-based security applications as an open problem, outlining that SDN-based network requires the development and creation of security applications that interact with the controller API to accomplish the desired security functions [9]. The reported

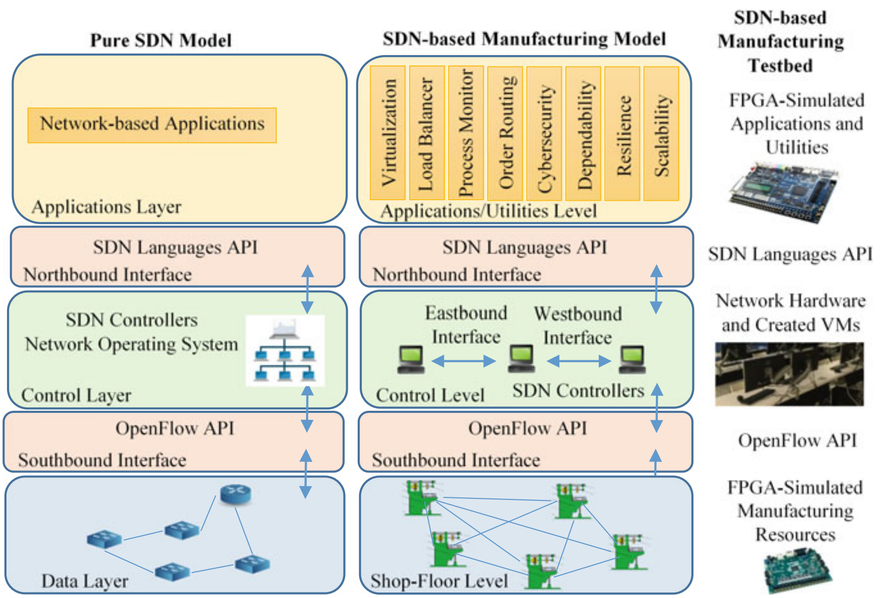


Fig. 1 SDN-based manufacturing model and testbed, adapted from [1]

manufacturing-specific SDN applications are still scarce; a few of them are mentioned next. As manufacturing operations increase in their distributed locations, the susceptibility to cyber-attacks also increases. SDN-based logical operations may be better positioned to detect those attacks and identify new routes for the flow of proprietary information. Remote 3D printing is employed through the use of SDN-based network operations centres that enhance responsiveness and reliability of manufacturing-based networks [10]. Probably the first environments for SDN-based manufacturing applications are presented in [1, 11]. The first environment considers manufacturing systems as aggregation of cyber-physical systems (CPS), which transmit/receive production orders and implement production processes through the use of information and communication technologies (ICT). Algorithms for routing time-sensitive communication flows are derived and evaluated [11]. The second framework [1] implements the main components of the SDN-based environments, including a populated application level layer, which briefly addresses the cybersecurity aspects, but leaves out any reference to network resilience. Figure 1 presents an updated version of the previously defined SDN-based manufacturing testbed environment [1]. Besides the network level applications defined for SDN environments, the updated model includes the cybersecurity and resilience application/utilities features introduced in this paper.

2.2 Brief Review on Resilience and Cybersecurity Aspects

Resilience was always studied as one of the most important characteristics of network-based systems, as failures due to attacks of links or nodes reduces the overall system performance. In traditional IP-based systems, local failures of nodes or links are addressed through the low-level routing performed by network devices. If failover paths are pre-programmed into neighbouring network devices, those paths can be activated upon failure detection and the attack is dealt with at network devices lower-level. This is not the case anymore for SDN-based networks, where routing decisions are removed from the network devices and are now logically centralized on the SDN Control Layer.

Following the above logic, for SDN-based network resilience is defined as the ability to recover the control logic after the attack. This architecture makes the SDN network resilience a challenging problem, as it needs to deal with local failures detection on a fast time decision-based framework. OpenFlow API and language-based API solutions for general SDN networks are presented in the literature [6]. Resilience is intrinsically related to notions such as risk and vulnerability. Modelling and simulation is used extensively in the literature to evaluate the risk, vulnerability, and resilience of interdependent infrastructure systems. Several of the modelling and simulation approaches found in the literature are: identification of frequent and significant failure patterns, quantification of interdependency related indicators, empirically risk analyses, multi-agent systems approaches, system dynamics approaches, economic theory approaches, and network science approaches [12]. Two types of measure of resilience were identified [13]:

- *Equilibrium resilience*: This measure is proportional with the inverse of the time necessary for the system to return to its equilibrium state, where equilibrium state implies that external perturbation deviating system from equilibrium state decays and the system returns to the vicinity of the state.
- *General resilience*: This measure is proportional with the inverse of the time necessary for the system to reach its original state, which not necessarily is its equilibrium state.

3 Proposed Cybersecurity-Resilience Modelling Environment for Manufacturing Applications

3.1 Combined System Resilience-Cybersecurity Ontology

This work investigates the adoption of a combined system resilience-cybersecurity ontology to identify and classify system threats, vulnerabilities and risks. In the same way as cybersecurity, resilience is intrinsically related to notions such as threat, risk, and vulnerability. However, there is also an important difference between the two: while cybersecurity is designed most of the time to account for known threats and has its purpose of preventing a security breach, systems are designed for resilience to account for unknown threats and the recovery process takes place after the security breach. System models need to be exposed to cyber threats in isolated laboratory environments to investigate and identify cyber vulnerabilities, as well as investigate and measure their exhibited resilience.

Ontologies are generally defined as shared formal representations of particular domains, and they define a common understanding of the information communicated between people and applications. In order to increase the effectiveness of addressing incoming threats, ontologies are necessary for the timely and effective sharing and processing of data, especially between computers, hence the need for the development and utilization of a combined resilience-cybersecurity ontology. Ontologies can be defined with widely accepted languages like Resource Description Framework Schema (RDFS), DARPA Agent Markup Language (DAML), and Web Ontology Language (OWL), among others. The proposed combined ontology scheme is depicted in Fig. 2, and includes specific ontologies coming from engineering, security-specific (vulnerability/threat/risk), and human behaviour/social influence domains. A reasoning engine is used to combine them together in a consolidated ontology to be used for the resilience-cybersecurity aspects of SDN-based manufacturing applications.

There is a rich literature addressing the ontologies for the manufacturing domain, out of which three papers were reviewed [14–16]. An adaptation of existing

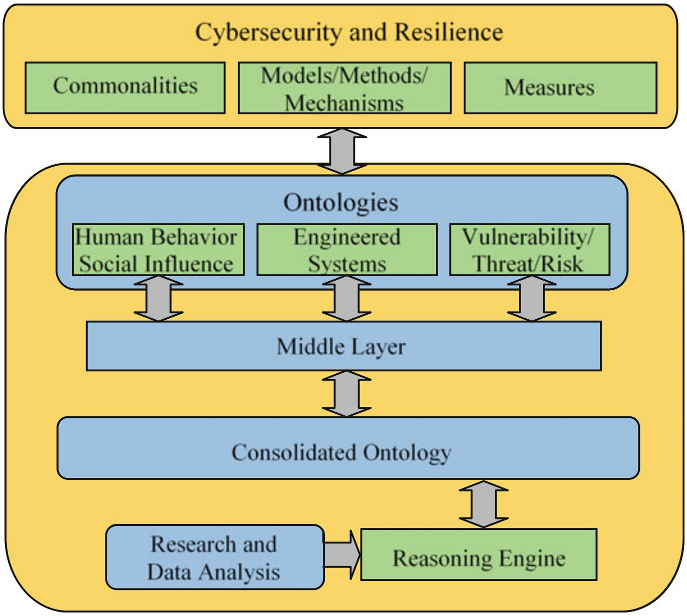


Fig. 2 Combined resilience-cybersecurity ontology

ontologies is envisioned to be included in the "Engineered Systems" module of Fig. 2.

While the ontologies for engineered systems and threats are already available in different formats as mentioned above for the manufacturing ontologies case, this section addresses the less studied human behaviour and social influence aspects. A good starting point is the work presented in [17], which defines social influence in terms of state dependence: how one individual's state impacts other individuals' states and also the opposite influence, how a certain group of individuals' states impact an individual's state. At the end, a good estimate of social influence is obtained by measuring the amount of interaction between two individuals. Given a set of individuals C, every individual c in the set C can be associated with a social influence I, defined as the conditional dependence (probability) between the individual's current state $h_t^{(c)}$ at time t and the previous states of all entities $h_{t-1}^{(c)}$ at time $t-1$ [17].

$$P\left(h_t^{(c')}|h_t^{(1)}, \ldots, h_t^{(C)}\right) = \sum_{c=(1,\ldots,C)} R_{c',c} P\left(h_t^{(c')}|h_{t-1}^{(C)}\right)$$

where R is a matrix that models the tie strengths between individuals.

3.2 SDN-Based Resilience-Cybersecurity Mechanisms for Manufacturing Applications

Most of the world's critical infrastructures are controlled by Supervisory Control and Data Acquisition (SCADA) systems that automatically monitor and adjust process control activities and control physical pieces of equipment that do everything from route trains, to distribute power throughout a city, to adjust the appropriate mix of chemicals for water purification [18]. The fact is that these systems are increasingly connected to the Internet; however they were not designed for cybersecurity. According to the 2015 Dell Security Annual Threat Report [19], the number of SCADA attacks doubled in 2014 compared to 2013, with 51,258 attacks in the Unites States.

Using the ontology defined in the previous section, this work aims at providing a platform for further study of the resilience and cybersecurity concepts and achieving system resilience in the face of cybersecurity threats. However, this work is not trying to solve the detailed challenges of building resilient systems from the security point of view. Rather, this work is attempting to set the framework for addressing system cybersecurity and resilience for SDN-based manufacturing applications right from the manufacturing applications network design stages, keeping in mind that security was not specifically designed into network applications from the beginning. For example, the widely used TCP/IP protocols do not include encryption, so security needs to be added as an extra-layer for applications needing it.

Scalability of the SDN-based networks is related to resilience as the network is expected to provide its quality of service also in the context of increasing flows of packets. The decoupling of data and control planes can result in a bottleneck at the control plane level due to increase traffic towards the same controllers when new flows are processed, as opposed to low decentralized traffic at the network devices level. Basically, by introducing a central authority to control, albeit a logically only one, to control the network traffic of a large network, the growth of network traffic may not scale with the performance of the controllers [20].

In addition to the IP-based vector threats such as various attacks on network devices, security of SDN-based networks comes with specific vector threats resulted from the decoupling of data and control planes, The SDN controllers on the control plane are vulnerable to attacks with the more sophisticated attacks resulting in the attacker gaining total control of the network. To address the view of resilience as defined in this paper, it is necessary to design the SDN-based system to proactively analyse the potential threats. Viewing resilience as restoring network functionality after attacks is not sufficient, regardless how fast it is done. To preserve the required network state of security, SDN-based resilience models must provide dynamically network reconfiguration solutions always ready to be employed in a much faster time than reactive restoring of network functionality. Figure 3 presents the schematic mechanisms for cybersecurity-resilience response in the face of a security attack. As defined in a previous section using the framework defined in [5], to account for its resilience, the SDN-based manufacturing system must be maintained in a required state of security through resilience countermeasures or by employing security mechanisms, if the applied countermeasures proved non-effective. The first option is preferred though, from the timely execution point of view of specific manufacturing applications.

Attacks in cyber-physical environments such as the SDN-based manufacturing environment are malicious actions that exploit one or more vulnerabilities, and can

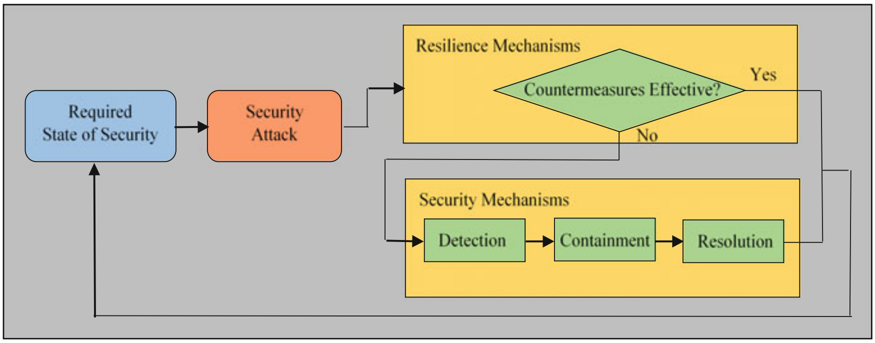


Fig. 3 Resilience-cybersecurity mechanisms to maintain the SDN-based manufacturing system in a required state of security

occur in cyber world, physical world, or both [21]. Cyber world attacks can result in corrupted data, unauthorized access to systems, and denial of service of networks and computers, employing access, use, disclosure, disruption, modification, or destruction of data and/or data interfaces. Physical world attacks can result in contaminated environment, unauthorized intrusions, hardware and infrastructure destruction or unauthorized relocation, and human coercion [21].

4 Conclusions and Future Work Directions

This work presents a cybersecurity and resilience framework for software-defined networking-based manufacturing applications, which include a combined resilience-cybersecurity ontology model and mechanisms to maintain the manufacturing system in a required state of security. SDN-based manufacturing applications need to bring together new equipment that is more adapted to be networked over the Internet (IoT-ready equipment), as well as legacy systems that were built a while back without specifically considering security as a design requirement. These legacy systems, also known as industrial control systems were designed with safety and reliability as major design considerations. No security and privacy aspects were considered in the design as the systems were designed to be open and interoperable within isolated environments. The challenge for SDN-based manufacturing applications is now to add security layers to systems that run on older processors, do not tolerate downtime, and were specifically designed to be subject to very rare software changes.

The SDN-based manufacturing testbed will be implemented in an external network-isolated cybersecurity laboratory that allows for a large number of manufacturing resources, applications, and utilities to be simulated using FPGA devices connected to the testbed network. The resilience-cybersecurity mechanisms will be tested to evaluate the levels of simulated production processes at which the SDN-based manufacturing system is kept in the required state of security. Performance of routing algorithms will also be evaluated from the flow of information point of view only.

References

1. Babiceanu, R.F., Seker, R.: Secure and resilient manufacturing operations inspired by software-defined networking. In: Borangiu, T., Thomas, A., Trentesaux, D. (eds.) Service Orientation in Holonic and Multi-Agent Manufacturing, SCI, 640, pp. 285–294. Springer, Heidelberg (2016)
2. Lee, J., Bagheri, B., Kao, H.-A.: A cyber-physical systems architecture for Industry 4.0-based manufacturing systems. Manuf. Lett. **3**, 18–23 (2015)
3. Department of Homeland Security, NIPP 2013: Partnering for Critical Infrastructure Protection and Resilience. https://www.dhs.gov/national-infrastructure-protection-plan

4. Francis, R., Bekera, B.: A metric and frameworks for resilience analysis of engineered and infrastructure systems. Reliab. Eng. Syst. Safety **121**, 90–103 (2014)
5. Thompson, M.A., Ryan, M. J., Slay, J., McLucas, A.C.: A new resilience taxonomy. In: Proceedings of the INCOSE International Symposium (2016)
6. Kreutz, D., Ramos, F.M.V., Verissiomo, P.E., Rothenberg, C.E., Azodolmolky, S., Uhlig, S.: Software-defined networking: a comprehensive survey. Proc. IEEE **103**(1), 14–75 (2015)
7. Xia, W., Wen, Y., Foh, C.H., Niyato, D., Xie, H.: A survey on software-defined networking. IEEE Commun. Surv. Tutor. **17**(1), 27–51 (2015)
8. Jammal, M., Singh, T., Shami, A., Asal, R., Li, Y.: Software defined networking: state of the art and research. Comput. Netw. **72**, 74–98 (2014)
9. Yeganeh, S.H., Tootoonchian, A., Ganjali, Y.: On scalability of software-defined networking. IEEE Commun. Mag. **51**(2), 136–141 (2013)
10. Kim, D., Gil, J.-M.: Reliable and fault-tolerant software-defined network operations scheme for remote 3-D printing. J. Electron. Mater. **44**(3), 804 (2015)
11. Nayak, N. G., Durr, F., Rothermel, K.: Software-defined environment for reconfigurable manufacturing systems. In: Proceedings of the 5th International Conference on Internet of Things (2015)
12. Ouyang, M.: Review on modeling and simulation of interdependent critical infrastructure systems. Reliab. Eng. Syst. Saf. **121**, 43–60 (2014)
13. Erdi, P.: Complexity Explained. Springer, Heidelberg (2010)
14. Borgo, S., Leitao, P.: The role of foundational ontologies in manufacturing domain applications. Lect. Notes Comput. Sci. **3290**, 670–688 (2004)
15. Leimagnan, S., Siadat, A., Dantan, J.-Y., Semenenko, Y.: MASON: a proposal for an ontology of manufacturing domain. In: Proceedings of the IEEE on Distributed Intelligent Systems: Collective Intelligence and its Applications, pp. 195–200 (2006)
16. Usman, Z., Young, R.I.M., Chungoora, N., Palmer, C., Case, K., Harding, J.A.: Towards a Formal Manufacturing Reference Ontology. Int. J. Prod. Res. **51**(22), 6553–6572 (2013)
17. Pentland, A.: Social Physics, How Good Ideas Spread: The Lessons from a New Science. Penguin Group (2014)
18. Goodman, M.: Future Crimes: Everything is Connected, Everyone is Vulnerable, and What We Can Do About It. Random House New York (2015)
19. Dell Security Annual Threat Report. https://software.dell.com/docs/2015-dell-security-annual-threat-report-white-paper-15657.pdf (2015)
20. Jarraya, Y., Madi, T., Debbabi, M.: A survey and a layered taxonomy of software-defined networking. IEEE Commun. Surv. Tutor. **16**(4), 1955–1982 (2014)
21. Sampigethaya, K., Poovendran, R.: Aviation cyber-physical systems: foundations for future aircraft and air transport. Proc. IEEE **101**(8), 1834–1855 (2013)

Part IV
Holonic and Multi-agent System Design for Industry and Service

Formal Specification of a Self-sustainable Holonic System for Smart Electrical Micro-grids

Adriano Ferreira, Paulo Leitão and José Barata Oliveira

Abstract Stand-alone micro-grids have emerged within the smart grids field, facing important challenges related to their proper and efficient operation. An example is the self-sustainability when the micro-grid is disconnected from the main utility, e.g., due to a failure in the main utility or due to geographical situations, which requires the efficient control of energy demand and production. This paper describes the formal specification of a holonic system architecture that is able to perform the automation control functions in electrical stand-alone micro-grids, particularly aiming to improve their self-sustainability. The system aims at optimizing the power flow among the different electrical players, both producers and consumers, to keep the micro-grid operating even under adverse situations. The behaviour of each individual holon and their coordination patterns were modelled, analysed and validated using the Petri net formalism, allowing the complete verification of the system correctness during the design phase.

Keywords Holonic system · Smart electrical micro-grid · Self-sustainability

A. Ferreira (✉) · P. Leitão
Polytechnic Institute of Bragança, Campus Sta Apolónia,
5300-253 Bragança, Portugal
e-mail: a.ferreira@ipb.pt

P. Leitão
e-mail: pleitao@ipb.pt

P. Leitão
LIACC - Artificial Intelligence and Computer Science Laboratory,
Rua Campo Alegre 1021, 4169-007 Porto, Portugal

J. Barata Oliveira
Faculty of Sciences and Technology, New University of Lisbon,
Quinta da Torre, 2825-114 Caparica, Portugal
e-mail: jab@uninova.pt

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_16

1 Introduction

The proliferation of Distributed Energy Resources (DER) [1] leads to a change of the energy consumers' role, moving from passive consumers to active "prosumers", i.e. being also able to generate power on site and to manage their surplus, feeding it to the grid or storing it for later usage. Moreover, the increasing number of electric vehicles will have a profound impact on the power system, since storage systems, grid-to-vehicle and vehicle-to-grid features will offer tools that ease the loads management and improve the control of the market pricing. As prosumers become more and more independent, the unidirectional power system will be replaced by a prosumer-dominated system where power will be primarily exchanged locally between prosumers in a bidirectional way. This supports the appearance and operation of small portion of distribution networks called micro-grids. A micro-grid is a small-scale power system comprising DERs and suitable control systems able to work in both grid-connected and islanded operating modes [2]. Islanded, or stand-alone (besides the forced islanding in case of failure of the main utility), are used in case of unavailability to connect the main utility due to financial or geographical reasons. These stand-alone micro-grids tend to rely mainly on Renewable Energy Sources (RES) turning them into a complex and difficult system to predict and control, which require intelligent control and management systems to turn these complex systems as reliable and efficient as possible. In particular, this problem arises in case of unpredicted underproduction events (due to production device failure or unpredicted weather event) where it is necessary to improve the self-sustainability of a micro-grid working in isolated mode.

Emergent ICT technologies and paradigms, and particularly multi-agent systems (MAS) and holonic systems, are being adopted in several domains of smart grids field due to their capabilities to offer modularity, flexibility, robustness, scalability and re-configurability. Different approaches reported in the literature show the benefits of decentralized structures and particularly holonic principles to address the challenges imposed by the smart grids being mainly centred in answering the problems of power scheduling and structural smart grid organization (surveys of several applications can be found in [1, 3]). However, these approaches do not explore important aspects, namely the self-sustainability in stand-alone micro-grids, where a tighter control of the available resources and enhanced cooperation among entities is needed.

Having this in mind, the paper proposes a holonic architecture addressing the self-sustainability in micro-grids that operate disconnected from the main utility grid, being focused the formal specification of the structural and behavioural models of the ecosystem holons, as well as the associated coordination models. The Petri nets formalism [4] is used taking advantage of its powerful mathematical foundation to model, analyse and validate the behaviour of complex event-driven systems.

The rest of the paper is organized as follows: Sect. 2 describes the main guidelines of the proposed holonic system architecture for the self-sustainability in stand-alone micro-grids. Section 3 discusses the formal modelling of the holonic system by using the Petri nets formalism, and Sect. 4 presents the coordination models among distributed holons. Section 5 describes the validation of the designed holons' behaviour models, namely by performing a qualitative and quantitative analysis using the PnDK tool. Finally, Sect. 6 rounds up the paper with the conclusions.

2 Holonic System Architecture for Micro-grid Self-sustainability

The proposed holonic architecture is based upon a set of autonomous and cooperative holons that share resources and responsibilities in order to achieve self-sustainability within the stand-alone micro-grid with the lowest degradation of the user QoS. Each holon represents an autonomous electrical entity, such as producer units (e.g., photovoltaic panel or wind turbine), storage units (e.g., battery bank) and controllable loads (e.g., refrigerator and lamps). Three types of holons where identified, namely *Consumer Holon* (CH), *Producer Holon* (PH) and *Storage Holon* (SH), with each one owing a specific role, proper knowledge and a set of functionalities.

CHs are responsible to manage the controllable loads, ensuring that the delivery of power is performed according to the consumer needs. This type of holon also implements scheduling and prioritization functions that assume particular relevance when the available power is below the load demand. PHs are responsible to ensure a stable power supply, executing power production forecasting and condition health monitoring functions. SHs are responsible to ensure a stable grid absorbing production surplus and to provide power in under-production periods.

The use of holonic principles takes advantages of the distributed nature of this paradigm, and their intrinsic provided features, namely in terms of modularity, flexibility and robustness. Additionally, the recursive property, inherent to holonic systems, allows simplifying the design of such complex large-scale systems. In fact, a single holon can comprise several others in a recursive and fractal manner: as example, a micro-grid holon can comprise multiple holons (e.g., house holons and producers), with each house holon comprising, in a recursive manner, multiple loads (i.e. CHs), producers (i.e. PHs), and storage elements (i.e. SHs). Additionally, a cluster of micro-grid holons can be aggregated forming a larger micro-grid holarchy.

The micro-grid control applications based on this holonic approach are usually complex, due to their distributed nature, making them difficult to understand even when using high levels of abstraction. The design of complex systems, as micro-grids are, requires the formal modelling and validation of the system behaviour, guaranteeing the system correctness and the detection of malfunctions and misunderstandings earlier during the design phase.

3 Modelling the Holons' Behaviour

The holonic system architecture distributes control functions by several autonomous and cooperative holons. The behaviour of each individual holon is formally specified by using the Petri nets formalism [4, 5], which, based in its powerful mathematical foundation, allows to analyse, validate and verify the behaviour of complex event-driven systems exhibiting parallelism, concurrency, synchronization and resource sharing.

3.1 Behaviour Model of the Producer Holon

PHs represent the resources producing energy, their behaviour model being illustrated in Fig. 1. The initialization procedure, represented by the transition t2, comprises the actions related to loading the holon's profile, registering its skills in the yellow pages service and creating its local database. This procedure is similar to other holons.

In normal operation, PH is continuously monitoring the status of its connected devices (periodically triggered or when an event occurs, e.g., a condition change in the device). This monitoring sub-behaviour comprises a direct access to the physical device to gather production data (represented by t5), such as the instantaneous production power and meteorological parameters (e.g., temperature, irradiation and wind speed according to the type of production device). The historical and current production data is deeply and online analysed (represented by t6) to extract valuable knowledge, e.g., to detect deviations and patterns, namely the detection of anomalies in the device operation (underproduction due to cell damage or bad panel orientation, and detection of surplus in the energy production). For this purpose, data mining algorithms are used.

Forecasting algorithms (represented by t9) are running in parallel to predict the future production of energy (particularly important for RES), anticipating the occurrence of under-production/over-production events. The generation of these unexpected events will trigger the implementation of proper actions to mitigate the possible undesirable impacts, e.g., storing the energy surplus through a proper negotiation process. The result of this activity also affects the power scheduling process (represented by t10).

Also in parallel, the holon is continuously waiting for requests from CH, related to the power delivery availability that starts a negotiation process (t17), and the power supply cancel that triggers a re-scheduling process (t18) aiming to relocate the power elsewhere. In case of a surplus detection (t12), a negotiation process to store the surplus is initiated (t13); in case the storage devices are unavailable, the power dissipation process is started (t15). The sub-behaviour control function handles the switch on/off of the production device, which depends of the inputs

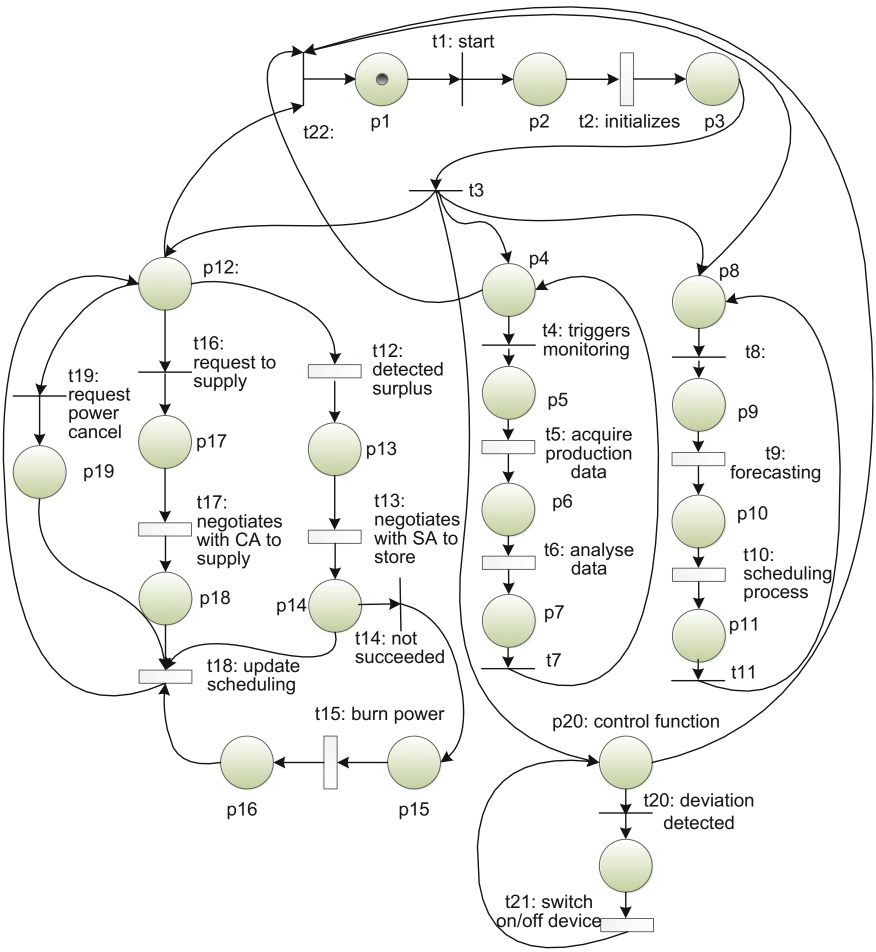


Fig. 1 Petri nets model for the PH

from other sub-behaviors, ensuring a proper physical connection/disconnection (t20).

3.2 *Behaviour Model of the Consumer Holon*

The Petri nets behaviour model for the CH, which represents the controllable loads, is illustrated in Fig. 2.

After the initialization process, similar to the one described for PH, four sub-behaviours are launched to run in parallel: one related to the monitoring process, another to the power acquisition and handling process, another related to load

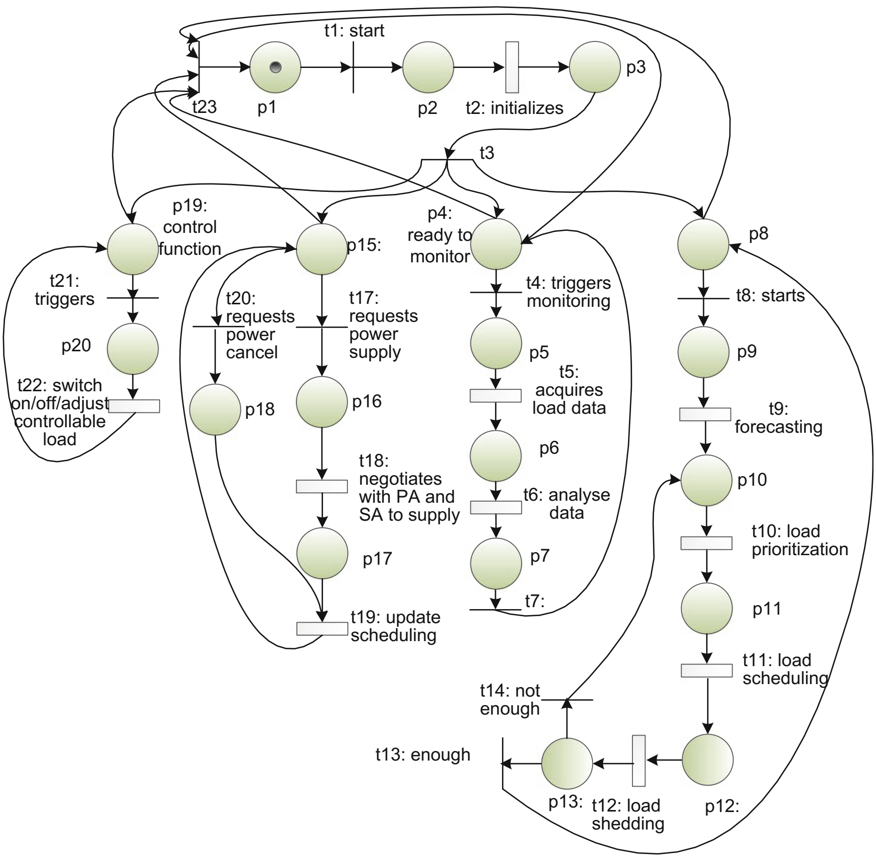


Fig. 2 Petri nets model for the CH

forecast scheduling, prioritization and shedding, and the last one related to the physical control. In the first one the holon monitors the status of the controllable loads and their instantaneous power consumption. As in the PH, the historical and current data is analysed (t6) in order to detect unexpected events, e.g., unexpected consumption profiles.

In the second sub-behaviour, triggered periodically or due to an external event detected in the monitoring process, the load demand is initially forecasted (t9) and then prioritized (t10), being necessary to ensure that the critical loads need to be attended first, and others can be turned-off. The prioritization process is modelled in detail by the Petri nets model illustrated in Fig. 3, where the influence of the production and storage levels as well as the usability of each load are considered. The usability level ensures that no load that is being used or will be used in the near future will be down prioritized, making it highly eligible for the shedding or the scheduling process.

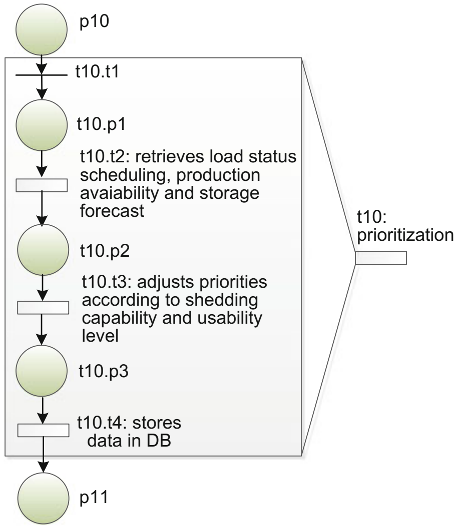


Fig. 3 Petri net model for the prioritization timed transition

The load scheduling, represented by transition t11, tries to shift the non-priority loads to less crowded and less expensive periods, enabling a better distribution of the loads over the time and leading to a better load balance and cost savings. In case of under-supply situations or over-consumption of non-critical loads, a shedding process (t12) is implemented aiming to reduce power consumption of un-priority loads and parasite consumptions, such as equipment in stand-by mode. In parallel, the negotiation process between CH and PHs and SHs (t18) is performed, taking into consideration the load demand after prioritization, scheduling and shedding processes. The physical control, held in t22, handles the loads' switch on/off, as well as the diming power adjustment according to the scheduling and shedding processes.

3.3 Behaviour Model of the Storage Holon

SHs represent storage devices, e.g., battery banks and electric vehicles; its behaviour model is illustrated in Fig. 4. This holon assumes a strategic role in the micro-grid system as it handles the storage of the energy surplus and the supply of energy in empty production time windows, managing the fluctuations of power within the micro-grid.

The forecasting takes place in transition t9 ensuring a proper forecast of the battery levels enabling a better fitment of charging and discharging. The scheduling, represented in transition t10, is responsible to schedule the charging and discharging cycles to maximize the available energy and battery life. At last, the

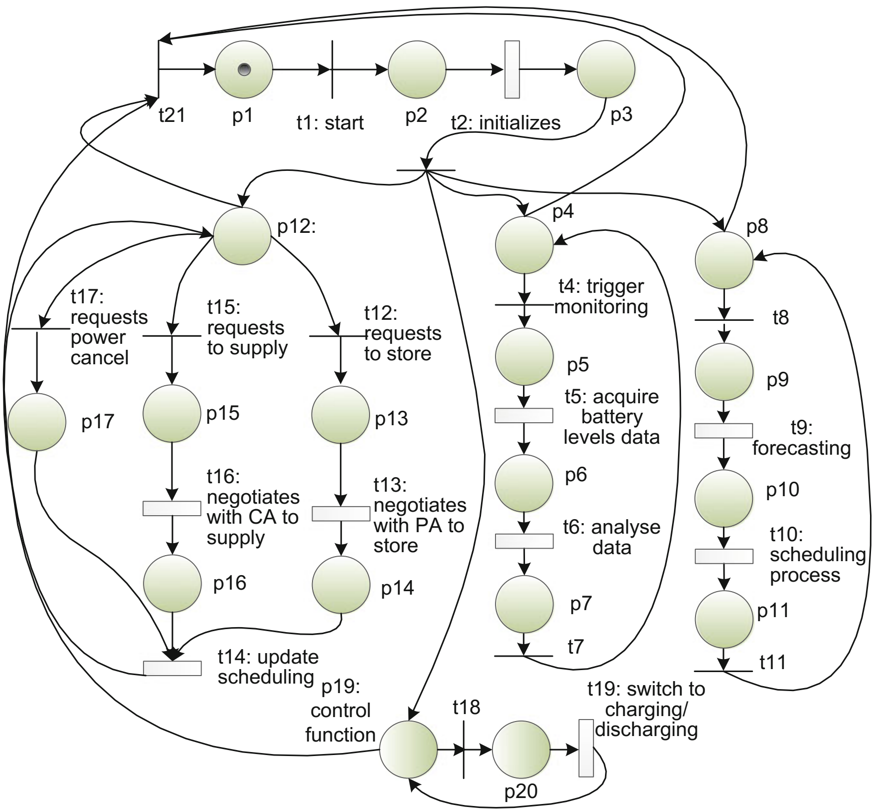


Fig. 4 Petri nets model for the SH

physical control sub-behaviour (t19) is responsible to manage the battery's charge and discharge cycles.

4 Coordination Models

In such distributed systems, the global system operation emerges from the interaction among the individual autonomous entities, which share their knowledge and skills aiming to reach the global system objectives. In this case, the interaction is performed by synchronizing the individual Petri net models described previously. In the literature, several methods can be found to handle these kind of interactions namely MUTEX (Mutual Exclusion Object), Queue Theory, Mailbox and Rendezvous [6]. In this work, the synchronization of the evolution of each Petri net model uses places playing the role of Mailboxes. Figure 5 illustrates the

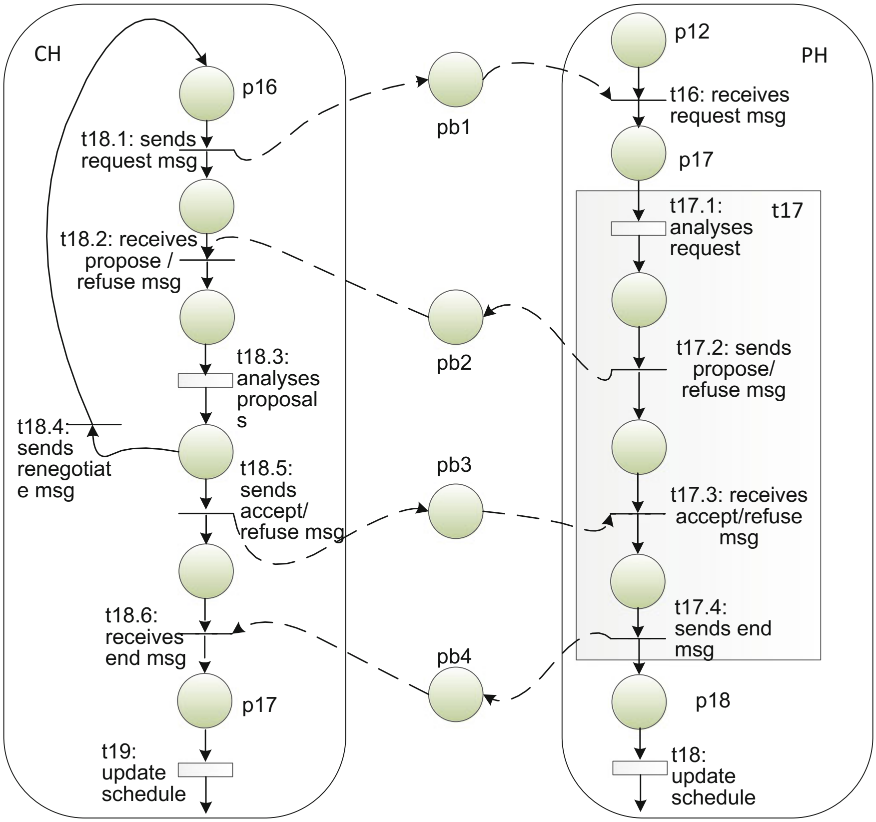


Fig. 5 Coordination of individual models of PH and CHs during the negotiation process

coordination model between a CH a PH that are interacting during the negotiation process for energy acquisition.

During this process, the CH sends a message requesting energy to the PH (indicating the requested amount of energy and the baseline conditions). For this purpose, the place pb1 is marked, enabling the transition t16 in the PH model. When fired, the request is analysed and a proposal is elaborated and sent to the CH by marking the place pb2.

The timed transitions *t*18.3 and *t*17.1 of CH and PH, respectively, model the step-by-step analysis of the proposals and comprise the context aware adaptation as well as learning mechanisms that allows the holons to adapt the negotiation conditions. The negotiation process is finished when PH sends an “end”/“refuse” message, which triggers the scheduling update in each holon. Similar coordination models where developed for other interaction patterns representing cooperation and collaboration processes.

5 Validating the Holon Behaviours' Models

The validation of the behaviour of the holonic system was performed through the analysis and simulation of the behavioural models of each individual designed holon, using the Petri net Development toolkit (PnDK) [7]. Initially, the described Petri nets models and sub-models for each holon were edited in the PnDK tool, being ready to be validated through qualitative and quantitative analysis.

5.1 *Qualitative Analysis*

The qualitative analysis enables the verification of both structural and behavioural properties of the model, allowing observing the system's operation by analyzing the existence of deadlocks, the structural and behavioural conflicts and the capacity of resources.

Figure 6 illustrates the result of the behavioural analysis for the consumer holon model. The behavioural properties analysis was supported by the linear algebra methods provided by the PnDK software tool.

This analysis allows verifying that the model is:

- *Bounded*, i.e. any place holds a maximum of 1 token.
- *Safe*, i.e. all places hold at maximum 1 token.
- *Conservative*, i.e. the number of tokens is constant for all marking, which means that no resources are added or removed during the holon's life-cycle.
- *Deadlocks-absent*, i.e. no matter what marking was reached from the initial marking, there is always a possibility to fire a transition to change to another state.

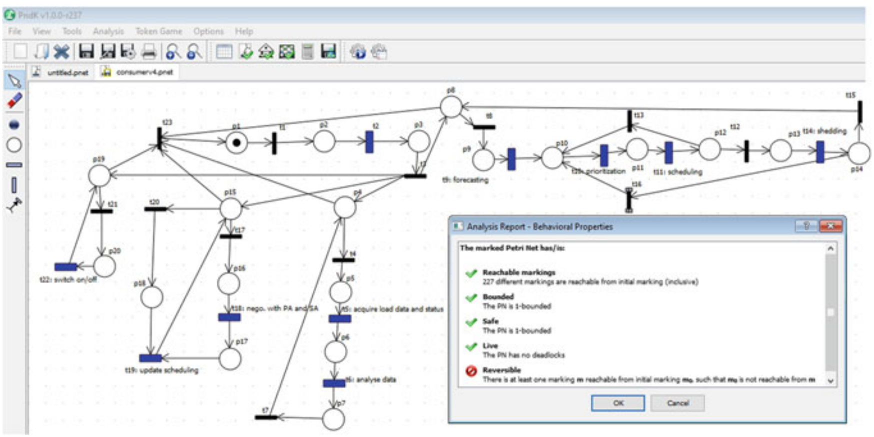


Fig. 6 Behavioural analysis of consumer holon model

The invariants analysis, extracted from the incidence matrix, provide more valuable information about the model. The place invariants analysis allows verifying mutual exclusion relationships among places, functions and resources involved in the structure and behaviour of the holon. For example, the invariants x1 = {p1, p2, p3, p4, p5, p6, p7}, x2 = {p8, p9, p10, p11, p12, p13, p14} and x3 = {p19, p20}, confirm that during the execution of each sub-behaviour, only one of the places/functions can be simultaneously marked. Transition invariants represent the several sequences of operations exhibited by the behaviour model. The existing invariants have the following physical meaning (note that invariants that are linear dependent are not considered):

- $y_1 = \{t_1, t_2\ t_3, t_{23}\}$ represents the initialization and conclusion cycle.
- $y_2 = \{t_4, t_5, t_6, t_7\}$ represents the acquisition, analysis and monitoring of data cycle.
- $y_5 = \{t_8, t_{9,} t_{10}, t_{11,} t_{12,} t_{14,} t_{15}\}$ represents the load prioritization and scheduling.
- $y_6 = \{t_{21}, t_{22}\}$ represents the cycle related to the physical control of the loads.

As previously described, this model comprises several timed transitions (representing the execution of time consuming actions), each one being exploded into a more detailed Petri nets sub-model. According to the Vallete theorem [8], if all sub-Petri nets models of a larger Petri nets are absent of deadlocks and bounded, then the larger Petri nets model is also bounded and absent of deadlocks. Having this in mind, besides the validation of the Petri nets models for the three holons, all sub-Petri nets models for the timed transitions were validated to ensure the correctness of the holons' behaviour.

5.2 *Quantitative Analysis*

The quantitative analysis allows performing an evaluation of the performance, responsiveness and resource utilization, requiring introducing time parameters associated to each transition. For the CH model, it comprises 12 logic transitions (representing the execution of non-time consuming actions), namely t23, t1, t3, t4, t7, t8, t13, t15, t16, t17, t20 and t21 with an estimated time of 1 t.u. (time unit) per transition. Regarding the timed transitions, some functions need more time to be executed and other less time to be executed. In this way, transitions t6, t9, t10, t11, t14 and t18 have associated 5 t.u., and the transitions t5, t19 and t22 have associated 2 t.u. Aiming to simulate the Petri nets model and observe the evolution of the tokens over the time, a token-game simulation was used, allowing to identify important characteristics of the holon's behaviour, such as bottlenecks, cyclic evolution and potential conflicts. The retrieved information contributes to improve the system performance.

The above described analysis has been repeated for the Petri nets models of the three individual holons, ensuring a proper evaluation of the system performance.

Coordination models representing the interaction protocols among the holons were also validated to achieve a complete computational model ready to be deployed.

6 Conclusions

An intelligent and adaptive control must be performed to guarantee that a micro-grid is operating with quality of service with low energy resources, especially when operating disconnected from the main utility. Therefore, this paper describes the formal specification of a holonic system that is able to perform automation control functions in electrical stand-alone micro-grids, particularly aiming to improve their self-sustainability. For this purpose, the Petri nets formalism was used to model, validate and simulate the behaviour of the different designed holons. The validation of the structural and behavioural properties, and posterior simulation of the holons' models, allowed evaluating the correctness of the system behaviour.

Future work will be focused on the specification of appropriate functions to be embedded in the holons, namely forecasting, prioritization, scheduling and shedding, and lately its implementation using a MAS development framework, e.g., JADE [9].

References

1. Merabet, G.H., Essaaidi, M., Talei, H., Abid, M.R., Khalil, N., Madkour, M., Benhaddou, D.: Applications of multi-agent systems in smart grids: a survey. In: International Conference on Multimedia Computing and Systems (ICMCS), pp. 1088–1094 (2014)
2. Katiraei, F., Iravani, M.R., Lehn, P.W.: Micro-grid autonomous operation during and subsequent to islanding process. IEEE Trans. Power Deliv. **20**(1), 248–257 (2005)
3. Leitao, P., Karnouskos, S., Ribeiro, L., Lee, J., Strasser, T., Colombo, A.W.: Smart agents in industrial cyber-physical systems. Proc. IEEE **104**(5), 1086–1101 (2016)
4. Murata, T.: Petri nets: properties, analysis and applications. Proc. IEEE **77**(4), 541–580 (1989)
5. Zurawski, Zhou, M.: Petri nets and industrial applications: a tutorial. IEEE Trans. Ind. Electron. **41**(6), 567–583 (1994)
6. Dijkstra, E.W.: Solution of a problem in concurrent programming control. Commun. ACM **8** (9), 569 (1965)
7. Mendes, J.M., Bepperling, A., Pinto, J., Leitao, P., Restivo, F., Colombo, A.W.: Software methodologies for the engineering of service-oriented industrial automation: the continuum project. In: 33rd Annual IEEE International Computer Software and Applications Conference, pp. 452–459 (2009)
8. Valette, R.: Analysis of petri nets by stepwise refinements. J. Comput. Syst. Sci. **18**(1), 35–46 (1979)
9. Bellifemine, F., Caire, G., Greenwood, D.: Developing Multi-Agent Systems with JADE (2007)

Erlang-Based Holonic Controller for a Modular Conveyor System

Karel Kruger and Anton Basson

Abstract Holonic systems have been a popular approach to face the challenges of the modern manufacturing environment. Holonic control implementations have predominantly made use of the Java Agent DEvelopment framework (JADE). This paper presents, as an alternative, a case study implementation based on Erlang. Erlang is a functional programming language with strong scalability, concurrency and fault-tolerance characteristics, which prove to be beneficial when applied to the manufacturing control context. The case study used in this paper is the holonic control of a modular conveyor system; this implementation was chosen to demonstrate the advantages that Erlang can offer as implementation language for holonic systems.

Keywords Erlang/OTP · Holonic manufacturing systems · Reconfigurable manufacturing systems · Manufacturing execution systems · Automation

1 Introduction

The modern manufacturing environment is governed by a new set of requirements, driven by unpredictability in market and technology trends. Modern manufacturing systems must adhere to shorter lead times and enhanced adaptability, all while remaining competitive in a global market.

To address these challenges, many studies have identified holonic systems—based on the theories of Koestler [1]—as a suitable solution. Holonic systems are based on the idea of dividing a complex system into smaller functional entities—autonomous components which, through cooperation, constitute the system functionality [2]. The holonic theory fits especially well in the manufacturing context,

K. Kruger (✉) · A. Basson
Department of Mechanical & Mechatronic Engineering, Stellenbosch University,
Stellenbosch 7600, South Africa
e-mail: kkruger@sun.ac.za

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_17

with multi-agent systems (typically using JADE [3]) being used for the implementation; this paper proposes Erlang as an alternative.

Erlang is a functional programming language which was developed for programming large-scale distributed control applications [4]. Erlang was developed specifically for the control of telecommunications switching systems [5], but the inherent characteristics of Erlang—namely concurrency, scalability and fault-tolerance—could prove greatly beneficial in the implementation of holonic control in modern manufacturing systems. The Erlang programming environment is supplemented by the Open Telecommunications Platform (OTP) [6, 7]—a set of robust Erlang libraries and design principles providing middle-ware to develop Erlang systems.

The case study presented in this paper is the control of a modular palletized conveyor system, which uses motor-driven belts along with stop gates and lifting-transverse stations to move pallets (from here on referred to as *carriers*). The conveyor system is also equipped with RFID read/write modules installed on several locations on the conveyor, while the carriers are fitted with RFID tags. The RFID readers provide feedback when a specific carrier arrives at a RFID reader location. The RFID readers were installed at the various stop gate locations.

As an extension to the modularity of the conveyor hardware, several small PLCs are used as opposed to using one centralized PLC. Segments of the conveyor were allocated to a dedicated controller, with all the interfacing between segments being handled at the higher level of control implementation [8].

This paper presents an implementation of Erlang-based holonic control for a modular conveyor system. The implementation is aimed at exploiting the advantages offered by Erlang, thus emphasizing modularity and concurrency in the application. This paper discusses the holonic architecture on which the implementation is based and explains the different components of the holonic controller for the conveyor system.

2 Holonic Control Architecture

The Conveyor holon presented in this paper forms part of a holonic cell control implementation. The cell control architecture is based on PROSA [9], a simplified schematic representation of the architecture being presented in Fig. 1. Detailed discussions of similar implementations are given in [10, 11].

The architecture of the cell control implementation consists of three levels: High Level Control (HLC), Low Level (station) Control (LLC) and hardware control. The communication and coordination of the system holons occur within the HLC. The HLC purely exists in the virtual environment, as the Product, Order and Staff holons are all software entities. Resource holons, which consist of both hardware and software entities, must also be represented in the HLC—it is therefore necessary that these resource holons incorporate a component to handle the HLC functions.

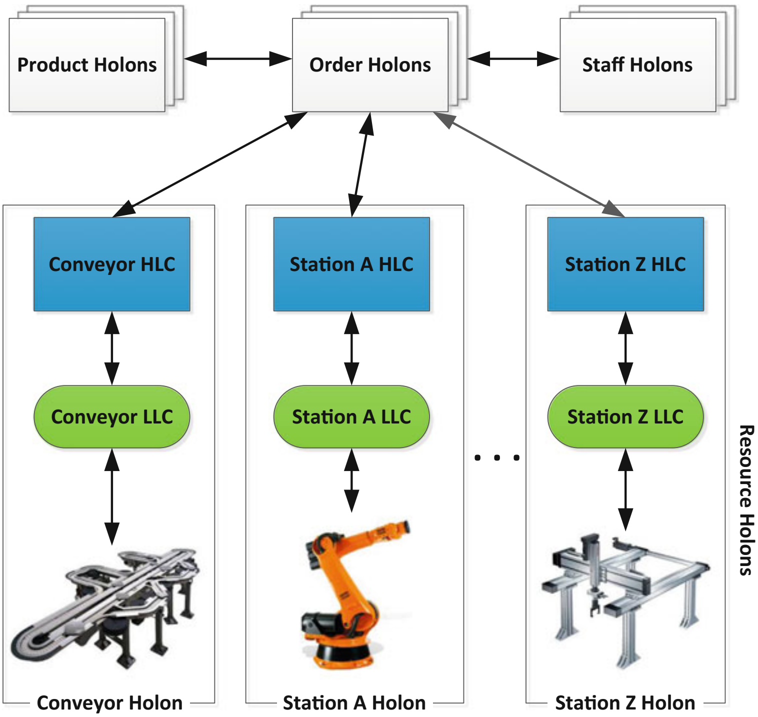


Fig. 1 Schematic of the holonic control architecture for the manufacturing cell

Where Resource holons consist of physical hardware entities, the station and hardware levels of control are encountered. Station LLC enables the coordination of the hardware functions for a station, to perform the service that the Resource holon advertises in the HLC. Hardware control refers to the control of actuators and sensors to successfully perform the various tasks included in the Resource holon's service.

3 Erlang-Based Conveyor Holon

3.1 Conveyor Holon Architecture

As mentioned in Sect. 2, the Conveyor holon forms part of the implementation of the holonic control for a cell. The Conveyor holon is itself implemented using a holonic architecture, i.e. the functions of the holon are mapped to several autonomous and cooperating entities which work together to perform complex transportation tasks. This holonic implementation then constitutes the HLC component of the Conveyor holon (as can be seen from Fig. 1)—the LLC implementation is

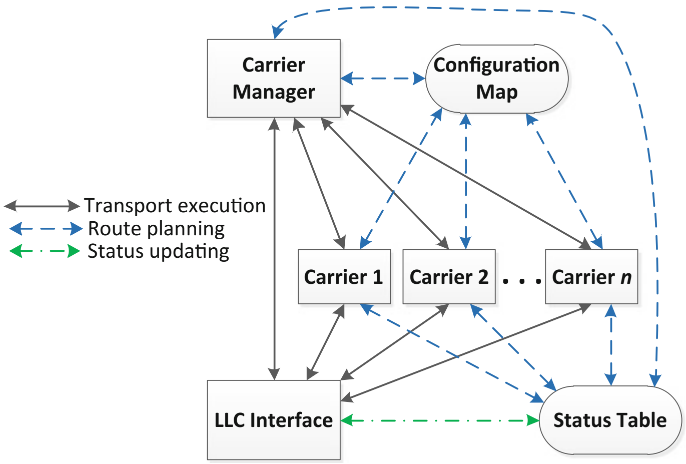


Fig. 2 Intra-holon communication within the Conveyor holon

distributed over the number of PLCs that control dedicated segments of the conveyor hardware.

The holons which comprise the Conveyor holon are shown in Fig. 2 and the respective roles and functions are discussed the following sections. The Conveyor holon entails three main functions: inter-holon communication within the HLC and intra-holon coordination within the Conveyor holon, execution of transportation tasks and the interaction with and virtual representation of the conveyor hardware.

3.2 *Communication*

Inter-holon Communication. The Carrier Manager holon is responsible for handling all communication with the other holons in the cell controller. The Conveyor holon interacts with three types of holons in the cell controller: the staff holon handling the service directory (a list of service-providing Resource holons), other Resource holons which have a physical interaction with the Conveyor holon, and Order holons.

As is usually encountered in the implementation of holonic systems, this presented implementation tries to mirror the physical system as far as possible. An example of this is when Resource holons must physically remove products from the conveyor or place products on it. This interaction is mirrored in the virtual environment—before removing or placing a product on the conveyor, Resource holons must first send a *release request* or *binding request* message to the Conveyor holon. This allows the Conveyor holon to ensure that a suitable carrier is present at the location of placing, or that the intended product is available to be removed by the resource holon. After the Conveyor holon replies to the request, the Resource holon can continue with the physical operation.

The approach of mirroring the physical interactions in the virtual system means that when an Order holon requires a transport service to the next booked service-providing station, the physical product instance for which it is responsible will already be present on a carrier on the conveyor. The Order holon can then proceed to send the Conveyor holon a *service start* message to perform the transportation service.

The Carrier Manager receives requests from Order holons to perform a transport service from some start position to a specified destination. The Carrier Manager then checks if a suitable Carrier holon is available at the requested starting position. If a Carrier is available, the Carrier manager sends a start message to the selected Carrier; if no Carrier is currently available the Carrier Manager will search for a compatible, idle Carrier holon and direct it to the designated starting location.

Intra-holon Communication. Three types of communication occur between the holons of the Conveyor holon: transportation service execution, route planning and status update communication—these interactions are shown in Fig. 2.

Transportation service execution communication requires interaction between holons in order to coordinate and execute the transportation tasks that the Conveyor holon must perform for Order holons. As the Carrier Manager receives requests from Order holon, the requests are allocated to suitable Carrier holons. The Carrier Manager sends *service start* messages to the relevant Carrier holons—these messages specify the end destination to where the Carrier holons must navigate. To execute the movement between conveyor nodes along the selected route, the Carrier holons request actuation from the specific PLCs by sending *request* messages to the LLC Interface holon. The LLC Interface holon then in turn replies with a confirmation that the requested actuation has been performed by the conveyor hardware. When a Carrier holon has completed its assigned transport task, it sends this confirmation to the Carrier Manager and awaits a new task.

Route planning communication entails the gathering of information by holons to aid the route finding process. Predominantly, this communication is performed by Carrier holons; when Carrier holons are assigned a transportation task, they are responsible for planning their own route. The Carrier holons request information of the physical conveyor configuration from the Configuration Map holon and status information of the conveyor nodes and transitions from the Status Table holon; this process is discussed in more detail in Sect. 3.6. The Carrier Manager holon will also occasionally initiate route planning communication—this occurs when the Carrier Manager must control Carrier holon movement for coordination purposes.

Finally, the status updating communication involves the LLC Interface holon passing status information, received from the PLCs, to the Status Table holon.

3.3 Virtual Conveyor Representation

The physical configuration and run-time status of the conveyor nodes and transitions are represented in the virtual environment by two holons: the Configuration Map holon and the Status Table Holon.

The Configuration Map holon contains the functions to read the configuration information from an operator-defined description into an accessible data structure (in this case, an Erlang Term Storage (ETS) table). The configuration information is described in terms of nodes and transitions, and is further supplemented by additional information for the nodes (e.g. which stations of the cell are located at which nodes) and transitions (e.g. transition capacity and transition time).

The Status Table holon maintains an ETS table of the conveyor node and transition status based on messages received from the LLC Interface holon—i.e. the status information is dynamically updated as carriers move along the conveyor.

The Configuration Map and Status Table holons handle all request messages from other holons, searches for and replies with the desired configuration and status information.

3.4 Carrier Manager

The Carrier Manager holon maintains the interface for inter-holon communication with the other PROSA holons (as discussed in Sect. 3.2). The Carrier Manager also handles intra-holon communication—i.e. messages from Carrier holons or the LLC Interface holon. The Carrier Manager thus functions as a server; messages are received and, according to message type and content, the appropriate functions are executed. Examples of such functions are *handleStartRequest()* or *handleCarrierDone()*.

An important function of the Carrier Manager is to allocate transportation tasks received from Order holons to the most suitable Carrier holon. A *start* message is then sent to the selected Carrier holon, upon which the transport service will be performed. Once the service is completed, the Carrier holon notifies the Carrier Manager, which in turn notifies the relevant Order holon.

Usually, the Carrier holons perform movements according to the Order holon request allocated to them by the Carrier Manager. However, the Carrier Manager also has the functionality to make decisions regarding the movement of carriers directly. This functionality is needed to ensure flow on the conveyor (i.e. not having carriers block certain segments) and to store carriers when they are no longer required.

3.5 Conveyor Low Level Control Interface

The LLC Interface holon is responsible for maintaining the interface between the Erlang control programs and the low level control PLCs; this is depicted in Fig. 3.

The communication with the PLCs is done over Ethernet, with messages encoded as XML strings. The PLCs can parse the XML strings to extract the necessary information pertaining to the actuation that must be performed. The LLC Interface holon maintains a TCP socket connection to each of the PLCs.

The LLC Interface holon receives messages from both Carrier holons and the Carrier Manager holon. As the Carrier holons execute their delegated transport services, they must send request messages to the relevant PLCs via the LLC Interface. This occurs every time a Carrier holon arrives at a node—the message will request the actuation at the given node to direct the carrier towards the next desired node (according to the planned route). The LLC Interface interprets this message to determine which PLC the message is intended for (according to the segment of the conveyor where the node is located). The message is then compiled into an XML string and is send over the correct TCP socket to the PLC. Messages from the Carrier Manager holon are handled in the same way.

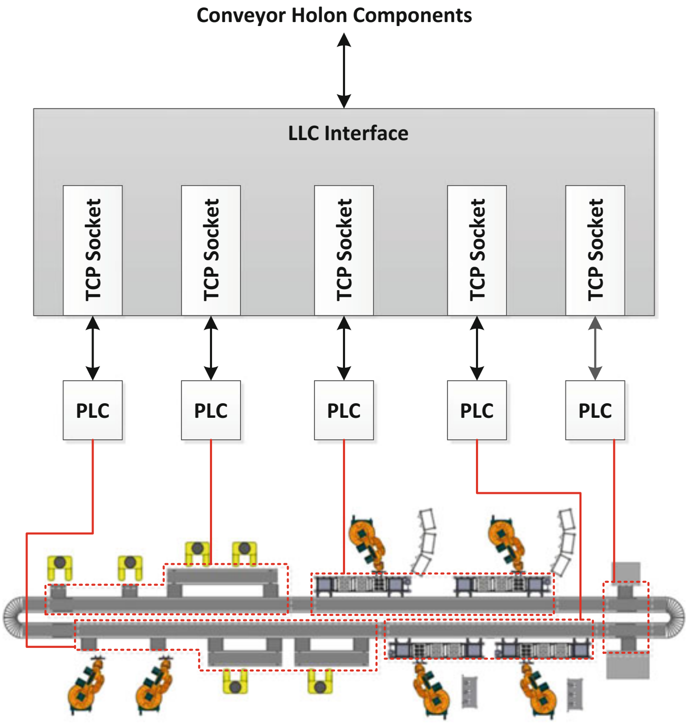


Fig. 3 LLC interface of the Conveyor holon

To maintain a representation of the conveyor status during operation, the LLC Interface holon sends messages to the Status Table holon when it receives notifications from or sends actuation commands to the PLCs.

3.6 Carrier Holon

Each carrier that is unloaded onto the conveyor is represented in the holonic system by a Carrier holon. Every time a carrier is unloaded, the Carrier Manager spawns a new instance of the Carrier holon Erlang process. The Carrier holon encapsulates the functionality to perform transportation services by controlling the movement of the physical carrier on the conveyor system. Although the physical carrier has no actuators or sensors, control of the movement is performed through communication between the Carrier holons and the controlling PLCs, via the LLC Interface holon.

Behaviour. The control logic of the Carrier holon is implemented using the standard OTP finite state machine behaviour. The Carrier holon transitions between states based on the occurrence of events (in this case, the arrival of messages).

The Carrier holon behaviour is described by two states: *stopped* and *moving*. The *stopped* state is entered when the holon awaits its next transportation task and when it reaches a node while travelling towards its destination. The behaviour enters the *moving* state once the LLC Interface holon confirms that the carrier has been physically routed towards the next node on the route. Once the LLC Interface holon notifies the Carrier holon of arrival at the next node, the state transitions to *stopped*.

Communication. As shown in Fig. 2, the Carrier holon engages in communication with other holons during transport service execution and route planning. In the transport execution activity, Carrier holons receive messages from the Carrier Manager holon to initiate a new transport service that must be performed by the carrier. The Carrier holons then send a notification message back to the Carrier Manager when the service is done and await the next service to be awarded. When the Carrier holons travel along their route, they send requests to the LLC Interface which interprets the messages and forwards it to the correct PLC to perform the necessary actuations to direct the carrier along its desired route. The Carrier holons also receive notification messages from the LLC Interface when the carriers arrive at conveyor nodes.

For route planning, Carrier holons must exchange messages with the Configuration Map and Status Table holons. When a Carrier holon is awarded a transportation task, it first determines which route to follow from its current location to its desired location. The Carrier holon can obtain the conveyor configuration and status information, which allows for the implementation of route finding algorithms and strategies.

4 Discussion

The Erlang-based holonic controller presented in this paper has been successfully implemented for two simulated conveyor configurations—initially for a small manufacturing cell (comprising of four different workstations) and a medium-sized cell (incorporating ten different workstations, as shown in Fig. 3). In future work, the research will focus on establishing benchmarks for a formal evaluation of this implementation and an equivalent multi-agent system for comparison. At this stage, the following preliminary remarks can be offered:

- The inherent modularity and concurrency of Erlang programming provides a natural facilitation for the implementation of holonic principles.
- The holonic controller exhibits good scalability and reconfigurability with very little effort.
- The compact, readable code, along with the modularity of Erlang programs allow for a reduction in programming complexity.
- The standard libraries offered by OTP contribute greatly to the simplicity and robustness of the control implementation, with potential for further improvement.

5 Conclusion

The paper presents an Erlang-based holonic control implementation for a modular conveyor system. The conveyor is implemented as a Resource holon in the PROSA holonic architecture upon which the control of the manufacturing cell is based.

The Conveyor holon is responsible for the movement of carriers (which transport products or work pieces around the cell) by controlling the actions of the conveyor hardware via low level control PLCs. The holon performs several functions—communication with other cell level holons, route planning and route execution through hardware coordination. The Conveyor holon is itself implemented as a holarchy, with the involved functions performed through the cooperation of the collection of holons.

The described implementation exploits the advantages that are offered by Erlang, namely modularity, scalability and concurrency. The resulting implementation displays reduced complexity and enhanced reconfigurability, while the potential exists to achieve increased fault-tolerance with further development.

References

1. Koestler, A.: The Ghost in the Machine. Arkana Books, London (1967)
2. Kotak, D., Wu, S., Fleetwood, M., Tamoto, H.: Agent-based holonic design and operations environment for distributed manufacturing. Comput. Ind. **52**, 95–108 (2003)
3. Bellifemine, F., Caire, G., Greenwood, D.: Developing Multi-Agent Systems with JADE. Wiley (2007)
4. Armstrong, J.: Erlang. Commun. ACM **53**(9), 68–75 (2010)
5. Armstrong, J.: Making reliable distributed systems in the presence of software errors. Doctor's Dissertation, Royal Institute of Technology, Stockholm, Sweden (2003)
6. Logan, M., Merrit, E., Carlsson, R.: Erlang and OTP in Action. Manning Publications Co., Stamford (2011)
7. Get Started with OTP. [S.a.]. http://www.erlang.org. Accessed 18 July 2013
8. Kotze, M.J., Basson, A.H.: Control of a Modular Conveyor System Using Object Oriented Programming (submitted for review)
9. Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P.: Reference architecture for holonic manufacturing systems: PROSA. Comput. Ind. **37**, 255–274 (1998)
10. Kruger, K., Basson, A.H.: Implementation of an Erlang-based Resource Holon for a Holonic Manufacturing Cell, Service Orientation in Holonic and Multi-Agent Manufacturing. Studies in Computational Intelligence. Springer International Publishing (2015)
11. Kruger, K., Basson, A.H.: Erlang-based control implementation for a holonic manufacturing cell. Int. J. Comput. Integr. Manuf. http://www.tandfonline.com/doi/full/10.1080/0951192X.2016.1195923

On Rescheduling in Holonic Manufacturing Systems

Carlos Pascal and Doru Pănescu

Abstract This paper treats a topical issue for present manufacturing systems, which is the needed adaptability when manufacturing orders are received in an unpredictable way. To face such cases, a rescheduling mechanism is necessary and such a possibility is investigated for a holonic system that materializes its coordination through a combination between the Contract Net Protocol and Distributed Constraint Satisfaction Problem. The proposed method is investigated for a case study, by doing simulation experiments with the system coloured Petri net model. The results are analysed for the cases when the new command received during execution is more or less important than the ongoing ones.

Keywords Holonic manufacturing systems · Rescheduling · Multiagent systems · Coloured Petri nets

1 Introduction

Present manufacturing systems are facing new and clear challenges; they have to be adaptable to dynamic changes as imposed by customers, new technologies and global market. With respect to the control theory, this is still in a transient regime from classical hierarchical control schemes to distributed or semi-heterarchical approaches. The last ones were mainly intended to create an increased adaptability; thus, nowadays control systems must be able to deal with rapid modifications of manufacturing goals and circumstances and provide the optimal response. This

C. Pascal (✉) · D. Pănescu
Department of Automatic Control and Applied Informatics,
"Gheorghe Asachi" Technical University of Iasi, D. Mangeron 27,
700050 Iasi, Romania
e-mail: cpascal@ac.tuiasi.ro

D. Pănescu
e-mail: dorup@ac.tuiasi.ro

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_18

paper analyses one issue from this area, namely rescheduling when manufacturing orders are received in an unpredictable manner.

The literature is abundant in research papers on scheduling/rescheduling. The important role of rescheduling for manufacturing systems is highlighted in [1, 2, 3]; though being considered a major issue, it is to notice that a gap exists between theory and practice of manufacturing scheduling. Our paper is aiming at reducing this weakness by proposing a modelling instrument that can be used in developing simulation experiments. Moreover, in [1] several differences are specified between the theoretical study of scheduling problems and the real-life scheduling cases. Some of them, namely the constant arrival of new orders, the necessity of rescheduling and the importance of the jobs' priorities are addressed by our paper, too. In [2], it is underlined that current planning and scheduling systems have poor performance with respect to their reactive capability and the ability to obtain a detailed solution in a reasonable time. Distributed approaches, as the one proposed in this paper, can improve both reactivity and computational efficiency.

Different architectures were proposed so that to enable rescheduling. In [2], a combination between a centralized scheduling system (this is based on constraint programming, a mechanism close to our solution that uses the Constraint Satisfaction Problem—CSP method) and a Holonic Manufacturing Execution System (HMES) is considered. The additional problem in this case is about the data exchange between the centralized component and the HMES, this being explained in [2]. In our proposal, scheduling and rescheduling are integrated in the Holonic Manufacturing System (HMS) and thus no additional interface is needed. Nevertheless, a problem can be with the optimality of scheduling that has to be managed by the distributed system. With respect to this, our paper is treating problems with few variables for which the optimal solution can be found in reasonable time, supposing that the disturbances that imply rescheduling create such cases; the proposed method can be also useful for more complicated situations, but the proposed coloured Petri net (CPN) model may be difficult to use. Furthermore, our method can be applied in a way similar with that proposed in [4], namely to have a mechanism for complexity reduction, which can be implemented by a centralized component—e.g., a staff holon, and then to employ rescheduling for problems of lower complexity. From the other proposed solutions for integrating rescheduling in manufacturing control schemes, one can notice the use of service-oriented architectures [3, 5, 6]. In [3] a service-oriented approach adapted for small lot manufacturing is proposed, being named Intelligent Service based architecture. In [6], it is shown how a service-oriented architecture can facilitate the integration of DisCSP based planning/scheduling within HMESs. Cyber physical systems were also used to enable rescheduling [4]; in this case, rescheduling could be done at the global level of a company, the rescheduling component being linked to the Enterprise Resource Planning system.

One can discover that a great deal of the new proposed approaches on rescheduling is based on artificial intelligence methods. Thus, in [7] genetic algorithms are involved so that, in the same way as in our proposal, rescheduling can modify only part of a previous schedule in order to accommodate the new arrived orders. The experiments developed in [7] showed that it is possible to improve the performance of the scheduling scheme with a non-reshuffle strategy. A literature survey on how genetic algorithms can be used in manufacturing scheduling, and what are the strong and weak points can be found in [8]. It must be also remarked that the usefulness of Petri net models is observed in [8], these being involved to represent the problems and to increase the efficiency of genetic algorithms. In our paper, CPNs are used to develop model prototypes for multiagent based coordination mechanisms. A possibility of using multiagent systems in scheduling and rescheduling is investigated in [9]. In the same way as in our case, an iterative agent bidding scheme is involved, but with the limitation of dealing with orders only one by one. A rescheduling mechanism that is able to deal with rush orders in an HMS is presented in [10], being based on a specific dynamic scheduling algorithm (Earliest Deadline First), which is integrated in the Contract Net Protocol (CNP); our method is also using CNP, but this is connected with Distributed Constraint Satisfaction Problems (DisCSP). In [11], more methods from the area of knowledge representation and reasoning (a subfield of artificial intelligence) are enumerated as valuable for solving the rescheduling problems; between the tools indicated as helpful for modeling, there are the Petri nets, the ones preferred by us, too.

The goal for this paper was to develop and evaluate a rescheduling method applicable for HMSs and to test it for a case study. The paper is organized as follows. First, the coordination scheme is introduced with the focus on how rescheduling can be integrated and holonic communication reorganized; then the CPN model of the HMS is briefly introduced, this allowing us to make simulation experiments for a case study being presented in the next section. Finally, a few conclusions are drawn and the possible development is commented.

2 A Holonic Coordination Mechanism that Allows Rescheduling

For holons' coordination, a combination between the CNP and DisCSP is involved [12, 13]. As it can be seen in Fig. 1, there is a first stage devoted to planning and scheduling that is based on CNP. Here, product/order holons are managers and issue goals according to the manufacturing orders to be solved. We took into account our previous research, i.e., a holonic scheme derived from PROSA [14], named HAPBA, which is characterized by a flexible way of holarchy construction, starting from order or product holons [15, 16, 17]. In what follows we discuss about holons, but actually we think about the decisional component of a holon, i.e., its agent; thus, coordination is judged at the level of a multiagent system. For sure, the execution part must be also involved, this one receiving the command for execution (according to the awarded contract) and supplying the feedback when an action is finalized. We suppose that managers (initially, there can be one or more managers, meaning the treatment is made at the batch level) use predefined plans and according to these they identify potential contractors among resource holons. This is done by a bidding process, within CNP [12, 13]. Then, DisCSP is involved in order to obtain schedules for product/order holons by considering two types of constraints: *ordering constraints* that take into account imposed orders among actions of plans, and *overlapping constraints* that appear from resource holons that cannot carry out more actions in the same time. The first phase of Fig. 1 implies a succession of sub-phases, namely: CNP for goal announcement and bidding (this is carried out between managers and contractors), DisCSP based resource allocation and scheduling (this is a negotiation only among managers), and finally the awarding of contracts according to CNP, so that the execution phase to be started.

Taking into account the goal of this research, it is supposed that a new manufacturing order appears after a while, being handled by a corresponding order/product holon (in Fig. 1 this is manager M_{new}). As Fig. 1 suggests, the coordination mechanism tries to integrate a replanning/rescheduling stage within execution, in a real time manner. All managers (order/product holons) are announced about the new manufacturing goal. Then, the following specifications are considered:

(1) Managers should keep their plans, with respect to the order of actions;
(2) An action that is in execution will not be stopped;
(3) For the real time operation, at the shop floor level after the completion of a plan's action the following actions can be delayed as needed by rescheduling;
(4) The new involved order/product holon becomes a manager and applies CNP in order to find potential contractors for the actions of its plan;
(5) A rescheduling phase is launched and all previous managers consider the part of their plans that is not already carried out; to solve this stage, first DisCSP is involved in order to establish the new schedule and then, through CNP, some contracts are cancelled and new ones are awarded;
(6) According to the new contracts, execution is continued.

From the above stages, we focused on the fifth one. About this, there are distinct cases depending on the importance of the new manufacturing order that must be additionally solved. The importance of a command will be decided depending on each manufacturing environment; for example, criteria to be considered may be: the importance of clients, requested deadlines or maximization of benefit. If this is less important than the orders being in execution, then rescheduling is done without modifying the schedules already established. In the other case, if the new order is more important, a negotiation is launched between all managers, so that the new introduced manager is considered as having the maximum priority. These two situations are analysed through the case study that will be presented later. In the next section, the CPN model that allowed us to make simulation experiments with the proposed coordination scheme is briefly introduced.

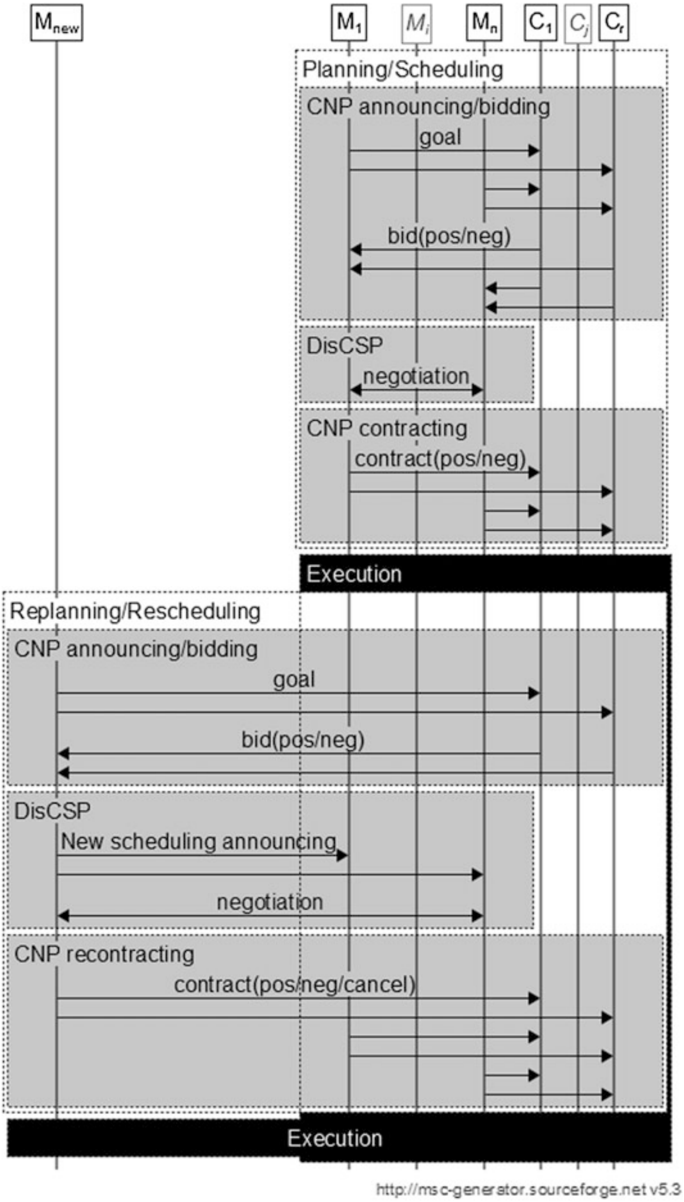


Fig. 1 Holonic coordination with rescheduling

3 A CPN Model for the Holonic Coordination Scheme

Considering our previous work [12, 13, 18, 19], a hierarchical CPN model was developed addressing HMSs that involve both DisCSP and CNP for their coordination; this is presented in Fig. 2. The CPN Tool was used for model construction [20]. The common notation of Petri nets is used, namely rectangles represent transitions and ellipses denote places. Each token keeps, in accordance with the place where it is present, a certain type of information. For example, in place *In MSGS* each token is a buffer for messages received by a holon; the same is true for the place *Out MSGS* with respect to the sent messages. These two places make the connection with the upper layer of the CPN where managers, contractors and the communication network are modelled (see Fig. 2b). Figure 2a depicts the model for a holon being a manager within CNP. The CPN model for a contractor is simpler than the one of Fig. 2a, namely it contains transitions for the events of transmitting bids and feedbacks after the completion of actions, while places are for the sent bids, received contracts, received and sent messages.

Places *Agent view, Current start point, Current priority, Received* and *Sent nogoods* and transitions *Choose start point*, *Backtrack*, and *Positive/Negative* result are needed for modelling the DisCSP mechanism (Fig. 2a). Three types of DisCSP algorithms can be tested with this model, i.e., Asynchronous and Synchronous

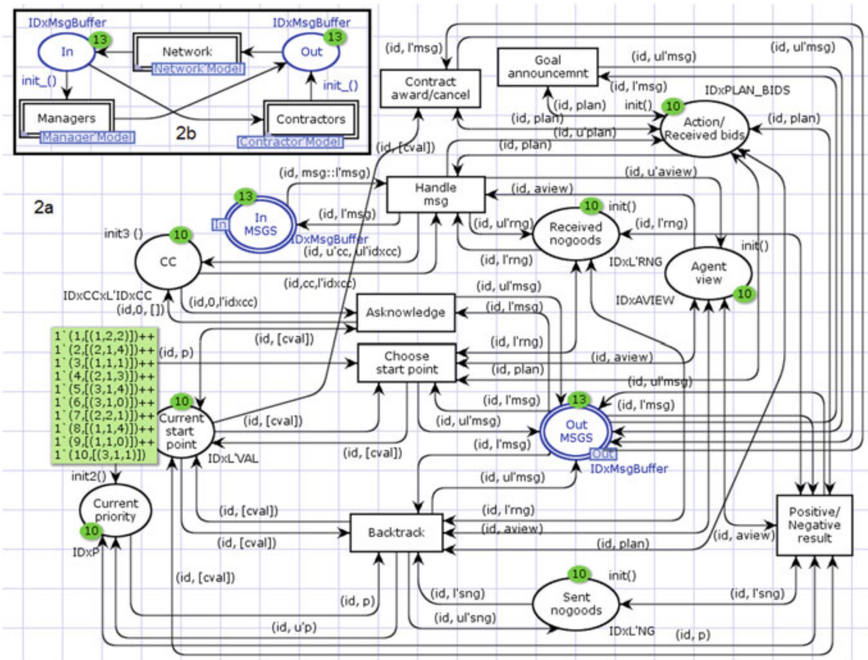


Fig. 2 CPN model for a holonic system that combines CNP and DisCSP

Backtracking, and Weak Commitment Search [21]. A distinct issue for the application of distributed algorithms is the detection of termination moment [22]. In the case of DisCSP, there are two types of endings: one for the case when a problem has no solution (this is detected by the appearance of the empty *nogood* [21]) and the other for a problem with solutions. For this last case, a specific mechanism should be implemented. In our approach, we developed an extension of the Dijkstra-Scholten algorithm [22]. Briefly, this supposes for an agent to count the number of messages used to announce its neighbours about assigning a value for its variable (let this number be A), and the number representing messages of acknowledgement received from agents having a lower priority (let this number be B). When the difference $A - B$ becomes zero, it means all agents having a lower priority are in a stable state and thus the agent itself will issue a message of acknowledgement towards its neighbour with higher priority. As a special case, the agent with the lowest priority sends a message of acknowledgement to those agents that announced it about their established value, as soon as this value is consistent with its agent view. When the agent with the highest priority obtains the difference $A - B$ equal to zero, it means all agents are in a stable state and the solution of problem was found. This mechanism is materialized through the CPN by the place *CC* and the transition *Acknowledge*. In the place *CC* the values A and B are kept, while transition *Acknowledge* is fired if $A - B$ is zero. A received message of acknowledgement is treated by the transition *Handle msg*. Determining the termination for the DisCSP based negotiation is important, because this is the moment when the HMS should start its execution stage (see Fig. 1).

About the CNP part, in the CPN model the following places and transitions are used: *Goal announcement* and *Contract award/cancel* for the corresponding events, and the place *Action/Received bids* that stores states of actions (an action may or not have assigned its actor), and bids. Anticipating the next section, as an implementation detail, the number of tokens depends on the number of actions from manager holons' plans. That is why in Fig. 2 there are ten tokens in the places of an agent model (corresponding to the total number of actions). The values to be assigned during scheduling/rescheduling through DisCSP regard the starting points for actions, their durations being known from the bids received from resources. In the places *In* and *Out* of the upper layer there are thirteen tokens, because the three tokens for resource holons are included, too.

4 A Case Study with Different Solutions for Rescheduling

The case study used in this paper was inspired by the one presented in [23]. While in [13, 23] three manufacturing commands are used, we considered that a new order appears (handled by holon M4 in Fig. 3). Each command is managed by an order/product holon and the plan for its solving is a priori known, with the actions marked as a_i in Fig. 3. The arrows represent ordering constraints, while edges

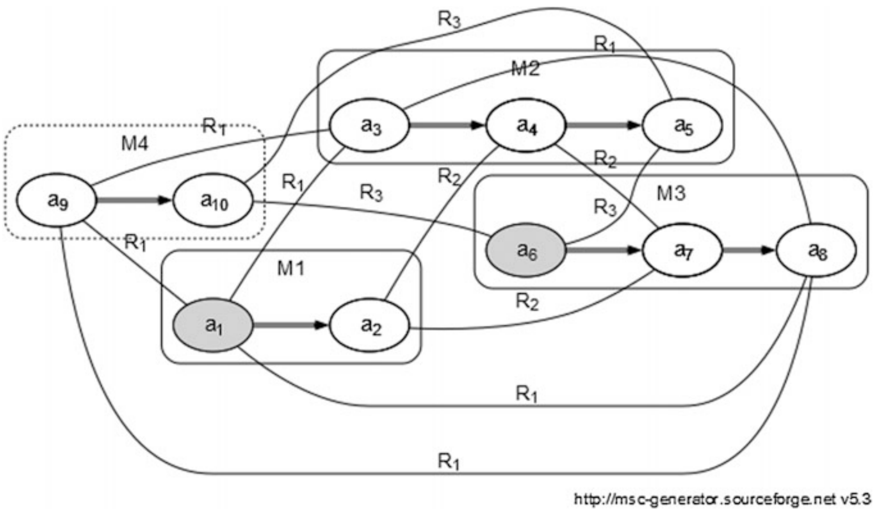


Fig. 3 Plans and constraints for manager holons

regard overlapping constraints because they refer to actions that can be executed by the same resource holon, but not in the same time.

In our experiment, we initially considered that the managers M1 ÷ M3 begin their activity at the batch level and through the coordination scheme they arrive to the schedules presented in Fig. 4, which is an optimum solution from their point of view (the actions from one manager's plan are marked with the same type of colour). One may notice that actions a_1 and a_7 need two time units, while all other actions need one time unit. It is to remark that for each action type there is a single resource that can achieve it. Thus, for each goal a single contractor will be able to provide a bid and in such cases a simple bidding mechanism, as the one provided by the common CNP, is adequate. We suppose that at a time moment within the interval [0 1] manager M4 enters the system with its manufacturing goal. In this situation, a_1 and a_6 are ongoing actions and they will not be interrupted. For the case when the new manager has a lower priority in comparison with the managers already working, the obtained schedules are the ones of Fig. 5. One can notice that this is an optimum schedule with respect to the makespan, but manager M4 is the

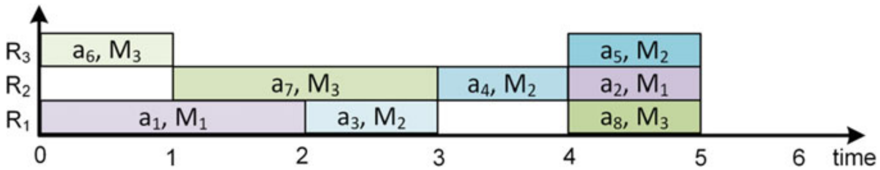


Fig. 4 Schedules for three manager holons

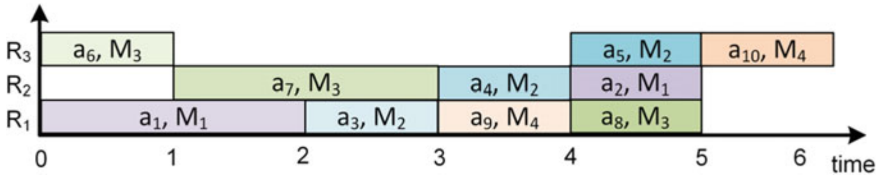


Fig. 5 Rescheduling for the new order of lower priority

last one that solves its goal. By comparing Figs. 4 and 5, it is to remark that the schedules of the other managers are not disturbed.

If manager M4 is the most important, then the obtained schedules are displayed in Fig. 6. One can see that M4 plans its first action (a_9) as soon as possible, namely after the end of a_1. The negotiation between managers determines holons M1 and M2 to change their schedules (M1 makes only a partial change, while M2 has a new schedule), and M3 does not make any change. Holon M4 ends its order the first, and now M2 will be the last. For comparison, Fig. 7 presents the schedules that are obtained if all managers would begin their activity at the same time, keeping the highest priority for M4. It is to notice that in this case a better solution is possible, rescheduling determining a makespan longer with one time unit, due to the constraint of not interrupting a running action.

To materialize this rescheduling method for the case when the new order is of the highest importance, in the DisCSP phase the Asynchronous Backtracking algorithm (ABT) is used, with the new manager having assigned the lowest priority. This salience regards ABT and not the priority of managers and in this way for the new introduced holon the place *Agent view* will be first updated with the values chosen by the other agents, as a consequence of the message *New scheduling*

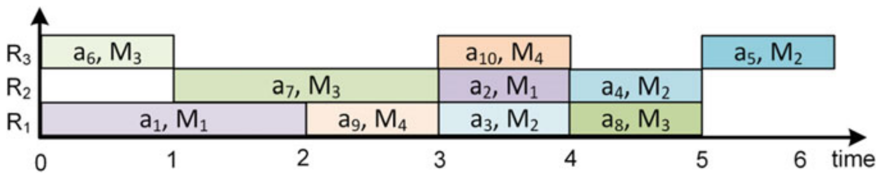


Fig. 6 Rescheduling for the new order of higher priority

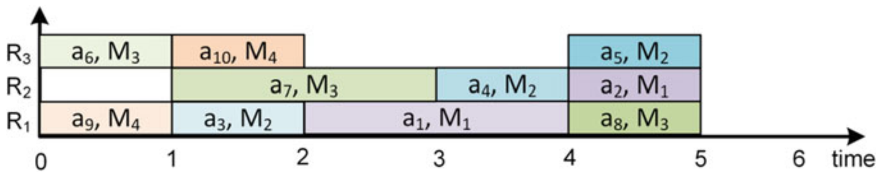


Fig. 7 Optimum schedules for all orders being received from the beginning

Table 1 Results of simulation experiments obtained with the CPN model—random choice of values

		Without a higher priority for M4			With a higher priority for M4		
		Min	Max	Avg.	Min	Max	Avg.
a.	#msgs	59	126	88.82	56	116	81.417
	#cycles	16	37	25.949	13	29	19.58
	#backs	6	13	8.589	3	10	5.361
	#solutions/time	0	24.617 s		0	19.534 s	
b.	#msgs	45	125	73.381	47	142	83.674
	#cycles	10	37	20.396	11	35	20.064
	#backs	0	8	2.22	0	9	3.582
	#solutions/time	30	24.041 s		6	22.715 s	
c.	#msgs	45	150	90.866	48	158	96.736
	#cycles	10	48	26.498	11	39	22.739
	#backs	0	15	5.799	0	12	4.74
	#solutions/time	21	27.051 s		1	25.458 s	

announcing (see Fig. 1). After having the *Agent view* filled in, the new manager applies specific constraints trying to minimize the starting time for its actions, without violating the constraints that ongoing actions must remain unchanged. If the new order is not of a higher priority, the same mechanism is applied, except for the constraints on minimizing the starting time for the new manager's actions, which are no longer used.

The model allowed us through the CPN Tool to make series of 1000 simulation experiments, the results being displayed in the next tables. These results regard only the DisCSP based negotiation phase, the one being the most important for coordination. The experiments were considered for two strategies regarding the way an agent chooses a value. Thus, in Table 1 the case when agents randomly choose their values is presented. The experiments were carried out in two stages. First, it was tried to obtain a solution with the new manager while keeping the same makespan of 5 time units. As Table 1a shows, in this case no solution could be obtained, both in the case when manager M4 is of a higher priority or not. Table 1 shows the communication load (number of messages), computation load (number of cycles and backtracks) and the total time to run the 1000 simulations. If the makespan is increased with one time unit, the results of Table 1b are obtained. One can notice that the model revealed the existence of 30 distinct solutions (schedules) if the new manager is of the same priority with the others, while 6 solutions appear if the new manager is of higher priority. We can conclude that the total computation/communication effort is obtained by adding the values of Table 1a and 1b, because the HMS will first try to obtain a solution in the minimum makespan and then iteratively increase this. For comparison, Table 1c shows the results for

Table 2 Results of simulation experiments obtained with the CPN model—greedy choice of values

		Without a higher priority for M4			With a higher priority for M4		
		Min	Max	Avg.	Min	Max	Avg.
a.	#msgs	106	126	110.462	75	125	100.975
	#cycles	30	40	33.98	17	30	24.433
	#backs	10	13	10.532	4	10	7.146
	#solutions/time	0	31.98 s		0	26.739 s	
b.	#msgs	110	119	114.334	108	114	111.016
	#cycles	31	40	35.72	27	30	28.458
	#backs	5	7	5.886	6	7	6.417
	#solutions/time	2	38.155 s		1	32.886 s	
c.	#msgs	130	147	135.669	116	162	133.239
	#cycles	39	49	43.06	25	37	29.942
	#backs	10	14	11.478	5	9	6.56
	#solutions/time	1	40.778 s		1	36.768 s	

the case when all managers begin planning and scheduling in the same time. In this case, if manager M4 is with the highest priority only one solution exists for the makespan of 5 time units.

In Table 2 the results for the same set of experiments are displayed, but for the case when agents choose their values in a greedy way; this means that each agent tries first the best possible value, i.e., the value for the minimum starting point of its action. One can remark that for all the three cases (a, b, and c), this strategy conducts the search towards less solutions and with increased computation and communication loads. This can be explained by the fact that in this strategy agents make the search only within a certain area of the solution space (thus less of the possible solutions are found) and it appears a greater chance to make a wrong choice; one can see the increased number of backtracks in Table 2 in comparison with Table 1. In this way, the number of messages is increased due to the additional backtracking events, as well as the computational load is amplified in accordance with an increased number of decisional cycles. This behaviour is specific for this case study. The conclusion we can derive is that the random choice of values within the DisCSP phase is to be preferred for scheduling because it determines a more efficient solution search.

The developed CPN model allows the simulation of different behaviours with respect to the order between distinct events. Thus, the results of Tables 1 and 2 were obtained for the case when agents take decisions in a rush way. Namely, agents react after each received message and their decision can be taken based on an un-updated system state; this appears when agents with a higher priority have already changed their values and these were not taken yet into account by the current agent. This behaviour can be changed by increasing the priority of transition *Handle msg* (see Fig. 2) and thus the decisional phase of an agent's activity will be done after all received messages are handled. This system behaviour would be an ideal one, although difficult to obtain in real operation.

5 Conclusion and Future Work

This paper shows promising results for obtaining adaptability through rescheduling in HMSs if coordination is carried out by combining the CNP and DisCSP. Our method implies an iterative process, i.e., the solution is obtained with a repeated increasing of the makespan; even so, the mechanism is practical if few variables are involved due to the small time needed for an iteration; for example, according to the results presented in Tables 1 and 2 a simulation experiment lasts less than one second. A main point for the proposed coordination scheme is the negotiation between manager holons, and about this different strategies can be implied, so that the managers' priorities to be changed in accordance with the manufacturing context. Moreover, if a holon is endowed with a library of constraints then its behaviour could be changed on-line, with an appropriate switching mechanism that determines the set of constraints to be involved.

The CPN model is useful in showing the computational and communication complexity; it allows the correctness of method to be tested, and new hypotheses to be checked by adjusting the model and constraints. A CPN model is important because it permits tests dealing with distributed systems and this can be done before launching the operation on real equipment.

As future work, we intend to test some other DisCSP algorithms in the negotiation among managers and to check the efficiency of the proposed coordination scheme on more complex problems. Moreover, the method is to be further tested for the cases when there are more resources that can achieve the same action and in such situations auction mechanisms more refined than CNP should be evaluated.

References

1. Katragjini Prifti, K., Vallada Regalado, E., Ruiz García, R.: Flow shop rescheduling under different types of disruption. Int. J. Prod. Res. **51**(3), 780−797 (2013)
2. Novas, J.M., Van Belle, J., Saint Germain, B., Valckenaers, P.: A collaborative framework between a scheduling system and a holonic manufacturing execution system. In: Borangiu, T. h., Thomas, A., Trentesaux, D. (eds.) Service Orientation in Holonic and Multi Agent Manufac. and Robotics, p. 3–17. Springer (2013)
3. Marin, C.A., Mönch, L., Leitao, P., Vrba, P., Kazanskaia, D., Chepegin, V., Liu, L. Mehandjiev, N.: A conceptual architecture based on intelligent services for manufacturing support systems. In: 2013 IEEE International Conference on Systems, Man, and Cybernetics, pp. 4749–4754 (2013)

4. Lanza, G., Stricker, N., Moser, R.: Concept of an intelligent production control for global manufacturing in dynamic environments based on rescheduling. In: 2013 IEEE International Conference on Industrial Engineering and Engineering Management, pp. 315–319 (2013)
5. Borangiu, T., Morariu, C., Morariu, O., Drăgoicea, M., Răileanu, S., Voinescu, I., Militaru, G., Purcărea, A.A.: A Service Oriented Architecture for total manufacturing enterprise integration. In: International Conference on Exploring Services Science, pp. 95–108. Springer (2015)
6. Varvara, G.: Service architecture for CSP based planning for holonic manufacturing execution systems. In: Borangiu, T.H., Dragoicea, M., Nóvoa, H. (eds.) Exploring Services Science, LNBIP, vol. 247, pp. 403–416. Springer (2016)
7. Zalmiyah, Z., Petrovic, S.: Genetic algorithms for match-up rescheduling of the flexible manufacturing systems. Comput. Ind. Eng. **62**, 670–686 (2012)
8. Godinho Filho, M., Barco, C.F., Tavares Neto, R.F.: Using genetic algorithms to solve scheduling problems on flexible manufacturing systems (FMS): a literature survey, classification and analysis. Flex. Serv. Manuf. J. **26**(3), 408–431 (2014)
9. He, N., Zhang, D.Z., Li, Q.: Agent-based hierarchical production planning and scheduling in make-to-order manufacturing system. Int. J. Prod. Econ. **149**, 117–130 (2014)
10. Borangiu, T.H., Răileanu, S., Roşu, A., Pârlea, M., Anton, F.: Management of changes in a holonic manufacturing system with dual-horizon dynamic rescheduling of production orders. IFAC Proc. **42**(4), 624–629 (2009)
11. García-Mata, C.L., Márquez-Rodríguez, P.R., Burtseva, L.: Rescheduling in Industrial Environments: Emerging Technologies and Forthcoming Trends. Int. J. Comb. Optim. Prob. Inform. **6**(3), 34–48 (2015)
12. Pănescu, D., Pascal, C.: Holonic coordination obtained by joining the contract net protocol with constraint satisfaction. Comput. Ind. **81**, 36–46 (2016)
13. Pascal, C., Pănescu, D.: On applying DisCSP for scheduling in holonic systems. In: International Conference on System Theory, Control, and Computing (ICSTCC), Sinaia, Romania, 13–15 Oct (2016)
14. Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P.: Reference architecture for holonic manufacturing systems: PROSA. Comput. Ind. **37**(3), 255–274 (1998)
15. Pascal, C., Panescu, D.: HAPBA—A BDI Agent based solution for holonic manufacturing execution systems. In: Borangiu, Th., Thomas, A., Trentesaux, D. (eds.) Service Orientation in Holonic and Multi Agent Manufacturing and Robotics, pp. 57–70. Springer (2013)
16. Pănescu, D., Pascal, C.: On a holonic adaptive plan-based architecture: planning scheme and holons' life periods. Int. J. Adv. Manuf. Technol. **63**(5), 753–769 (2012)
17. Pănescu, D., Pascal, C.: HAPBA—a Holonic Adaptive Plan-Based Architecture. In: Borangiu, Th., Thomas, A., Trentesaux, D. (eds.) Service Orientation in Holonic and Multi-Agent Manufacturing Control, pp. 61–74. Springer (2012)
18. Panescu, D., Pascal, C.: A constraint satisfaction approach for planning of multi-robot systems. In: Proceeding of International Conference on System Theory, Control, and Computing (ICSTCC), Sinaia, Romania, pp. 157–162 (2014)
19. Pascal, C., Pănescu, D.: A petri net model for constraint satisfaction application in holonic systems. In: IEEE International Conference on Automation, Quality and Testing, Robotics (AQTR), Cluj-Napoca, Romania (2014)
20. Jensen, K., Kristensen, L.M.: Coloured Petri Nets: modeling and Validation of Concurrent Systems. Springer, New York (2009)
21. Yokoo, M., Durfee, E., Ishida, T., Kuwabara, K.: The distributed constraint satisfaction problem: formalization and algorithms. IEEE Trans. Knowl. Data Eng. **10**(5), 673–685 (1998)
22. Fokkink W.: Distributed Algorithms: An Intuitive Approach. MIT Press (2013)
23. Bongaerts, L., Monostori, L., McFarlane, D., Kádár, B.: Hierarchy in distributed shop floor control. Comput. Ind. **43**(2), 123–137 (2000)

Customisation in Manufacturing: The Use of 3D Printing

Rengarajan Srinivasan, Vaggelis Giannikas, Duncan McFarlane and Mudassar Ahmed

Abstract An increasing demand to provide customised products creates challenges for manufacturing organisations. This poses a need to understand the characteristics required for manufacturing systems to handle customisation. In this study, 3D printing technology is assessed as an enabler for customisation. Additionally, the requirements of manufacturing systems with respect to configuration and control co-ordination are explored. A demonstrator is implemented to integrate 3D printing with conventional manufacturing, using an agent based distributed control system that co-ordinates the customisation of products and the order management.

Keywords 3D printing · Mass customization · Service orientation

1 Introduction

Dynamic changes in customer requirements and preferences are driving the needs for highly customised products [1]. This customer-oriented focus allows organisations to stay competitive in the global market and to satisfy varying customer demands. This requires manufacturers to deliver high variety of products, individually designed at a low cost [2].

Nevertheless, the need for a high degree of customisation creates disruptions in the manufacturing system and poses problems in scheduling. As a result, manu-

R. Srinivasan · V. Giannikas (✉) · D. McFarlane · M. Ahmed
Distributed Information and Automation Laboratory, Institute for Manufacturing, University of Cambridge, 17 Charles Babbage Road, Cambridge, UK
e-mail: v.giannikas@eng.cam.ac.uk

R. Srinivasan
e-mail: rs538@eng.cam.ac.uk

D. McFarlane
e-mail: dcm@eng.cam.ac.uk

M. Ahmed
e-mail: ma603@eng.cam.ac.uk

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_19

facturing systems are required to be flexible and resilient to disruptions [7]. Additionally, customisation leads to small batch sizes and requires frequent and dynamic re-configuration of the manufacturing system. This creates additional burden on the control system to dynamically alter process steps and change tools, routing and material handling.

Therefore, a manufacturing system should be able to handle rapid changes and product variety utilising flexible resources and intelligent control.

In this paper, the ability of 3D-printing as an enabler to handle customisation is explored. The configurations to integrate 3D printing technology with conventional manufacturing systems are proposed. A distributed agent based control system is implemented in a laboratory system to demonstrate the applicability of 3D printing in handling customisation in a conventional manufacturing system.

2 Manufacturing Systems Supporting Mass Customization

Mass customisation can be defined as the ability to deliver a wide range of products that meet specific requirements of individual customers, at a cost equivalent to mass production [1, 2]. Generally, the ability to provide mass customisation requires manufacturers to have flexible resources that can handle product variety and have the ability to change over quickly. In this section we review such requirements and we argue on the suitability of 3D printing to support mass customisation.

2.1 Handling Customisation in Manufacturing

In order to handle the issues related to customisation, the following characteristics of the manufacturing system are required [1, 2, 5]:

R1 *Customer driven manufacturing*: The customer needs to be involved in the specification, design and manufacture of the customised parts. Generally, two options exist here. Firstly, the customer can specify the requirements and preferences of the product and the manufacturer focuses on design and production. Alternatively, the customer can himself design the product or parts and provide the design to the manufacturer to produce the product. In both instances, the manufacturing system should have the ability to handle this.

R2 *Integration of design and manufacturing systems*: Involvement of customers in the design process requires seamless integration of design and manufacturing systems to quickly transition from design to manufacture. The decoupling point of order, the point where customer influences the manufacturing process, varies depending on the level of customisation or customisation offered. This requires better integration of Computer Aided Design (CAD) and

Computer Integrated Manufacturing (CIM), which will allow in ease of customised manufacturing.

R3 *Flexibility*: The manufacturing system should be flexible enough to make a wide range of product variations and the resources should have multiple capabilities to handle this. Furthermore, the manufacturing system should have process/material flow flexibility to handle various customisation aspects.

R4 *Management of inventory of great variety*: High degree of customisation creates a wide range of product variety and this leads to increasing levels of inventory to tackle the uncertainty with regards to customer preferences. Therefore, the manufacturing systems should have the ability to minimize inventory levels whilst handling customisation requests.

R5 *Management of customised orders*: Customisation leads to varying order due-dates and reduced batch sizes. Additionally, the customer orders might arrive at random times. This requires the manufacturing system to have the ability to manage the orders by rescheduling dynamically and to have the capability to handle rush orders.

In order to address these requirements, the following are needed:

- Flexible resources with capability to make or handle customised parts.
- Manufacturing system configuration able to utilise the flexible resources.
- Control system for co-ordinating customised orders and flexible resources.

In this paper, 3D printers enabling rapid, additive manufacturing are evaluated as an example of a flexible resource for handling customisation issues. This is discussed more in the following section. Subsequently, we discuss the configuration and co-ordination issue in the section that follows.

2.2 3D Printing Technology Enabling Mass Customisation

In this section, the suitability of rapid manufacturing, especially 3D printing technology, for the management of customisation disruptions is assessed. Rapid manufacturing is defined as the use of CAD-based automated additive manufacturing process for making parts that can be used as a finished product or as a component that can be assembled into a final product [6]. 3D printing technology is a form of an additive manufacturing process, where products are manufactured by adding layers of materials [3].

With regards to the first requirement (*R1*) the ability of 3D printing machines to directly utilise 3D models of designed products allows the customer to design his preferences related to products directly [4]. Additionally, the customers can co-design products or choose the design by other customers in a marketplace. This allows for high degree of customisation. Additionally, 3D laser scanners can be used to map the products to get a digital design of the product directly, which enable the customers to design customisation aspects directly.

Furthermore, the 3D design models from the customers can be directly transferred to the 3D printer for manufacturing.

The integration of design and manufacturing system requirement (*R2*) is addressed by 3D printing as CAD drawings can be directly imported or converted into appropriate instructions for additive manufacturing automatically. This allows quick transition from design to manufacture and provides rapid customisation capabilities.

As regards flexibility (*R3*), 3D printing based manufacturing removes the tooling requirements and thereby allows components of any geometry to be manufactured in a single resource without too much change over time [4, 8]. 3D printing eliminates the need for having a wide range of tooling and the associated costs. Furthermore, multiple materials can be combined to produce a part, rather than products or parts made of homogeneous materials.

Additionally, 3D printing allows for small batch sizes and there is no change over of tools, thus providing flexibility to cater for various customisation requests. Similarly, 3D printing based manufacturing offers the possibility of reducing inventory levels of customised parts as some of them can be produced on demand based on actual customer orders (*R4*).

2.3 Configurations and Co-ordination for 3D Printing-Based Customisation

There are a number of different configurations that allow the usage of 3D printers in manufacturing depending upon the level of customisation required. These different configurations will require different co-ordination capabilities to control and manage customisation requests. Three possible configurations are depicted in Fig. 1.

The first configuration illustrated in Fig. 1 allows 3D printed parts and components produced by printers belonging to different organisations to be used in the production of a product. These 3D printed parts are manufactured in different

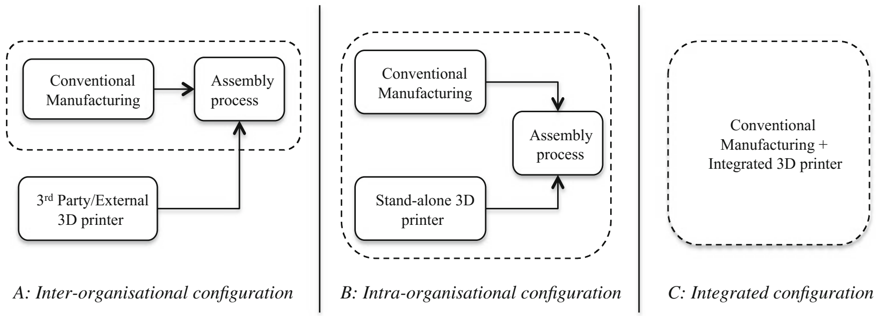


Fig. 1 Configurations for the usage of 3D printing in manufacturing

geographical locations and need to be transported to the location of the main manufacturer of a product for final assembly. The co-ordination needed in this case refers to the assignment of a printing job to external organisations and to the physical transportation of a part in a manufacturing line.

The second configuration describes the case where the conventional manufacturing line and the 3D printer belong to the same company. Here, conventional and flexible resources are located in the same geographical area and they are managed by the same company. However, the 3D printer is used only for making customised parts. The co-ordination aspects in this configuration are similar to the ones of the first configuration although the manufacturer will have greater control over both processes.

The third configuration depicts an integrated approach where 3D printers are part of a production line along with other resources (e.g., robots) and tools. It illustrates the case where 3D printers have seamlessly been integrated in existing manufacturing processes and are considered part of the overall manufacturing system. In this configuration, the 3D printer is also involved in the making of conventional parts as part of the manufacturing system. One of the main issues here is introducing 3D printing resources in existing systems and controlling them along with other manufacturing resources.

In order to examine the introduction of 3D printing in conventional manufacturing in more detail, a demonstrator was developed. The demonstrator uses the intra-organisational configuration as a configuration that allows a manufacturer to use 3D printing without the need to outsource to external parties (inter-organisational configuration) and without the need for significant changes in the existing manufacturing line (integrated configuration). This demonstrator is described next.

3 Demonstrator

In this section, we describe the developed demonstrator as well as our key findings in terms of meeting the customisation requirements discussed in Sect. 2.

3.1 Set-up

The production system used in the demonstrator produces a gearbox (Fig. 2) consisting of: (i) a metal casing made of two parts, top and base, (ii) a plastic top cover, (iii) gears which go into the casing (metal or customisable plastic), (iv) (optional) customisable cap. The customers have two options for customisation. First, the customer can have a choice of different coloured gears with different gear ratios. Additionally, the customers can prefer to have a custom made cap for the gearbox with added text.

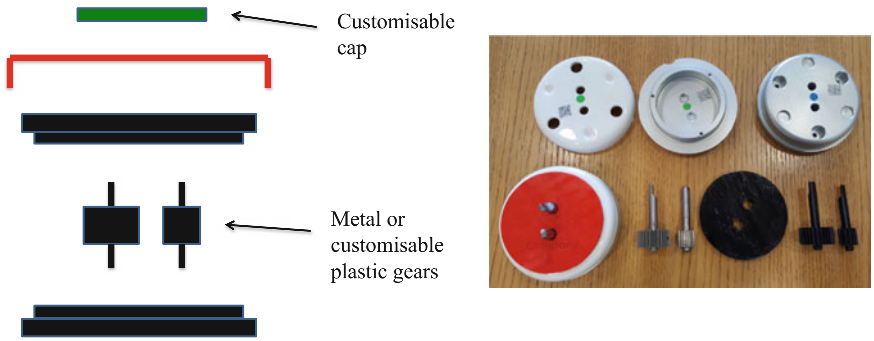


Fig. 2 Gear box

The production system for manufacturing the gear box consists of three cells. Cell 1 is a manufacturing cell, where the metal casing is machined by a 5-axis CNC machine and the plastic part is formed by a vacuum forming machine. Cell 2 is the sub-assembly process where the metal top and the plastic cover are aggregated. Cell 3 is the final assembly cell associated with gear meshing and fastening operations. The customised parts (i.e., gears and cap) are printed and delivered in Cell 3 by a standalone 3D printer.

3.2 *Implementation*

This section illustrates how the three key essential features required for handling customisation, discussed previously are implemented.

Firstly, a 3D printer is used as the flexible resource to make the customised parts (i.e., gears and caps), which delivered to Cell 3 and are integrated into the final product. The use of 3D printing allows easy transition from design to manufacture and offers additional flexibility by removing the change-over time between products. Furthermore, the use of 3D printers eliminates the need to have customised parts being maintained as inventory.

With regards to configuration, Fig. 3 shows the one implemented in the demonstrator. The 3D printer is implemented as a stand-alone resource that is dedicated to making customised parts only (i.e., intra-organisational configuration). Cell 1 and cell 2 are conventional manufacturing systems which deliver the metal base and the sub-assembly (metal top and plastic cover) to Cell 3. The 3D printer delivers customized parts to cell 3 to be integrated into final assembly.

For co-ordination between the conventional manufacturing and the 3D printer, a distributed control system was implemented. The distributed control was based on multi-agent systems and was implemented using the JADE framework (see Fig. 3). The agents and their roles in co-ordination are:

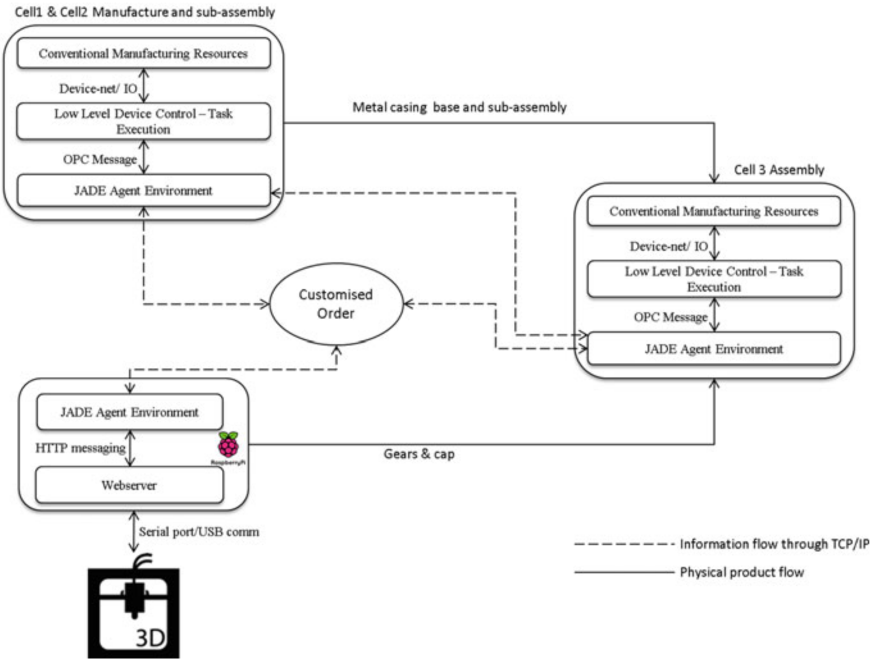


Fig. 3 Demonstrator configuration and control

- *Order agent*: Responsible for negotiating the task allocation with the various manufacturing resources. Additionally, the orders will have the customisation requests associated with a design file. The order agent then negotiates and reserves the job with the 3D printer.
- *Customisation agent*: This agent represents the 3D printer and is responsible for managing customisation request and for scheduling the printing operations based on order due dates.
- *Resource agents*: These agents represent the typical manufacturing resources such as CNC machine and robots for material handling. These agents are responsible for managing task allocation and order management.

The stand-alone 3D printer is implemented as an intelligent resource capable of making its own decision with respect to customisation and scheduling. This is enabled by having a Raspberry Pi computer integrated with the 3D printer. The integrated computer then has the capability to receive and process messages from the orders, and also control the additive manufacturing process.

Table 1 Customisation requirements implemented in demonstrator

Requirements	Implementation
Customer driven manufacturing	The customers can decide the choice of options for the gears and ratios, along with the optional customised caps. The manufacturer has the required design in CAD which is then transferred to the 3D printer for printing
Integration of design and manufacturing systems	3D printer is directly linked to the manufacturing system. Orders are associated with the required design information and are transferred directly to the 3D printer, which converts the design into appropriate commands for printing the parts
Flexibility	3D printer can print both customised gears and caps from the same resource, thereby offering flexibility to use a single resource for all customisation needs
Management of inventory of great variety	The ability to make parts quickly and on request eliminates the need to hold inventory. Additionally, the customised parts can be produced from a single material, thereby reducing the complexity and level of inventory needed
Management of customised orders	Intelligent orders and resources having distributed decision making ability allow the orders to negotiate and schedule tasks independently. This provides flexibility to handle small batch sizes and re-schedule dynamically to cater for rush orders on customisation

3.3 Meeting Customisation Requirements

We conclude this section with a discussion on the way the five customisation requirements identified in Sect. 2 have been met in our demonstrator. These are summarised in Table 1.

4 Conclusions

In this paper we investigated the suitability of 3D printing to handle customisation needs. 3D printing was chosen as a technology providing flexible and rapid manufacturing capabilities. Our analysis shows that the technology can indeed be used for enhancing the customization capabilities of conventional manufacturing systems. Nevertheless, the integration of 3D printing with conventional manufacturing systems poses challenges in:

- **Automation**: Lacks ability to automatically transfer materials in and out of the 3D printer. In-process sensing of product quality is not well developed.
- **Communication**: Existing 3D printers have different communication interfaces (e.g., Ethernet, serial port) and are not inter-operable with standard manufacturing devices (e.g., PLC).

- **Interfaces**: There is limited monitoring of control interfaces and task execution in 3D printers. Ability to control and communicate with 3D printers varies between manufacturers (open source vs. proprietary tools). Furthermore, the conversion of CAD files into machine codes is not standardized.

References

1. Da Silveira, G., Borenstein, D., Fogliatto, F.S.: Mass customization: literature review and research directions. Int. J. Prod. Econ. **72**(1), 1–13 (2001)
2. Fogliatto, F.S., da Silveira, G.J.C., Borenstein, D.: The mass customization decade: an updated review of the literature. Int. J. Prod. Econ. **138**(1), 14–25 (2012)
3. Gao, W., Zhanga, Y., Ramanujan, D., et al.: The status, challenges, and future of additive manufacturing in engineering. Comput. Aided Des. **69**, 65–89 (2015)
4. Hague, R., Campbell, I., Dickens, P.: Implications on design of rapid manufacturing. Proc. Inst. Mech. Eng. Part C: J. Mech. Eng. Sci. **217**(1), 25–30 (2003)
5. Hart, C.W.L.: Mass customization: conceptual underpinnings, opportunities and limits. Int. J. Serv. Ind. Manag. **6**(2), 36–45 (1995)
6. Hopkinson, N., Hague, R., Dickens, P.: Rapid manufacturing: an industrial revolution for the digital age, Wiley (2006)
7. Srinivasan, R., McFarlane, D., Thorne, A.: Identifying the requirements for resilient production control systems. In: Service Orientation in Holonic and Multi-Agent Manufacturing, pp. 125–134, Springer (2016)
8. Tuck, C., Hague, R., Ruffo, M., Ransley, M., Adams, P.: Rapid manufacturing facilitated customization. Int. J. Comput. Integr. Manuf. **21**(3), 245–258 (2008)

Analysing the Impact of Rescheduling Time in Hybrid Manufacturing Control

Jose-Fernando Jimenez, Gabriel Zambrano-Rey, Abdelghani Bekrar, Damien Trentesaux and Paulo Leitão

Abstract Hybrid manufacturing control architectures merge the benefits of hierarchical and heterarchical approaches. Disturbances can be handled at upper or lower decision levels, depending on the type of disturbance, its impact and the time the control system has to react. This paper focuses particularly on a disturbance handling mechanism at upper decision levels using a rescheduling manufacturing method. Such rescheduling is more complex that the offline scheduling since the control system must take into account the current system status, obtain a satisfactory performance under the new conditions, and also come up with a new schedule in a restricted amount of time. Then, this paper proposes a simple and generic rescheduling method which, based on the satisfying principle, analyses the trade-off between the rescheduling time and the performance achieved after a perturbation. The proposed approach is validated on a simulation model of a realistic assembly cell and results demonstrate that adaptation of the rescheduling time might be beneficial in terms of overall performance and reactivity.

Keywords Rescheduling time · Manufacturing control · Reactivity

J.-F. Jimenez · A. Bekrar · D. Trentesaux
LAMIH, UMR CNRS 8201 University of Valenciennes and Hainaut Cambrésis UVHC, Le Mont Houy, 59313 Famars, France
e-mail: abdelghani.bekrar@univ-valenciennes.fr

D. Trentesaux
e-mail: damien.trentesaux@univ-valenciennes.fr

J.-F. Jimenez (✉) · G. Zambrano-Rey
Pontificia Universidad Javeriana, Bogotá, Colombia
e-mail: j-jimenez@javeriana.edu.co

G. Zambrano-Rey
e-mail: gzambrano@javeriana.edu.co

P. Leitão
Polytechnic Institute of Bragança, Bragança, Portugal
e-mail: pleitao@ipb.pt

P. Leitão
LIACC—Artificial Intelligence and Computer Science Laboratory, Porto, Portugal

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_20

1 Introduction

Manufacturing control architectures have evolved from the traditional centralised and/or rigid hierarchical configurations to more flexible ones based on partial or full heterarchies. While the first ones are focused on optimality, the second ones are based on highly autonomous and cooperative entities, making decisions closer to the process in order to react to unexpected changes of conditions [1]. However, such reactivity, fault tolerance and dead-lock avoidance provided by heterarchical relationships require sacrificing optimality. In fact, since the 1940s, Hebert Simon [2], a pioneer in artificial intelligence, proposed a satisfying principle where models based on satisfactory or good enough decisions work much better than laboriously optimal ones. Moreover, current studies are focused on demonstrating that modelling aiming at providing satisfactory decisions may result in more sustainable, reliable and realistic models [3–5]. Particularly, hybrid control architectures (HCA), also known as semi-heterarchical architectures, provide such balance between satisfactory efficiency and reactivity, since industrials require more realistic models to reduce such a gap [6].

In HCAs, global decisional entities e.g., supervisors, schedulers, coordinators, are in charge of supervising, generating and/or coordinating pre-execution schedules with updated information, focused on one or more production objectives. Afterwards, schedules are shared with local entities which, depending on their level of autonomy, execute and keep track on these schedules [7–9]. Finding good enough schedules is highly constrained by the complexity of the scheduling problem, the required amount of information, and the limited amount of time to come up with a decision. Hence, most of HCAs advocate for artificial intelligence techniques such as metaheuristics, agent orientation, holonic or bio-inspired approaches, or a combination of these with classic mathematical formulations to ensure a certain level of reactivity [10–12].

The disruption handling in HCA's dynamics can be entirely entitled on local decision entities or shared between global and local entities. Out of these two, the latter stands out because of its benefits in terms of overall performance and responsiveness [13]. In such a case, the global entity takes care of the rescheduling of undone tasks. However, if the rescheduling time is too long and perturbations are quite frequent, this configuration may have serious consequences on the overall performance. Two insights to reduce the risk of inefficient rescheduling are: (i) to avoid fully centralized scheduling as proposed in [8], meaning partial centralized scheduling, and (ii) to analyse the impact of the rescheduling time so that this time can become a kind of governance parameter [14] for the system and be adjusted depending on the system configuration and production status; thus it evolves and adapts to new conditions.

This paper focuses on analysing the impact of the rescheduling time when rescheduling is managed jointly by global and local decisional entities. However, assuming a predictive offline and a reactive online scheduling, herein the analysis is only made on the rescheduling executed by a single global decisional entity while the local entities just take such a schedule as an input to complete it with local decisions. In this case, the time spent for the local decision is being neglected.

This paper is organised as follows. Section 2 briefly describes the rescheduling concept. Then, Sect. 3 performs a generic method to analyse the rescheduling time impact as a rescheduling strategy. A case study based on flexible manufacturing system is presented in Sect. 4 to validate the proposed strategy. Conclusions and further research opportunities are highlighted at the end.

2 Rescheduling in HCA

In HCAs, the global entity starts generating optimal or satisfactory predictive schedules based on information coming from the manufacturing execution system (MES) and the current state of the shop floor. Afterwards, when the schedule execution starts, the reactive phase takes care of disturbances using two possible approaches: (i) classic scheduling repair, which may include a complete rescheduling or (ii) switching to a fully reactive configuration in which local entities are autonomous to make online schedules [7, 15] as shown in Fig. 1. The main benefit of complete/partial scheduling repair is that it re-evaluates the conditions of the system and optimises the performance given the restrictions and degradation caused by disruption. However, besides the need of defining a strategy of execution while the schedule is being recalculated, it has synchronization issues when the solution is going to be applied [16]. The main benefit of fully reactive configurations is the instantaneous reaction and thus the capability of the system to deal with various types of perturbations, dropping the rescheduling time to zero and passing to online/real-time rescheduling. Unfortunately, given the myopic behaviour of

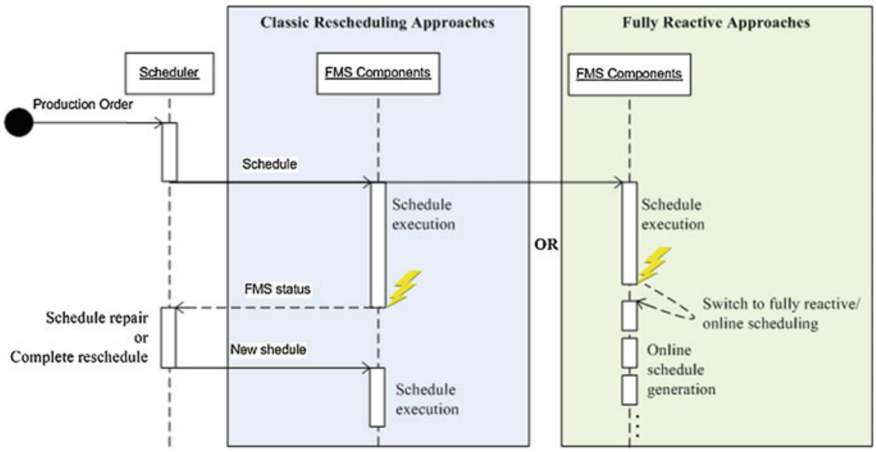


Fig. 1 Two strategies to respond to online perturbations

local entities, it becomes hard to predict the system's overall performance for the remaining production time [17].

Scheduling repair modifies only the affected portion of the original schedule. It was proposed to limit changes to the original schedule and also to reduce the rescheduling time. For instance, ElMaraghy and ElMekkawy [18] proposed a rescheduling algorithm that uses time Petri-Nets and the minimal siphons concept to deal with machine breakdowns in real-time. Similarly, Ahmadi et al. [19] used a right-shift rescheduling method, and Zakaria and Petrovic [20] employed a genetic algorithm based on a match-up approach to modify only a part of the initial schedule within a rescheduling horizon when a change occurs. When complete rescheduling is executed, it is important to ensure the applicability of the schedule by obtaining up to date information to conceive the new schedule. To do so, Novas et al. [21] proposed a collaborative framework between centralised scheduling and holonic manufacturing execution systems (HMES). The role of the HMES is to provide rightful information, in time, to centralised scheduling and use the new schedule as guidance for holon decisions. Although the authors mentioned that the updated schedule has to be as fast as possible, rescheduling time is not addressed.

The scheduling strategy has been widely addressed by researchers looking for balancing the schedule performance and stability. Event-based, periodically, or hybrid-driven strategies have been reported in literature [22]. Hybrid-driven strategies synthesize advantages of event-driven and periodic-driven because it cannot only deal with unexpected events but it also maintains a certain stability of the system. The scheduling horizon sets the rescheduling frequency and is intended to reduce complexity by dividing a wide scheduling range into small segments, which reduces the rescheduling time as reported by H. A. ElMaraghy and T.Y. ElMekkawy [18]. For instance, in this study, the author sets the rescheduling resolution to 1 s. In [23], the rescheduling time is limited as a fraction of setup and processing times and low complex heuristics are used to respect such constraint.

The rescheduling point also impacts the rescheduling time since it determines what needs to be scheduled and when. For instance, if the breakdown occurs nearly at the end of the considered scheduling horizon, low complex techniques can be used because there will be no much impact on scheduling performance. However, if the disturbance occurs right after creating the schedule, in the normal (re)scheduling point or in the middle of the scheduling horizon, it is advisable to apply a more complex technique, as proven by Pfeiffer et al. [24]. Fattahi and Fallahi [25] also studied the impact of the rescheduling point for job arrival disturbances, having the best results when efficiency and stability objectives are taken into consideration. As it has been already mentioned, an analysis on the rescheduling time has not been particularly addressed and most of the studies just mention that it must be short enough compared to processing times. In fact, in some studies, this time is ignored [26]. Since this is not necessarily true and the rescheduling times affect the quality of the schedule, in the next section a generic strategy to analyse the rescheduling time in flexible manufacturing systems (FMS) is proposed. Among the different manufacturing configurations, the FMSs were chosen because of the heterogeneity of the manufacturing resources and their high flexibility, which imply the high complexity of such systems. This strategy is intended to work with a HCA in which a global entity (scheduler) manages rescheduling, either repair or complete rescheduling, and local entities take care of local decisions (i.e., transport between machines).

3 Proposal to Analyse the Rescheduling Time in FMS

From an Operations Research perspective, scheduling in FMS is a more complex version of the classical flexible job-shop scheduling problem, which is known to be NP-hard [27]. This complexity is caused by versatile manufacturing lines, redundant and reconfigurable machines, alternate routings, and flexibility in operation sequencing [28]. Basically, FMS scheduling consists of ordering products for dispatching, allocating each operation to a machine out of a set of capable machines, and sequencing the assigned operations on all machines in order to obtain a feasible schedule. These decisions are taken in the predictive phase (① in Fig. 2), being possible to estimate a certain production time (*EPt* in Fig. 2), and/or other production indicators if needed. Once the scheduler finishes, the order is released into the FMS to follow the schedule (② in Fig. 2). If a disruption arrives (③ in Fig. 2), the rescheduling process is triggered. At first, the scheduler needs the current FMS and product status (④ in Fig. 2) to launch the rescheduling technique (⑤ in Fig. 2). Then, a new schedule is ready for execution either until a new disruption arrives and rescheduling is again needed, or the production order is processed (⑥ in Fig. 2). As a result of disruption(s), the actual production time (*APt* in Fig. 2) differs from the *EPt* in a called lateness time *Lt* indicator; it is possible to determine the impact of the disruption, the capacity of the system to absorb it and the impact of the rescheduling time.

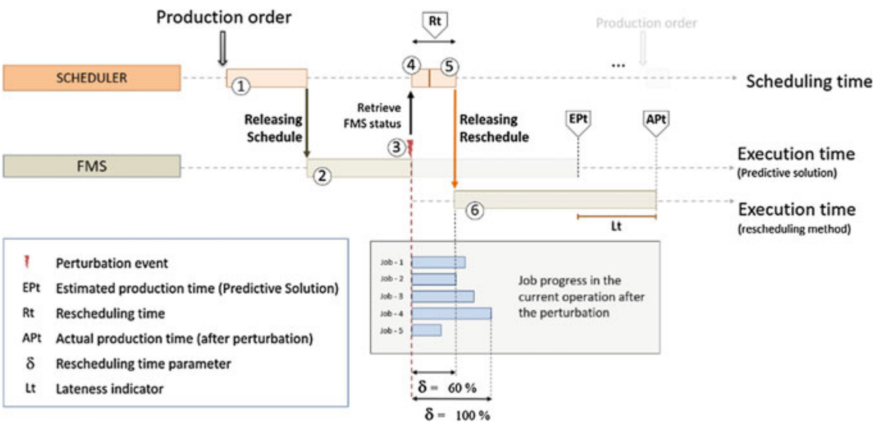
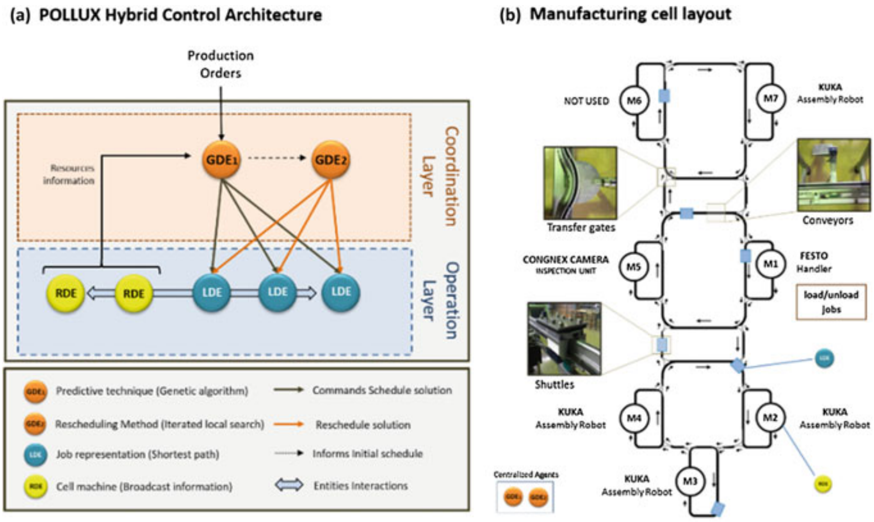


Fig. 2 Rescheduling calculation time analysis

The strategy proposed herein to analyse the rescheduling time is based on a rescheduling time parameter than can be used to limit the time the scheduler spends recalculating a new schedule or repairing it. Normally, when a disruption arrives, jobs are in the middle of a task, e.g., manufacturing, transportation, inspection, queuing, etc.; so when a job finishes the current task, it would be ideal to have the new schedule ready. Not having the schedule ready results in idleness, hence a waste of time and energy. For example, as seen in Fig. 3, J_5 is the job with the shortest remaining time, so if the new schedule were ready right after J_5 finishes, then there would be no idleness at all. In such case, the rescheduling time parameter, denoted by δ, is 40% and rescheduling may run only until J_5 finishes its current task. The opposite scenario happens when the scheduler takes the longest job remaining time, J_4 in Fig. 2. Consequently, δ is set at 100%, meaning that all jobs have reached idleness waiting for the new schedule. Then, the scheduler must monitor the number of jobs reaching idleness and stop rescheduling when such count equals the following expression:



(c) Operations for each job (Letter)

Code	Letter	Operations Sequence
901	B	07 - 01 - 01 - 01 - 02 - 02 - 05 - 03 - 06 - 08
902	E	07 - 01 - 01 - 01 - 02 - 02 - 04 - 06 - 08
903	L	07 - 01 - 01 - 01 - 05 - 05 - 03 - 03 - 06 - 08
904	T	07 - 01 - 01 - 02 - 04 - 06 - 08
905	A	07 - 01 - 01 - 01 - 02 - 04 - 05 - 03 - 06 - 08
906	I	07 - 01 - 01 - 05 - 03 - 06 - 08
907	P	07 - 01 - 01 - 02 - 04 - 06 - 08

(d) Processing times at each machine

	Processing times per machine (Seconds)				
Operation	M1	M2	M3	M4	M5
Operation 1 (O1)		20	20		
Operation 2 (O2)		20	20		
Operation 3 (O3)				20	
Operation 4 (O4)		20		20	
Operation 5 (O5)			20	20	
Operation 6 (O6)					5
Operation 7 (O7)	10				
Operation 8 (O8)	10				

Fig. 3 **a** Architecture of the case illustrated, **b** layout of the manufacturing cell, **c** operations sequence for each job, and **d** processing times of each operation per machine

$$no.jobs\ reaching\ idleness = \delta * no.of\ unfinished\ jobs \tag{1}$$

The number of unfinished jobs means the number of jobs that have not been discharged. Generally speaking, small δ values aim at reactivity while greater δ values aim at better global performance. In addition, δ values can be greater than 100% if needed. Thus, the maximum rescheduling time for idleness, i.e., the longest job remaining time, is taken as reference. For instance; if the remaining time of J_4 in Fig. 2 is 25 s, a δ value of 150% would be around 37 s. After the 25th second, all jobs are idle and waiting for the new reschedule for 12 s. This particular case may arrive when the rescheduling problem is very complex and a certain level of performance must be ensured.

As presented in Fig. 2, δ can become a parameter that needs to be setup, either statically or preferably in a dynamic way, for instance depending on the rescheduling technique, scope, strategy, rescheduling point and type of perturbation. Then, the rescheduling time parameter can be part of an adaptive FMS control. To validate the proposed strategy, the next section presents an experimental study carried out by simulating a realistic FMS.

4 Illustration of the Proposed Rescheduling Method

This section illustrates the inclusion of the proposed rescheduling method into a previously proposed HCA named Pollux [29]. It shows a flexible manufacturing system based on the real flexible assembly cell presented in Trentesaux's benchmark [30]. The manufacturing cell consists of four (4) partially redundant machines (M2, M3, M4, M7), one loading/unloading station (M1) and an automated inspection unit (M5) connected through a conveyor system. It can process seven different jobs ('B', 'E', 'L', 'T', 'A', 'I', 'P') composed of a unique operation sequence of a subset of 8 operations. The manufacturing problem of a given order to address is the jobs dispatching to the cell, the machine allocation for each job and the route path of each job through the conveyor system.

The hybrid control architecture model that contains the proposed rescheduling method is based on the Pollux reference control system [29] and is customised to manage the manufacturing operations within the defined FMS. The model is divided into two layers: the coordination layer, responsible for job dispatching and machine allocation, and the operation layer responsible for path routing. The coordination layer contains two global decisional entities, named GDE_1 and GDE_2. While GDE_1 is a decisional entity that executes the offline scheduling through a predictive decision-making approach, GDE_2 is a decisional entity responsible for the rescheduling process. Located in the operation layer, the local decisional entities or LDEs are responsible for guiding the jobs in the cell and coordinating the online scheduling either when the instruction are imposed by GDE_1 in the scheduling phase or by GDE_2 in the rescheduling phase. For this reason, the number of LDEs corresponds to the number of jobs in the production order. Figure 3 illustrates the architecture of the control model and the manufacturing cell layout.

Predictive scheduling (GDE_1): Even though the studied FMS can be resolved as a Flexible Job-Shop Scheduling Problem (FJSP) in Optimization software, the reported research solves this problem through a hybrid technique using a genetic algorithm (GA) and simulation-optimization. The representation of chromosomes in the GA is a schedule that specifies the job dispatching and machine sequence to run the production execution. The GA chromosome is divided into two parts. The first part is an array where each position represents a job to process and the value in this position defines the order to be released into the cell. The second part is also an array in which each position corresponds to a job and contains a sub-array with a sequence of machines to follow. An example of a chromosome (the representation of which is also used in rescheduling) for two jobs is

$$Chromosome_i = \{[T, P][(M_1, M_2, M_2, M_3, M_4, M_5, M_1), (M_1, M_7, M_7, \; M_3, M_4, M_5, M_1)]\}.$$

While the selection in the GA is done by a tournament selection, the crossover and mutation are separately executed for each part of the chromosome by a one-point and an integer randomization method, respectively. The fitness function used in this instantiation is the makespan of the production order. For this, this paper presents a simulation-optimization technique because the evaluation of the fitness function is evaluated in a simulation of the manufacturing cell. A simulation model of the FMS studied is programmed in NetLogo agent-programming software [31].

Even though NetLogo is designed to simulate agents' environments, this paper uses the commands and report features of this software to run both the genetic algorithm and evaluate the fitness function by its simulation. After this process is executed, the best chromosome with minimum makespan is used for execution.

Rescheduling method (GDE_2): the rescheduling method, which is activated when a disruption is detected (e.g., machine breakdown, urgent order arrival,…), is executed by the GDE_2.

In this research, an *iterated local search* limited to fulfil a Satisfying principle was used. The iterative local search (ILS) is a hill-climbing method for discrete optimisation problems that improves searches over discrete variables. For this, starting from an initial solution it explores in a reduced space (called neighbourhood of the solution) by following a single chain of explicit set moves [32].

ILS is built as follows: when a disruption is detected, the process starts by retrieving the state of the schedule during execution and setting the initial schedule and representation as a current solution of the ILS algorithm. Then, in an iterated search, it starts improving the current solution in the two parts of the solution. In the first part, its swaps two positions of the solution changing the dispatching order. In the second part, it changes randomly the machine allocation according to the feasibility in the redundant machines.

Certainly, these moves consider the reparation of the solution to correct the consequences of the disruption and are limited to the jobs and operations not processed in both the job dispatching and the machine allocation.

Reactive scheduling (LDE$_i$): The reactive scheduling is executed by each LDE. The LDE has two different states according to the manufacturing environment. On one side, when the intensions are imposed by GDE_1 or GDE_2, the LDE receives the instruction of the job dispatching and machine allocation and executes the processing of jobs accordingly.

In addition, in the path routing decision, the LDE uses a shortest path algorithm in order to get to the next machine. On the other hand, when a disturbance occurs, the LDE is in charge of finalising the on-going task (i.e. manufacturing, transporting, queuing, or waiting to be released) and passing to a stand-by state where the jobs loops within the cell until new instructions are received (GDE_2 instructions).

For this case study, the disruption handling of the proposed HCA is analysed by comparing different values for the rescheduling time (δ) parameter after a fixed perturbation (Fig. 4).

For this, four scenarios were tested as defined by Trentesaux et al. [30]: B0 (2xAIP), C0 (1xAIP and 1xBELT), D0 (1xAIP and 3xBELT) and E0 (3xAIP, 2xBELT and 1xLATE). The disruption considered in the experiment is the breakdown of resource M_3 and is fixed for all scenarios at 50 s after execution kick-off. The rescheduling time (δ) parameter is settled in percentage (%) and is tested from 50 to 400%; here, 100% represents the time elapsed from perturbation occurrence until all available jobs get to the stand-by state. The control model of the FMS was implemented in NetLogo on a PC Intel(R) Core(TM) i7-4770U CPU @ 3.40 GHz with 32.00 GB of RAM memory. Figure 5 illustrates the execution process for the C0 scenario (1 x AIP, 1xBELT).

As seen in Fig. 5, having a δ value of 100% for scenario B0 and 150% for other scenarios helps finding a better Cmax. In scenario B0, the perturbation arrives a

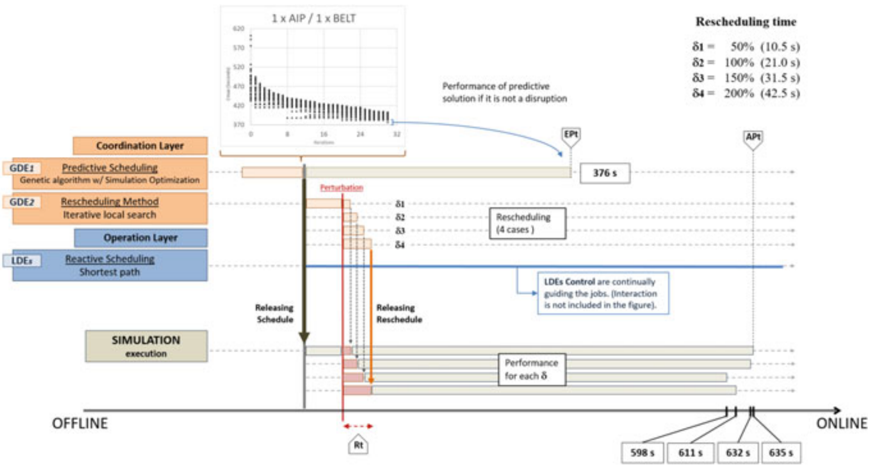


Fig. 4 Illustration of the rescheduling method (ILS) in a hybrid control architecture (Pollux) for scenario C0

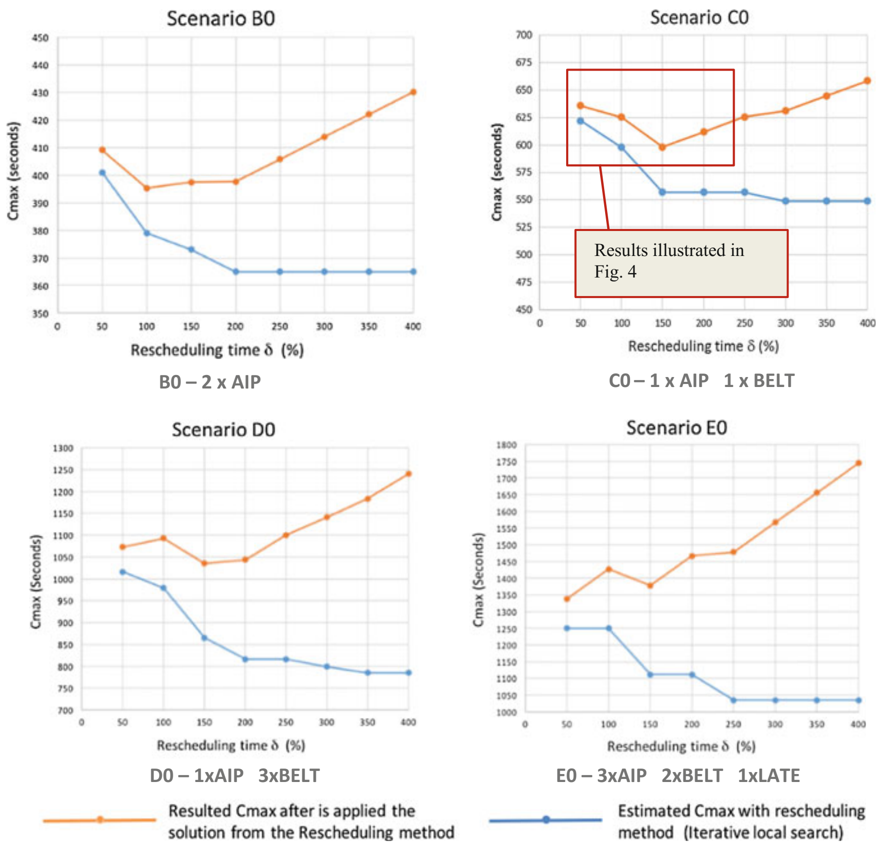


Fig. 5 Experimental results

t = 50 s and at t = 67 s all jobs have finished their tasks and the new schedule is ready. For the other scenarios, jobs finish their current manufacturing tasks and have to wait a certain amount of time for the new improved schedule. For instance, in scenario E0, the perturbation arrives at t = 50 s, all jobs finish their current tasks at t = 224 but a new schedule with δ = 150% is ready at t = 316 s.

From these preliminary results, it can be seen that actually a specific rescheduling time could be tuned that mitigates the degradation caused by certain perturbations. Even though by minimising the degradation caused by the perturbation and proving that it is worth to implement a satisfactory solution that is limited by a rescheduling time, some benefits are expected to be obtained in this research direction, it is clear that this needs to be extensively proven in different scenarios and modelling cases. For this reason, this study encourages our research to continue to explore the trade-off between the rescheduling time and the improvements achieved by the method.

5 Conclusions

Until now, in most works rescheduling times have been neglected and their impact on the performance indicators of manufacturing system's production has not been studied yet. Indeed, having a short rescheduling time it is needed to ensure reactivity but larger times aim at better performance. Hence, in this paper, a generic methodology was proposed to analyse the rescheduling times in terms of the current system conditions.

Using four different scenarios with increasing complexity, it was possible to show that this rescheduling time is an important parameter; moreover, it can become a dynamic parameter for the entire manufacturing control system. Therefore, the rescheduling time can be tuned up depending on the type of perturbation, current system state, maximum performance deviations, etc. Since for these particular cases an iterated local search heuristic was used to fix the schedule and find an alternative machine, future work will try to produce a thorough study using different rescheduling algorithms.

References

1. Dilts, D.M., Boyd, N.P., Whorms, H.H.: The evolution of control architectures for automated manufacturing systems. J. Manuf. Syst. **10**(1), 79–93 (1991)
2. Russel, S., Norvig, P.: Artificial Intelligence: A Modern Approach, 3rd edn. EUA Prentice Hall (2014)
3. De Lara, M., Martinet, V., Doyen, L.: Satisficing versus optimality: criteria for sustainability. Bull. Math. Biol. **77**(2), 281–297 (2015)
4. Hansson, O., Mayer, A.: The optimality of satisficing solutions. In: Proceedings of the 4th workshop on uncertainty in artificial intelligence (1988)
5. Cardin, O., Mebarki, N., Pinot, G.: A study of the robustness of the group scheduling method using an emulation of a complex FMS. Int. J. Prod. Econ. **146**(1), 199–207 (2013)
6. Mourtzis, D.: Challenges and future perspectives for the life cycle of manufacturing networks in the mass customisation era. Logistics Res. **9**(1), 1–20 (2016)
7. Pach, C., Berger, T., Bonte, T., Trentesaux, D.: ORCA-FMS: a dynamic architecture for the optimized and reactive control of flexible manufacturing scheduling. Comput. Ind. **65**(4), 706–720 (2014)
8. Zambrano-Rey, G.: Reducing myopic behavior in fms control, PhD Dissertation. University of Valenciennes and Hainaut-Cambrésis (2014)
9. Shen, W., Maturana, F., Norrie, D.H.: MetaMorph II: an agent-based architecture for distributed intelligent design and manufacturing. J. Intell. Manuf. **11**(3), 237–251 (2000)
10. Barbosa, J., Leitão, P., Trentesaux, D., Adam, E.: Enhancing ADACOR with biology insights towards reconfigurable manufacturing systems. In: IECON 2011-37th Annual Conference on IEEE Industrial Electronics Society, pp. 2746−2751 (2011)
11. Li, X., Zhang, C., Gao, L., Li, W., Shao, X.: An agent-based approach for integrated process planning and scheduling. Expert Syst. Appl. **37**(2), 1256–1264 (2010)
12. Nejad, H.T.N., Sugimura, N., Iwamura, K.: Agent-based dynamic integrated process planning and scheduling in flexible manufacturing systems. Int. J. Prod. Res. **49**(5), 1373–1389 (2011)
13. Vieira, G.E., Herrmann, J.W., Lin, E.: Rescheduling manufacturing systems: a framework of strategies, policies, and methods. J. Sched. **6**(1), 39–62 (2003)

14. Jimenez, J.F., Bekrar, A., Trentesaux, D., Rey, G.Z., Leitao, P.: Governance mechanism in control architectures for flexible manufacturing systems. IFAC-PapersOnLine **48**(3), 1093–1098 (2015)
15. Wong, T.N., Leung, C.W., Mak, K.L., Fung, R.Y.K.: Integrated process planning and scheduling/rescheduling—an agent-based approach. Int. J. Prod. Res. **44**(18–19), 3627–3655 (2006)
16. Jimenez, J.F., Bekrar, A., Trentesaux, D., Leitão, P.: A switching mechanism framework for optimal coupling of predictive scheduling and reactive control in manufacturing hybrid control architectures. Int. J. Prod. Res. 1−16 (2016)
17. Zambrano-Rey, G.Z., Bonte, T., Prabhu, V., Trentesaux, D.: Reducing myopic behavior in FMS control: A semi-heterarchical simulation–optimization approach. Simul. Model. Pract. Theory **46**, 53–75 (2014)
18. ElMaraghy, H.A., ElMekkawy, T.Y.: Deadlock-free rescheduling in flexible manufacturing systems. CIRP Ann. Manuf. Technol. **51**(1), 371–374 (2002)
19. Ahmadi, E., Zandieh, M., Farrokh, M., Emami, S.M.: A multi objective optimization approach for flexible job shop scheduling problem under random machine breakdown by evolutionary algorithms. Comput. Oper. Res. **73**, 56–66 (2016)
20. Zakaria, Z., Petrovic, S.: Genetic algorithms for match-up rescheduling of the flexible manufacturing systems. Comput. Ind. Eng. **62**(2), 670–686 (2012)
21. Novas, J.M., Van Belle, J., Saint Germain, B., Valckenaers, P.: A collaborative framework between a scheduling system and a holonic manufacturing execution system. In: Proceeding of SOHOMA Service Orientation in Holonic and Multi Agent Manufacturing and Robotics. pp. 3–17. Springer (2013)
22. Yu, G.D., Yang, Y., Zhao, X., Li, G.: Multi-objective rescheduling model for product collaborative design considering disturbance. Int. J. Simul. Model. **13**(4), 472–484 (2014)
23. Gao, K.Z., Suganthan, P.N., Tasgetiren, M.F., Pan, Q.K., Sun, Q.Q.: Effective ensembles of heuristics for scheduling flexible job shop problem with new job insertion. Comput. Ind. Eng. **90**, 107–117 (2015)
24. Pfeiffer, A., Kádár, B., Monostori, L., Karnok, D.: Simulation as one of the core technologies for digital enterprises: assessment of hybrid rescheduling methods. Int. J. Comput. Integr. Manuf. **21**(2), 206–214 (2008)
25. Fattahi, P., Fallahi, A.: Dynamic scheduling in flexible job shop systems by considering simultaneously efficiency and stability. CIRP J. Manufact. Sci. Technol. **2**(2), 114–123 (2010)
26. Gao, K.Z., Suganthan, P.N., Pan, Q.K., Tasgetiren, M.F., Sadollah, A.: Artificial bee colony algorithm for scheduling and rescheduling fuzzy flexible job shop problem with new job insertion. Knowl. Based Syst. (2016)
27. Conway, R.W., Maxwell, L., Miller, L.W.: Theory of scheduling. Dover
28. Shirazi, B., Mahdavi, I., Mahdavi-Amiri, N.: iCoSim-FMS: An intelligent co-simulator for the adaptive control of complex flexible manufacturing systems. Simul. Model. Pract. Theory **19** (7), 1668–1688 (2011)
29. Jimenez, J.F., Bekrar, A., Zambrano-Rey, G., Trentesaux, D., Leitão, P.: Pollux: a dynamic hybrid control architecture for flexible job shop systems. Int. J. Prod. Res. 1–19 (2016)
30. Trentesaux, D., Pach, C., Bekrar, A., Sallez, Y., Berger, T., Bonte, T., Leitao, P., Barbosa, J.: Benchmarking flexible job-shop scheduling and control systems. Control Eng. Pract. **21**(9), 1204–1225 (2013)
31. Wilensky, U.: NetLogo: Center for Connected Learning and Computer-based Modelling, Northwest. University, Evanston (1999)
32. Glover, F., Kochenberger, G.: Handbook of Metaheuristics 1 (2003)

Big Data Analysis to Ease Interconnectivity in Industry 4.0—A Smart Factory Perspective

Pedro Lima-Monteiro, Mafalda Parreira-Rocha, André Dionisio Rocha and Jose Barata Oliveira

Abstract Thriving and challenging market trends led to changes in the manufacturing industry. Production lines that need to adapt to customisable products on the fly emerged. By applying communication and sensors to the shop-floor, along with Industry 4.0 principles, this became a possibility. The growing amount of sensors led to an exponential boom of the amount of data available, creating the concept of Smart Factory. By applying Big Data Analysis to this data, it may be possible to optimise Smart Factories. There are technologies capable of doing this, even though only some are capable of guaranteeing Smart Factory requirements, such as real-time. A study of these technologies, based on SME's experts' opinion, is hereby presented to assess the most suitable ones to analyse Big Data in a Smart Factory environment.

Keywords Big data analysis · Smart factory · Industry 4.0 · Internet of things

1 Introduction

With the integration of more and more sensors into the traditional shop-floor, a new sort of factory as emerged. Smart Factory is a quite new concept that emerged from the German Program, Industry 4.0 [1]. It aims at exploiting the trend in electronics development, taking advantage of devices becoming smaller, faster, cheaper and

P. Lima-Monteiro (✉) · M. Parreira-Rocha · A.D. Rocha · J. Barata Oliveira
Faculdade de Ciências r Tecnologia, Departmento de Eng. Electrotécnica,
CTS, UNINOVA, Universidade Nova de Lisboa, 2829-516 Caparica, Portugal
e-mail: pedro.monteiro@uninova.pt

M. Parreira-Rocha
e-mail: mafalda.rocha@uninova.pt

A.D. Rocha
e-mail: andre.rocha@uninova.pt

J. Barata Oliveira
e-mail: jab@uninova.pt

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_21

mobile, and new convergent technologies such as automation, robotics, vision, multi-agent control and holonic organisation allowing to add intelligence to products and resources and to distribute intelligence in processes and production control. This so called fourth industrial revolution also intends to benefit from the worldwide growth of the IT infrastructure (WLAN, WAN, etc.) which is now capable of enabling the connection of any electronics to this infrastructure [2]. The usage of 3rd parties' technologies such as Cloud Services, along with the usage of deployment and innovation accelerators such as Internet of Things (IoT) allowed to integrate operations technologies (OT) into Smart Factory [3].

These smaller devices are allowing SMEs and industry players in general to come across with high amounts of data. In a market where the smallest enhancement on, for instance, energy consumption can make all the difference, companies are now attempting to transform this stored data into something useful. In this shifting scenario, there are already several European-funded projects that attempt to bridge the gap between existing production lines and the new Industry 4.0 concept of a Smart Factory. The European H2020 Programme Project openMOS is one of those. It aims at developing a generic manufacturing service bus that allows for existent machinery to provide relevant data to a Cloud that is responsible to perform data analysis for the best optimisation output.

In this context, this paper presents an evaluation of several tools for both data storage and data processing to be used in a Smart Factory environment, in its lower layer, the shop-floor, where the real-time requirements are extremely important. For this evaluation to be properly introduced, a state of the art is given in Sect. 2, followed by the explanation, implementation and obtained results of the method used to evaluate the technologies in Sect. 3. In Sect. 4, some conclusions and further work on this area will be presented.

2 State of the Art

This chapter presents a concise state of the art of smart factory, its relation to big data analysis methods and the available technologies to perform RTBDA (Real-Time Big Data Analysis).

2.1 Smart Factory

With the current market the companies are facing a huge challenging shift paradigm. In this sense the concept of Industry 4.0 [4] presented a fully new approach based on an industry environment fully connected and sharing information among all the components and smart entities. In this interconnected world, the traditional production plants are no long able to deal with the new value chain activities and intelligent and flexible suppliers, such as supply chain, water supply and energy supply [5]. Hence, in order to make a cost-effective interconnection among these entities and optimise the usage of each one it is really important to analyse the information extracted from each one.

All the shared information remains most of the time stored without being processed and without generating new knowledge to optimise the new factories to this new world. So, Data Analysis represents an important topic for factories and industry of today.

2.2 Big Data Analysis in Manufacturing

Big Data problems remote back to the late 90s when the first search engines began to deal with big amounts of information in their searches. The first solution came from Google, by creating its own File System and the MapReduce programming model. This algorithm consists of two stages: Mapping the keys in the dataset and Reducing them by coupling all elements with a given key together. A known framework to use this algorithm is Hadoop [6], one of the benchmarks in Big Data Solutions.

There are already several usages of Big Data Analysis in Manufacturing (Fig. 1). To be able to perform Big Data Analysis in the manufacturing environment, there are some requirements that must be met, real-time being the most important since one cannot expect an entire production line to wait few minutes for a decision.

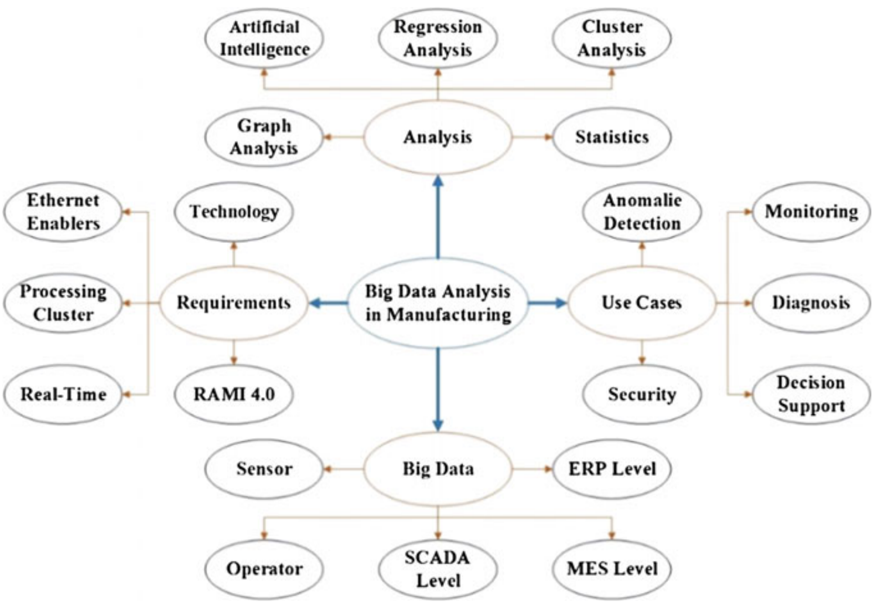


Fig. 1 Big data analysis in manufacturing diagram

In an attempt to solve this, a paradigm emerged: RTBDA, which aims at real-time processing of the data. However, to perform RTBDA, one must first define what real-time is. For the purpose of this research, real-time will be considered the "ability to process data as it arrives" [7]. It is also important to state the issues related with RTBDA: (1) it is important to guarantee that no bottlenecks occur on the production line [8]—by performing RTBDA: if the analysis of the data is not swift enough, the feedback may not reach the production line when it is required; (2) to apply learning methods in RTBDA, these mechanisms must be fast so that the learned knowledge to be put into context [9]. Concerning **Big Data Technologies**, these are varied and not all do the same nor have the same objective. They can be divided into three subjects: Data Storage, Data Processing, and Machine Learning.

Concerning **Data Storage Technologies**, the earlier years of data storage were marked by RDBMS (Relational Database Management System). With the growing amount of data, RDBMS's limitations became visible. It became impossible to process the amount of information; it is relatively easy to convert data to tables, when the amount of data is small; this does not turn out to be the case when the data amount grows exponentially [10]. A new type of database became imperative; this new type of databases [11] is suitable for distributed systems and parallel access. NoSQL databases are scalable and provide better performance rates. Some of the most famous are MongoDB [12], Cassandra [13] and HBase [14]. The turning point towards non-Relational databases occurred in 2007, when Amazon published an introduction to its Dynamo System [15].

In what concerns **Data Processing Tools and Engines**, tools are used by those who want to extrapolate new knowledge from their data. Frameworks are used by developers to extend their applications with data mining capabilities. Both use the MapReduce algorithm; this algorithm was implemented in the free Apache Hadoop framework. Some of these frameworks announcing to be capable of performing RTBDA are Apache's Spark [16], Flink [17] and Storm [18], among others. According to the original Spark's developer, it is capable of outperforming Hadoop by 10x to 100x [19].

Machine Learning frameworks are often considered as Data Processing Engines libraries. These are composed of learning algorithms, based on several natural behaviours, such as our immune system [20]. Despite one can use ML algorithms in a small set of data, the results will always be more trustworthy if the algorithms use broader sets [21]. Examples are Apache Mahout [22], Apache SAMOA [23] and MLlib [24]. ML frameworks can be used with several data processing engines.

3 Technologies Assessment

One presumption was assumed in order to apply the designed methodology: the model considers that the criteria are self-contained. The NoSQL Databases assessed in this research were: Cassandra (T_1), CouchDB (T_2), HBase (T_3), MongoDB

(T_4), Redis (T_5). The Data Processing Frameworks assessed were: Flink (T_1), Storm (T_2) and Spark (T_3).

To select the appropriate technologies to perform a specific task requires an analysis of their main characteristics. To do so, a five steps methodology was developed.

***Step 1*: Criteria definition and description**

These criteria will be used to classify each technology. Each criterion C_i, where $i \in \mathbb{N}$, was collected based on literature review. The criteria are defined as follows:

- NoSQL Databases: **Free Data Model** (C_1)—Freedom of choice on how the data is stored (tables, key-value pair; **Data Type** (C_2)—Defined data types, such as integer or float; **Proprietary Access** (C_3)—API to access it, for updates; **Partition Methods** (C_4)—Methods for distributed storage; **Replication** (C_5)—Replication to ease consistency insurance; **Consistency** (C_6)—Methods that enable consistency.
- Data Processing Engines: **Low Latency** (C_1)—Processing data fast; **Throughput** (C_2)—Being able to process big data at a time; **Streaming Model** (C_3)—Constant input and output of data; **Fault Tolerance** (C_4)—Methods of fault tolerance.

Each criterion is defined as "–" or "X" in Table 1 representing the existence or not of each specific feature. To apply the model, "–" or "X" are defined as "0" or "1".

***Step 2*: Relevance definition**

This step defines the importance each end user associates to each criterion. This factor is nominated as weight (W), and is defined as W_i, where $i \in \mathbb{N}$. This weight is defined by a scale, from 1 to 10. A group of end users (distributed along different industrial areas), defined as E_m, where $m \in \mathbb{N}$, were invited to answer a small questionnaire. For this specific case $m = \{1, \ldots, 6\}$. The end users' evaluation is presented in Table 2.

Table 1 Analysed technologies and the associated differentiation criteria

	NoSQL databases						Data processing engines		
	T_1	T_2	T_3	T_4	T_5		T_1	T_2	T_3
C_1	–	X	–	X	X	C_1	X	X	–
C_2	X	–	–	X	X	C_2	X	–	X
C_3	X	–	–	X	X	C_3	X	X	–
C_4	X	X	X	X	X	C_4	X	–	X
C_5	–	X	–	X	X				
C_6	X	–	–	X	–				

Table 2 Importance of the criteria for the end users

	NoSQL databases				Data processing engines		
	E_1	E_2	E_3		E_1	E_2	E_3
C_1	5	4	5	C_1	1	5	5
C_2	3	2	10	C_2	7	10	10
C_3	6	10	10	C_3	2	2	1
C_4	1	6	1	C_4	7	5	10
C_5	2	8	10				
C_6	4	7	10				

***Step 3*: Technology assessment**

The process is defined by (T_{Score_k}), where $k \in \mathbb{N}$. The evaluation factors are combined to create a score. Each technology is evaluated according to the W_i each end user gives to each criterion C_i. To proceed to the technology evaluation, Eq. 1 is applied:

$$T_{Score_k} = \frac{C_1W_1 + \cdots + C_nW_n}{\sum_{i=1}^{n} \max(W_i)} = \frac{1}{\sum_{i=1}^{n} \max(W_i)} \sum_{i=1}^{n} C_iW_i \quad (1)$$

where $n \in \mathbb{N}$.

The results from the use of Eq. 1 correspond to the evaluation of one technology by one end user.

So, in order to have a global validation of the technologies by all end users, it is necessary to aggregate each of their opinions. Table 3 summarizes the opinions of the respective end user for the evaluation of each technology.

***Step 4*: Data aggregation process**

The fourth step aims to set the aggregation method by defining two new factors: the average $(\bar{X}_{T_{Score_k}})$ and the Standard Deviation $(S_{T_{Score_k}})$ defined by Eq. 2.

$$\bar{X}_{T_{Score_k}} = \frac{\sum_{i=1}^{n} T_{Score_k}}{n} \wedge S_{T_{Score_k}} = \sqrt{S^2_{T_{Score_k}}} = \sqrt{\frac{\sum_{i=1}^{n} (T_{Score_k} - \bar{X}_{T_{Score_k}})^2}{n-1}} \quad (2)$$

The results are summarised by Table 4.

Table 3 Technology assessment (per end user)

	NoSQL databases				Data processing engines		
	E_1	E_2	E_3		E_1	E_2	E_3
T_1	0.23	0.42	0.52	T_1	0.43	0.55	0.65
T_2	0.13	0.30	0.27	T_2	0.08	0.18	0.15
T_3	0.02	0.10	0.02	T_3	0.35	0.38	0.5
T_4	0.35	0.62	0.77				
T_5	0.28	0.50	0.60				

Table 4 Data aggregation

	NoSQL databases			Data processing engines	
	$\bar{X}T_{Score_k}$	$S_{T_{Score_k}}$		$\bar{X}T_{Score_k}$	$S_{T_{Score_k}}$
T_1	0.39	0.14	T_1	0.54	0.11
T_2	0.23	0.09	T_2	0.13	0.05
T_3	0.04	0.05	T_3	0.41	0.08
T_4	0.58	0.21			
T_5	0.46	0.16			

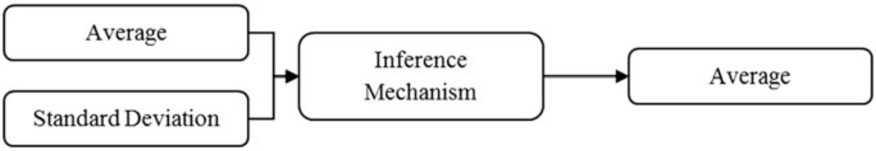


Fig. 2 Consensus-based model (adapted from [25])

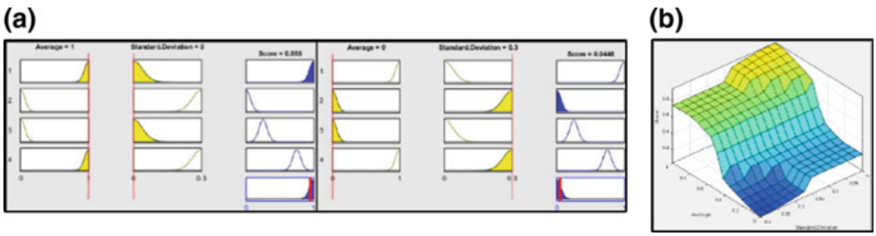


Fig. 3 **a** Extreme conditions' test, **b** behavioural test

Step 5: Ranking of the assessed technologies

The consensus based model results from the combined opinion of the end users. To rank the technologies a fuzzy inference system (FIS) is used (Fig. 2).

The FIS model is validated by the non-existence of any anomaly (Fig. 3).

The model is used to analyse the data in Tables 4 and 5 presents its results.

As can be seen in Table 5, NoSQL Database number 4 MongoDB, and Data Processing Engine number 1 Flink, were the preferred ones by the end users.

Table 5 Score generated by the consensus based model and technologies

	NoSQL databases			Data processing engines	
	Technology	Score (%)		Technology	Score (%)
1	T_4	66.0	1	T_1	64.0
2	T_5	36.0			
3	T_1	32.6	2	T_3	35.5
4	T_2	25.5			
5	T_3	25.0	3	T_2	25.0

4 Conclusions and Further Work

Current data technologies can now be used in a Smart Factory environment which features some restrictions, from which the most important are the real-time constraints. The technology assessment highlighted **MongoDB** for data storage and **Flink** for data processing as the ones which are more suitable for a Smart Factory environment.

However, a tool that is capable of performing this analysis in a generic and distributed way, whilst caring for real-time constraints, is still to be developed. Such a tool will be developed for the H2020 openMOS Project, in order to optimise the system, both energy and production wise. This tool will use the technologies that were hereby determined as the most suitable for a Smart Factory Environment.

Acknowledgements This research was supported by the openMOS (Open Dynamic Manufacturing Operating System for Smart Plug-and-Produce Automation Components) project of European Union's H2020. This document does not represent the opinion of the European Community, and the European Community is not responsible for any use that might be made of its content.

References

1. The Fourth Industrial Revolution, Foreign Affairs (2016). https://www.foreignaffairs.com/articles/2015-12-12/fourth-industrial-revolution. Accessed 24 June 2016
2. Bloem, J., van Doorn, M., Duivestein, S., Excoffier, D., Maas, R., van Ommeren, E.: The Fourth Industrial Revolution, Things Tighten (2014)
3. China Automotive Industry IT Application Market Forecast (2015–2019). http://www.idc.com, http://www.idc.com/getdoc.jsp?containerId=CHE40707015. Accessed 27 June 2016
4. Gilchrist, A.: Introducing Industry 4.0, in Industry 4.0, A press, pp. 195–215 (2016)
5. Seliger, G., Kohl, H., Mallon, J., Stock, T., Seliger, G.: 13th Global Conference on Sustainable Manufacturing—Decoupling Growth from Resource Use Opportunities of Sustainable Manufacturing in Industry 4.0, Procedia CIRP, vol. 40, pp. 536–541 (2016)
6. Shvachko, K., Kuang, H., Radia, S., Chansler, R.: The hadoop distributed file system. In: 2010 IEEE 26th Symposium on Mass Storage Systems and Technologies (MSST), pp. 1–10 (2010)
7. Barlow, M.: Real-Time Big Data Analytics: Emerging Architecture. O'Reilly Media, Inc. (2013)

8. Roser, C., Nakano, M.: A quantitative comparison of bottleneck detection methods in manufacturing systems with particular consideration for shifting bottlenecks. In: Umeda, S. et al. (eds.) Advances in Production Management Systems: Innovative Production Management Towards Sustainable Growth, pp. 273–281. Springer (2015)
9. Condie, T., Mineiro, P., Polyzotis, N., Weimer, M.: Machine learning for big data. In: Proceedings of the 2013 ACM SIGMOD International Conference on Management of Data, New York, NY, USA, pp. 939–942 (2013)
10. Jung, M.G., Youn, S.A., Bae, J., Choi, Y.L.: A study on data input and output performance comparison of MongoDB and PostgreSQL in the big data environment. In: 2015 8th International Conference on Database Theory and Application (DTA), pp. 14–17 (2015)
11. Moniruzzaman, A.B.M., Hossain, S.A.: NoSQL Database: New Era of Databases for Big data Analytics—Classification, Characteristics and Comparison, Cs (2013). arXiv:13070191
12. Chodorow, K.: MongoDB: The Definitive Guide. O'Reilly Media, Inc. (2013)
13. Wang, G., Tang, J.: The NoSQL principles and basic application of cassandra model. In: 2012 International Conference on Computer Science Service System (CSSS), pp. 1332–1335 (2012)
14. George, L.: HBase: The Definitive Guide. O'Reilly Media, Inc. (2011)
15. Leavitt, N.: Will NoSQL databases live up to their promise? Computer **43**(2), 12–14 (2010)
16. Shanahan, J.G., Dai, L.: Large scale distributed data science using apache spark. In: Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, New York, NY, USA, pp. 2323–2324 (2015)
17. Alexandrov, A., Bergmann, R., Ewen, S., Freytag, J.-C., Hueske, F., Heise, A., Kao, O., Leich, M., Leser, U., Markl, V.: The Stratosphere platform for big data analytics. VLDB J. **23** (6), 939–964 (2014)
18. Marz, N.: History of Apache Storm and Lessons Learned. Thoughts Red Planet (2014)
19. Zaharia, M., Chowdhury, M., Franklin, M.J., Shenker, S., Stoica, I., Spark, : Cluster computing with working sets. HotCloud **10**, 10–10 (2010)
20. Rocha, A.D., Monteiro, P.L., Barata, J.: An artificial immune systems based architecture to support diagnoses in evolvable production systems using genetic algorithms as an evolution enabler. Flex. Autom. Intell. Manuf. **25** (2015)
21. Landset, S., Khoshgoftaar, T.M., Richter, A.N., Hasanin, T.: A survey of open source tools for machine learning with big data in the Hadoop ecosystem. J. Big Data **2**(1), 1–36 (2015)
22. Eluri, V.R., Ramesh, M., Al-Jabri, A.S.M., Jane, M.: A comparative study of various clustering techniques on big data sets using apache mahout. In: 2016 3rd MEC International Conference on Big Data and Smart City (ICBDSC), pp. 1–4 (2016)
23. De Francisci Morales, G.: SAMOA: a platform for mining big data streams. In: Proceedings of the 22nd International Conference on World Wide Web, New York, NY, USA, pp. 777–778 (2013)
24. Meng, X., Bradley, J., Yavuz, B., Sparks, E., Venkataraman, S., Liu, D., Freeman, J., Tsai, D. B., Amde, M., Owen, S., Xin, D., Xin, R., Franklin, M.J., Zadeh, R., Zaharia, M., Talwalkar, A.: MLlib: Machine Learning in Apache Spark, Cs Stat (2015). arXiv:150506807
25. da Rocha, M,.P.S.V. et al.: Risk of employing an evolvable production system, MSc. Thesis, UNINOVA, Lisbon (2015)

Part V
Should Intelligent Manufacturing Systems be Dependable and Safe?

Application of Measurement-Based AHP to Product-Driven System Control

William Derigent, Alexandre Voisin, André Thomas, Sylvain Kubler and Jérémy Robert

Abstract This paper presents an application of the measurements-based AHP to define a two-stage algorithm for product-driven systems control, in case of an unexpected event. This algorithm is made of two stages: the first one aims at defining which kind of strategy the product should adopt (*wait*¸ *react by itself* or *switch back to centralized mode*) while the second one helps to choose the most appropriate resource able to fulfil the product requirements. The methodology is detailed on a simple case study.

Keywords Product-driven systems · Resource allocation · Measurement-based AHP

W. Derigent (✉) · A. Voisin · A. Thomas
Université de Lorraine, CRAN, UMR 7039, 2 avenue de la forêt de Haye,
Vandoeuvre-lès-Nancy, 54516 Cedex, France
e-mail: william.derigent@univ-lorraine.fr

A. Voisin
e-mail: alexandre.voisin@univ-lorraine.fr

A. Thomas
e-mail: andre.thomas@univ-lorraine.fr

S. Kubler · J. Robert
Interdisciplinary Centre for Security, Reliability & Trust,
University of Luxembourg, 4 rue Alphonse Weicker,
2721 Esch-sur-Alzette, Luxembourg
e-mail: sylvain.kubler@uni.lu

J. Robert
e-mail: jeremy.robert@uni.lu

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_22

1 Introduction

PDS (Product Driven Systems) can provide reactive solutions to unexpected events and significantly improve robustness and adaptation of local decisions on the shop floors. The products processed by a PDS are considered as intelligent in the sense of [1], i.e. "linked to information and rules governing the way [they are] intended to be made, stored or transported and capable to support or influence these operations" (for more information about intelligent products, the reader is advised to read the comprehensive review made by [2]). Products can thus make real-time decision in unexpected situations according to production status. In [3], the authors suggest that performance of product-driven control depends highly on the nature of the local decisions. Indeed, when nothing happens, the initial production plan is followed while products stay inactive. However, in case of unexpected events (in this paper, resource breakdowns), the product has 3 choices:

- *Wait and do nothing:* the product waits and does nothing until the resource is repaired;
- *React by itself* (*distributed mode*): the product tackles the emergency by itself;
- *Switch back to centralized mode*: the product switches back to the centralized control to ask the higher level to globally optimize the re-scheduling.

It is thus vital, at first, to be able to correctly select and switch from a strategy to another when appropriate. The product thus executes the following two-stage algorithm:

(1) **Stage 1: Select the strategy to use (wait, react by itself or ask for help);**
(2) **Stage 2 [only in the "react by itself" strategy]:**

   (a) Evaluate and rank the resources than could respond to its needs;
   (b) Select the most appropriate alternative among the ordered list of resources, using consensus strategy or negotiation protocols with other products/resources.

This two-stage algorithm is the subject of this paper. However, in this paper, the point (2b)—resource selection—is not detailed.

The problem of *strategy switching* (stage 1 of the algorithm) is addressed in [4], where a complete solution dedicated for mixed planning and scheduling is described. Under an unexpected event, they consider three different strategies: hierarchical (centralized) strategy, negotiated heterarchical (decentralized) strategy and non-negotiated heterarchical strategy. In their experiments, they compare the performances of each strategy on the makespan for a particular FMS platform, and conclude that the second strategy is the best for this system.

The problem of *resource evaluation and selection* (stage 2 of the algorithm) has been treated extensively via several theories and methods (goal programming, stochastic approaches, etc.). One interesting way to handle this problem is to employ Multi-Criteria Decision Making (MCDM) algorithms [4, 5]. Trentesaux et al. [6] argue that MCDM algorithms suit well to production activity control because they are interfaced with the human operator and can handle conflicting objectives that can arise in task reallocation (such as minimizing the lead time vs. the production costs). They can also support qualitative and quantitative criteria. Trentesaux et al. [6] uses indicators related to 3 criteria (*Lead Time*, *Quality* and *Cost*) and 7 indicators (like *Running Time*, *Move Time*, *Setup Time*, …) to evaluate resource via the ELECTRE method. In [7] the concept of 'potential' associated with a resource-machine as the basis for reassignment of tasks in case of breakdowns is defined. This potential is a value between 0 and 1, evaluated by means of three criteria (*Time*, *Cost* and *Reliability*). Each criterion comprises a certain number of indicators (among them, *Upstream Storage Cost* or *Reliability*).

In the present work, the Analytic Hierarchy Process (AHP) [8] is chosen as a MCDM technique. This choice is based on the simplicity and flexibility of AHP among multi-criteria decision making (MCDM) techniques. Indeed, AHP allows ranking a set of alternatives, here the finite set of resources, via the use pairwise comparisons matrices (see Sect. 2 for further explanations). Each alternative receives a 'potential' value (or *weight*). The 'potential' concept is closed to the concept of 'potential field' used for product/resource interaction, for dynamic product routing and task allocation in FMS [9, 10]. When arriving at a location *d,* the product uses the value of each resource potential field to select the resource the most suitable for its situation.

From this short state-of-the-art, some remarks can be done: (1) Even if the performances of the different execution strategies have been studied, the question of when switching from a strategy to another one is still an issue (2) Resource evaluation is related to the production context (location of the product, required operation, resource queue at time *t*, machine states, …). This very specific choice should be based on human knowledge, which makes MCDM techniques the most suitable ones because they are flexible, capable of handling a wide range of information, and overall interfaced with the human operator. In this regard, AHP, via its hierarchical structure, seems to be a very convenient alternative. However, it needs some adaptation to be able to handle dynamic data originating from the shop floor.

The rest of this document is organized in 3 sections. Section 2 introduces the mathematical background, needed to structure the different stages of the methodology. The AHP process as well as its adaptation for PDS is described. Section 3 presents in details both stages of the algorithm. Finally, in Sect. 4, an illustration of the proposed algorithm is done on a small case of study. Finally, some conclusions are provided.

2 Adaptation of AHP to PDS

AHP is a simple but cumbersome process, which is time-consuming mainly because the pairwise comparisons. Hence, when a breakdown occurs, each product being manufactured may need to find an alternative resource, autonomously, which is not possible with the classical AHP methodology. The main idea is then to modify the last level of the AHP structure, when alternatives are compared to each other, to use measurements made on resources. Indeed, all the other levels (criteria and sub-criteria) are the same for all the products, and supposed to be fixed by the human operator before the breakdown occurs (each week for instance). It represents the global context, which is supposed to be the same for all products. However, the choice of alternatives is highly contextual and the preferences of one product may not be similar to the ones of another product.

As a result, *Product preference functions* are introduced to transform parameters measured on resource into a preference scale. A product preference function is defined for each criterion as in Eq. 1:

$$\begin{aligned} f: R \quad &\mapsto [0;1] \\ x &\mapsto p \end{aligned} \tag{1}$$

where:

- x, the value measured on a resource;
- p, is the corresponding preference value between 0 and 1.

For each resource A_i, a preference value p_i is obtained, corresponding to the preference for **a given criterion** for **the considered product**. This value is then used to build the pairwise comparison matrix (PCM) of the resources as in Eq. 2:

$$PCM\,final\,level = \begin{array}{c} \\ A_1 \\ A_2 \\ \vdots \\ A_n \end{array} \begin{array}{c} \begin{array}{cccc} A_1 & A_2 & \vdots & A_n \end{array} \\ \begin{bmatrix} 1 & \frac{p_2}{p_1} & & \frac{p_1}{p_n} \\ \frac{p_2}{p_1} & 1 & & \frac{p_2}{p_n} \\ \vdots & & 1 & \vdots \\ \frac{p_n}{p_1} & \frac{p_n}{p_2} & \cdots & 1 \end{bmatrix} \end{array} \tag{2}$$

3 Description of the Proposed Algorithm

3.1 Stage 1: Strategy Switching

From a product's perspective, the objective of this first stage of the algorithm is to determine which actions should be undertaken at time t. In a first attempt, two parameters have been considered as relevant for strategy switching, i.e. *Residual Slack Time* (*RST*) and *Event Duration* (*ED*). The *RST* is the difference between the latest possible completion time of product production (the date which will not delay the completion of the overall command), and the earliest possible completion time.

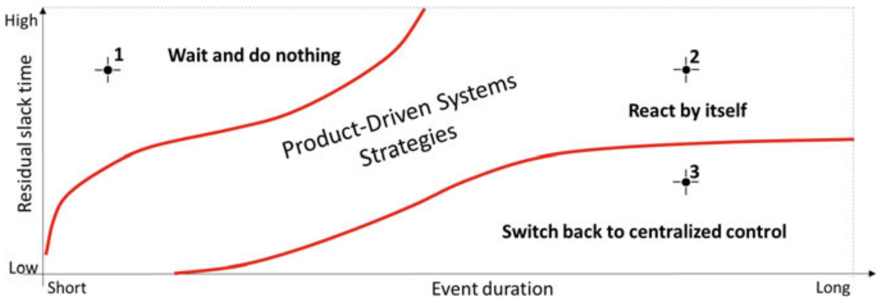


Fig. 1 Different strategy domains according to residual slack time and event duration

The *ED* is an estimated duration of the event causing the breakdown, given by the human operator when the breakdown is detected. It can change over time. Figure 1 depicts different zones depending on the *RST* and *ED*: in zone 1 (*Wait and do nothing*), the *RST* is high and estimated *ED* low, the ratio *RST/ED* is far above 1. After recovery, the failed resource would still have time to produce the item. In zone 2 (*React by itself*), the ratio *RST/ED* is approximately 1, meaning the *ED* is equal to the *RST*. The product can be done on time, but a slight additional problem could result in important delivery delays. When the ratio is really under 1, which corresponds to zone 3 (*Switch back to centralized*), the product needs to ask for help. The limits between each zone are clearly an issue that is solved in this paper via the use of the AHP methodology combined with product preference functions.

The proposed AHP structure is presented Fig. 2a. The Goal is to select the most appropriate strategy, and this choice is based on two criteria which are: the ratio *RST/ED* (also named *Product Ratio*) and the *ED duration*. The different alternatives are the candidate strategies (*Centralized*, *Wait*, *PDS*). Two product preference functions are defined for each criterion (Fig. 2b).

3.2 Second Stage: Resource Evaluation

In this second stage, the resource evaluation is done via AHP as well. The AHP structure is shown Fig. 2c. Some of the parameters of this structure have been selected from the literature. However, to emphasize on the importance of quality and sustainability, two more criteria: *machine precision* and *power consumption* have been added to the structure. For each parameter, product preference functions are defined and presented in Fig. 2d. For the sake of simplicity, we consider only one type of product, hence only one type of preference function has been considered for each criterion.

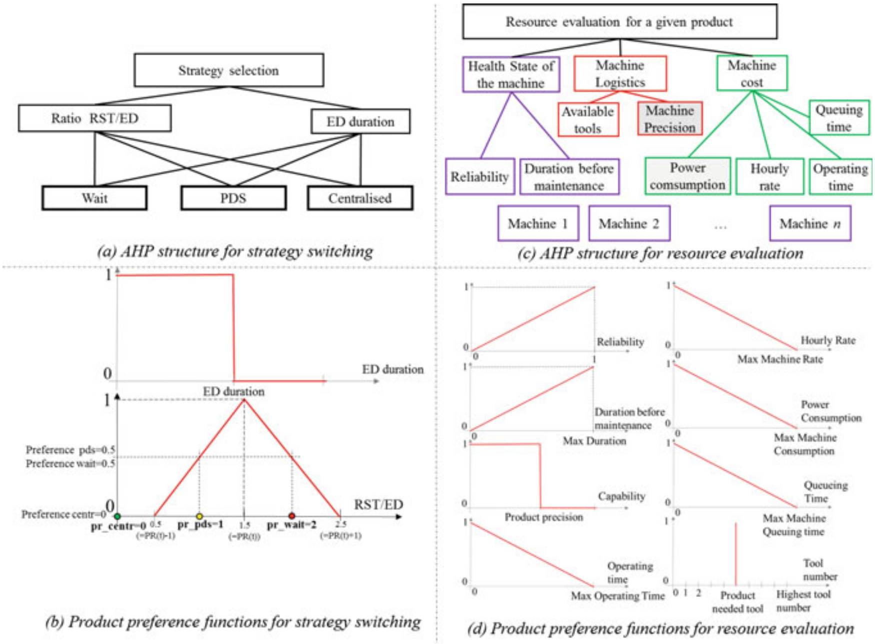


Fig. 2 AHP structures and product preference functions

4 Case Study

In our scenario, a breakdown occurs at time *t* in the shop floor composed of intelligent products and resources. Products affected by this breakdown have to decide or not to react themselves or with the help of the centralized control (*1st stage of the algorithm*). To do so, the following values have been assessed by experts: the time needed to react autonomously (defined as *pds_time*) is 10 min, to setup a centralized response (defined as *centr_time*) is 20 min; PDS is ideal when the product ratio is equal to 1 (this value is referred to as *pr_pds*), whereas centralized control should be used when the product ratio is near 0 (*pr_cent*). If the product ratio equals or exceeds 2 the wait strategy should be preferred (value of *pr_wait*). In our case, the product ratio is equal to 1.5 at time *t*, with a *RST* of 15 min and assessed *ED* of 10 min. If an autonomous reaction is required, each concerned product then decides to evaluate and select a resource among the 3 available machines. Each of these machines is supposed to send every 15 min the information needed for the 2nd stage of the algorithm (as depicted in the extract shown in Fig. 3). The AHP is then launched with the information available at time *t*. The length of our experimentation is 480 min (8 h). In this experiment, the product is supposed to require a machine precision of 5 mm, and the tool number 3.

INFORMATIONS FOR MACHINE 1						*MTTF :=*	*300*	*h*
Time (min)	Fiability	Duration before maintenance (h)	Power consumption (kW)	Available tool	Queuing time (min)	Operating time (min)	Hourly Rate (€/h)	Machine Precision (mm)
0	0,71	1,5	0	5	0	30	40	1
15	0,71	1,5	0	5	0	30	40	1
30	0,71	1,5				30	40	

Fig. 3 Example of information obtained for machine 1

4.1 Algorithm 1st Stage: Strategy Selection

As explained earlier, this part of the algorithm uses the AHP structures and the preference functions plotted Fig. 2a, b. In a first step, the weights of the criteria are defined. Then, the weights of the alternatives are determined and finally, all these are aggregated.

(a) *Determination of criteria weights*

The two concerned criteria are *RST/ED* (*or Product ratio—PR*) and *ED*. The expert gives a stronger importance to the criteria *Product Ratio* than the criteria *ED*, and defines the following 2 × 2 PDM, which leads to the weights expressed in Eq. (3):

$$\begin{array}{c} PR \\ ED \end{array} \overset{\begin{array}{cc} PR & ED \end{array}}{\begin{bmatrix} 1 & 2 \\ 0.5 & 1 \end{bmatrix}} \Rightarrow \begin{array}{c} PR \\ ED \end{array} \begin{bmatrix} 2/3 \\ 1/3 \end{bmatrix} \tag{3}$$

(b) *Determination of the weights of alternatives* via *product preference functions for each criteria*

Then, for each criterion, the alternatives' values are transformed into preference via the preference functions shown in Fig. 2b. As an example, let consider the criteria *PR* (*RST/ED*). The product ratio at time *t* is 1.5. By replacing 'product ratio' by 1.5, it is then possible to use the *PR* product preference function to compute the preference associated with the 3 alternatives which are 'Wait', 'PDS' and 'Centralized'. Because preferences of 'PDS' and 'Wait' are equal, both strategies are equally preferred before 'Centralized'. Via the use of a 3 × 3 final level PCM, the weights are determined. The same kind of process is done with the criteria *ED*, by using the values *pds_time*, *centr_time* and the product preference function associated to *ED*.

(c) *Aggregation and ranking of alternatives*

For a given *PR* and *ED*, it is then possible to compute the weights of alternatives and rank them. For the example expressed above, the final weights are:

$w_{wait} = 0.25$; $w_{pds} = \mathbf{0.75}$; $w_{centr} = 0$. This result means that, for this given situation, the PDS strategy should be used.

4.2 Algorithm 2nd Stage: Resource Evaluation

As explained earlier, this second part of the algorithm uses the AHP structure and the preference functions detailed in Fig. 3c, d. The steps are similar to the ones of the first part of the algorithm.

(a) *Determination of criteria weights*

The expert has attributed the same importance to all the criteria of a given level. Indeed, the computed weights for the 3 criteria of level 1 are all equal to 0.33, the weights of the 2 sub-criteria related to the 'Health State of the machine' and 'Machine logistics' are equal to 0.5, and the 4 sub-criteria of 'Machine cost' are all equal to 0.25.

(b) *Determination of the weights of alternatives* via *product preference functions for each criteria*

The process is the same than the one described earlier. For each criterion of level 2, the corresponding product preference function is used to transform the alternative's value into a preference. The alternative's values are taken from the data sent by the resource itself. Once done for all criteria, the resulting PCM is constituted and weights computed. Note that these weights depend on the instant $t_¸$ when the breakdown occurs.

(c) *Aggregation and ranking of alternatives*

The aggregation of the different levels allows the computation of the weights of the alternatives. As said previously, the weights of each machine are time-dependent, as shown in Fig. 4, plotting the evolution of each alternative's weight over the 480 min of our scenario. Obviously, at a certain time, a decision should be done by consensus or negotiation between products (step 2b of the algorithm).

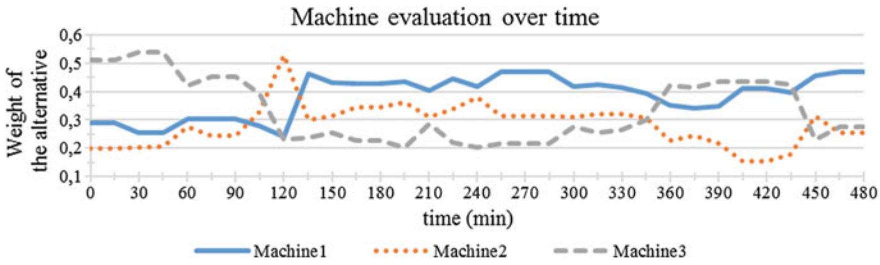


Fig. 4 Machine evaluation over time

5 Conclusions and Future Works

In this paper, a two-stage methodology to handle resource breakdowns has been presented. The proposed method is based on AHP coupled with product preference functions, that make possible to compute easily and on-the-fly the weights of the different alternatives. The theoretical foundations of this work and the tools needed to proceed to experimentation have been made, and a first experiment on a very simple scenario has been conducted. The future works will consist in first analyzing more precisely the results obtained with this scenario and then proceed to tests on a real production system like the TRACILOGIS Platform,[1] based in Epinal, France.

References

1. McFarlane, D., Giannikas, V., Wong, A.C., Harrison, M.: Product intelligence in industrial control: theory and practice. Annu. Rev. Control **37**(1), 69–88 (2013)
2. Meyer, G.G., Främling, K., Holmström, J.: Intelligent products: a survey. Comput. Ind. **60**(3), 137–148 (2009)
3. Pannequin, R., Morel, G., Thomas, A.: The performance of product-driven manufacturing control: an emulation-based benchmarking study. Comput. Ind. **60**(3), 195–203 (2009)
4. Borangiu, T., Răileanu, S., Berger, T., Trentesaux, D.: Switching mode control strategy in manufacturing execution systems. Int. J. Prod. Res. (2014). doi:10.1080/00207543.2014.935825
5. Lopez, P., Roubellat, F.: Ordonnancement de la production. Hermès Science Publications (2001)
6. Trentesaux, D., Dindeleux, R., Tahon, C.: A multicriteria decision support system for dynamic task allocation in a distributed production activity control structure. Int. J. Comput. Integr. Manuf. **11**(1), 3–17 (1998)
7. Ounnar, F., Ladet, P.: Managing breakdowns machines: a Petri Nets model and a decision-making process. J. européen des systèmes automatisés **33**(8–9), 977–994 (1999)
8. Saaty, T.L.: How to make a decision: the analytic hierarchy process. Eur. J. Oper. Res. **48**(1), 9–26 (1990)

[1]For more information, visit http://www.tracilogis.uhp-nancy.fr/.

9. Pach, C., et al.: An effective potential field approach to FMS holonic heterarchical control. Control Eng. Pract. **20**(12), 1293–1309 (2012)
10. Zbib, N., et al.: Heterarchical production control in manufacturing systems using the potential fields concept. J. Intell. Manuf. **23**(5), 1649–1670 (2012)

Product Driven Systems Facing Unexpected Perturbations: How Operational Research Models and Approaches Can Be Useful?

Alexis Aubry, Hind Bril, André Thomas and Mireille Jacomino

Abstract Production planning and control and more generally taking a decision in the context of production systems often consider that input information are known, static and predictable. However, uncertainties on data and perturbations are recorded in the genetic of every production system. For instance, it is impossible to know exactly the level of the demand for a product, the availability of resources, etc. Dealing with this issue raises the question of the ability to take robust decisions against uncertainty (off-line) or the ability to be flexible (on-line). This paper proposes to analyse how Product Driven Systems—as reactive systems against unpredicted perturbations—can be part of operational research solution process against perturbations. Moreover, an overview of models and approaches for dealing with uncertainty in Operational Research is given and a first proposition is made to apply these elements into PDS as decision-making-against-perturbations engines.

Keywords Production planning and control · Product Driven Systems · Uncertainty · Robustness · Flexibility · Operational research

A. Aubry (✉) · H. Bril · A. Thomas
Université de Lorraine, CRAN, UMR 7039, Campus Sciences, BP 70239, Vandœuvre-lès-Nancy Cedex 54506, France
e-mail: alexis.aubry@univ-lorraine.fr

H. Bril
e-mail: hind.el-haouzi@univ-lorraine.fr

A. Thomas
e-mail: andre.thomas@univ-lorraine.fr

A. Aubry · H. Bril · A. Thomas
CNRS, CRAN, UMR 7039, Vandœuvre-lès-Nancy, France

M. Jacomino
G-SCOP Laboratory, University Grenoble Alpes, CNRS, Grenoble 38000, France
e-mail: mireille.jacomino@grenoble-inp.fr

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_23

1 Introduction

Since some years, Product Driven Systems (PDS) are becoming a credible way to control product flows in shop-floors and in supply chains thanks to Auto-ID and especially RFID technologies. PDS are defined as systems in which *products are able to be active and are at the core of an enterprise architecture which integrates every actor in the company, from the central systems to the processes, products and also operators, into the same ambient information system* [1]. The Intelligent Product (IP) paradigm [2, 3] was a step further to the use of RFID-tagged products, and not only for identification objectives. IP has demonstrated the efficiency of speeding up and making more visible information flows in logistic operations [4]. Following, a new interpretation of the stigmergy concept was put forward in [1], where the cooperation between production actors (products, resources and all used manufacturing appliances) was achieved thanks to attributes called informational pheromones and carried by RFID-tagged products. Product collaboration is a concept on which many researchers [5, 6] have focused their works.

Thus, this Product Driven Systems (PDS) paradigm has been established and its relevance has been highlighted precisely to deal with new product flows control approaches, especially to react to unpredicted events occurring on the shop-floors and in the supply chains. Indeed, anybody with production or logistic experience knows that often, a predicted plan or schedule cannot be exactly executed! Events occurring in the shop-floor or in the supply chain lead to uncertainties in daily activities, inducing needs for adjustments in plans. That is one of the main reasons that justify PDS.

On the other hand, in the previous decades, a lot of research teams have proposed a huge number of models, built on exact approaches or on heuristics, useful to schedule or re-schedule production and logistic activities. In the design or management of these various types of control systems, and particularly for production and logistic ones, a number of variables have to be frequently determined so as to optimize a given performance criterion. Optimization models are used to search for the best possible values of a vector of variables (e.g. number of resources, production planning in a manufacturing system and so on). In this context, operational research approaches are privileged approaches when dealing with optimization in production systems. However, in most dedicated works, the optimization model is built from data that characterize given and fixed environmental conditions (e.g., assumed nominal processing times, fixed transportation delays, exact amount of demand etc.). However, in practice, perturbations can modify these values, and thus change the data that were initially assumed to be known and static. The expected values (also called reference instance in [7] or base environmental scenario in [8]) are supposed to model the environment that the domain expert thinks relevant to characterize the actual operating conditions of the studied system. In practice, good or even optimal performances achieved by the system in given conditions can be drastically deteriorated if these conditions change. For instance, a delivery delay, evaluated through an optimization model, can be considered as acceptable with regards to the customer demand, if breakdowns occur at a given rate, but can become unsatisfactory if the breakdown rate increases. To deal with this issue, it is necessary to take the data uncertainty and the perturbations into account. Since several years, operational researchers are aware of this issue and have proposed some process and approaches to deal with uncertainty and perturbations.

The goal of this paper is to analyse how the PDS paradigm can be involved inside the process proposed by operational researchers to deal with perturbations and to identify the knowledge and information about operational research works dealing with scheduling under uncertainties and to highlight if some of them could be useful to deal with PDS re-scheduling acting in uncertain production and/or logistic environment.

The rest of the paper is organized as follows. Section 2 discusses how optimization can take into account inherent uncertainties and perturbations and proposes a 3-steps solution process to deal with uncertainties and perturbations. Section 3 deals with the step 1 and more specifically the concepts of robustness in operational research. Section 4 focuses on the step 2 and presents how the concept of Product Driven System can be a good candidate for implementing this step as a reacting system against perturbations. Moreover, this section proposes a first overview of models and approaches coming from Operation Research that can be useful for implementing efficiently these types of system. Finally, a conclusion and perspectives are formulated in Sect. 5.

2 Dealing with Uncertainty: A Full Solution Process

2.1 Classical Optimization

Classically, solving an optimization problem consists in building a solution S that optimizes a criterion z (that is assumed to be a minimization criterion in this section without loss of generality) and satisfies some constraints assuming that the problem data are certain and sure. This particular string of data can be seen as a forecast instance I^{ref} that is used to compute S. Getting an optimal solution is still a problem that is often hard. An optimal solution to the problem for an instance I is denoted S_I^* and its corresponding performance is denoted z_I^*. Otherwise, the performance of a solution S applied to an instance I relatively to an optimization criterion z is denoted $z_I(S)$.

The classical way to solve an optimization problem without uncertainty is the predictive approach. An off-line algorithm builds an optimal solution $S_{I^{ref}}^*$ for the forecast instance I^{ref}, and guarantees an optimal performance for this instance only, valued by $z_{I^{ref}}^*$. In practice, the real system is subject to perturbations such that the solution $S_{I^{ref}}^*$ is applied to the actual instance I that may be different from the forecast instance I^{ref}, and $S_{I^{ref}}^*$ may even no longer be admissible. In the most optimistic case (when the solution remains admissible for I), the actual performance $z_I\left(S^*_{I^{ref}}\right)$ can be "far" from the forecast performance $z^*_{I^{ref}}$, and also far from the optimal value for I, z^*_I. Then, a costly solution step may return a poor-quality solution.

2.2 A Full Solution Process for Dealing with Uncertain Context

There are two complementary ways for taking uncertainties into account. The first one is concerned with uncertainties that are explicitly depicted. Those uncertainties are somehow expected. The second way concerns unexpected perturbations which the system has undergone. Expected uncertainties can be taken into account more or less in an anticipative optimisation structure; this is the robust approach. Unexpected uncertainties have to be taken into account in a reactive procedure. In [9], the generic structure for dealing with uncertainties is addressed. It is formally defined for addressing scheduling problems. It is composed of three steps.

Step 0: Static Problem Definition

Classic specifications of the optimisation problem are given along with the optimisation criterion z. Moreover, when uncertainties are identified, it is necessary to produce a model of these uncertainties. Several possibilities exist. Uncertainties can be modelled as stochastic parameters [10] or parameters belonging to fuzzy intervals [11] or belonging to a set of scenarios or instances [7, 12]. This set of instances can be a continuous or a discrete interval. A specific robustness criterion has to be defined related to the expected risk answering the question: "what must be guaranteed despite these perturbations?"

Step 1: Calculation of a Set of Anticipative Solutions by an Off-line Algorithm

From the available knowledge about perturbations (the risk to be covered defined in step 0), one or more algorithms can be used to build a set of robust solutions guaranteeing a performance on the considered risk as defined in step 0. This set can include one or more solutions. Instead of computing an optimal solution for a single forecast instance, a global performance on the risk to be covered is addressed. Such solutions if they exist are said to be robust for the considered risk.

Step 2: Calculation of the Applied Solution by an On-line Algorithm

The on-line algorithm uses the progressive knowledge about environment to implement a solution knowing the anticipative solutions. This solution can be chosen among the set of solutions calculated in step 1 or can result from a reactive adaptation of anticipative choices. The gap between the implemented solution and the anticipative one depends on the perturbations that occur: if they are included in the expected risk to be covered then the reactive procedure consists in choosing a solution among the robust previously computed solutions, otherwise all changes are possible. Moreover, this asks the question of the detection of the perturbation.

In this dynamic procedure, the problem of guaranteed performance is open.

3 Step 1: Taking Robust Decisions

3.1 What Is Robustness?

Developing robustness features for the decision to be made has appeared to be an efficient way of coping with deterministic uncertainties as in [12–16] even though researchers use different measures of robustness. Beyer and Sendhoff [17] identifies several types of uncertainty involved in robustness optimization and several ways of formulating and addressing the corresponding problems in the field of parameter design. In [16], robustness is defined as *a capacity for withstanding "vague approximations" and/or "zones of ignorance" to prevent undesirable impacts, notably the degradation of the properties to be maintained*. It is considered that the so-called properties to be maintained and the considered "approximations" and "zones of ignorance" are strongly application dependent, which has led researchers to develop a large variety of robustness approaches.

Many robustness definitions can be found in the literature. The five following papers are devoted to such definitions: [12, 15, 16, 18, 19]. A lot of other papers are devoted to finding a robust solution for a given problem and explicitly proposing or not a measure of robustness. It is possible to sort the robustness definitions into two categories. The first category defines a robust solution as a solution that optimizes a robustness criterion. This point of view has the advantage that classical models and approaches used in Operational Research can address this problem. The second category defines a robust solution as one that satisfies a condition (or a set of conditions). Such a vision for robustness allows practitioners to keep their solution processes and to check for the robustness of any solution with no restriction.

4 Step 2: React After the Occurrence of Perturbations

4.1 The Product Centric Paradigm/Product Driven Systems

When perturbations don't concern directly data (like variation in demand) but are relative to specific events occurring on the shop-floor or in the supply chain, it is necessary to react on-line. As previously introduced, Product Driven Systems (PDS) are seen today as a good alternative to address this challenge. The idea is to deal directly with products whose information content is permanently bound to their material content and which are able to influence the decisions made about and for them. Therefore, product-driven control has an impact on decision-making procedures as well as on information exchange and storage. PDS are thus good candidates for covering the step 2.

4.2 Re-optimization or Online Optimization

In the production step, an in-process product can manage its own manufacturing, dispatching or operating lists and priorities, according to the real state of the production system and possible dysfunctions occurring on the shop floor or modifications concerning the due dates or the demand volume. It can ask for specific production services to resources and communicate with the other products to negotiate priorities and production deadlines. In the logistic phase, active products in a system with safety constraints can supervise their own operations, triggering alarms when required by cooperating with other active components, thus improving the availability and maintainability of the system and obtaining an ambient system [20].

All these re-scheduling cases could be solved in a more reactive way by switching down from the centralized control system to PDS. Very quickly PDS could find, if there exists, a feasible solution to perform production or logistic activities remaining after the specific event. Obviously if no solution emerges at the operational level it would be still necessary to switch back to the centralized way to find an optimal solution. Ideally, to be the most possible reactive, this kind of process must be "on-line" to be sure that the real system won't change during the decision process as highlighted in [21].

The main issue for researchers is then to provide accurate mechanisms to define "on-line" the best switching dates (and/or the best switching decision-making levels) for control holons/agents so that they behave in a sense that the behaviour of the hybrid architecture stays globally optimized despite disturbances. In this context, flexibility and resilience are two performance indicators that are interesting for evaluating a PDS.

4.3 Models and Approaches for Reacting After Perturbations

Models

Regarding Operational Research, two main techniques are used for modelling optimization problems: graphs and mathematical programs. When dealing particularly with operations scheduling problems, it is easy to check that the mathematical programs are more used in this context. According to the type of the mathematical program (linear or not, with integer variables or not…), associated solution methods are existing for finding the optimal solution. And these methods are implemented in commercial tools like Cplex, Lingo. The efficiency of the methods for solving such models is particularly sensitive to the type of variables, the number of variables and the number of constraints.

When dealing with operations scheduling (and so re-scheduling), two sub problems must be solved. The first one concerns the allocation of operations to machines and the second one concerns the sequencing of these operations with possible precedence constraints to be satisfied. We can note that the allocation problem can be sometimes fixed a priori when only one machine is qualified for executing a given operation. We will see in the next paragraph that the fact that the allocation problem is fixed or not has an incidence on the solving method.

Solution Approaches

Classical Operational Research approaches for solving optimization problems can be split into two categories:

- *Exact methods*: Branch and Bounds algorithms (for solving Mixed Integer Linear Programs), dedicated algorithms. These approaches have the advantage to be able to find the optimal solution but are often not efficient for real-sized problems.
- *Approximation methods*: mainly heuristics and metaheuristics. These approaches have the advantage to be efficient even for real-sized examples and to be able to find good solution regarding the optimization criterion.

In order to identify the relevant approaches for solving re-scheduling problems in the context of PDS, it is mandatory to take into account at least two inputs: the available time for finding the schedule and the type of decision to be taken.

The available time is clearly dependent on the duration of the unpredicted perturbation. In fact, if the time for taking the decision becomes bigger than the time for solving directly the perturbation, re-scheduling becomes useless. For instance, imagine that a machine failed and that we can predict that the machine can be repaired within a given duration; this will fix the available time for finding a new schedule. It is precisely here that the PDS concept is the most useful because its application must help to evaluate this available time through the information that can give the product the closest from the production system. The product can help detecting the unpredicted perturbation and then the merged information coming from different products can give an overall situation and help to decide which type of decision must be taken.

The type of decision to be taken concerns mainly whether re allocation is needed or not. If re allocation is needed, an integer variable is needed to represent the allocation of each operation to a machine. That means that the mathematical program for modelling the problem becomes a Mixed Integer Linear Program (MILP). This type of problem is known to be NP-Complete, meaning that no efficient solution method is known for solving efficiently a MILP.

Solution Approach	Criterion/Need	Optimality need	Available time for finding a feasible solution		Participation of a decision expert	Several solutions needed	Fixed or Flexible route	Need of homogeneity of the route	Possible splitting or not	Possible overlapping or not
Metaheuristics	Genetic Algorithm	Feasible solution accepted	real-time decision			Yes (population)			Yes	Yes
	Simulated Annealing									
	Tabu Search									
	Ant Colony Optimization							Yes	Yes	Yes
	Particle Swarm Optimization								Yes	Yes
	BAT Optimization			Very quick convergence						
Exact methods	MILP	Optimal of feasible solution			Process acceleration		Flexible route			
	LP						Fixed route			

Fig. 1 Identified solution methods according to several criteria

According to several inputs, we define a first table giving the preferable solution methods depending on some criteria arbitrarily defined. To our best of knowledge, such a work has never been done such that the identified criteria/needs should be enriched and discussed. The first table is given in Fig. 1. Future works must be done to: (i) fill the gaps in the table, (ii) better justify the criteria in the table.

5 Conclusions

This paper investigates the product-driven systems as a candidate for reacting against perturbations and for taking part in a full solution process existing in the operational research context. Moreover, this paper is a first attempt to identify the solution approaches from operational researchers that are relevant for implementing a PDS.

Valuable efforts should still be done for better identifying the different situations that can occur when dealing with perturbations, and thus to associate better solution approaches to deal with these situations.

References

1. Pannequin, R., Thomas, A.: Another interpretation of stigmergy for product-driven systems architecture. J. Intell. Manuf. **23**(6), 2587–2599 (2011)
2. Wong, C.Y., McFarlane, D., Zaharudin, A.A., Agarwal, V.: The intelligent product driven supply chain. In: 2002 IEEE International Conference on Proceedings of Systems, Man and Cybernetics, vol. 4 (2002)
3. Morel, G., Panetto, H., Zaremba, M., Mayer, F.: Manufacturing enterprise control and management system engineering: paradigms and open issues. Annu. Rev. Control **27**(2), 199–209 (2003)

4. Klein, T., Thomas, A.: Opportunities to reconsider decision making processes due to Auto-ID. Int. J. Prod. Econ. **121**(1), 99–111 (2009)
5. Yoshimura, M.: System design optimization for product manufacturing. Concur. Eng. Res. Appl. **15**(4), 329–343 (2007)
6. Wenyan, S., Ming, X., Wang, P.: Collaborative product innovation network: status review, framework, and technology solutions. Concur. Eng. Res. Appl. **21**(1), 55–64 (2013)
7. Espinouse, M.-L., Jacomino, M., Rossi, A.: On the robustness of multi-purpose machines workshop configuration. In: Flexibility and Robustness in Scheduling. ISTE Ltd, London, United Kingdom (2008)
8. Pierreval, H., Durieux-Paris, S.: Robust simulation with a base environmental scenario. Eur. J. Oper. Res. **182**, 783–793 (2007)
9. Billaut, J.-C., Moukrim, A., Sanlaville, E.: Introduction to flexibility and robustness in scheduling. In: Flexibility and Robustness in Scheduling. ISTE Ltd, London, UK (2008)
10. Dauzère-Pérès, S., Castagliola, P., Lahlou, C.: Service level in scheduling. In: Flexibility and Robustness in Scheduling. ISTE Ltd, London, UK (2008)
11. Dubois, D., Fargier, H.: Fuzzy scheduling: modelling flexible constraints vs. coping with incomplete knowledge. Eur. J. Oper. Res. **147**, 231–252 (2003)
12. Kouvelis, P., Yu, G.: Robust Discrete Optimization and Its Applications. Kluwer Academic Publishers, Dordrecht, The Netherlands (1997)
13. Vincke, P.: Robust solutions and methods in decision-aid. J. Multi-Criteria Decis. Anal. **8**, 181–187 (1999)
14. Perny, P., Spanjaard, O., Storme, L.X.: A decision-theoretic approach to robust optimization in multivalued graph. Ann. Oper. Res. **147**, 317–341 (2006)
15. Kalaï, R., Aloulou, M.A., Vallin, P., Vanderpooten, D.: Robust 1-median location problem on a tree. In: Proceedings of the ORP3 (Euro Conference for Young Researchers and Practitioners), Valencia, Spain
16. Roy, B.: Robustness in operational research and decision aiding: a multi-faceted issue. Eur. J. Oper. Res. **200**(3), 629–638 (2010)
17. Beyer, H.G., Sendhoff, B.: Robust optimization—a comprehensive survey. Comput. Methods Appl. Mech. Eng. **196**, 3190–3218 (2007)
18. Aubry, A., Rossi, A., Jacomino, M.: A generic off-line approach for dealing with uncertainty in production systems optimisation. In: Proceedings of the 13th IFAC Symposium on Information Control Problems in Manufacturing INCOM'09, Moscow, pp. 1464–1469 (2009)
19. Galand, L., Spanjaard, O.: OWA-based search in state space graphs with multiple cost functions. In: Proceedings of the 20th International Florida Artificial Intelligence Research Society Conference (FLAIRS'07), pp. 86–91 (2007)
20. Parlikad, A., K., McFarlane, D.: RFID-based product information in end-of-life decision making. Control Eng. Pract. **15**, 1348–1363 (2007)
21. Li, M., Bril El-Haouzi, H., Thomas, A., Guidat, A.: Fuzzy decision-making method for product holons encountered emergency breakdown in product-driven system: an industrial case. Springer Series Studies in Computational Intelligence. In: Proceedings of SOHOMA'14, Nancy (2014)

Holonic Facility Environment Monitoring and Control for Radiopharmaceutical Agent-Based Production

Theodor Borangiu, Andrei Silişteanu, Silviu Răileanu and Octavian Morariu

Abstract The paper presents design and implementing considerations for an agent-based environment monitoring and control system dedicated to a radiopharmaceuticals production line. This system is distributed on two layers: a centralized one, responsible for updating the status of the facility's parameters and the global control strategy for cascade room parameters, and a distributed one that applies the global strategy to local environment control units—Heat Ventilation and Air Conditioning units (HVAC). The paper describes the environment parameters and defines the HVAC process models. The facility environment control system is developed in holonic approach, with three basic holons and one expertize holon. An implementing solution and integration of the environment control system with the radiopharmaceutical production management system is described. Experimental results and conclusions are finally presented.

Keywords Holonic paradigm · Multi-agent system · Environment monitoring · Distributed intelligence · SCADA · Radiopharmaceuticals production

1 Introduction

The production of radiopharmaceuticals, i.e. products that contain radioactive materials called radioisotopes, is based on two processes: (1) obtaining the radionuclides i.e. the radioactive isotopes of elements with atomic numbers less than that of bismuth; (2) portioning and packaging the manufactured bulk radio-

T. Borangiu (✉) · A. Silişteanu · S. Răileanu · O. Morariu
Department of Automation and Industrial Informatics,
University Politehnica of Bucharest, Bucharest, Romania
e-mail: theodor.borangiu@cimr.pub.ro

A. Silişteanu
e-mail: andrei.silisteanu@cimr.pub.ro

S. Răileanu
e-mail: silviu.raileanu@cimr.pub.ro

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_24

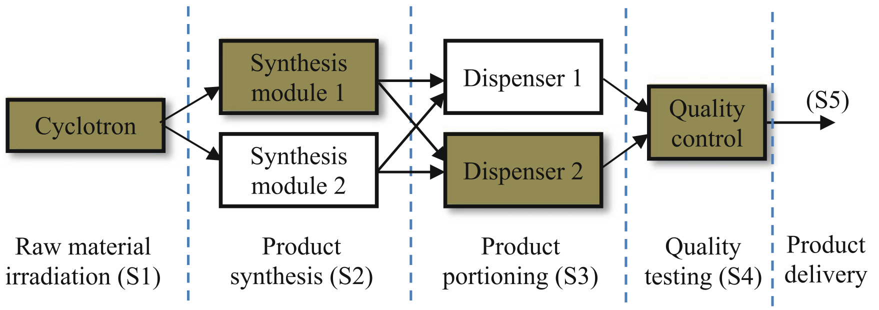


Fig. 1 Manufacturing stages of radiopharmaceutical production

pharmaceutical product [1, 2]. Radiopharmaceuticals obtained using a cyclotron and dedicated radiochemistry equipment and laboratories are produced inside nuclear facilities, the environment of which must be very strictly controlled both from the point of view of radiologic security and of cleanness of the areas where the pharmaceuticals for human use are dispensed and packaged (the clean rooms) [3].

A cyclotron-based radiopharmaceutical production line, including facilities, control and monitoring systems, manufactures small batches of nuclear medicine products in the shortest possible time [4], safely for employees and the facility environment. Each ordered quantity of product has a specific chemical structure, radioactivity and usage and follows the same manufacturing cycle: radio isotopes are produced in a particle accelerator (cyclotron), then transferred into technology isolators for chemical synthesis followed by portioning (vial dispensing) of the bulk product, and quality control of the final product by conformity tests on multiple parameters; in the last stage, valid products are packed and transported to the clients in shielded containers. The manufacturing stages (S1)–(S4) for the cyclotron-based production line of radiopharmaceuticals are represented in Fig. 1, and involve one functional block (filled in with light grey) for each stage; a fifth stage (S5) is considered for the transport of the final products to the client (e.g., hospital, analysis laboratory).

Manufacturers grant high reliability for the cyclotron, whereas for single point of failure avoidance and fault tolerance reasons, stage 2 (product configuring) and stage 3 (portioning of the bulk product) may use respectively any of two identical resources (synthesis modules 1, 2 and robotized dispenser 1, 2), capable to replace each other in the event of breakdown or maintenance work [5–7]. The three production resources together with the quality testing equipment are disposed in *flow shop* processing mode [8]. Raw materials enter the cyclotron and are irradiated, the output being fed to one synthesis module where chemical reactions occur; the resulting bulk radiopharmaceutical product is then portioned and eventually diluted in one robotized dispenser box; samples of vials are sent to the quality control room for multi-parameter tests; finally, the vials are packed and labelled for transport to the client. The product passes from one resource to the next through capillary tubes for successive processing, the operations carried out in stages (S1)–(S4) being totally ordered.

Special conditions must be continuously fulfilled during the manufacturing process of radiopharmaceuticals; these conditions can be classified in two main categories: *radioprotection safety conditions* (radioactivity doses, pressure cascades and the number of air changes per hour) and *environment manufacturing conditions* (physical environment parameters) as defined by Good Manufacturing Practice guides (GMP) or imposed by National Radioprotection Regulations [6, 9].

The area inside a nuclear facility where radiopharmaceuticals are being produced is called the cleanroom area. A cleanroom is a special closed space in which the concentration of airborne particles is controlled for specified limits in order to eliminate sub-micron airborne contamination induced by people, processes and equipment. Air flow rates and direction, pressurising, temperature, humidity and specialized filtration need to be strictly controlled:

- Permanent input and filtering of fresh air; air flow recirculation is forbidden;
- Cascades of negative pressure in closed spaces relative to the exterior, in order to evacuate the potentially contaminated air from the production facilities;
- Cascades of positive pressures between adjacent closed spaces preventing the rapid air penetration from one room to another at access doors opening and slowing down the diffusion of air particles in the clean rooms;
- Monitoring temperature, pressure, humidity and radioactivity level in each one of the three rooms: cyclotron, production (synthesis, dispensing) and environment parameter preparation (HVAC—Heat, Ventilation and Air Conditioning).

The monitoring and control of cyclotron-based radiopharmaceutical production environment parameters use HVAC automation systems. The goal of HVAC automation is in general either to minimize energy consumption for the entire facility (production structure, office or home building) while keeping the environment parameters in a selectable value range, or to maintain the critical environment parameters (pressure, humidity, temperature and airflow quality) in a strict imposed range, irrespective of the energy consumption. The research literature reports several optimization solutions for HVAC systems' automation strategies using different methods: tuning of the PID controller parameters using classical or genetic algorithms [10], fuzzy logic control [11] or artificial neural networks [12]. One recent approach consists in collecting environment and HVAC control data over a long period of time, data mining, identifying the HVAC model and synthetizing the control algorithm [13]. Some authors report HVAC control implementations using software agents for distributed parameter monitoring and control in large spaces [14].

An agent-based, holonic facility environment monitoring and control solution conditioning the production of radiopharmaceuticals will be described in this paper.

The remainder of the paper is organized as follows: Sect. 2 describes the environment parameters and defines the HVAC process models. Section 3 presents the structure of the holonic facility environment system (HFES) and composing holons with functions and data structures. Section 4 is devoted to the HMES implementation with intelligence distribution and integration with the radiopharmaceutical production management system. Section 5 reports experimental results and presents conclusions.

2 Environment Parameters and HVAC Process Models

The radiopharmaceuticals production environment must comply with the ISO 14644-1 standard and is continuously monitored during production cycles according to GMP regulation. The key environment parameters are:

- *Pressure* (P): a minimum positive difference of 8–15 Pa is required between any clean room and its adjacent rooms, such as the production area where the access to the clean rooms is possible by passing first through a locker room and then through a clean corridor. This is a global "cascade pressure model". Inside closed spaces surrounding production areas (synthesis, dispensing) a negative pressure (−30 to −70 Pa) must be permanently kept to prevent inside air entering the clean-room. In the cyclotron vault, a negative pressure cascade is critical relative to the production area to prevent the inside air spreading to the entire facility. Notations: P_c, P_h, P_v, P_A are pressures respectively in the clean room, clean corridor, locker room and adjacent area to the clean room (see Table 1).
- *Temperature* (T): for the comfort of clean room operators wearing gowning fabrics, the desired temperature range will be from 18 to 21 °C. Keeping temperature in this range reduces microbial growth (viruses, spores, fungi, bacteria) and will compensate the heat produced by electronic devices inside the room.
- *Relative humidity* (RH): must be kept between 35% and 55%. Very low humidity can generate static electricity which can then destroy expensive measurement instruments, and high humidity favours the growth of bacteria and germs.

Table 1 Environment constraints for cleanness in radiopharmaceutical facilities

Parameter value	P	T	RH	NP_c	$H_c^*(10)$
[Meas. unit]	[Pa]	[°C]	[%]	#	μSv/h
cl.r.-cl.corr.	$P_c – P_h \geq 10$ Pa	18 °C $\leq T_c \leq$ 21 °C	35–55	$\leq$3520 (0.5 μm) $\leq$20 (5 μm)	$\leq$0.8
cl.corr.-lock.r.	$P_h – P_v \geq 10$ Pa	18 °C $\leq T_h \leq$ 21 °C			
lock.r.-adj.z.	$P_v – P_A \geq 10$ Pa	18 °C $\leq T_v \leq$ 21 °C			
Alert value	5 Pa	m.17 °C, M.23 °C	m. 30, M.60	4000/25	1
Alarm value	8 Pa	m.15 °C, M.25 °C	m.25, M.65	4520/30	10

- *Number of airborne particles* (NP_c): this parameter is continuously monitored in the robotized dispensing isolator box, using a particle counter. If the number of particles is not in range for class C cleanness, the dispensing process is suspended for a maximum timeout of 20 min to allow this parameter to re-enter in range; if it does, the process continues and dispensing (dilution and portioning) is delayed with the corresponding amount of time—which also delays delivery; otherwise production is abandoned and the production order fails.
- *Radioactivity level* $\left(H_c^*(10)\right)$: represents the flow of ambient dose produced by radiations in the clean rooms for: production (synthesis), control and dispensing.

Table 1 summarizes the environment constraints imposed to maintain cleanness of class A in the dispensing isolator box, of class B in the clean corridor of the dispensing isolator box and of class C inside the clean room and the synthesis box:

The parameter values in the first three lines of Table 1 correspond to the *normal state* (coded 0) of the facilities' environment. In the *alert state* (coded 1) the environment parameters are very close to the imposed limits and the Facility Environment Monitoring and Control System (HFES) triggers investigation actions; if this diagnosis identifies the reason for entering the alert zone, then appropriate corrective actions are taken by HFES. NPc is the only parameter which, entering the alert state suspends the production process for maximum 20 min. In the *alarm state* (coded 2) parameters exceed the imposed safety limits; the production processes are stopped and corrective actions are initiated to bring back the faulty parameter in the normal value range.

Two HVAC units are used inside the nuclear facilities to maintain environment parameters within the range of prescribed normal state values: (1) one working in negative-pressure mode inside and around the cyclotron vault in order to avoid the radioactive air spreading in the building, and (2) the other working in over-pressure mode inside clean rooms to avoid the penetration of the building air and deteriorating the quality of the purified air inside the production and dispensing closed spaces.

The two HVAC units are integrated into a Holonic Facility Environment Monitoring and Control System (HFES), which analyses environment data, evaluates its current state (0, 1 or 2) and computes accordingly the global facility environment control strategy and environment parameter set points in *centralized* mode; the application of this strategy is then *decentralize*d by distributing intelligence in a multi-agent framework based on: holons collecting environment parameter data (T, NPc, RH, Hc*(10), P) from room sensors, holons transposing the global, cascade strategy in HVAC operating modes and control laws and computing the reference values of the HVAC parameters (environment tasks), and resource holons executing environment tasks via HVAC specific models and processes (air flow and cooling water conditioning).

The HFES coordinates at MTU (Master Terminal Unit) level a physical SCADA (Supervisory Control and Data Acquisition) environment monitoring and control system, providing the necessary commands for the two HVAC units (see Fig. 2) acting as SCADA RTU (Remote Terminal Units).

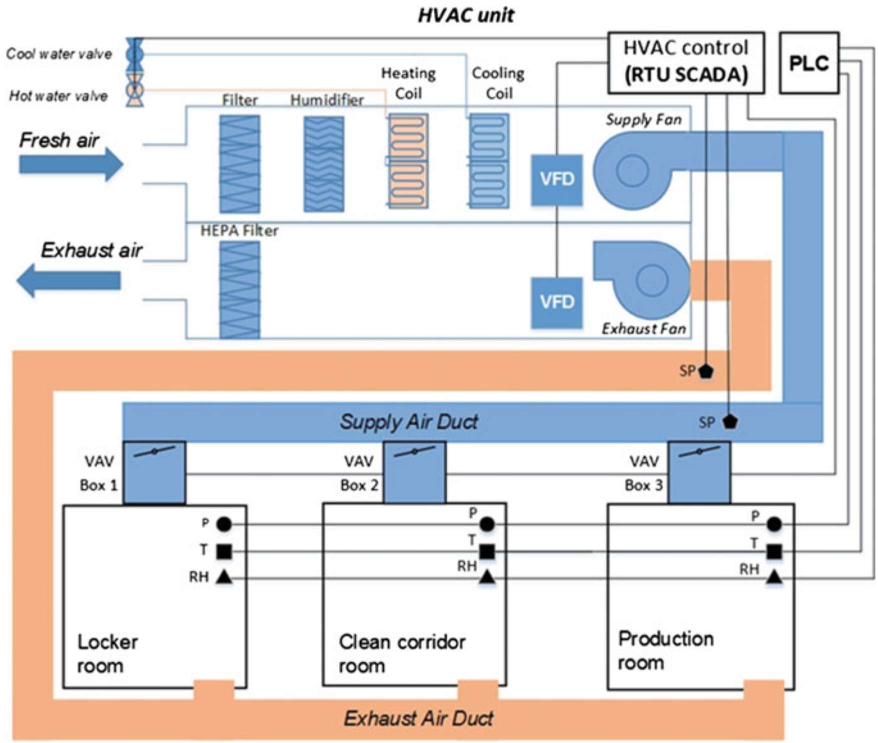


Fig. 2 Structure of a HVAC unit for radiopharmaceutical production environment conditioning

The proposed solution for safe environment control in the cleanroom area is composed by a classic HVAC unit combined with a VAV (variable air volume) box for each room of the multi-zone area. As shown in Fig. 2 the main HVAC components are the air filters, the humidifier block, the heating and cooling coil and the supply and exhaust fans driven by two VFD (variable frequency drives) [10, 12].

While the two VFD can vary the total airflow passing through admission and exhaust ductwork, the VAV boxes are setting the required airflow entering each room. If the total supply airflow exceeds the total extracted airflow, overpressure will be created in the clean-room area. In order to create a pressure cascade between rooms, VAV boxes are used to set a specific airflow for each room according to the requested pressure conditions (the cascade model previously defined).

The room temperatures are set to the desired values by comparing the value read by local sensors with the supplied air temperature. Outside air is treated inside the HVAC by the heating and cooling coils that act upon water supplied, so that its temperature will match the set-point (reference) value for the room air. Temperature regulation is made by varying the water flow passing through these coils using variable opening valves. Since the cooling process is similar to the heating one (only the water temperature is different) only the first one will be further modelled.

To establish the HVAC model for the process of room airflow cooling, one must derive the expression of the mass flow $\dot{m}_{\text{water}}$ of the cooling water let in the cooling coil area function of the admission valve opening, *pos* (its limit opening values are: 0 for completely closed and 1 for completely opened). The flow coefficient specific for the water admission valve has the expression CV_{valve} provided by the manufacturer:

$$CV_{\text{valve}} = 0.0165 \cdot pos^3 + 0.0076 \cdot pos^2 + 0.0024 \cdot pos + 0.0001$$

Considering as input parameters the pressure difference between the water supply pipe and the water evacuation pipe, $\Delta\text{P}_{\text{mains}} = 40000\,\text{Pa}$, constant and the coefficient of losses in the water circuit inside the HVAC unit, $\text{k}_{\text{coil}} = 7000$, constant, then the admitted water flow is $\dot{m}_{\text{water}} = CV_{\text{valve}} \cdot \sqrt{\Delta P_{\text{valve}}/\text{g}}$, where $\text{g} \cong 9.81\text{N/kg}$ is the gravitational constant and $\Delta P_{\text{valve}} = \Delta\text{P}_{\text{mains}} - \dot{m}^2_{\text{water}} \cdot \text{k}_{\text{coil}}$ represents the pressure difference between the admission valve's input and output. The computational model of the HVAC cooling mass flow function of the valve opening becomes:

$$\dot{m}_{\text{water}} = \sqrt{\frac{CV^2_{\text{valve}} \cdot \Delta\text{P}_{\text{mains}}}{1 + CV^2_{\text{valve}} \cdot \text{k}_{\text{coil}}}}$$

This model will be implemented by the task agent responsible for model-based room temperature control with set-point temperature value computed at centralized,

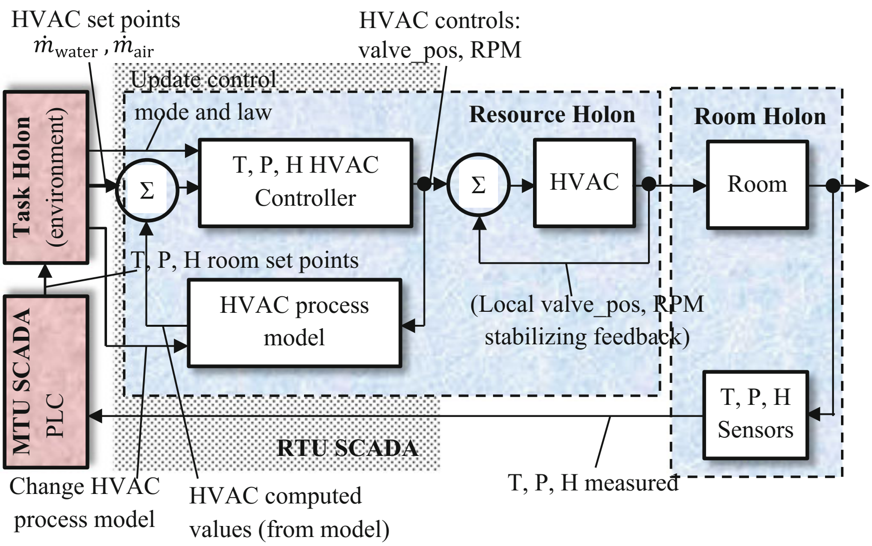


Fig. 3 Generic agent-based control, distributed at room sensors and HVAC level, tracking T, P, H room set points computed at global facility environment level by the centralized HFES

SCADA MTU level of the HFES and temperature regulation carried out by the resource agent representing the HVAC process. Similarly, HVAC models have been derived for single room pressure and relative humidity regulation.

In order to have the room humidity in the prescribed range according to season, the outside air is dried in summer using the heating coil followed by the condensation process in the cooling coil. In winter, moisture is added in the form of air steams in the humidifier block.

Figure 3 shows the generic architecture for HVAC agent-based control with distributed intelligence (at parameter channel level), providing tracking of the room temperature, pressure and humidity set points computed at centralized level to satisfy the global environment facility model.

3 Design of the Holonic Facility Environment System (HFES)

The dual control architecture proposed for the manufacturing system producing radiopharmaceuticals in shop floor processing mode (Fig. 1) has two control systems working in subordination: (a) the holonic Manufacturing Execution System (HMES) layer—for production planning and resource allocation, data storage and reports generation is subordinated to (b) the holonic Facility Environment Monitoring and Control System (HFES) layer—environment monitoring and maintaining safe operating conditions by help of a SCADA architecture distributed over sensor data collecting and HVAC control channels for the cyclotron vault and production rooms with their adjacent spaces.

The radiopharmaceutical production line controlled by the HMES described in [4] is authorized to work only if the values of the environment parameters described in Sect. 2 ($\mathrm{P, T, RH}$ and $\mathrm{H}_{\mathrm{c}}^{*}(10)$) are within their normal or alert variation ranges corresponding to states 0 and 1 for their imposed cleanness classes. The $\mathrm{NP_c}$ parameter must be in state 0 to allow bulk product dispensing (dilution and portioning in vials); once it enters state 1, dispensing is suspended for at most 20 min; if, during this time interval, $\mathrm{NP_c}$ re-enters state 0 the process is resumed, otherwise it is stopped.

The environment monitoring and control system is aimed at providing continuously a safe environment for the radiopharmaceutical production processes, human and material resources. It has been designed according to the holonic paradigm, combining a centralized behaviour to assure global, cascade pressure model and differentiated radioactivity, airborne particle and standard (temperature and humidity) production conditioning with a decentralized behaviour at parameter channel level (sensors and HVAC units) [15]. The environment monitoring and control holarchy contains three basic holons: room holons, resource holons and task holons (Fig. 4).

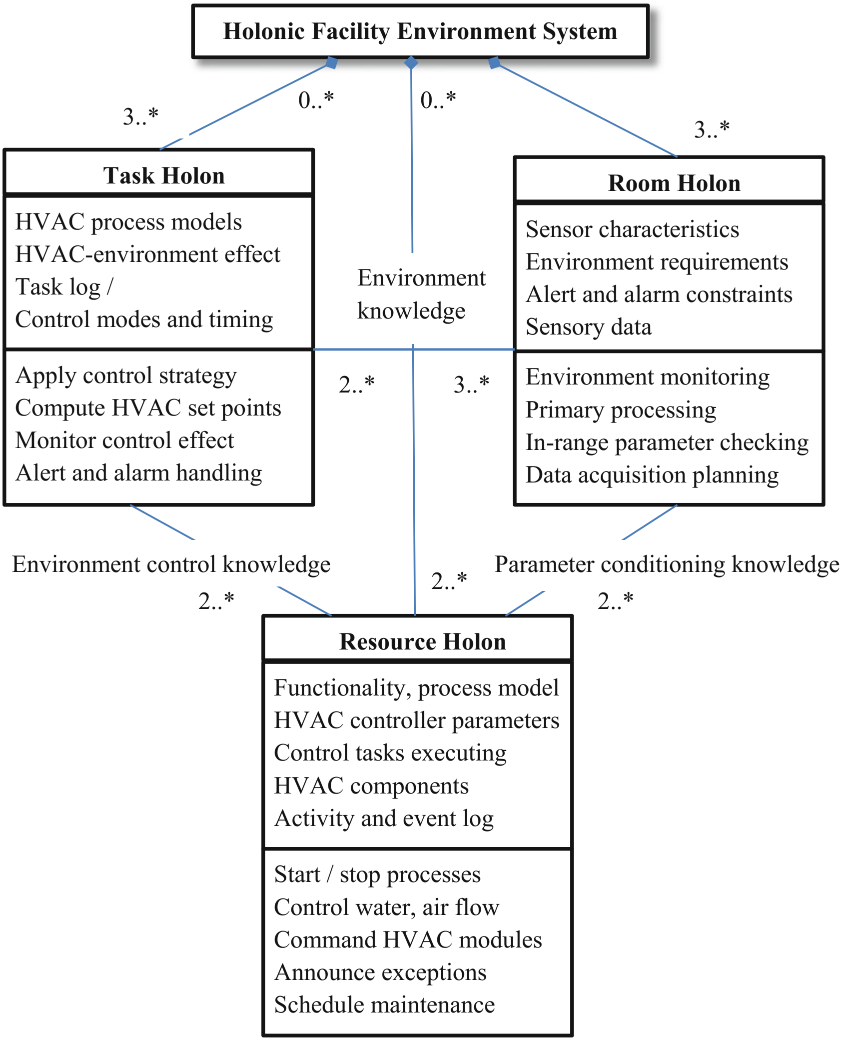


Fig. 4 The HFES holarchy, data and functions mapping to the room, resource and task holons

The objectives of the Holonic Facility Environment System are: (1) maintaining clean room parameters in the normal state for radiopharmaceutical product's quality, human and material resource safety: this problem concerns a number of technological aspects of *environment monitoring and control in the clean rooms* and adjacent closed spaces (cyclotron vault, production room, corridor and locker), i.e. which parameters to track $(P, T, RH, NP_c, H_c^*(10))$, sensor type, number and layout, operations for environment conditioning, cleanness classes for product quality; (2) *resource management and control* to efficiently provide the normal environment state: this aspect concerns the HVAC operating modes (summer/winter), control laws of the compensator, stability, controllability and sensitivity, response time, a.o.; (3) *logistics* concerning strict timeliness of production operations (operations might be delayed by environment parameters entering the alert state), due dates of product delivery and levels of product radioactivity decreasing during stages 2–5 (synthesis, portioning, quality testing and transport to the client).

The structure of the HFES architecture is built around three types of PROSA-type basic holons interacting in the holarchy represented in Fig. 4 [16]:

1. *Room holon*: contains a physical part represented by the set of sensors that measure the environment parameters, from which $H_c^*(10)$ is critical for the safety of human and material resources, whereas P, T, RH, and NP_c are vital for the product's quality; it also contains an information processing part, represented by the parameter acquisition and primary processing (linearization, compensation, unit conversion, in-range limit checking, averaging, timing, fault-tolerant data transmission, a.o.) software programs that control the monitoring of environment parameters. The room holons hold the parameter knowledge to assure the safe making of radiopharmaceutical products in processes with imposed cleanness classes; they contain the "safety and quality model" of the product's environment, in terms of "cleanness compliance".
2. *Resource holon*: contains a physical part, which is the environment conditioning resource—the HVAC unit, and an information processing part that controls the resource (the RTU SCADA: the model-based device controller, see Fig. 3). This entity holds the techniques to allocate the environment conditioning resources—the cooling water and fresh air conditioning modules (filter, heating/cooling coil, humidifier, fan), and the knowledge, operating modes and procedures to select, control and use these sub-resources to maintain the facility environment.
3. *Task holon*: represents an environment conditioning task in the HFES; it uses environment data collected from one or more sensors, and specifies the environment conditioning processes to be carried out by one HVAC sub-resource to ensure the assigned environment condition (e.g. temperature in one room) as imposed and on time. A task holon includes the HVAC control mode and law, the process model (specific for any parameter conditioning), the HVAC set point value for that individual parameter, the timing for the control process (sample period of the control process), and the way in which exceptions and alarms must be handled. A task holon can be interpreted as a single-channel "environment conditioning order" and manages the execution framework, the way this task is accomplished and its effects.

The task holon acts as a short term scheduler for environment conditioning operations. As shown in Fig. 4, the three types of basic holons exchange knowledge about the facility environment system. Room holons and resource holons communicate and exchange knowledge about parameter conditioning at single channel level; room holons and task holons exchange environment knowledge at cascade level of the global facility, while resource holons and task holons share environment control knowledge for effective execution.

- *Parameter conditioning knowledge* comprises the information, methods and techniques on how to perform a channel environment monitoring process, how to maintain that parameter in the desired value range (in the normal state) and how to bring it back in the normal state from alert and alarm states. It represents the knowledge about: (a) the characteristics of the room sensors and their disposal, (b) the capabilities of the HVAC resources and their sub-resources (filter, fan, coils and humidifier), the conditioning functions they can perform and the particular precision and time performances, (c) the estimated effects and possible exceptions that may arise.
- *Environment knowledge* represents the information and methods on how to maintain certain environment conditions imposed by the compliance with classes of cleanness A, B and C for different irradiation and production stages; these conditions are expressed at the global facility level through "cascade models" that prevent irradiated air flows contaminate the outside of the facility, and impure air flows to spread inside clean rooms degrading the controlled air. It is knowledge about the parameter set points $T_{ref}, P_{ref}, RH_{ref}, NP_{c,ref}, H^{*}_{c,ref}(10)$ that must be provided in time to maintain/re-enter the normal value range, the control modes, laws and combination of water and air flow HVAC conditioning sequence to be configured, the methods of handling exceptions and alarms and of monitoring the effects of control process execution.
- *Environment control knowledge* contains the information and procedures which allow starting the execution of HVAC control processes and tracking their progress. This knowledge refers to how to install and run process models, how to set up the timing for cooling water and air flow conditioning, how to convert room parameters set points to HVAC set points, how to install, change and update control modes and laws.

The HFES architecture includes a number of staff (or expertise) holons that assist the three basic holons in accomplishing their activity, as in the PROSA holonic reference architecture [16]. The staff holons have a global perception about the entire radiopharmaceuticals production facility in terms of layout, functionalities, adjacencies, safety constraints, classes of cleanness related to production stages, production conditioning by environmental conditions, functioning hours, inter-room pipes allowing product flows, backup devices and alarm strategies, being thus able to compute the parameter set points for individual environment monitoring and control channels.

Such functions can be solved in centralized manner; in our design, the staff holon is placed at the MTU level of the SCADA system and communicates both with the room holons to supervise monitoring and with the task holons to launch control [17].

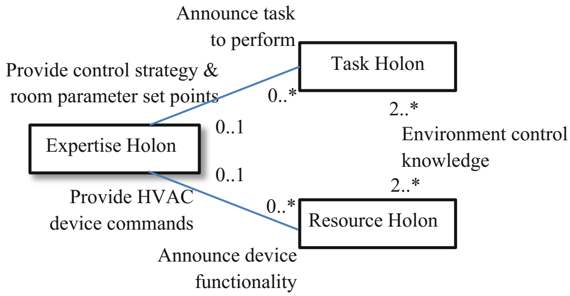


Fig. 5 The HFES expertise holon for global environment monitoring, cascade model update and set up of HVAC environment conditioning tasks

Figure 5 shows the centralized interpreter of the global facility environment data from which it prescribes the channel set points and control modes for the task holons, acting as an expertize entity.

The centralized environment observer of the radiopharmaceuticals production line is one instance of the expertise holon. Another one is represented by the centralized computing of the room parameter set points using the global cascade model of room pressures. These set point values are transferred to the task holons which compute the HVAC set points $\dot{m}_{\mathrm{water,\,ref}}, \dot{m}_{\mathrm{air,\,ref}}$ for the cool water and fresh air flow conditioning processes. All the computing tasks performed by the expertise holon can be however distributed to the task holons for local execution; in this way, values computed or decisions taken at centralized MTU SCADA level could be interpreted as advices rather than commands, thus eliminating single points of failure and providing agility in case of alert or alarm events [18].

4 HFES Implementation and Production Conditioning

The Facility Environment Monitoring and Control System designed in a dual holonic semi-heterarchical approach is currently in implementing stage in the building of the Radiopharmaceutical Research Centre of the IFIN-HH Institute in the Bucharest Măgurele area.

The architecture of the HFES, shown in Fig. 6, includes two parts: (1) the centralized subsystem responsible with monitoring and global facility environment control of cascaded parameters, maintaining a process database where historical data logs are kept, displaying real time environment parameters, product traceability information and production data, and conditioning of the radiopharmaceutical production management system; (2) the decentralized subsystem collecting environment data and controlling the HVAC units for environment conditioning [19].

The ambient parameter monitoring subsystem uses a network of pressure, temperature and relative humidity sensors, which are distributed in the closed rooms of the cyclotron-based radiopharmaceutical production line. The temperature/humidity sensors of Lumel P-18 type are interconnected in a star

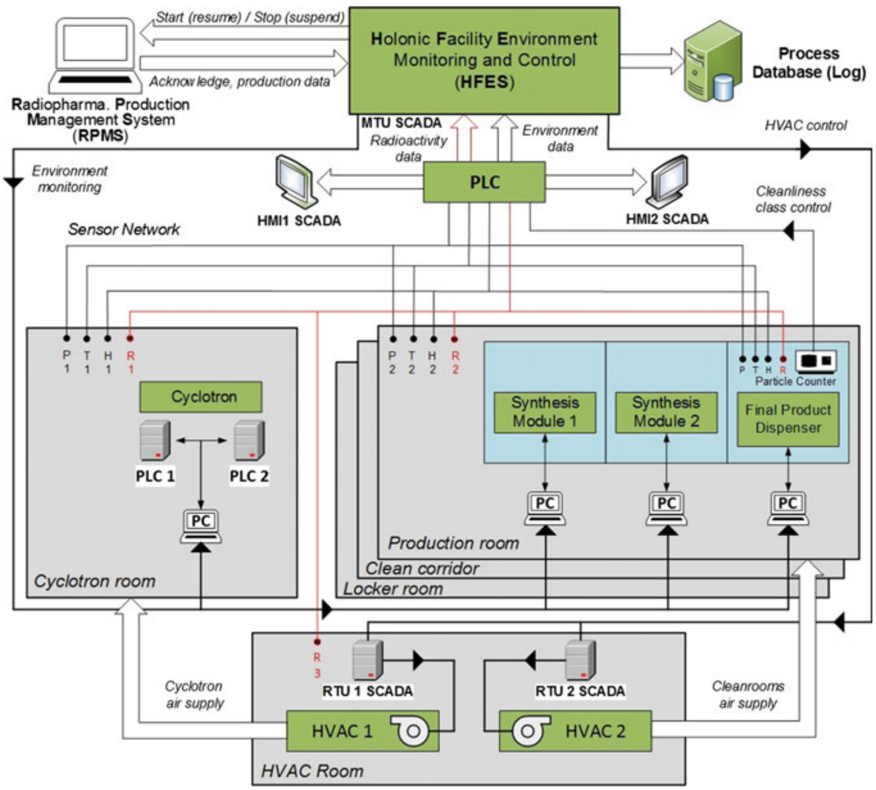


Fig. 6 The architecture of HFES with dual SCADA subsystems: MTU and channel distribution

topology using RS-485 communication; after primary processing and in-range checking, the values of the environment parameters are retrieved from a PLC data concentrator and:

- Sent to the central processor running the expertise holon to update the environment control strategy and compute the set points of cascaded parameters;
- Displayed on two Delta HMI monitors providing real time information for the human operators of the production line; the HMI1and HMI2 screens display respectively the current values of pressure and temperature/humidity room parameters.

A number of pressure sensors of analogic differential type subtract the reference (atmospheric) pressure from the ones measured in the facility's room points. The display of the differential pressure values on the HMI screens requires A/D conversion of the analogue signals in RS-485 format using Lumel SM1 devices.

Also, the values of the pressure cascades between the cyclotron vault, respectively irradiation chamber and their adjacent spaces are computed and displayed by a differential system, checking permanently whether the imposed relationships are met.

A network of Geiger-Müller sensors, denoted by R, performs radiologic monitoring of the environment by registering the equivalent flow of ambient dose (in Sv/h) in the areas inside the facility which are intensively exposed to ionising radiations; the measured values are displayed on a dedicated HMI screen and sent directly to a central HFES application for highest priority interpretation and decision support.

The central PLC acting as field data concentrator is also distributed and physically replicated in the cyclotron and production areas for system redundancy, making possible a complete heterarchical environment monitoring and control; in this operating mode, the cascade pressure set points are locally computed and correlated through communication between the local area PLCs.

The hierarchical MTU SCADA processor collects the values of ambient parameters, process and product data (environment data, information about the product during all processing stages, events occurring in the production cycle), organizes and stores them in a data storage server for historical evidence, reports, analysis and audit.

The cascaded room set points and decisions about the strategy to be instantaneously adopted by the environment control units—the HVAC of VAV (Variable Air Volume) type are decentralized being transferred to the RTU SCADA processors responsible for: establishing the timing of control actions, applying the recommended control strategy and mode, and converting the room parameter set points to HVAC set points for cooling water and air flow conditioning processes and VAV air flow distribution (admission and evacuation) to/from cascaded closed spaces. The motors of the two SCADA units have VFD (Variable Frequency Drives), the rpm signals of which are fed back to the RTU controllers for stabilization purpose (see Fig. 3).

Product execution is conditioned by room parameters monitored by HFES (Fig. 7).

The radiopharmaceuticals production management system has been designed as a Holonic Manufacturing Execution System (HMES) performing centralized, off-line production planning and distributed control of product making [4]. Figure 7 shows the relation of subordination exerted by the environment monitoring and control system over the holonic production management system.

The setup of room environment parameters is done through a Graphical User Interface (GUI); these parameter values are transferred to the expertise holon running on the MTU SCADA, which distributes them to the environment task holons. The real time measured room parameters are collected by the central and production area PLCs and compared by the task holons with their setup values. As long as the values of the environment parameter remain close to their setup values (within the "normal state" range), any started production process proceeds.

The room parameters are read at different sampling time periods; once an alert is raised for at least one of P, T and RH, production continues and simultaneously appropriate actions are initiated to bring back their values within the range

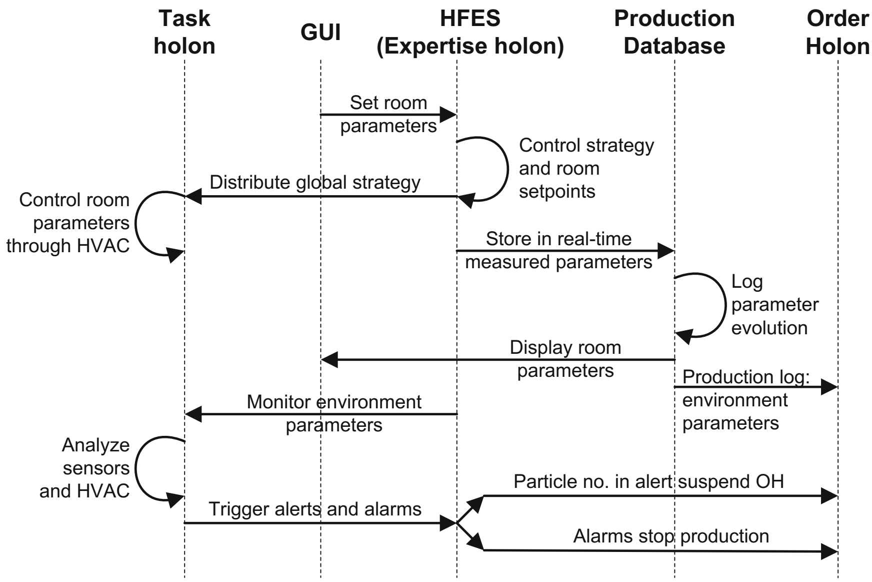


Fig. 7 HFES integration with the radiopharmaceuticals production management system

corresponding to the normal state; if these actions have not effect and the alarm state is entered, production is stopped, and the delivery of products at the off-line planned time is cancelled. The same decision is taken when the level of radioactivity expressed by the parameter $H_c^*(10)$ enters the alert value range.

A special situation is generated when the number of airborne particles in the dispenser isolating box, NP_c becomes greater than the predefined alert value; in this case, two decisions are taken:

- The execution of the order holon (OH) currently pointing at the dispensing stage for the bulk product currently in fabrication (already irradiated, chemically processed and currently subject to dispensing) is suspended. This decision is taken upon detecting that the number of airborne particles exceeds the upper limit value of the class A normal range (3520 particles of 5 μm/20 particles of 0.5 μm), irrespective of the bulk product awaiting to enter the dispensing isolator box or the dispensing process already started. This decision is sent to the HMES.
- The rate of air changes (admission of fresh airflow, evacuation of impure air) is increased in the HVAC unit connected to the dispensing isolator box, to eliminate as much as possible airborne particles.

If the value of the NP_c parameter re-enters the normal state in less than 20 min, the OH is re-activated and dispensing is started (resumed). Otherwise, the OH is definitively stopped and the current production is cancelled. Depending on the recovery time for NP_c, some of the remaining product orders for that day will be re-planned.

5 Experimental Results and Conclusions

Several experiments and tests were performed with some of the HFES components. Among these, a recovery time test for HVAC VFD performance in reducing particle concentration in a clean room was done. "Recovery time" is defined by ISO Standard 14644-3 as the time required for reducing an (artificially increased) initial particle concentration in a "class A" clean room up to the "normal state" value range.

For such cleanrooms with more than 20 air changes/hour a recovery time of less than 20 min can be expected. This test was carried out for non-unidirectional airflow systems because the recovery performance is a function of air re-circulation ratio, inlet-outlet airflow geometry, thermal conditions and the air distribution characteristics within the controlled zone.

A laser particle counter PMS model Lasair III 310C was placed inside the final product radiopharmaceutical dispenser. When the cleanroom was in "work state" with the HVAC system running and the dispenser ready for daily production process, the particle counter was started. The first read after 1 min indicated 3220 particles (0.5 µm) and 12 particles (5 µm), within the limits imposed by GMP specification.

The HVAC VFD for supply fan/exhaust fan were set at 38 kHz/26 kHz; an overpressure of 35 Pa was thus created inside the cleanroom, with the airflow set up to provide 20 air changes/hour (the number of times the air enters and exits a room from the HVAC system in one hour). The HVAC system was then turned off for 5 min and then switched back on showing 4870 particles (0.5 µm) and 80 particles (5 µm). After 20 min functioning time, the particle counter displayed 3410 particles (0.5 µm) and 16 particles (5 µm), meaning that the radiopharmaceuticals dispenser has successfully recovered its "A grade" cleanliness.

This sequence was repeated for HVAC supply/exhaust fan set at 32 kHz/20 kHz and overpressure of 25 Pa created inside the clean room with an airflow set up to provide a number of 10 air changes/hour. After 5 min switch off the number of particles was 4860 (0.5 µm)/78 (5 µm), and then, after 20 min the values shown by the particle counter were 4630 (0.5 µm)/40 (5 µm), meaning that for 10 air changes/hour the dispenser was unable to recover its "A grade" class. With these last settings, the recovery time was 43 min—unacceptable for routine production.

From these experiments it was concluded that higher supply air flows and implicit a higher number of air changes/hour significantly reduce the dispenser's recovery time. The task holons can impose for the two RTU SCADA controllers different setup coefficients, so that VFD will operate at higher frequencies until the cleanliness class is successfully restored, and then restore its normal settings to avoid high energy consumption.

The advantages of using a model-based environment monitoring and control system with distributed intelligence are: flexibility in adopting the environment control strategy (at centralized level) and applying it at facility room levels through task configuring, decoupling process execution from the physical structure (sensors, HVAC), high availability through semi-heterarchical control mode and fast recovery times at disturbances altering the environment of the clean rooms. Future research will be directed towards extended equipment instrumenting for predictive maintenance.

Acknowledgements This research work has been partially supported by the IBM FA 2016 DTM Project: “Big Data, Analytics and Cloud for Digital Transformation on Manufacturing”.

References

1. Iverson, Ch., et al. (eds.): 15.9.2 Radiopharmaceuticals, AMA Manual of Style 10th edn. Oxford University Press, Oxfordshire (2007). ISBN 978-0-19-517633-9
2. Schwochau, K.: Technetium. Wiley-VCH (2000). ISBN 3-527-29496-1
3. Ell, P., Gambhir, S.: Nuclear Medicine in Clinical Diagnosis and Treatment, Churchill Livingstone (2004). ISBN 978-0-443-07312-0
4. Raileanu, S., Borangiu, Th., Silisteanu, A.: Centralized MES with environment adaptation for production of radiopharmaceuticals. In: Borangiu, Th., Trentesaux, D., Thomas, A., McFarlane, D. (eds.) Service Orientation in Holonic and Multiagent Manufacturing, Chapter 1. Springer series Studies in Computational Intelligence, vol. 640, pp. 3–20. Springer (2016). doi:10.1007/978-3-319-30337-6_1. ISBN 978-3-319-30337-6
5. http://www.ansto.gov.au/_data/assets/pdf_file/0019/32941/Nuclear_Medicine_Brochure_May08.pdf
6. Mas, J.C.: A Patient’s Guide to Nuclear Medicine Procedures: English-Spanish. Society of Nuclear Medicine (2008). ISBN 978-0-9726478-9-2
7. Medema, J., Luurtsema G., Keizer, H., Tjilkema, S., Elsinga, P.H., Franssen, E.F., Paans, A. J., Vaalburg, W.: Fully automated and unattended [^{18}F] Fluoride and [^{18}F] FDG production using PLC controlled systems. In: Proceedings of the 31st European Cyclotron Progress Meeting, Zurich (1997)
8. Kusiak, A.: Intelligent Manufacturing Systems. Prentice Hall, Englewood Cliffs, N.J. (1990). ISBN 0-13-468364-1
9. Tang, F.: HVAC system modeling and optimization: a data mining approach. M.Sc. thesis, University of Iowa. http://ir.uiowa.edu/etd/895 (2010)
10. Hadjiski, M., Sgurev, V., Boishina, V.: Multi agent intelligent control of centralized HVAC systems, energy saving control. In: IFAC Workshop on Energy Saving Control in Plants and Buildings ESCPB’06, vol. 1, pp. 195–200 (2006)
11. Gopika, S.: Agent based HVAC optimization model for building energy efficiency. In: IEEE 9th International Conference on Intelligent Systems and Control (2015). doi:10.1109/ISCO.2015.7282239
12. Nassif, N., Kajl, S., Sabourin, R.: Simplified model-based optimal control of VAV air conditioning systems. In: Proceedings of 9th International IBPSA Conference, Montréal, Canada (2005)
13. Tsai, W.T.: Service-oriented system engineering: a new paradigm. In: Proceedings of the 2005 IEEE International Workshop on Service-Oriented System Engineering (SOSE’05), IEEE Computer Society. (2005). ISBN 0-7695-2438-9/05
14. De Deugd, S., Carroll, R., Kelly, K.E., Millett, B., Ricker, J.: SODA: service-oriented device architecture. IEEE Pervasive Comput. **5**(3), 94-C3 (2006)
15. McFarlane, D., Giannikas, V., Wong, C.Y., Harrison, M.: Product intelligence in industrial control: theory and practice. Annu. Rev. Control **37**, 69–88 (2013)

16. Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P.: Reference architecture for holonic manufacturing systems: PROSA. Comput. Ind. Spec. Issue Intell. Manuf. Syst. **37** (3), 255–276 (1998)
17. Leitao, P., Restivo, F.: ADACOR: a holonic architecture for agile and adaptive manufacturing control. Comput. Ind. **57**(2), 121–130 (2006)
18. Răileanu, S., Anton, F., Iatan, A., Borangiu, Th., Anton, S., Morariu, O.: Resource scheduling based on energy consumption for sustainable manufacturing. J. Intell. Manuf. (2015). doi:10.1007/s10845-015-1142-5 Print ISSN: 0956-5515
19. Novas, J.M., Bahtiar. R., Van Belle, J., Valckenaers, P.: An approach for integration of a scheduling system and a multiagent MES. Towards a collaborative framework. In: Proceedings of 14th IFAC Symposium on INCOM'12, Bucharest, pp. 728–733, IFAC PapersOnLine (2012)

Disruptions Are the Norm: Cyber-Physical Multi-agent Systems for Autonomous Real-Time Resource Management

Petr Skobelev and Damien Trentesaux

Abstract This paper analyses the new requirements for real time resource management systems based on multi-agent technology. It shows the growing demand for developing autonomous systems which combine resource allocation, scheduling, optimization, communication with users and control in one cycle and can respond rapidly to unexpected events in real time. To solve the problem, cyber-physical multi-agent systems are considered. The paper also analyses the new impact which such systems bring into design of modern systems on the way from smart Internet of Things—to new organizations and ways of user motivation.

Keywords Autonomous systems • Multi-agent technology • Resource management • Cyber-physical • Real time

1 Introduction

In the last decade, global economy has generated such levels of complexity, uncertainty and dynamics that rend real-time decision-making for resource management critical for business efficiency, growth and sustainability [1].

Under these circumstances disruptive unpredictable events of any kind become more the norm rather than the exception, for example: new VIP order arrives, the already scheduled and half-implemented order is cancelled, discrepancies between

P. Skobelev (✉)
Smart Solutions, Ltd, Moskovskoye shosse 17, office 1201,
443013 Samara, Russia
e-mail: petr.skobelev@kg.ru

P. Skobelev
Samara University, Samara, Russia

D. Trentesaux
LAMIH, UMR CNRS 8201, University of Valenciennes et Hainaut-Cambrésis,
UVHC, Le Mont Houy, 59313 Valenciennes Cedex, France
e-mail: damien.trentesaux@univ-valenciennes.fr

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_25

planned actions and reality increase, resources are not available or become more expensive, traffic jams, delays and breakages occur, etc.

The typical reaction of current businesses to such disruptive events is to bring more oversized resources, to increase costly safety stocks and to inject slack-time. Such measures dramatically reduce the utilization and efficiency of business and push the increase of costs onto the customer [2], the risk being thus a drastic decrease in competitiveness of companies regarding the international competition.

This tendency is already a changing game for managing resources in many domains: manufacturing and transport, supply chains, aerospace, railways, etc. [3].

As a result new generations of growing businesses would like to become fully on-line and event-driven to make resource allocation, scheduling, optimization and control "on the fly" in real time (in one "cycle"—when all these processes can be triggered by events and combined), to avoid peaks and to control bottlenecks in resource utilization, to reduce idle time and empty miles, etc.

To support this global shift of paradigm to real-time economy the new generation of industrial systems for resource management need to become more intelligent, fast-responsive, adaptive and flexible, scalable and reliable while being able to optimize their operations as much as possible and at a longer time span.

Gartner, the world's leading information technology research and advisory company, says: "future is a digital thing" [4]. We would like to add: "a smart thing" as well—as a key factor to provide increase in levels of autonomy and efficiency.

From our point of view, the introduced typical reaction of current businesses to disruptive events comes from the centralized, highly hierarchical, monolithic and deterministic architectures inherited from the 70s with the principles of Computer Integrated Manufacturing (CIM). Current ERP systems are aligned with these principles. In the past years, disruptive events were the exception and the principles of CIM were consistent. Nowadays, disruptive events are the norm and the old principles find their limits. Since the 90s, concurrent paradigms have emerged, based on a more reactive, local management of operations, assuming that disruptions are no more negligible: distributed control, multi-agent control, holonic paradigm to name a few. Meanwhile, these concepts hardly penetrate the industry and the real market because of their lack of predictability and guarantee of performances (in terms of cost, return of investments, reliability, safety…) [5]. In industries, similar approaches are developed, based on the assignment of more autonomy to lower level controls, such as the Kanban system or the concept of Autonomous Production Entity (ex: EAP in the French car industry first rank suppliers). But all these evolutions remain dependent on a strong relationship with a hierarchically organized decision-making process still inherited from the CIM principles.

Nowadays, the constantly changing environment, the constraints from competition, the nervousness in customers' behaviour and the complexity of industrial processes rend these disruptions still more important in number and consequences. From our point of view, disruptions are now the norm and this should be considered as is from the beginning of the design of a manufacturing management system. Some research concepts are currently being improved to take this new norm into consideration, for example using bio-inspired fundamental principles of self-organization and evolution based on multi-agent technology [6, 7]. Meanwhile, in our opinion, this is not sufficient since the principle "disruptions are the norm" is not explicitly considered during the design process of such systems.

In this context, this paper suggests a new concept of autonomous systems based on cyber-physical multi-agent technology for real time resource management considering that disruptions are the norm. To argue in favour of this statement in the first part of this paper we will make a brief overview of existing models, methods and tools for decision-making and will show their limitations and constraints. The second part will be focused on the new generation of multi-agent systems, which are organized as cyber-physical systems and support full cycle of resource management—to bring new levels of autonomy in decision-making.

In the third part we will discuss the concept and its first results and make recommendations for future developments.

2 New Requirements for Real-Time Resource Management

The traditional approach for resource management inherited from the CIM principles includes strategic and operational resource allocation, scheduling and optimization which are usually made separately and on different levels.

Strategic resource management refers to long-term plans according to information about future demand, capacities and costs and is made periodically in batches for a specific time period from several months to years. Operational resource management provides more granularity and deals with months-weeks-days. It also takes into consideration specific technology for order implementation, availability of equipment and workers, their skills, materials and instruments, etc.

Traditionally both approaches are focused on planning problems which are addressed by using heuristic, rule-based and mathematical optimization solvers, which run in batch mode under the strong assumption that all orders and resources are given in advance and do not change during computations.

Currently a number of mathematical optimization solvers are available on the market, e.g., IBM ILOG CPLEX Optimizer, Xpress Optimizer, etc. [2]. These solvers implement different methods of linear or dynamic programming, constraint programming and other methods, based on combinatory search of options, for example, the branch-and-bound methods. Such methods and algorithms are usually centralized and deterministic, top-down and sequential in analyzing all possible options which is very time-consuming and based on sequential achievement of solutions. Heuristics and meta-heuristics methods may combine different concepts and techniques, for example, tabu search, simulation annealing or ant optimization, to reduce the number of combinatorial options and to provide local optimizations to reduce the computational time. To compensate the loss of quality of such solutions, these methods try to use randomization and non-deterministic algorithms to explore and find ways to avoid local optimums.

Table 1 Key issues and consequences in resource management

The issue of resource management	The consequences when disruptions are the norm
Planning is usually considered as top-down centralized batch process but in practice it is continuous, distributed and interactive	Unpredictable events must be propagated to the top of organization and one needs to collect and prepare manually all the data for planning and decision-making. The independency (partition) of problems principles can no longer be assumed: an issue emerges when previous ones are still not solved
Delays in event processing cause growing gaps between plans and reality	Decision lags increase, thus leading to possible instability (control theory principle)
A number of decision-makers with different decision-making criteria, logic of decision-making, constraints and preferences need to be involved	Events may come at any time from any of decision-makers involved; this will require triggering of chain of decision revisions and changes. Impact of serial disruptions is seen globally, thus it is hard to evaluate and to test "what if?" solution scenarios
High variety of objects and relations, factors, situations, orders and resources, technological operations, etc., mass customization and high competition	A number of different types of events need to be considered: new order arrives, order cancelled, resource broken, etc.
Combinatorial search of options is NP-hard and very time-consuming and may require long hours even for small businesses	Many events require immediate reaction (seconds and minutes), otherwise money and time will be totally lost with the loss of efficiency. Increased complexity increases reaction time
Orders and resources are not usually given in advance	Events are coming at unpredictable time and dynamically change orders and resources, company strategy, etc. Dynamic and stealth bottlenecks are faced with
Many specific individual criteria, preferences and constrains for orders, products and technology, workers and equipment are often ignored	There is no "optimal" solution for processing of new event—new balance of many interests of decision-makers must be found (consensus)
Interconnectivity of decisions may trigger long ripple effects	New events may trigger full reconstruction of routes, plans and schedules
Situation-dependencies: the number and value of criteria depend on results and could change during computations	New events may change criteria, preferences and constraints for creating and revising schedules
Domain-specific knowledge is usually hidden and difficult to formalize and continuously evolve during day-to-day operations	It is hard to learn from experience and user interventions in decision-making process and manual adjustments of schedules, incl. constraints overriding Big data approaches need time to be completed, which is no more possible in a constantly changing environment of data and parameters
Part of knowledge for decision-making is available only on "ground" level	It requires strong communication, interaction not only with managers, but also with workers, drivers, etc.

Other key issues of traditional resource management under circumstances of uncertainty and dynamics are presented in Table 1.

As a result, the majority of traditional solutions require hours to make computations for real businesses but automatically generated plans are in practice only 40% feasible, at the same time requiring full re-run in case of user intervention or arriving of new events which always take place.

The new generation of industrial software for autonomous resource management must take into consideration that now "disruptions are the norm" when dealing with the complexity of the problem and when bringing radically new models, methods and tools for industry applications.

3 Multi-agent Technology for Resource Management

Multi-agent technology usually is considered as one of new software engineering approaches for developing smart applications combining object oriented programming with parallel computations, AI components and telecommunications.

Multi-agent systems (MAS) differ from traditional software by distribution, decentralization, modularity, flexibility and robustness. But the more important is that it offers new ways of solving complex problems, including resource management with the use of bio-inspired principles of self-organization and evolution—when classical mathematical methods work not very well or don't work at all [8].

In spite of the fact that MAS are mainly developed by academia, universities and research organizations, there are a number of first industrial applications of MAS which are already bringing real value for business [9–11].

In some applications, it has been show that MAS technology helps to increase efficiency up to 20–40% [3].

But a new challenge is coming with growing tendency of total "uberization of all businesses" from forward-thinking lead users which are requiring new kind of "Google-mobile for business".

Such type of applications will require fully autonomous solutions with minimum number of managers and will automatically set priorities, forecast new orders and other events, process new orders and serve resources individually, generate and analyse options for orders execution, recognize patterns, make schedules and adjust them in case of not-foreseen events, communicate, negotiate and coordinate decision with users, learn from experience, etc.

Current multi-agent solutions and applications are mainly used on the level of the basic support of strategic decision making; first of all, it means generation of options, simulations and analysis of results, making choice for further manual controlling of plans execution.

But the new objectives require fully autonomous cycle of decision making at the operational level whose core part is the reaction on events, continuous re-scheduling and execution of plans—as it works for any living organisms.

In this paper we outline the possible solutions by introducing the concept of cyber-physical multi-agent systems for autonomous resource management.

4 Cyber-Physical Multi-agent Systems for Autonomous Resource Management

Multi-agent technology can be considered as way of building Smart Internet of Things for Industry 5.0—where all things will not only represent physical objects but will be able to react on events, make decisions and plans and communicate and coordinate these plans with other agents and end-users.

This step is considered as the key differentiation from Industry 4.0.

Such a paradigm suits the "disruption is the norm" principle. Moreover, from our point of view, it generates an interesting paradox: there won't be any disruptions anymore since everything is a disruption.

In such non-deterministic systems, based on parallel and asynchronous computations and coordinated decision-making processes, the agents representing the real world entities not only continuously interact with each other trying to achieve a quasi-optimized solution in a more robust, flexible and agile manner but also simulate the behaviour in real world. In this case MAS can be considered as self-completed virtual world which is in fact the computer model of real world where the software agents make the main decisions instead of people as decision-makers. As a result, it could be possible to design autonomous resource management system as a cyber-physical system (CPS).

An example of an autonomous multi-agent solution which is organized as a CPS for trucks management is shown in Fig. 1.

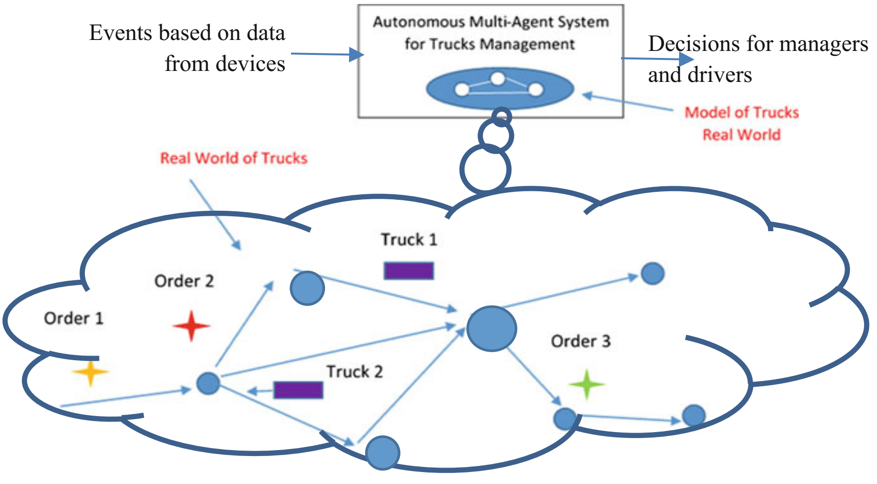


Fig. 1 Cyber-physical system architecture: an example for trucks management

To support such kind of decision-making, it is required to provide the full cycle of resource management including fast reaction on events, allocation of orders to resources, scheduling of orders/resources, optimization of orders execution (if time is available), communication with users, monitoring of plan execution, re-scheduling in case of growing gap between the plan and reality, and, finally, learning from experience to get and make available new domain knowledge which affects decision-making.

As shown in Fig. 1, the system considers the multi-agent world as a model of real world which is continuously updated by events and is used for generating decisions through negotiations of agents representing orders, trucks, etc. The decisions made are forwarded to mobile phones of users in real world as instructions and schedules which can be adapted by users.

Such cycles help autonomous system to keep schedule up-to-date with reality and generate reasonable and feasible decisions.

In this context the multi-agent technology is suitable to face the current challenges associated with industrial strategic and operational planning problems, especially when facing the need to be extremely adaptive.

5 New Functionality of Autonomous Solutions When Disruptions Are the Norm

The new functionality of autonomous multi-agent solutions will face challenges and exert impact on the following levels:

- *Smart Internet of Things*—it will be required to support the full cycle of decision-making on the level of physical or abstract entities.
- *Models of Technology Processes and/or Business Processes*—the classical approach for control and accounting will be extended to the models of physical world objects and processes.
- *Decision-Making Support Systems*—start with support of decision-making but will move to fully autonomous systems.
- *Knowledge Bases*—extremely important for decision-making as a tool to separate domain knowledge from source code and provide learning from experience for the future.
- *Organizations*—will be significantly changed by "uberization" of all resources, discovering new opportunities for people, motivating them by sharing profit in case of more efficient decision-making.

As a result, not only the software solutions will become more open, less centralized and hierarchical, but also the organizations. At the next stage they will become more open, distributed, flexible and efficient overriding barriers between departments for supporting teamwork and developing human-centric organizations with full focus on talented, active, knowledgeable and well-performed people.

Cyber-physical multi-agent systems for autonomous real-time resource management must be considered as a decision-making tool for such innovations helping to handle unpredictable disruptions which are becoming the norm now.

6 Conclusion

The aim of this paper is to develop the new concept of autonomous systems based on the principles of cyber-physical systems, multi-agent technology and Internet of Things where disruptions are considered as the norm.

New generation of autonomous multi-agent systems will allow enterprises to move to real-time economy and help improve efficiency of resources, quality of the service, reduce expenses and time, risks and penalties.

Acknowledgements This paper was prepared with the financial support of the Ministry of Education and Science of the Russian Federation—contract No. 14.578.21.0137, project unique ID is RFMEFI57815X0137.

References

1. IBM: Capitalizing on Complexity? Insights from the Global Chief Executive Officer Study. IBM USA (2010)
2. Leitão, P., Skobelev, P., Pazienza, G., Kazanskaia, D., Barbosa, J.: Agent-based production planning and optimization. In: Adaptive Rump Up Management. Springer (2016) (in publishing)
3. Rzevski, G., Skobelev, P.: Managing Complexity. WIT Press, London-Boston (2014)
4. Gartner: Top strategic predictions for 2016 and beyond: the future is a digital thing (2016). https://www.gartner.com/doc/3142020?refval=&pcp=mpe. Accessed 10 June 2016
5. Trentesaux, D.: Distributed control of production systems. Eng. Appl. Artif. Intell. **22**(7), 971–978 (2009)
6. Wooldridge M.: An Introduction to MultiAgent Systems. John Wiley & Sons (2009)
7. Luck M., et al.: Agent technology: computing as interaction (A Roadmap for Agent Based Computing). http://www.agentlink.org/roadmap/. AgentLink (2005) Accessed June 15 2016
8. Skobelev, P.: Multi-agent systems for real time adaptive resource management. In: Industrial Agents: Leitão, P., Karnouskos, S. (eds.) Emerging Applications of Software Agents in Industry. Elsevier (2015)
9. Leitao, P., Vrba, P.: Recent developments and future trends of industrial agents. In: Proceedings of HoloMAS-2011, Holonic and Multi-agent Systems for Manufacturing, vol. 6867, Lecture Notes in Computer Science. Springer (2012)
10. DeLoach, S.A.: Moving multi-agent systems from research to practice. Int. J. Agent-Orient. Softw. Eng. **3**(4) (2009)
11. Műller, J., Fisher, K.: Application impact of multiagent systems and technologies: a survey. In: Agent-Oriented Software Engineering. Springer (2013)

Exploring the Design Space for Myopia-Avoiding Distributed Control Systems Using a Classification Model

Tianyi Wang, Henning Blunck and Julia Bendul

Abstract Avoiding myopia, suboptimal behaviour, caused by the limited information horizon and computation capacity of agents, has been recognized as a major design challenge for the future academic development and industrial adoption of distributed production control systems. In [3] existing literature from various research streams has been reviewed to classify design decisions that can be made to avoid myopic decision making. In the present paper, this model will be validated by mapping different paradigms of distributed control onto it. Through this exercise, an initial validation of the proposed classification model can be attained and a starting point for a classification of existing distributed production control approaches based on design features is provided. This will help designers of distributed architectures in production control to better understand their design space, take deliberate steps towards the avoidance of myopic behaviour, and identify unexplored areas within the design space.

Keywords Production control · Myopia · Distributed decision making · Classification model

1 Introduction

Roughly 30 years after the idea of agent-based, heterarchical production control architectures was first proposed (e.g. [9]), the industrial adoption of distributed control architectures is still lacking behind [7, 15]. This gap has been attributed to a

T. Wang · H. Blunck (✉) · J. Bendul
School of Mathematics & Logistics, Jacobs University Bremen gGmbH,
28759 Bremen, Germany
e-mail: h.blunck@jacobs-university.de

T. Wang
e-mail: t.wang@jacobs-university.de

J. Bendul
e-mail: j.bendul@jacobs-university.de

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_26

series of factors, including a lack of standards, development tools, high necessary investments, and a continuing prevalence of reductionist thinking and training among production system designers and planners [6, 14–16, 20].

In this contribution, we focus on another feature of distributed, agent-based approaches: the presence of *myopic behaviour*. Myopic behaviour, defined as an agent's inability to assess the consequences of its decisions in the future and on other agents [31] can lead to undesirable emergent system properties such as lack of performance, unpredictability, etc. (e.g. in [15, 20, 23]). Myopia in agents' decision making may arise from (c.f. [2, 3]):

- the planning problem decomposition,
- selfish decision making,
- decision making based on local information only, and
- bounded rationality (capacity-constraints on decision making).

The desire to avoid myopic behaviour has led to an early abandonment of purely heterarchical control systems [6] and since then, the assumption that features of heterarchy and hierarchy should both be present in a control architecture has become an often re-iterated tenet of production control research (c.f. [5, 8, 17, 30]).

It is then no surprise that many proposed distributed control approaches have tried to address this trade-off and improve the performance or reliability of distributed control systems. However, as noted in [7, 25], existing attempts to design distributed control systems have largely been based on experience and intuition, lacking the intention and analytical tractability to systematically investigate the relationship between the degree of autonomous decision making and emergent system properties such as performance, reliability, etc.

In [3], the authors have made a first step towards the goal of characterizing a design space (in the sense of [13]) of distributed production control systems, by reviewing contributions from the field of production control itself, as well as from multiple other streams of research that are concerned with the investigation of the dynamics of decision making entities (such as Game Theory and Statistical Physics). The proposed classification model is shown in Fig. 1.

While in [3], the classification model was primarily interpreted as a visual aggregation of previous research in the domain of distributed control and myopia avoidance, this contribution seeks to validate the model as a *design tool* and discussion base for the future development of distributed production control systems. Leaning on [19], a model may be considered validated if it exhibits sufficient accuracy within the domain of applicability and the intended application context. The intended application scenario here is to allow designers of distributed production control approaches to classify existing approaches and visualize commonalities, differences, and unexplored design alternatives. This paper seeks to validate the model by mapping a selection of popular distributed control architectures onto it. The focus will deliberately be expanded beyond agent-based production control approaches to the more general class of distributed production control architectures. The validation

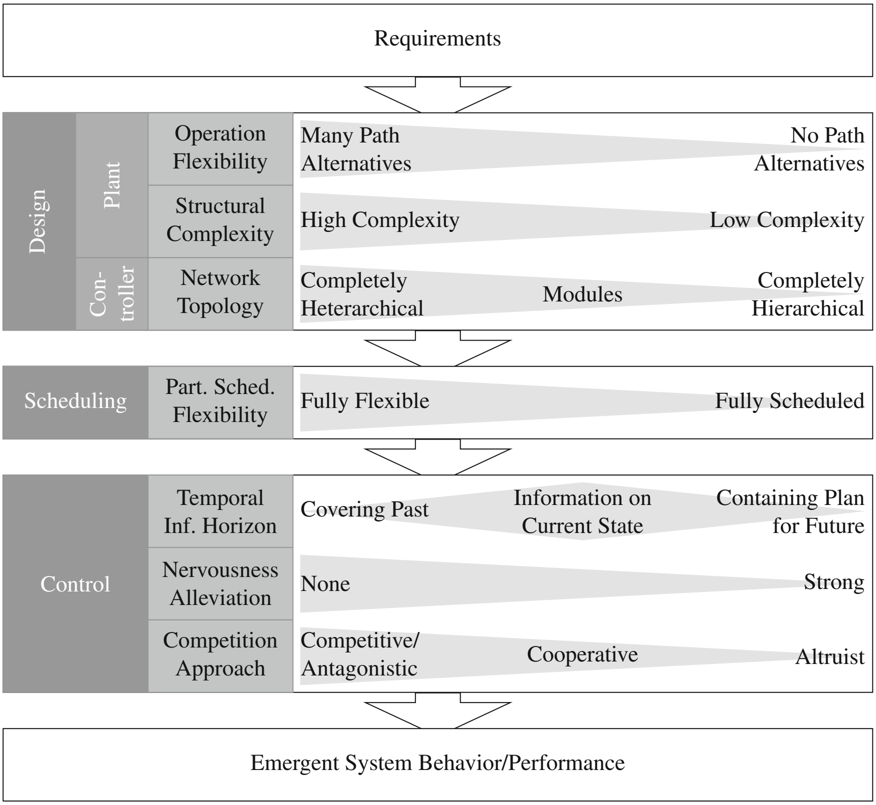


Fig. 1 Classification model of design choices affecting the degree of myopic behaviour in distributed production control as proposed in [3]

attempt shall be considered successful if (1) different proposed control architectures are differentiable in the classification and (2) related control architectures (that have evolved from another, etc.) show similar signatures. If both goals are reached, we may conclude that the classification model is a feasible vehicle to understand, compare, and discuss various proposed architectures for distributed production control and that it may be used for the selection of existing and/or the development of novel architectures in the field.

The remainder of this paper is structured as follows: In Sect. 2, the choice of control archetypes mapped onto the classification model in Sect. 3 will be motivated. The result is shown in Fig. 2 and the implications are discussed in Sect. 4.

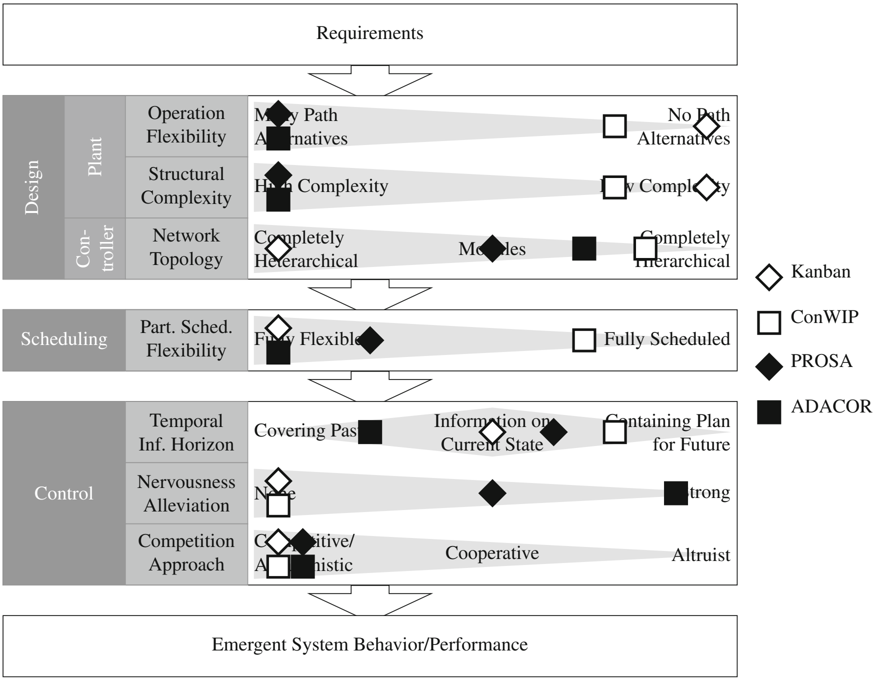


Fig. 2 Mapping of reviewed control architectures onto the classification model. *White nodes* represent classical PULL approaches. *Black nodes* represent HMS architectures

2 Selection of Distributed Control Architectures

We will consider four archetypes of distributed production control: the first two (Kanban and ConWIP) are non-agent based, distributed control architectures, that have gained popularity as part of the Lean-Movement. From the domain of Holonic Manufacturing Systems (HMS), we will consider the PROSA [24] and ADACOR [1, 12] reference architectures, given their wide application and good coverage of requirements posed on agent-based manufacturing control systems [8]. With these four architectures we are able to make both inter-group and intra-group comparisons and can validate our model beyond the realm of agent-based control approaches.

3 Mapping the Control Architectures onto the Classification Model

For reasons of brevity, we cannot give a detailed introduction into the four architectures here. The reader is referred to the given literature for a more in-depth introduction. We will restrict ourselves to discussing the architectures with respect to the dimensions of myopia control of the proposed classification model.

3.1 Kanban

In terms of physical plant design, Kanban is usually applied in systems with low material-flow flexibility and complexity [5, 18, 21]. The topology of the control network in a Kanban system is completely heterarchical with only very limited information exchange on same hierarchy levels [22]. Scheduling systems within a Kanban system are entirely made online, not relying on pre-calculated schedules. Consequently, Kanban systems are generally suitable for a stable, leveled production environment [10].

Since Kanban systems exchange information only through Kanban cards (representing a quantized information about recent demand), the information horizon of such systems is very short [21], covering neither the long-term demand development, nor estimates of future behaviour. They also possess little nervous alleviation measures since (again) they assume stable production environments. Finally, at the different inventories along the production process, Kanban cards are released (and queued at the supplying process) without consideration of cross-effects on other products. Kanban systems should hence be characterized as competitive since they do not consider the benefit of other agents (workstations or inventories).

3.2 ConWIP

The deficits of Kanban in more challenging environments have lead to the development of hybrid approaches between Kanban and more classical PPC systems (c.f. e.g. [10]), with ConWIP [21] among the most famous offsprings. Contrary to Kanban, ConWIP is less restrictive w.r.t. the physical layout. It can be implemented in environments where demand is less stable and the product mix is more diverse [18, 21]. ConWIP then allows to extend PULL production principles to (somewhat) complex material flow patterns, with academic experiments going as far as job shop environments [18]. The focus however still lies in the realm of line and flow shop productions with relatively stable demand [11, Chap. 10.6.2].

ConWIP has a more hierarchical control network topology, because the card-based information exchange only occurs between the last and first operation, where orders are assigned to cards based on a centrally held and optimized *backlog list* [21]. This centrally computed order-release list pre-determines the allocation and many sequencing decisions. ConWIP systems hence exhibit, by comparison to the other archetypes reviewed here, a low degree of schedule flexibility.

Where these release lists rely on demand forecasts, they can provide an information horizon extended towards the future (Spearman and Hopp suggest that the backlog list could draw on master production scheduling [21]). As in Kanban, there are no nervousness alleviation methods applied in ConWIP systems because system entities take no autonomous decisions. There are also no recognizable efforts towards cooperative behaviour, beyond potential considerations taken in the generation of the backlog list.

3.3 PROSA Reference Architecture

Our discussion of the PROSA reference architecture (originally proposed in [24]) here is based on the descriptions/applications in [26, 28, 29].

The PROSA architecture aims to pose no restrictions on physical layout. It is commonly applied to job shop style production systems (e.g. in [28]), with both numerous and interwoven (i.e. complex) material flow options, where orders may interfere with each other significantly and repeating or parallel structures do not prevail.

The quality, roles, and communication methods of PROSA holons have been explained exhaustively in [24, 26]. These discussions show that in a PROSA architecture, information is exchanged in both vertical and horizontal ways. Vertically, holons in higher hierarchy advise other independent subordinates; horizontally, holons on the same level also exchange various information or belong to multiple holarchies simultaneously. Holarchies represent (by their very definition) an intermediate state between hierarchy and heterarchy (c.f. [24, 26] and [4, Chap. 2]).

In [28], a method to include a predefined production plan into the PROSA agents' consideration was demonstrated. The research paper shows PROSA's capability to cooperate with traditional scheduling systems. However PROSA is still independent from schedules as they are not a required input for the system. Therefore, we shall not consider partial schedules as a commonly applied measure of myopia reduction.

With "delegate MAS", a system to incorporate distributed planning within the context of PROSA was presented in [26], yielding a consistent (across agents) perception of the near future.

System nervousness has been identified as an issue within the PROSA community. A common approach is to require intention changes to create benefits that exceed a certain threshold, thus limiting frequent plan changes [29].

Where agents can make autonomous decisions (i.e. dot not follow an agreed upon schedule), other agents' utility functions are generally not included into an agent's decision making function, rendering their competitive behaviour non-cooperative.

3.4 ADACOR Reference Architecture

The ADACOR reference architecture was proposed in [12] and has been extended over time [1] (also known as ADACOR^2). In this section, ADACOR will primarily be compared with PROSA to determine its relative position in the design space.

ADACOR and PROSA are very similar with respect to the physical material flow layout of typical application settings: Like PROSA, ADACOR is designed for applications in job shop environments [12]. Therefore, in terms of both physical complexity and flexibility, ADACOR and PROSA systems take the same position, at the least restrictive (and most myopia prone) end of the scale.

When comparing the control network topology, the most significant difference between ADACOR and PROSA is the introduction of supervisor holons in ADACOR. While PROSA's staff holons are an optional feature for the integration of legacy systems, the supervisor holons in ADACOR act as constantly present coordinators within holarchies [12, 24, 26]. ADACOR supervisor holons can coordinate the behavioural and structural evolutions of the system, maintaining a more hierarchical control approach in the absence of environmental changes [1]. This is in contrast to the fluid approach PROSA takes, placing ADACOR more to the hierarchical end of the control network topology scale. This strategy can be viewed as a counter-myopia measure as it tackles potential chaos situations in heterarchical structures [1].

Similar to PROSA systems, a production schedule is not necessary for an ADACOR system to function, as the production is coordinated within the system by supervisor holons. As no method of integrating master production plans is mentioned for the ADACOR systems discussed, their schedule flexibility is set to maximum.

It can be observed that the temporal information horizon is different in the two systems. PROSA holons may take into account the future system status by accessing the resource reservation queue [27], whereas ADACOR holons rely on historical data for decision making. Specifically, the learning technique takes past decisions into consideration and the reorganization process may be simplified by adopting previously applied structures [1]. Compared to the delegate MAS approach, ADACOR implements a distinctly different approach to extend the information horizon and limit myopia.

PROSA and ADACOR also differ in nervousness control strategies: when the system senses a need to change, ADACOR[2] applies an elaborate, multi-layered evaluation approach, that takes into account the reaction time, the accepted error of the goal, and responsiveness to reach the goal [1]. Comparing to the simple threshold in PROSA, ADACOR's nervousness alleviation measure is more elaborate.

Finally, ADACOR agents are designed as selfish competitors. By default, they do not consider the benefits of other agents when making decisions.

4 Conclusion

The result of our discussion is shown in Fig. 2, mapped to the proposed classification model.

Across years of development and evolution in technology and fashions in production control research, distributed control architectures exhibit distinct traits of myopia control that can be classified along the dimensions proposed in [3]. Comparing the two PULL approaches alone clearly shows that aiming at more complex production environments requires increased "investment" in other forms of myopia control (here: hierarchical system structure and and an extended temporal information horizon). The transition from PULL principles towards agent-based production control architectures may be understood as a continuation of this effort to bring distributed control to ever more complex production environments: Comparing PULL principles with HMS reference architectures, a clear shift towards more complex production systems is seen, demanding in turn the implementation of more sophisticated myopia control approaches in the realm of production scheduling and control.

The discussion further underpins the hypothesis that, for a distributed control approach to work, some form of measure against myopic behaviour has to be taken. The proposed classification model can be applied to conceptualize a design space that includes measures against myopia. It can be used to differentiate between existing distributed control architectures and for the development of new production architectures that are both: (1) distinct from existing approaches, i.e. explore a so far untouched region of the design-space and (2) exhibit basic features to reduce myopic behaviour, making their industrial adoption more likely.

The paper is meant to provide an initial proof of concept, based on the "textbook" definitions of the architectures reviewed. The academic discussion on these concepts, let alone their implementations in practice, are more diverse and the differences more fluid then described here. The classification can be further developed into details and more distinctions can be identified, when specific systems are discussed.

The second obvious limitation is the abstract nature of most of the dimensions of the model, making it impossible to assign absolute values and positions. The discussion in Sect. 3 tried instead to reason for the relative position of each discussed architecture across the dimensions of the classification model.

References

1. Barbosa, J., Leitão, P., Adam, E., Trentesaux, D.: Dynamic self-organization in holonic multi-agent manufacturing systems: the adacor evolution. Comput. Ind. **66**, 99–111 (2015)
2. Blunck, H., Bendul, J.: Invariant-based production control reviewed: mixing hierarchical and heterarchical control in flexible job shop environments. In: Mařík, V., Schirrmann, A., Trentesaux, D., Vrba, P. (eds.) Industrial Applications of Holonic and Multi-Agent Systems, Lecture Notes in Computer Science, vol. 9266, pp. 96–107. Springer International Publishing (2015)
3. Blunck, H., Bendul, J.: Controlling myopic behavior in distributed production systems—a classification of design choices. Procedia CIRP 57, 158–163 (2016). Factories of the Future in the digital environment—Proceedings of the 49th CIRP Conference on Manufacturing Systems
4. Botti, V., Giret, A.: ANEMONA—A Multi-Agent Methodology for Holonic Manufacturing Systems. Springer Series in Advanced Manufacturing, Springer, London (2008)
5. Cochran, J.K., Kaylani, H.A.: Optimal design of a hybrid push/pull serial manufacturing system with multiple part types. Int. J. Prod. Res. **46**(4), 949–965 (2008)
6. Dilts, D., Boyd, N., Whorms, H.: The evolution of control architectures for automated manufacturing systems. J. Manuf. Syst. **10**(1), 79–93 (1991)
7. Farid, A., Ribeiro, L.: An axiomatic design of a multiagent reconfigurable mechatronic system architecture. Ind. Inf. IEEE Trans. **11**(5), 1142–1155 (2015)

8. Giret, A., Trentesaux, D.: Software engineering methods for intelligent manufacturing systems: a comparative survey. In: Mařík, V., Schirrmann, A., Trentesaux, D., Vrba, P. (eds.) Industrial Applications of Holonic and Multi-Agent Systems, Lecture Notes in Computer Science, vol. 9266, p. 11–21. Springer International Publishing (2015)
9. Hatvany, J.: Intelligence and cooperation in heterarchic manufacturing systems. Robot. Comput. Integr. Manuf. **2**(2), 101–104 (1985)
10. Ho, J.C., Chang, Y.L.: An integrated MRP and JIT framework. Comput. Ind. Eng. **41**(2), 173–185 (2001)
11. Hopp, W.J., Spearman, M.L.: Factory Physics, vol. 3. McGraw-Hill/Irwin New York (2008)
12. Leitão, P., Restivo, F.: ADACOR: a holonic architecture for agile and adaptive manufacturing control. Comput. Ind. **57**(2), 121–130 (2006)
13. Malone, T.W., Crowston, K.: The interdisciplinary study of coordination. ACM Comput. Surv. **26**(1), 87–119 (1994)
14. Mařík, V., Lažanský, J.: Industrial applications of agent technologies. Control Eng. Pract. **15**(11), 1364 – 1380 (2007). Special Issue on Manufacturing Plant Control: Challenges and Issues INCOM 2004 11th IFAC INCOM'04 Symposium on Information Control Problems in Manufacturing
15. Mařík, V., McFarlane, D.: Industrial adoption of agent-based technologies. IEEE Intell. Syst. **20**(1), 27–35 (2005)
16. McFarlane, D.C., Bussmann, S.: Holonic manufacturing control: rationales, developments and open issues, chap. 13, pp. 303–326. Springer, Berlin, Heidelberg (2003)
17. Philipp, T., Böse, F., Windt, K.: Evaluation of autonomously controlled logistic processes. In: Proceedings of 5th CIRP International Seminar on Intelligent Computation in Manufacturing Engineering, pp. 347–352. Ischia, Italy (2006)
18. Ryan, S.M., Baynat, B., Choobineh, F.F.: Determining inventory levels in a conwip controlled job shop. IIE Trans. **32**(2), 105–114 (2000)
19. Schlesinger, S., Crosbie, R.E., Gagné, R.E., Innis, G.S., Lalwani, C.S., Loch, J., Sylvester, R.J., Wright, R.D., Kheir, N., Bartos, D.: Terminology for model credibility. Simulation **32**(3), 103–104 (1979)
20. Shen, W., Wang, L., Hao, Q.: Agent-based distributed manufacturing process planning and scheduling: a state-of-the-art survey. Syst. Man Cybern. Part C: Appl. Rev. IEEE Trans. **36**(4), 563–577 (2006)
21. Spearman, M.L., Woodruff, D.L., Hopp, W.J.: Conwip: a pull alternative to kanban. Int. J. Prod. Res. **28**(5), 879–894 (1990)
22. Spearman, M.L., Zazanis, M.A.: Push and pull production systems: Issues and comparisons. Oper. Res. **40**(3), 521–532 (1992)
23. Trentesaux, D.: Distributed control of production systems. Eng. Appl. Artif. Intell. **22**(7), 971–978 (2009)
24. Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P.: Reference architecture for holonic manufacturing systems: Prosa. Comput. Ind. **37**(3), 255–274 (1998)
25. Van Dyke Parunak, H.: Agents in overalls: experiences and issues in the development and deployment of industrial agent-based systems. Int. J. Coop. Inf. Syst. **09**(03), 209–227 (2000)
26. Verstraete, P., Saint Germain, B., Valckenaers, P., Van Brussel, H., Belle, J., Hadeli, H.: Engineering manufacturing control systems using prosa and delegate mas. Int. J. Agent-Oriented Softw. Eng. **2**(1), 62–89 (2008)
27. Verstraete, P., Valckenaers, P., Brussel, H.V., Germain, B.S., Hadeli, K., Belle, J.V.: Towards robust and efficient planning execution. Eng. Appl. Artif. Intell. **21**(3), 304–314 (2008)
28. Verstraete, P., Valckenaers, P.: Towards cooperating planning and manufacturing execution systems. IFAC Proc. **39**(3), 393–398 (2006). Proceedings of the 12th IFAC Symposium on Information Control Problems in Manufacturing
29. Verstraete, P., Valckenaers, P., Saint Germain, B., Van Brussel, H., Hedeli, K.: Integration of planning systems and an agent-oriented mes. Int. J. Manuf. Technol. Manage. **8**(1–3), 159–174 (2006)

30. Zambrano Rey, G., Bonte, T., Prabhu, V., Trentesaux, D.: Reducing myopic behavior in FMS control: a semi-heterarchical simulation-optimization approach. Simul. Model. Pract. Theory **46**(0), 53–75 (2014). Simulation-Optimization of Complex Systems: Methods and Applications
31. Zambrano Rey, G., Pach, C., Aissani, N., Bekrar, A., Berger, T., Trentesaux, D.: The control of myopic behavior in semi-heterarchical production systems: a holonic framework. Eng. Appl. Artif. Intell. **26**(2), 800–817 (2013)

Part VI
Service-Oriented Management and Control of Manufacturing Systems

Dynamic Service Reconfiguration with Multi-agent Systems

Nelson Rodrigues, Paulo Leitão and Eugénio Oliveira

Abstract Most modern manufacturing systems rely on constantly seeking new solutions to better fulfil their manufacturing objectives. As reported in today's manufacturing literature, dynamic service reconfiguration is one solution that permits to endorse continuous service reconfiguration, flexibility and evolvable systems. In spite of the current research efforts, real reconfiguration solutions are still lacking automated tools that support dynamic and runtime reconfigurations by discovering new adaptation needs and opportunities and, thus, explore possible actions leading to new system configurations. To overcome these issues, it is essential to provide solutions that answer to the "when" and "what" to reconfigure questions. Most of the service changes triggers rely on reactive events, where decisions come from a centralized decision-maker and are performed manually. Based on these facts a service-oriented multi-agent systems architecture is described aiming at actively promoting service reconfiguration (e.g., improvement of the service's properties and/or update the services' catalogue) to cope with the unexpected and unpredictable condition changes. This paper describes the processes that decide which service reconfiguration should be applied to each circumstance. The developed prototype for a flexible manufacturing system case study allowed verifying the feasibility of the proposed dynamic service reconfiguration solution in different scenarios.

N. Rodrigues (✉) · P. Leitão · E. Oliveira
LIACC—Artificial Intelligence and Computer Science Laboratory,
Rua Dr. Roberto Frias, 4200-465 Porto, Portugal
e-mail: nrodrigues@ipb.pt

P. Leitão
e-mail: pleitao@ipb.pt

E. Oliveira
e-mail: eco@fe.up.pt

N. Rodrigues · P. Leitão
Polytechnic Institute of Bragança, Campus Sta Apolónia, 5300-253 Bragança,
Portugal

N. Rodrigues · E. Oliveira
Faculty of Engineering—University of Porto, Rua Dr. Roberto Frias,
4200-465 Porto, Portugal

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_27

Keywords Service reconfiguration · MAS · Flexible manufacturing system

1 Introduction

The growing of global markets and interest of customer satisfaction challenge manufacturing companies to deliver high-quality customized products while facing the growing requirements from the customers. To face this problem, several governmental initiatives are promoting the research and development towards the factories of the future, e.g., Industrie 4.0 [1] promoted by the German government that is recognized as the 4th industrial revolution. In these visions, digitization of manufacturing and particularly the Internet of Things and Services are recognized as crucial to support the deployment of more flexible, robust, responsive and reconfigurable systems.

Several reconfigurable paradigms were proposed during the last decades, introducing flexible and agile characteristics to react promptly to unexpected events, system failures, quality deviations, etc., avoiding the loss of financial revenue and trust from the customer viewpoint [2]. As an example, reconfigurable manufacturing system (RMS) [3] is a well know paradigm that provides the ability to repeatedly change and reorganize the components of a system in a cost-effective way. Consequently, the reconfiguration of a system is embraced when it brings benefits to the manufacturing control context.

More particularly in service-oriented manufacturing approaches the functionalities of manufacturing components are offered and consumed as services [2]. For example, ANEMONA-S+Thomas proposes a framework to implementing a service-oriented intelligent manufacturing system [3]. The current works of service-oriented manufacturing systems constitute the fundamental functionalities to cope the service reconfiguration problem. The adaptation of the system behaviour and its evolution are based in the service readjustment of the manufacture elements or settings of a system. Each one of the service reconfiguration is composed of several services; an automated service composition is proposed in [4], with the objective to maximize the overall quality of the final compositions using agents that adapt services processes, in a continuous form. A dynamic service reconfiguration is proposed by [5] that explores the use of agents to achieve consistent service reconfiguration solutions focusing fault-tolerant systems.

In addition to the software reconfiguration, hardware reconfiguration plays an important role. An approach that considers software and hardware reconfiguration is proposed by [6], employing a knowledge ontology and AI-planning for the service reconfiguration. In the manufacturing context, some projects already addressed the service reconfiguration problem, namely SOCRADES [7] that is oriented to the reconfiguration of smart embedded devices and PRIME [8] that lies on a plug and produce system architecture capable of reacting to unexpected disturbances to maintain the productivity and quality parameters. The IDEAS project [9] supports the use of reconfigurable production systems using agent technology to perform the on-line reconfiguration without the need of reprogramming efforts, and more recently, the PERFoRM project [10] is developing a system architecture for the seamless reconfiguration of machinery and robots as response to operational or business events.

However, most of the real service reconfiguration solutions are executed manually, reactively and in a centralized perspective. In fact, one drawback is that the decisive actions for the system reconfiguration are made after the occurrence of failure and sometimes involves stopping a running system, reconfiguring and then restarting the system which does not go according to industrial needs, pointing out to have a pro-active, distributed and online service reconfiguration.

Having this in mind, the paper describes a service reconfiguration approach that allows the identification of the opportunities for reconfiguration, in a pro-active and online manner, determines and implements on-the-fly the best strategies for the service reconfiguration that will lead to better production efficiency. For this purpose, a multi-agent system (MAS) is providing distributed intelligence to run the service reconfiguration, particularly embedding intelligent mechanisms for the early detection of reconfiguration opportunities, e.g. anticipating performance or quality degradation, and also advanced data analytics to support the service reconfiguration by selecting strategies for improving the service properties (e.g., QoS and execution time) or updating the catalogue of offered services (e.g., offering a new service that has more demand).

The remaining of this paper is organized as follows. Section 2 overviews the multi-agent system architecture for dynamic service reconfiguration, particularly addressing the "when and how to reconfigure" phases. Section 3 describes the mechanism for determining the alternative solutions for service reconfiguration and Sect. 4 presents the evaluation methodology to select the best service reconfiguration solution from the space of solutions previously created. Section 5 presents some preliminary experimental results aiming at validating the proposed approach. Finally, Sect. 6 rounds up the paper with the conclusions.

2 Multi-agent System Architecture for Service Reconfiguration

2.1 Agents to Provide Intelligence Supporting Service Reconfiguration

The literature suggests Service-Oriented Architecture (SOA) [2] as an excellent solution to face the many current industry challenges, namely providing interoperability in heterogeneous systems. The development of SOA-based solutions requires the implementation of several features, namely service-discovery, service-registration, service-composition and service-reconfiguration. In particular, service reconfiguration is crucial to facilitate the changes taking place

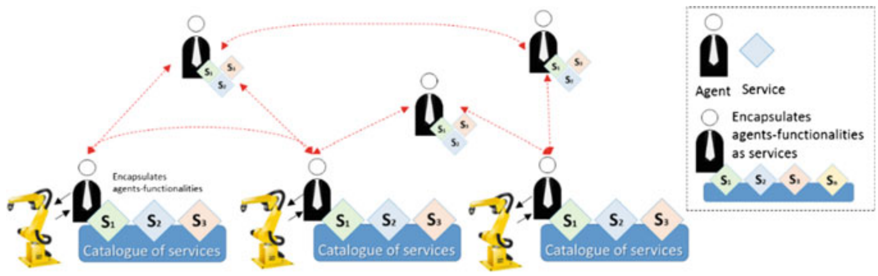


Fig. 1 Service-oriented multi-agent system

in unpredictable environments. As such, this task requires the development of an adaptive system that can regulate service management in response to events and changes in the environment, and particularly a continuous monitoring to dynamically support any change that might occur by reacting on-the-fly. Having this in mind, a MAS is proposed to embed intelligence and adaptive mechanisms in the distributed and autonomous agents to support the service reconfiguration on-the-fly, as shown in Fig. 1.

The service-oriented MAS architecture considers a network of intelligent and autonomous agents, each one exposing its functionality as services. These services are published in a repository and can be reached via discovery mechanisms. The question that arises is that, over the time, services can become less competitive (i.e. not being requested), e.g., due to their low QoS or high price, requiring the execution of proper actions to improve the services competitiveness. For this purpose, a continuous monitoring of the service performance is required to identify when a reconfiguration should be performed, and intelligent mechanisms should be implemented to determine the best solution for the service reconfiguration. In this approach, the service reconfiguration procedure is performed in a distributed way, with individual agents embedding intelligent mechanisms to run the reconfiguration of their catalogue of services.

2.2 *Service Reconfiguration Approach*

According to the architectural principles, each individual agent is running an online mechanism aiming at continuously collecting data, identifying opportunities to reconfigure, determining how the reconfiguration can be performed and deciding whether the reconfiguration should be performed, as illustrated in Fig. 2.

In this service reconfiguration approach, the first phase is related to the continuous collection of up-to-date information on the services' performance and the storage of the relevant data into a local database. A service reconfiguration trigger relies on the determination of when is the best moment to reconfigure, which can happen in different moments of the production cycle, e.g., in a case of performance or quality degradation, failure occurrence on the introduction of new products.

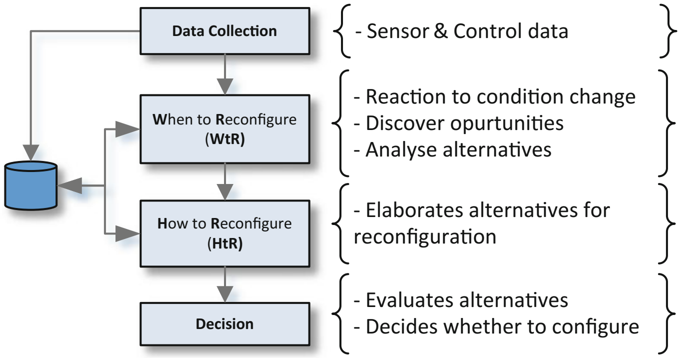


Fig. 2 Service reconfiguration module implemented in each agent

Table 1 Possible types of service reconfiguration

Reconfiguration types	Description	Effort
Improve the service's behaviour	Improve the behaviour of the service without replacement, e.g., calibrating tools and switching components of the process that executes the service aiming at reducing service time or improve the service quality	Low
Change the service's catalogue	The catalogue of offered services is changed, i.e. new services are provided by the existing agent or by new agents, e.g., offering a new drilling service	High

For this purpose, the *When to Reconfigure* (WtR) module is responsible for monitoring and analysing the collected data in order to identify the triggers for the reconfiguration. In the proposed approach, this model relies on events, periodic and trend triggering strategies to support the on-the-fly reconfiguration (see [11] for more details about the WtR module).

After being identified an opportunity to reconfigure, the *How to Reconfigure* (HtR) module determines how the service reconfiguration can be implemented. The process comprises the building of a pool of possible alternatives for the service reconfiguration, followed by a semantic checking that reduces the dimension of the alternative solutions. The elaboration of the alternative solutions considers two classes of service reconfiguration, namely improving the service's behaviour (as a weak-reconfiguration class) and changing the service's catalogue (strong-reconfiguration class), as shown in Table 1.

The decision module is responsible for evaluating the effectiveness of the service reconfiguration alternatives, considering different criteria set by the system managers. Only after recognizing the expected profits of such service reconfiguration, the selected solution is implemented. In this work, special attention is devoted to the HtR module.

3 Mechanism to Create Alternative Solutions

The HtR module contains a mechanism to create alternative solutions that works in two phases. The first phase is responsible for the creation of a pool of alternative service reconfiguration solutions and the second phase is in charge of testing the compatibility of the alternative solutions by using the semantic matching.

3.1 *Build a Space of Alternative Solutions Phase*

In order to produce several reconfiguration alternatives, each individual agent, after receiving the reconfiguration triggers from the WtR module can recommend the improvement/replacement of a specific service, as shown in Fig. 3, considering the pool of available services not installed (i.e. they exist, but are not offered at the moment).

The algorithm embedded in each agent to calculate the alternative solutions for the service reconfiguration is represented as follows.

```
S1 ← select the worst service from the catalog
If ((utilization rate of Service S1 ≥ α1) OR (rate of service bids of S1 ≥ α2))
Improve the Service(S1)
Else
  Snew ← select the potential best Service
  If (rate of service bids of Snew ≥ rate of service bids of S1)
  Perform Service Replacement (remove S1, add Snew)
```

A service with weak performance can be improved if its utilization rate or the missing bids for its usage are higher than certain threshold values (α1). This can involve the execution of a set of actions regarding the optimization of the process

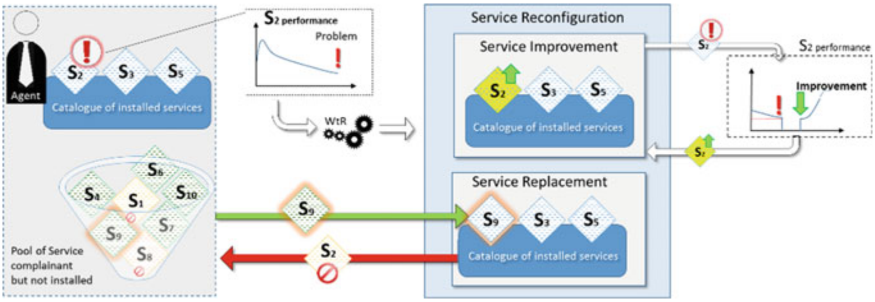


Fig. 3 Service reconfiguration alternatives (service replacement and service improvement)

encapsulated by the service, e.g., calibrating tools, optimizing operational parameters or replacing components. Otherwise, the best services from the pool of available services and not installed are selected to create alternative possibilities to replace the one with the weak performance.

Additionally, sometimes it is useful to consider services that are not available in order to discover new opportunities for reconfiguration. The decision to explore potential solutions is calculated by the nervousness control that adjusts the threshold values (i.e. α1, α2). Moreover, the learning module is capable of changing the exploration rate value, allowing controlling the exploration of different solutions. This process can generate an enormous volume of service configuration alternatives, resulting in a time-consuming process. In this way, the agent can run this process in the background, especially when the trigger for reconfiguration follows the periodic strategy.

3.2 Semantic Matching Phase

Inspired on the service description topology (i.e., manufacturing-service model [12]) the technical operator describes semantically each service, resource, and process that exists on the system (e.g., describing in a RDF/XML format). Therefore, each agent contains the entire device's information in its catalogue of services (e.g., gripper's characteristics).

Moreover, it also contains information about which processes the agent can produce (e.g., the resource *r* can make process *openGripper* using the service *gripper1*) under some constraints (e.g., physical limitations and QoS).

Figure 4 illustrates the particular agents, representing industrial robots, which are implementing the semantic reasoning about the logical configuration, for

Fig. 4 Semantic matching of service reconfiguration solutions

example, using JENA, to determine the feasibility of the service reconfiguration solution (i.e. semantic matching between the resource machines and all services from the pool of services).

The outcome of this process is set of feasible service reconfiguration solutions.

4 Evaluation of Service Reconfiguration Alternatives

The process of selecting the optimal system configurations is not consensual in literature. One novelty, besides the generation of alternative service reconfiguration solutions, is the distributed measurement of the reconfiguration alternatives effectiveness.

4.1 Evaluation of the Service Reconfiguration Phase

In general, an evaluation takes into account several criteria, e.g., processing time and quality, which can be evaluated individually or combined. The agents conduct the evaluation of the service reconfiguration solutions based on two metrics:

- Maximizing the service composition quality, which permits to select the service with the highest quality (see Table 2).
- Minimizing the reconfiguration index which permits to select the service configuration with best improvement values with minimal implementation effort.

The quality metrics denoted in Table 2 were based on the MAS architecture capable of evaluating, in run-time, several hypotheses of services composition [13].

In a next step, ranking criteria are created based on the selected indicators for the created configurations. Note that the measurements of these indicators can change over time according to each system manager's requirements.

In contrast, it is essential to understand the reconfiguration effort, i.e., to calculate for each potential solution the reconfiguration cost. However, these

Table 2 Representation of the agent and service variables—from the machine viewpoint

Name and equation	Description
QoS availability $f(\phi) = \frac{\sum\lambda}{\sum\lambda + \sum\psi} * 100$	The ratio of the service uptime of time period, where λ stands for service uptime and ψ for the service downtime
QoS response time $f(\theta) = \delta - \rho$	Performance of a service. Given by the difference between conclusion time δ, and ρ the request time
QoS throughput $f(\eta) = \frac{\sum\gamma}{\sum t}$	Provider performance index. Given by the maximum number of services to process a unit of time where γ stands for the complete request and t for the unit time

requirements do not support the dimension of the reconfiguration effort index. In the literature, the structure matrix analysis investigates the capability to reconfigure [14]; however the process relies on centralized decisions. This indicates a lack of research in measuring the reconfigurability effort and its impact on the decentralized way. Thus, the proposed model inspired from [15] and [13], takes a step forward by joining the following indicators.

The reconfiguration index (RI) considers the number of reconfigurations. Each agent contains a vector with the actual configuration of the services (CC) that are being currently executed. In addition, the agent contains other simulated vectors that represent the alternative configurations previously built (AR). The idea is to compare the CC vector and the simulated AR vector to understand the effort that is required [15].

$$\mathrm{RI} = 1 - \frac{\sum_{\mathrm{s_i}} (\mathrm{modificationCost}(\mathrm{s_i}))}{\#\,\mathrm{of\ services}} \tag{1}$$

where

$$\mathrm{modificationCost}\ (\mathrm{s_i}) = \begin{cases} 1, & \mathit{if}\ (\mathrm{cc}[\mathrm{s_i}] = \mathrm{AR}[\mathrm{s_i}] \\ 0, & \text{otherwise} \end{cases}$$

According to Formula (1), an evaluation process is created by the number of modifications. For example, if the alternative is equal to the current configuration, the RI is 0; RI becomes close to 1 as many modifications exist. Besides the RI, the reconfiguration cost is another important measure, as the Formula (1) considers the same weight for different modifications. The reconfiguration cost (RC) [16] considers different service modification costs between the services, calculated as follows:

$$\mathrm{RC} = nr \times \sum_{\mathrm{s_i}} (\mathrm{modificationCost}\,(\mathrm{s_i}) + \mathrm{lbcost}\,(\mathrm{s}_i)) \tag{2}$$

where *nr* represents the number of modification multiplied by the unitary cost of modifying the service $\mathrm{s_i}$, which includes the $\mathit{modificationCost}\,(s_i)$ of a particular service s_i and labor cost $\mathit{lbcost}\,(s_i)$. The reconfiguration effort is evaluated with this simple metric, which considers the number of modifications required for a specific service reconfiguration. The positive impact is defined if the expected profit is higher, which is obtained as shown in Formula (3), where the *Expected Benefit* value is being calculated at the planning phase:

$$\text{Expected Profit} = \text{Expected Benefit} - \mathrm{RC} \tag{3}$$

At the end the list of solutions is ordered according to an assessment with low reconfiguration costs, high levels of benefits ensuring high quality.

4.2 Implementation Phase

After the evaluation process, the *Decision* module is responsible for deciding if the best service reconfiguration alternative is implemented or not. Such decision takes into account several points, namely the expected benefit and quality of such service reconfiguration. Thus the automatic mechanism considers self-* properties like self-learning to support the decision-making about alternative service reconfiguration scenarios; in parallel this module provides valuable information in proper graphical user interfaces.

5 Experimental Results

The proposed approach for service reconfiguration was tested using a flexible manufacturing system case study comprising a set of 6 workstations (WS), interconnected through conveyors, each WS offering a limited set of operations (i.e. services). Several sub-products are made in this system, namely the parts in the form of the letters A, B, E, I, L, P and T, which combined can produce the final products BELT and AIP. More detail about the case study benchmark can be found in [17].

For this purpose, the proposed MAS was implemented using the JADE framework. Several agents were launched to represent several WS and the products requested to be manufactured in the system. Each one of the WS agents has embedded the WtR module (to identify opportunities to reconfigure) and the HtR (to determine the best strategy to reconfigure). In the proposed case study, the service reconfiguration in WS-3 is considered when the service's quality is not meet (trend) and also due to the occurrence execution failures (event). Each WS agent has a catalogue of 3 installed services and 2 services available but not installed. The time to replace one service installed by another service available is 60 s, the time to improve the service performance is 30 s. and the service recovery 120 s.

Table 3 summarizes the results considering the execution of different scenarios, each one considering different batch sizes.

Table 3 Experimental results

Scenarios	Cmax in s (without reconfiguration)	Cmax in s (with reconfiguration)	Improvement (%)	Details
#5 (BELT)	1595	1565	1.8	#1 change, #0 replace
#10 (BELT)	3234	3147	2.6	#1 change, #0 replace
#15 (BELT)	5245	5068	3.3	#2 change, #1 replace
#20 (BELT)	7576	7268	4.0	#1 change, #2 replace

The comparative analysis of each scenario is performed using the value of the makespan, known as Cmax. Once the reconfiguration decision is made according to the expected profit as is formulated in Formula (3), the value Cmax allows comparing different scenarios without performing reconfiguration, illustrating the number of performed changes and replacements.

The analysis of the experimental results shows the benefits of considering the proposed approach. In fact, all scenarios reported improvements in the Cmax values ranging from 1.8−4.0% when considering the automatic service reconfiguration. It is also worth noting that the bigger is the batch size the better is the achieved improvement.

6 Conclusions and Further Work

Companies are successfully using centralized techniques for analyzing up-to-date information collected from the shop floor to identify service failures or performance deviations. Once the need for service reconfiguration recognized, its implementation is often carried out in a manual mode, usually held as a recovery approach. With the objective to increase the opportunities to dynamically evolve and implement the service reconfiguration on-the-fly, this paper proposes an agent-based approach for the dynamic, distributed efficient and on-the-fly service reconfiguration. The proposed decentralized approach pro-actively generates several service reconfiguration solutions promoted by different triggering strategies. The developed modules are embedded in smart agents, that do not only recognize opportunities to change, but also assist engineers in exploring and deciding about different alternative configuration possibilities, to cope with disturbances or predicting production changeover. Another contribution of this work is the evaluation of the potential service reconfiguration solutions. The preliminary experimental results validate the feasibility of the mechanism that determines how to reconfigure services offered by the system leading to more efficient and agile systems.

As future work, there are some open questions that require a deep analysis, e.g., what would happen if all agents representing manufacturing resources adapt at the same time, and what are the necessary rules to control the system nervousness avoiding falling into a chaotic system.

References

1. Kagermann, H., Wolfgang, W., Helbig, J.: Securing the future of German manufacturing industry. Recommendations for implementing the strategic initiative INDUSTRIE 4.0. Final report of the Industrie 4.0 Working Group (2013)
2. MacKenzie, C.M., Laskey, K., McCabe, F., Brown, P.F., Metz, R.: Reference model for service oriented architecture 1.0. OASIS Stand. **12**, 1–28 (2006)

3. Giret, A., Botti, V.: ANEMONA-S+Thomas: a framework for developing service-oriented intelligent manufacturing systems. In: Service Orientation in Holonic and Multi-agent Manufacturing, pp. 61–69. Springer International Publishing (2015)
4. Lécué, F., Gorronogoitia, Y., Gonzalez, R., Radzimski, M., Villa, M.: SOA4All: An innovative integrated approach to services composition. 2010 IEEE International Conference on Web Services (ICWS), pp. 58–67 (2010)
5. Tsai, W.T., Song, W.W., Chen, Y.N., Paul, R.: Dynamic system reconfiguration via service composition for dependable computing. Reliable Syst. Unreliable Network. Platf. **4322**, 203–224 (2007)
6. Yang, T.-H., Lee, W.-P.: Intelligent service reconfiguration for home robots. In: Ding, X., Kong, X., Dai, S.J. (eds.) Advances in Reconfigurable Mechanisms and Robots II, pp. 735–745. Springer International Publishing, Cham (2016)
7. Colombo, A.-W., Karnouskos, S., Mendes, J.-M.: Factory of the future: a service-oriented system of modular, dynamic reconfigurable and collaborative systems. In: Benyoucef, L., Grabot, B. (eds.) Artificial Intelligence Techniques for Networked Manufacturing Enterprises Management. Springer, London, pp. 459–481 (2010)
8. Rocha, A., Di Orio, G., Barata, J., Antzoulatos, N., Castro, E., Ratchev, S.S., Ribeiro, L.: An agent based framework to support plug and produce. In: 12th IEEE International Conference on Industrial Informatics (INDIN), pp. 504–510 (2014)
9. Onori, M., Lohse, N., Barata, J., Hanisch, C.: The IDEAS project: plug & produce at shop-floor level. Assem. Autom. **32**, 124–134 (2012)
10. Leitão, P., Barbosa, J., Pereira, A., Barata, J., Colombo, A.: Specification of the PERFORM architecture for the seamless production system Reconfiguration. In: 42nd Annual Conference of the IEEE Industrial Electronics Society (IECON) (2016)
11. Rodrigues, N., Leitão, P., Oliveira, E.: Triggering strategies for automatic and online service reconfiguration. In: 11th Conference, pp. 76–82 (2016)
12. Gamboa Quintanilla, F., Cardin, O., L'Anton, A., Castagna, P.: A modeling framework for manufacturing services in service-oriented holonic manufacturing systems. Eng. Appl. Artif. Intell. **55**, 26–36 (2016)
13. Rodrigues, N., Leitão, P., Oliveira, E.: Self-interested service-oriented agents based on trust and QoS for dynamic reconfiguration. In: Service Orientation in Holonic and Multi-Agent Manufacturing (SOHOMA'14), Nancy, France (2014)
14. Farid, A.M.: Measures of reconfigurability and its key characteristics in intelligent manufacturing systems. J. Intell. Manuf. 1–17 (2014)
15. Neves, P.: Reconfiguration methodology to improve the agility and sustainability of plug and produce systems (2016)
16. Zhou, J., Djurdjanovic, D., Ivy, J., Ni, J.: Integrated reconfiguration and agent-based preventive maintenance decision making. IIE Trans. **39**(12), 1085–1102 (2007)
17. Trentesaux, D., Pach, C., Bekrar, A., Sallez, Y., Berger, T., Bonte, T., Leitão, P., Barbosa, J.: Benchmarking flexible job-shop scheduling and control systems. Control Eng. Pract. **21**(9), 1204–1225 (2013)

Caregivers Routing Problem in Home Health Care: Literature Review

Eric Marcon, Sondes Chaabane, Yves Sallez and Thérèse Bonte

Abstract In France, Hospital-at-Home Services (HHS) providers that employ their caregivers have to define the assignment of patients to caregivers and the planning of the caregivers' routes. In the literature, different approaches are proposed to deal with both these problems. Our aim in this paper is to present a literature review of such researches. The target is to show how operational research approaches and Multi-agent based methods can be combined and generate a "win-win" relationship between both approaches to solve HHS resources management in operational level.

Keywords Hospital at home services · Caregivers' routing problem · Assignment problem · Multi-agent

1 Introduction

Over the last decade, Home Health Care (HHC) services have significantly increased [1]. The most developed type of HHC in France is: "Hospital-at-Home Services" ("hospitalisation à domicile" in French), which shall hereinafter be referred to as HHS. HHS providers handle patients in their own homes the same way as in a hospital. Generally, patients have to be handled through the HHS following a stay in hospital, because they still require acute medical care. HHS is the interface between the hospital and the home, and is only provided for short, limited periods.

E. Marcon
Université de Lyon INSA-LYON, University Jean Monnet of Saint-Etienne, DISP, EA 4570, 69621 Villeurbanne, France
e-mail: marcon@univ-st-etienne.fr

S. Chaabane (✉) · Y. Sallez · T. Bonte
LAMIH UMR CNRS 8201, University of Valenciennes and Hainaut-Cambrésis, 59313 Valenciennes, France
e-mail: sondes.chaabane@univ-valenciennes.fr

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_28

This paper focuses on the literature and works on the operational level concerning "human resource and care activity scheduling problem", more precisely, the studies concerning: (1) patient-caregiver assignment (denoted "assignment problem": allocate each admitted patient to suitable caregiver) and (2) determining the caregiver's route (denoted "routing problem": plan the caregivers' routes). Two main research communities are highlighted: Operational Research and Multi-agent communities. The objective is to analyse existing works and propose new trends to cope with their identified limitations.

The remainder of the paper is organized as follows: the Sects. 2 and 3 present respectively the literature. Studied researches are analysed and discussed in Sect. 4. The conclusion and some perspectives are exposed in the last section.

2 Operational Research Based Approaches

2.1 Assignment and Routing Problems Solved Jointly

Most of the time, the problem is identified as a VRPTW (Vehicle Routing Problem with Time Windows) if the patient is assigned jointly with the planning of the caregiver's route [2, 3]. In these approaches, authors have introduced into their mathematical models both assignment and routing plan constraints. They optimize the daily planning of health care services by hypothesizing that the care planning for the patient is the same every day. Authors suggest a different way to solve this NP-hard problem based on MILP [2, 3] or with original heuristic [4–6], in order to cope with computational time limitation. Recently, [7] suggest a heuristic to tackle the mid-term and short-term planning problem. Some papers suggest the use of metaheuristics-based approaches. For example, [8] define a Particle Swarm Optimization (PSO)-based algorithm, with a combination of the different techniques; [9] propose a Group Genetic Algorithm (GGA), an extension of a basic GA adapted to grouping problems and comparing their approach with existing PSO and GA algorithms.

2.2 Assignment and Routing Problems Solved Separately

In other studies, both problems (assignment and routing) are separately treated because authors reach achieving a robust local solution for each. [10] focused on the assignment problem but they introduced a travel-time estimators into their model, for taking into account routing plan constraints. [11] define an assignment policy based on the cost of the home care services. In [12] the authors suggest a new approach for assignment to face such random events as: new patient's demand, nurses' workloads, etc. For the routing problem, a large part of the literature tackles this problem as a "Travelling Salesman Problem with Time Windows (TSPTW)" approach for designing the caregiver's route with a MILP [13] and/or heuristic [14, 15] approaches for a static and deterministic problem.

3 Multi-agent Based Approaches

An agent is defined as: an autonomous entity (physical or virtual), able to act within its environment and communicate with other entities [16]. In recent years, Multi-Agent Systems (MAS) have gained tremendous popularity in providing solution to transport planning and scheduling problems in healthcare systems in general, and in the management of home health care operations, in particular. Recently, [17] presented advanced methodologies and case studies in healthcare using MAS and artificial agents. The MAS paradigm attracts our attention because it allows modelling of distributed approaches where several entities (requesters) need services from other entities (providers) [18]. In the field of Home Health Care, for example, requesters would be patients and providers would be caregivers. This section highlights the related works in assignment and routing problems in the home health care field. From our review, it was apparent that the routing and assignment problems were systematically solved jointly.

The project K4Care (Knowledge-Based HomeCare eServices for an Ageing Europe) is worth of mention [19, 20]. This project aims at improving the assistance of senior patients that suffer chronic diseases, or disabled persons. A multi-agent system constitutes the heart of the K4Care platform and deals with the problems as a negotiation of meetings and tasks assignment for the care providers. The platform allows access to all the knowledge required by the agents to keep track of their current and pending activities, and address/request any necessary needs of their patients.

More recently, [21] consider homecare staff scheduling and task assignment decisions in a dynamic context, with multi-conflicted objectives. The authors propose a system framework incorporating intelligent agents for the different players (manager, supervisor, patient, nurse, resource and scheduler), internet services, wireless networks, and mobile devices. The multi-agent platform is located in the home healthcare centre, and there is no direct interaction between caregivers and patients.

In [22], IT systems are used to support cooperative work between relatives and home care workers, to take care of elderly persons. A prototype, called Carecoor, allows access at home via a tablet PC, and outside via Internet. Home care workers exchange information on the scheduling of tasks, suggestions for swapping, and messages used to coordinate the cares. If the paper describes the interaction among the different players, the problem of tasks scheduling is not really addressed.

To the best of our knowledge, only [23] propose a really "on line" distributed approach. They develop a support system for home-care scheduling, using a multi-agent system. Client and caregivers are equipped with PDAs, and can communicate with the HHS support system. This support system incorporates a database, and contains three types of agents (interface agent, scheduler agent and helper agent). A negotiation mechanism among agents, associated to clients and caregivers, is used to conduct the care schedule autonomously. The patients and caregivers use PDAs to confirm the schedules arranged by the agents. However, care schedules are computed in the home care centre in a centralized manner.

4 Literature Analysis and Discussion

In Table 1, the papers reviewed in both previous parts are classified according to the problem they address, and according to the type of resolution (either off-line or on-line).

Most assignment and routing sub-problems are solved jointly, i.e., simultaneously. In our opinion, there are different advantages to solve these sub-problems separately, but in sequential manner. Firstly, the assignment sub-problem must be solved less frequently than caregivers' routes, i.e. the assignment problems must be solved only when patients are admitted/discharged, while caregivers' routes must be solved more frequently (each day the patient's care plan changes). Secondly, the objective function of the assignment sub-problem balances and improves the workload for caregivers, whereas the objective function of caregivers' route is to find the shortest route. Thirdly, caregiver route problems evolve in dynamic, real-world context, and the main difficulties are: (1) taking into account dependency constraints between caregivers; (2) facing random events, e.g., traffic jams, road accidents; (3) to react quickly to unexpected disturbances; and (4) makong good decisions in reasonable time.

The assignment and/or routing problems are solved "off-line" in a centralized manner in most related works. A central authority, represented in this case by the

Table 1 Classification of the different reviewed papers

	Off-line approaches	On-line approaches
	Assignment and routing problems solved jointly	
OR	(Bredström and Rönnqvist [2]; Rasmussen et al. [3]; Allaoua et al. [4]; Mankowska et al. [5]; Coppi et al. [6]; Nickel et al. [7]; Akjiratikarl et al. [8]; Mutingi and Mbohwa [9])	
MAS	(Sánchez et al. [19]; Isern et al. [20]; Mutingi and Mbohwa [21])	(Itabashi et al. [23])
	Assignment and routing problems solved separately	
OR	(Yalçındag et al. [10]; Lanzarone and Matta [11, 12]; Kergosien et al. [13]; Redjem and Marcon [14])	

OR Operational Research approaches; *MAS* Multi-Agent System approaches

HHS manager, is responsible for all decisions, and a route plan is sent daily to each caregiver, without any "on-line" feedback. Most of the studies consider only one criterion for optimization, and sometimes the aggregation of two or three different criteria. Minimizing the cost, time, or distance travelled are the most widely-used objective. With a few exceptions, most researchers address assignment and/or routing problems in static and deterministic context.

In this case, the decision is calculated using centralized approaches, and optimal solutions are found in the majority of the cases. These kinds of approaches are efficient when the system is not changed frequently, and is not subject to disruptions. To solve caregivers routing problem in dynamic environment, the analysis of the literature review allows us to highlight some limitations:

- Firstly, the system is subject to uncertainties; even minor changes may have an impact on the system, on the performances and on the "optimal" solution. Moreover, it is impossible, a priori, to assess the impact of changes on the quality of the solution. In fact, HHS operates in dynamic environments. Dynamics is principally related not only to the uncertainty around on the travel durations, (for instance, due to traffic jams), but sometimes also to the variability of care duration. However, literature offers few studies dealing with disruptions and dynamic environments.
- Secondly, some approaches are very expensive in computational time; for example, the ones based on Mixed Integer Linear Program models. Although recent approaches based on heuristics require less computational time than linear programming ones, dedicated models must be designed to solve each specific problem. In addition, real world problems involve large systems, i.e., several hundred patients and caregivers, and more than hundreds of thousands of care activities daily.
- Thirdly, some works in the MAS field are distributed in terms of software platforms, but the decision mechanism remains "off-line" and centralized. Other papers describe interactions among different agents, but no decision mechanisms.

For that, we suggest to combine approaches based on an OR and approaches using Multi-agent systems. This idea gets inspiration from the ORCA control architecture presented in [24, 25], where one of the basic principles is to design a global optimizer (in our case, the HHS level) that interacts with local reactive controllers (in our case, the caregivers). Thus, the problem is sequentially solved: firstly, an off-line centralized process assigns the patients to the caregivers and defines for each workday the caregivers' patients list. Secondly, we suggest that each caregiver dynamically solves his/her own routing problem by applying a local rule of decision for choosing the order of patients he/she will visit. More precisely, we propose that the assignment problem is solved using OR tools, and the routing problem is solve in a dynamic and distributed way by the caregivers based on MAS system. The idea behind this is: firstly, OR tools face complexity limitations, but can provide efficient long-term performances. Thus, they can be used to solve the assignment problem while ignoring routing issues. Secondly, the routing problem, being more a dynamic problem subject to numerous disturbances and unpredicted events, can be easily simulated by a MAS system, in the same spirit as the paper [23]. Consequently, such a combination may generate a "win-win" relationship between both approaches.

Figure 1 shows the global framework solving both assignment and routing problems.

This framework is characterized by an off-line centralized process based on OR models (MILP, heuristic or metaheuristic). This off-line centralized process makes two kind of decisions: (1) Assigning patients to caregivers based on several criteria like the patient's preferences, the location, etc.; (2) Drawing up the ordered list of patients each caregiver will visit every workday according to the shortest route for each caregiver (under the unique constraint of the patient's and caregiver's locations). In that case, all the other constraints are relaxed, i.e., dependency constraints, patient's availability, etc., but still considered at the on-line level. As shown in Fig. 1, these decisions should be taken at least 24 h prior the start of the care treatment in our context.

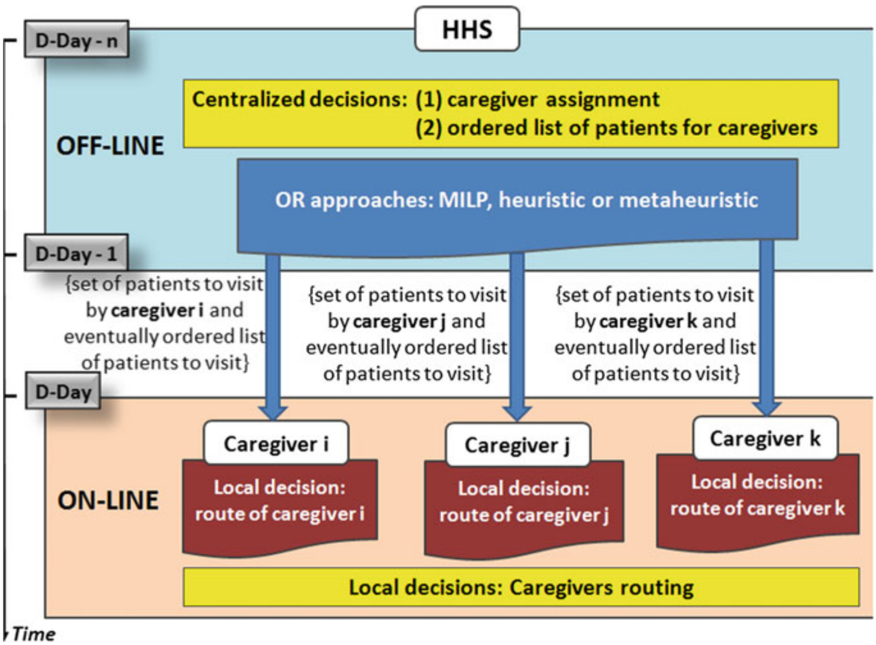


Fig. 1 OR and MAS Integrated architecture for solving the assignment and routing problems

5 Conclusion and Perspectives

A literature review in operational research and multi-agent communities on HHS operational level management is presented and discussed. To cope with limitation of studies in each community our proposal is to combine the advantages of the two kinds of approaches. The idea is to define an integrated architecture using the operational research and Multi-agent methods.

The future work in short time can be conducted to propose such architecture in real case study.

References

1. Benzarti, E., Sahin, E., Dallery, Y.: Operations management applied to home care services: analysis of the districting problem. Decis. Support Syst. **55**(2), 587–598 (2013)
2. Bredström, D., Rönnqvist, M.: Combined vehicle routing and scheduling with temporal precedence and synchronization constraints. Eur. J. Oper. Res. 191 **1191**, 19–31 (2008)
3. Rasmussen, M., Justesen, T., Dohn, A., Larsen, J.: The home care crew scheduling problem: preference-based visit clustering and temporal dependencies. Eur. J. Oper. Res. **219**, 598–610 (2012)
4. Allaoua, H., Borne, S., Létocart, L., Wolfler Calvo, R.: A metaheuristic approach for solving a home health care problem. Electron. Notes Discret. Math. **141**, 471–478 (2013)
5. Mankowska, D.S., Meisel, F., Bierwirth, C.: The home health care routing and scheduling problem with interdependent services. Health Care Manag. Sci. **17**, 15–30 (2014)
6. Coppi, A., Detti, P., Raffaelli, J.: A planning and routing model for patient transportation in health care. Electron. Notes Discret. Math. **41**, 125–132 (2013)
7. Nickel, S., Schröder, M., Steeg, J.: Mid-term and short-term planning support for home health care services. Eur. J. Oper. Res. **219**, 574–587 (2012)
8. Akjiratikarl, C., Yenradee, P., Drake, P.: PSO-based algorithm for home care worker scheduling in the UK. Comput. Ind. Eng. **53**, 559–583 (2007)
9. Mutingi, M., Mbohwa, C.: Home Healthcare worker scheduling: a group genetic algorithm approach. In: Proceedings of the World Congress on Engineering, London, UK (2013)
10. Yalçındag, S., Matta, A., Sahin, E., Shanthikumar, G.: A two-stage approach for solving assignment and routing problems in home health care services. In: Proceedings of the International Conference on Health Care Systems Engineering, Springer Proceedings in Mathematics & Statistics: Springer International Publishing (2014)
11. Lanzarone, E., Matta, A.: A cost assignment policy for home care patients. Flex. Serv. Manuf. J. **24**, 465–495 (2012)
12. Lanzarone, E., Matta, A.: Robust nurse-to-patient assignment in home care services to minimize overtimes under continuity of care. Oper. Res. Health Care **1**, **2** (2014)
13. Kergosien, Y., Lenté, C., Billaut, J.-C.: Home health care problem: an extended multiple traveling salesman problem. In: 4th Multidisciplinary International Conference on Scheduling: Theory and Applications (MISTA'09), Dublin, Ireland (2009)
14. Redjem, R.: Aide à la décision pour la planification des activités et des ressources humaines en Hospitalisation A Domicile, Ph.D. thesis, University of Saint Etienne, France (2013)
15. Redjem, R., Marcon, E.: Operations management in the home care services: a heuristic for the caregivers' routing problem. Flex. Serv. Manuf. J. (2015). doi:10.1007/s10696-015-9220-8
16. Ferber, J.: Multi-agent systems: An Introduction to Distributed Artificial Intelligence. Addison Wesley, London (1999)

17. Vaidya, I.B., Sachin, J.A.: Computational Intelligence in Healthcare, vol. 4. Springer (2010)
18. Zambrano, R.G., Pach, C., Aissani, N., Bekrar, A., Berger, T., Trentesaux, D.: The control of myopic behavior in semi-heterarchical production systems: a holonic framework. Eng. Appl. Artif. Intell. **26**(2), 800–817 (2013)
19. Sánchez, D., Isern, D., Moreno, A.: Agent technology and healthcare: possibilities, challenges and examples of application. In: Computational Intelligence in Healthcare, vol. 4. Springer (2010)
20. Isern, D., Sánchez, D., Moreno, A.: Agents applied in health care: a review. Int. J. Med. Inf. **79**, 145–166 (2010)
21. Mutingi, M., Mbohwa, C.: A home healthcare multi-agent system. In: A Multi-objective Environment, SAIIE25 Proceedings, Stellenbosch, South Africa (2013)
22. Bossen, C., Christensen, L., Grönvall, E., Vestergaard, L.: CareCoor: augmenting the coordination of cooperative home care work. Int. J. Med. Inf. 82; 189–199 (2013)
23. Itabashi, G., Chiba, M.,. Takahashi, K., Kato, Y.: A support system for home care service based on multi-agent system. In: 5th International Conference on Information, Communications and Signal Processing, Bangkok, Thailand (2005)
24. Kadri, F., Pach, C., Chaabane, S., Berger, T., Trentesaux, D.,. Tahon, C., Sallez, Y.: Modelling and management of the strain situations in hospital systems using an ORCA approach. In: IEEE International Conference on Industrial Engineering and Systems Management, IESM'13, I4E2, Rabat, Morocco (2013)
25. Pach, C., Berger, T., Bonte, T., Trentesaux, D.: ORCA-FMS: a dynamic architecture for the optimized and reactive control of flexible manufacturing scheduling. In: Computers in Industry, vol. 65, no. 4, pp. 706–720 (2014)

Semantic Model to Perform Pluggability of Heterogeneous Smart Devices into Smart City Environment

Andre Dionisio Rocha, Pedro Fernandes, Catiele Lima and Jose Barata Oliveira

Abstract Nowadays smart cities are becoming more and more a hot topic in the technological world. Some different approaches emerged and many city departments approved investigation and implementation of the smart city technology in their cities. The implementation of these processes is becoming a landmark for the modern cities. This paper proposes a generic semantic model to easily plug and manage heterogeneous smart devices and areas of cities, in order to integrate all the diverse components that constitute a fully integrated and functional smart city environment. This semantic model can be used in the most diverse city scenarios; to demonstrate it, a specific scenario is presented in this paper, describing the usage of the proposed semantic model to detect new components and share information among the smart components.

Keywords Smart city · Internet of Things (IoT) · Cloud computing · Plug-and-Play · Cyber physical systems · Semantic model · Ontology · Smart device

---

A.D. Rocha (✉) · P. Fernandes · J. Barata Oliveira
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa,
Costa da Caparica, Portugal
e-mail: andre.rocha@uninova.pt

P. Fernandes
e-mail: pc.fernandes@campus.fct.unl.pt

J. Barata Oliveira
e-mail: jab@uninova.pt

A.D. Rocha · C. Lima · J. Barata Oliveira
Centre for Technology and Systems, UNINOVA Institute,
Caparica, Portugal
e-mail: catiele.lima@uninova.pt

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_29

1 Introduction

The studied and commonly suggested smart cities are expected to bring into modern society a way of optimizing the available resources of the city, such as natural resources, human resources, street traffic, and many others available in an evolved city. Many city governments, companies and academic institutions started investing in this technology. The smart city goal is to adjust the city processes to the population living in it. The city has to interpret the population routines, the available resources and the external factors as meteorological aspects, population variations, and many others in order to optimize the city processes [1]. The emerged technologies based on distributed systems for industrial environments, cloud based platforms and Internet of Things are pushing the development of the smart city concept. A semantic model will be presented, based on distributed systems applied to a smart city. The goal is to create a technological organism capable of acquire information from the environment and act on the real world, working as a technological community that helps the living community of a city, improving the life quality of the population. Some smart city approaches will be described in the second chapter, as well as some concepts, in order to let the reader know about some important topics when talking about smart cities. The proposed semantic model is introduced and described in the third chapter. The fourth chapter will present an overview of an instantiation of the proposed semantic model, showing an example of an agent based implementation using the semantic model.

2 Related Work

2.1 Smart Cities

The process of adding intelligence to the city processes is accomplished by solving day-to-day problems related with the population routines, gathering real time information about the city resources, and implementing a feedback system between the city and the people in order to manage efficiently the existing processes and improve the life quality of the community in this way.

One of the pioneers on the smart city concept was IBM. This giant of technology, talking about gathering data from a wide variety of sources and using that data to solve new problems for clients and customers [2], defined the concept of connected environment or “smarter planet”. Directing this technological concept to the cities, claiming that cities are being empowered technologically, as the core systems on which they are based become instrumented and interconnected, enabling new levels of intelligence [3] brought up a new awareness of the city processes and their automation. The use of a technological approach to manage the city processes and resources became a turning point on unifying the city departments.

Masdar city [4] in Abu Dhabi was one of the pioneer attempts to create a smart city from scratch. The big goal was to free the city from carbon dioxide emissions, and create an environment in which one can walk, fresh, and car-free. The transportation system was based on an electric public underground net of self-driving cars. The main problem was that the city was designed having in mind the resources management and not the people behaviour. In other words, the city couldn't adapt to the population, but the population had to adapt to the city.

The Songdo project in South Korea [5], was more a people directed smart city project. There was taken in consideration the population feedback in order to improve the processes. The resource management and energy efficiency was controlled by the needs of the population. A symbiotic relationship between the smart city and the population inside has been created.

Both these cities above described were built from scratch, entirely new. It took some time to build them, and many cutting edge technologies that were incorporated in the early project became obsolete because the technology progression was too fast for the construction of the city to catch up, although it was an pioneer work for a lot of developers and researchers. The cities of the future had already arrived.

In a more recent approach to the smart city technology, developers work on already existing cities trying to improve their processes. Cities like Barcelona, Boston, Hong Cong, Stockholm [6], are developing smart processes and incorporating them in the city, getting immediate feedback from the populations which always reacts to changes on their environment and routines. This is a smart approach, because it forces the technology to grow around people and adapt to the community, improving its life style and quality.

Analyzing all of the cases of study above mentioned it becomes possible to create some general guidelines to build a safe, versatile, smart city capable of adapting its processes to the population. For that it is necessary to find the basic and structural city components, study the relationship between them or as it will be called from now on, the city processes establish a map of basic behaviours for each city component and city process, and based on these assumptions create a technological identity for each e one of them. Once created this identities, it is time to explore the potential associated to them, in order to optimize all the city processes, make a common line of communication between all the active technological members of the smart city creating a technological community that will be capable of evolving with the population itself.

2.2 Concept Awareness

2.2.1 IoT and Cloud Computing

The technology needed to create, host, and program the processes of a smart city is directly related to the cyber physical systems [7] because it is necessary to collect information from the physical environment and process it in order to create the suitable response to the most different kinds of situations.

The marriage of IoT with Cloud computing is a powerful cyber-physical system [8]. The IoT is the designation used for the connection of several devices capable of reading and/or acting in the environment where they are inserted, being all connected in a common network that allows them to communicate data. This common network can be the cloud, which brings a lot of advantages. Its structure explained in [9] allows virtually unlimited space and processing power. The cloud computing technic came to solve what was a big limitation in the IoT technology.

2.2.2 Pluggability

In industrial multi-agent platforms such as PRIME [10], there is a concern about agility and flexibility of the industrial process. It means that for a constantly and rapidly changing market it is necessary to implement fast changes in the production lines. The main goal is to create the ability of adding and removing components to the production line in order to fulfil the market needs.

Transposing this concept to the smart city, the goal is to come up with a standard model for a smart city where the components can be added, removed or created as needed; thus it is possible to adapt the model to a variety of different smart city objectives and ideas. The semantic model for this concept will be introduced in the next chapter.

3 The Proposed Semantic Model

The proposed semantic model is composed by a set of classes and subclasses with a set of data properties and connections between them or object properties. All the classes and subclasses which constitute the semantic model are firstly presented in Fig. 1.

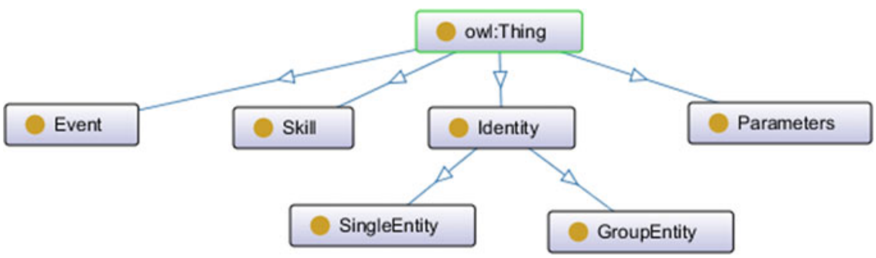


Fig. 1 Core concepts of the semantic language

3.1 Classes Description

This is the semantic model and each class has a specific purpose. The dynamics of the model is represented by the relations between them. The class descriptions are:

- **Parameters**: This class is composed by an ID (string), data type (string) and a value (string), which is used to define the input parameters of the skills that will be created.
- **Skill**: The skill class has the input parameters composing the skills that represent the behaviours and communications between the entities, as well as the ID that defines the skills.
- **Identity**: This class has the necessary parameters to properly identify and characterize a type of entity in terms of physical location and technological identity. This class has two subclasses, the single entity class and the group entity class.
- **SingleEntity**: This class represents an entity of any kind, specifically a singular entity. It is an entity capable of any kind of interaction with the environment; it can be a basic component of the smart city or device, such as an air conditioner or a smart TV. The single entity has a set of skills, and can communicate with another single entity or with group entities.
- **GroupEntity**: this class represents a set of single entities that can be organized or classified in a group to work together, to organize the entities with similar behaviour, to act in the same physical environment or any other organizational criteria. A group of entities can be a housing block—the devices inside a house, a group of micro-grids, or a combination of any other smart city components. A group entity can communicate with other group entity or with a single entity.
- **Event**: when a certain entity executes a certain skill or a combination of different skills, this class launches the event; the identity of the entity that emitted the event is stored in this class in order to allow the communication. It must be specified which skill or skills are being called and their parameters.

Since all the proposed and designed classes and subclasses are described, it is now crucial to specify how all these classes relate with each other in order to understand which information can and/or must influence other information stored in a different type of individual. In Fig. 2 one can see besides classes the object properties. Basically the object properties identify dependencies and/or usage of data from individuals of different classes. For instance, hasParameter indicates that each individual Skill has a list of parameters described according to the Parameters class.

Table 1 contains all the modelled properties for the semantic model.

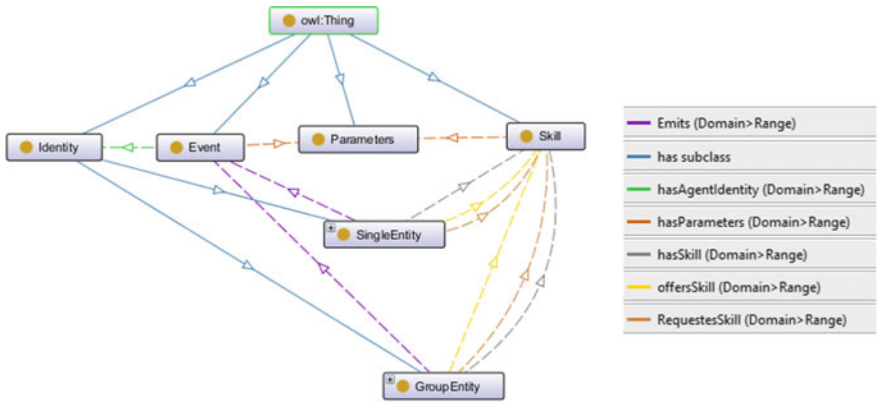


Fig. 2 Core concepts and relationships of the semantic language

Table 1 Properties that define the relationships between the classes

Properties	Description
hasParameter	Allows the connection between any class and the parameter class
hasSkill	Instantiates the relation between an entity, single or group, and a Skill
offersSkill	It relates an entity to the skills it has to offer to other entity
requestsSkill	It instantiates the skills a certain entity can request form other entity
hasEntityIdentity	Creates an instance of the entity's identity that called the event
Emits	Allows the entity class (SingleEntity or GroupEntity) to call for an event, that is a skill itself

4 Agent Based Instantiation

The proposed semantic model is to be used in the project phase, being the main tool to develop the smart city architecture, instantiating the basic processes, regardless of the programming platform or technic that will be used for the implementation. To exemplify the applicability of the proposed semantic model is used a multi-agent programming approach.

Figure 3 shows an example of the instantiation containing two local grids and three smart housing blocks each one abstracted as a Grouped Agent. On the other hand, the five micro grids and five houses are abstracted as Single Agents.

4.1 Agents Description According to the Semantic Model

The following description is based on the semantic model, translating the given hypothesis, proposed in 4. The description is simplified in order to better

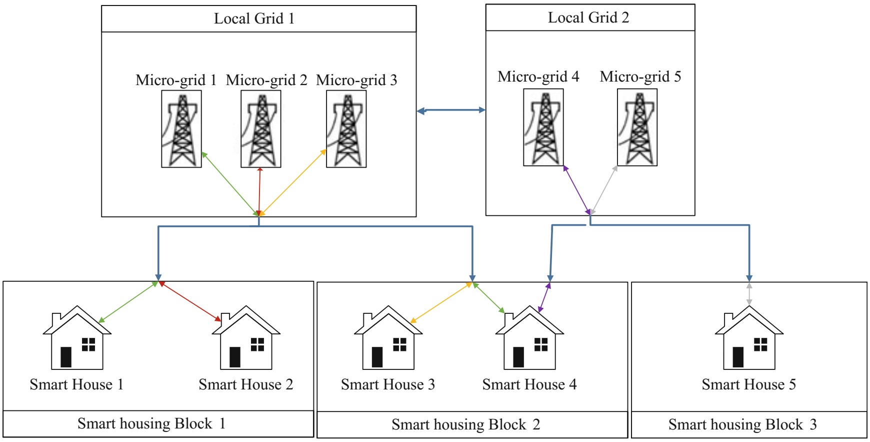


Fig. 3 Example of the proposed smart city environment and communication lines

understand the basic concept of creating the technological identity and prepare the agents for a simple communication task, through the events, using the existing skills.

4.2 *Agents Relations and Communication*

The first three micro-grids (micro-grid 1 to 3) are grouped into the first local grid (local grid 1), which means that each individual micro-grid can communicate with the local grid. In this case the micro-grids can't communicate directly with each other; the only way for them to communicate is through the local grid group agent. The same happens in local grid 2 which contains the micro-grids 4 and 5 (Fig. 4).

The local grid 1 can communicate with the local grid 2, which means that if the local grid 1 by any reason wants to know some information about the micro-grid 4

Smart House
Type: Single Entity
Identity: Smart House
Skills: "askForGridConnection"

Smart Housing Block
Type: Group Entity
Identity: Smart Housing Block
Skills: "ForwardTheHouseRequest"
"ForwardTheGridResponse"

Micro-grid
Type: Single Entity
Identity: Micro-grid
Skills: "ReturnAvailability"

Local Grid
Type: Group Entity
Identity: Local Grid
Skills: "AskForMicroGAvailability"
"ReturnConnectionStatus"

Fig. 4 Agent description

it can ask about it to the local grid 2, that in turn will ask to the micro-grid 4, and the information will make the inverse path reaching the local grid 1.

In the smart housing block the idea is identical. The smart house 1 and 2 are single agents part of the smart housing block (group agent); the smart houses cannot communicate with each other directly, only through the smart housing block agent.

The blue connection in Fig. 3 represents the connection between the group agents. The other colours are associated to the connections between the smart-houses and the micro-grids. The connections exist when the arrow has the same colour in both the micro-grid and the smart house (this represents a direct connection), which means direct communication. On smart house can be connected to multiple micro-grids and vice versa. The micro-grid 1 is connected to the smart house 1 and smart house 4. The smart house 4 is connected to both micro-grid 1 and micro-grid 4.

The communication process is done through events that represent the agents' calls for a certain skill or set of skills. The communication between the entities can comply with hierarchical rules based on location, clearance level or any other criteria.

As a practical example of agents communicating though their skills, let us consider that agent smart house 1 has the skill "ask for grid connection". Once smart house 1 needs to connect some device to the grid, this agent will check the established hierarchical rules, and send the request to the smart housing block. This agent will find out which local grid it can interact with and "forward the grid request" to the local grid that is more suitable. Once the local grid receives this request, the agent will ask the micro-grids, through the skill "ask for micro-grid availability", which ones are available. All the micro-grids will reply "I'm available" or "I'm not available". The local grid analyses the answers and forwards the one that is more suitable to the smart housing block which in turn will forward it to the smart house. This is a basic example of the communication between the agents using their innate skills.

5 Conclusions

The use of a pluggable solution such as the multi-agent system in a smart city can be a really agile and flexible option to the implicit needs of a fast response to scenario changes. The proposed semantic model can be a strong solution for the complex and always changing processes such as pluggability and constantly sharing of data. The Semantic model can be used in diverse approaches to any desired architecture of a modern smart city, allowing re-creating the city processes in a technological environment in order to control and optimize them. The built-in relations between this classes described in the semantics allow the developer to describe the process as he wants, in order to reach the desired outcome out of a specific process. This is a basic structure in a smart city technological approach. This semantic model is to be used for creating the technological concepts of the basic actors of a modern smart city, on the top of which the functionalities and relations between them can be programmed in order to achieve the desired outcome, optimizing the smart city processes. This approach allows the scalability of the modular city. Once the technological identity is created, it is virtually possible to scale infinitely, the only requirement being the existence of the infrastructures. This semantic model can be used to instantiate any city process.

References

1. Wang, Q., Meegan, J., Freund, T., Li, F.T., Cosgrove, M.: Smarter city the event driven realization of city-wide collaboration. In: Proceedings of 2010 International Conference on Management of e-Commerce e-Government, ICMeCG 2010, pp. 195–199 (2010)
2. At IBM Research: A constant quest for the bleeding edge—CNET. http://www.cnet.com/news/at-ibm-research-a-constant-quest-for-the-bleeding-edge/
3. Dirks, S., Keeling, M.: A Vision of Smarter Cities, p. 18. IBM Glob. Serv., New York (2009)
4. Lau, A.: Masdar city: a model of urban environmental sustainability. Stanf. Undergrad. Res. J. **11**, 77–82 (2012)
5. City of the Future: Songdo, South Korea. The Network. https://newsroom.cisco.com/songdo. Accessed 16 Apr 2016
6. Departament for Business innovation & Skills, Global Innovators: International Case Studies on Smart Cities Smart Cities Study, no. 135, p. 58 (2013)
7. Baheti, R., Gill, H.: Cyber-physical systems. Impact Control Technol. **1**, 161–166 (2011)
8. Mitton, N., Papavassiliou, S., Puliafito, A., Trivedi, K.S.: Combining cloud and sensors in a smart city environment. EURASIP J. Wirel. Commun. Netw. **2012**(1), 247 (2012)
9. Botta, A., de Donato, W., Persico, V., Pescapé, A.: Integration of cloud computing and internet of things: a survey. Future Gener. Comput. Syst. **56**, 684–700 (2014)
10. Di Orio, G., Rocha, A., Ribeiro, L.D., Barata, J.: The PRIME semantic language: plug and produce in standard-based manufacturing production systems. In: The International Conference on Flexible Automation and Intelligent Manufacturing (FAIM 2015), p. 8 (2015)

Active Monitoring of a Product: A Way to Solve the "Lack of Information" Issue in the Use Phase

Vivien Basselot, Thierry Berger and Yves Sallez

Abstract The small firms can be satisfied with their quick reaction to customer requests and their creativity but they meet big issues internally about the products sold. These issues put ahead a lack of information when the product is out of the SME. After an analysis of the origin of the problem, the paper proposes to use the active product concept as a key element to solve the problem. A preliminary model helping to support this concept is proposed, and a use case exemplifies how an active product can have an important impact to resolve the problem of "lack of information" in the use phase.

Keywords Active product · PLM · Closed-loop PLM · Intelligent system

1 Introduction

"Technological progress is not translated into economic benefits and jobs by governments, countries, or sectors, but by innovative firms" [1]. This is why SMEs play an important role in innovation. Today SMEs policies are reconsidering their focus on new technology. But they are confronted with one main issue: after their creations/products have been sold, they lose their view on it and lose a lot of information (e.g. how the product is used). In this paper we are going to try to highlight where they lose their control on the product and how it is possible to

V. Basselot (✉) · T. Berger · Y. Sallez
University Lille Nord de France, 59000 Lille, France
e-mail: vivien.basselot@univ-valenciennes.fr

T. Berger
e-mail: thierry.berger@univ-valenciennes.fr

Y. Sallez
e-mail: yves.sallez@univ-valenciennes.fr

V. Basselot · T. Berger · Y. Sallez
UVHC, LAMIH UMR no. 8201, 59313 Valenciennes, France

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_30

recover information from their products. This paper proposes early research on how an active product can respond to some issues encountered by SMEs. The Sect. 2 is dedicated to a presentation of a major issue: the lack of information from the product and how we can approach the problem to solve it. Focusing on the use phase, a functional modelling of an "active" product is proposed in Sect. 3. This model is applied in Sect. 4 for the "active" monitoring of refuelling pumps in a shop-floor. The last section gives conclusions and perspectives.

2 A Need of Information

2.1 Traditional PLM

The small firms can be satisfied with their quick reaction to customer requests and their creativity but they meet big issues internally about the products sold (and so outside the SME). For a better understanding of the involved problematic, the traditional Product Lifecycle Management (PLM) is briefly presented. According to Stark [2], Terzi et al. [3], three main phases are typically encountered in a PLM:

- *Beginning of life* (BOL), including design and manufacturing systems, and commercial service,
- *Middle of life* (MOL), including logistics (i.e. distribution), use (i.e. exploitation), and maintenance,
- *End of life* (EOL), including reverse logistics (i.e. collecting damaged product), remanufacturing (e.g. disassembly and other treatments) and recycling.

In a traditional PLM approach it is known to encounter several issues [4]. Some examples can be cited:

- In the BOL phase there is a lack of communication between the commercial, design and manufacturing services. All the information concerning the product is managed via CAD/CAM systems or product data management systems. However all the SME services do not use these information flows which cause an issue during the design phase.
- In the MOL phase, the multiplicity and diversity of customers makes difficult the collecting of information on the products. Moreover, the products are outside the SME. The SME can only expect to handle some information when customers ask for the support system (e.g. maintenance) when the product is damaged. This limited visibility on each product can imply a loss of effectiveness of all services/systems of the SME by a lack of information feedback from products.
- In the EOL phase, the limited visibility on the previous use phase introduces difficulties to determine efficiently the parts of the product to recycle.

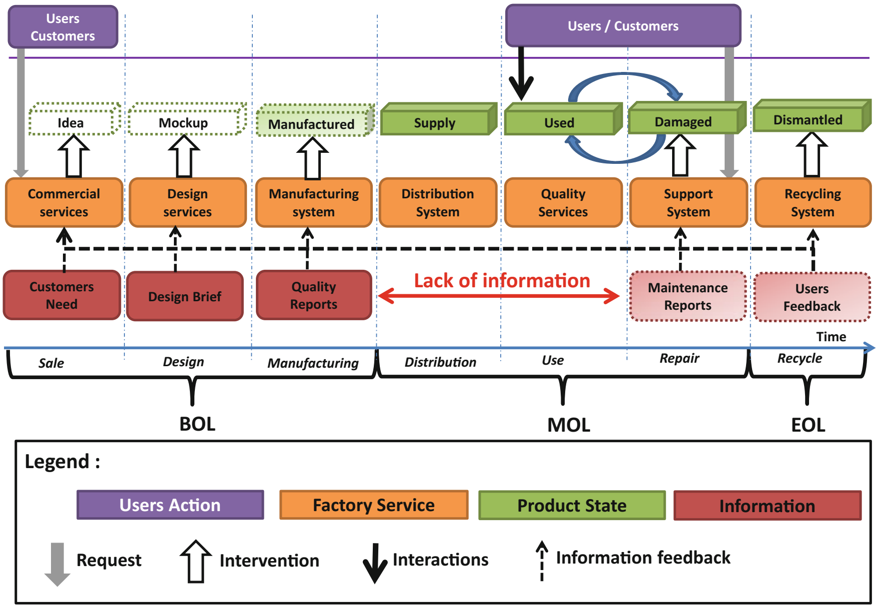


Fig. 1 Traditional product lifecycle with lack of information highlighted

Figure 1 shows during the BOL, MOL and the EOL phases the interactions between the users/customers (in purple) and the SME. In this figure, different SME services are depicted in orange, the product state in green, and, in red coloured, different product information recovered by actors in the SME.

The lack of information during the MOL phase causes breaks in the information feedback flow and prevents the expansion of SMEs. The SMEs cannot react quickly if they do not have information and cannot find effective solution if a problem appears during the “lack of information” period. During all the phases of the PLM, notably the MOL phase, the product is considered as passive and undergoes the different operations. To counterbalance this lack of information, a possible solution consists in transforming the “passive product” as “active product”. As a first step towards this direction some researchers proposed a new PLM approach presented in the following section.

2.2 *Closed-Loop PLM*

The closed-loop PLM focuses on the complete product life-cycle and was introduced in the PROduct life Management and Information tracking using Smart Embedded systems (PROMISE) project [5]. This concept initially relies on interactions among three entities: a PLM agent, a PLM system and a Product Embedded

Information Device (PEID). The PLM agent gathers product lifecycle data from the PEID using a mobile reader device. It sends the information gathered at each site (e.g., manufacturing sites, distribution sites) to a PLM system [6]. As explained by Jun et al. [4], unlike traditional PLM, closed-loop PLM focuses on the complete product lifecycle, with an emphasis on tracking and managing the information from the whole product lifecycle and potentially returning information to each phase in the process. These returns of information, or feedback, can be performed via several forward and backward closed-loops linking the BOL, MOL and EOL [7].

With the increase of embedded technology (e.g., Wireless communication, RFID, smart processors…), the concepts of ambient intelligence [8], pervasive Computing [9], Auto Identification [10], machine-to-machine intelligence [11], Intelligent Product [12], active product [13] and Cyber Physical Systems [14] were introduced. An analysis and a typology of most of these concepts, which are out the scope of this paper, can be found in [15]. In the present paper the concept of active product [13] is used. On the contrary to a passive product, an active product (with the help of a technological module, called "augmentation module", associated to the passive product) can hold information, make decisions, trigger information and, if needed, interact with its own users. So the active product can give information during all the phases, notably the use phase where an important lack of information was identified (cf. Fig. 1). An analogy with studies about Cyber-Physical Systems can be done to put ahead the interests of the active product. For example, based on the study of Herterich [14] three major interests and impacts can be highlighted:

- ***Data flow for services***: Data collected by an active product can be used to create and optimize the data flow in each service activities.
- ***Data analysis, product optimization***: The data collected from the active product allow SME to analyse the used-product (e.g. if it is well-used), create a new product with eventually less problem and better performances (e.g. better cost, energy, ergonomics…).
- ***Remote maintenance and diagnostic***: Continuous data collection might be used to trigger maintenance service. This service can repair or make a diagnostic with remote maintenance.

In Fig. 2, an active product during all its lifecycle is considered.

During the BOL the product follows the same way that a traditional PLM.

In the MOL phase, the "active" product can hold information and trigger its SME services/systems (predictive and remote diagnostics) like for example:

- The "distribution phase" to give a report on how it has been moved, stored.
- The "repair phase"; the customer service can get back all the information present on the product to repair it. This information can equally serve to take decisions to improve the maintenance support system.

As depicted on Fig. 2, the introduction of "active" products increases quantitatively and qualitatively the information flow between the different phases of the life cycle.

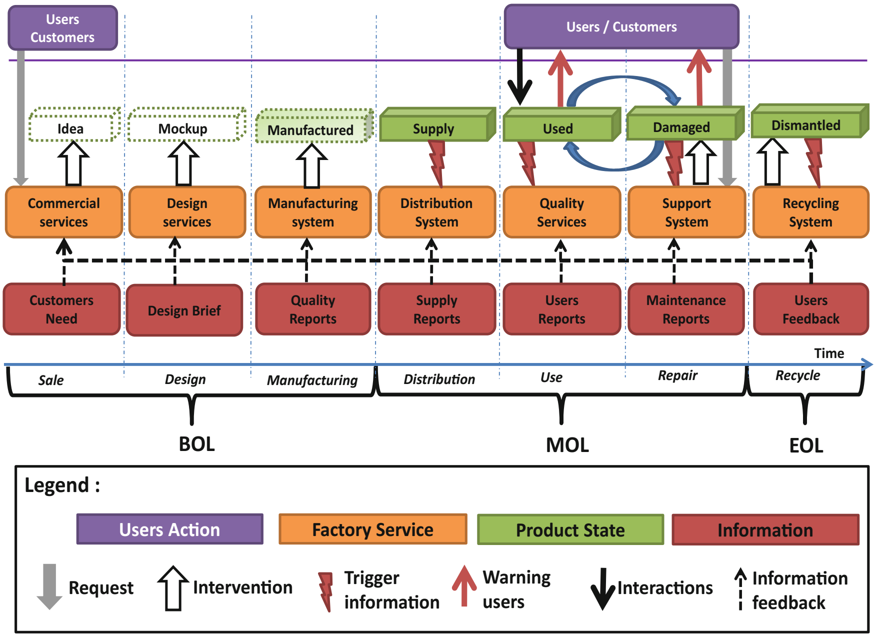


Fig. 2 Closed-loop PLM with an "active" product

The next section details the interest of the active product concept to solve the "lack of information" issue in the MOL phase, more particularly in the use phase, and proposes a model.

3 Focus on the MOL Phase and Proposed Model

3.1 Active Product in MOL Phase

An active product [15] is a product in which triggering is the minimal activity. In other words, in an event-oriented approach, the product is able to identify its state, compare its state with the desired one and, when certain conditions are met, store, send information to a support centre for example. From an external point of view, the requirements for an active product during its use phase (targeted by this paper) concern, for example:

- *Monitoring*, the product checks and self-diagnosis its state for a better availability,
- *Localization*, the product can be geo-localized to avoid to lose it,
- *Integrity*, the product checks the conditions of use (e.g., shock, temperature),

- *Traceability*, permitting to verify the history of the product (e.g., location, "health variables", dates of the maintenance operations).

3.2 Proposed Model

The model proposed in [15] is revisited and enhanced to introduce a more clear distinction between customers and provider of a product. As introduced in [15], the product, denoted target system $\sum^{T}$, provides services associated to the primary functions to users/customers. In addition to these primary functions, certain secondary functions are also needed during the MOL phase. These secondary functions allow $\sum^{T}$ to be effectively maintained. During the use phase, $\sum^{T}$ is operational and executes all the primary functions and some secondary functions, called internal secondary functions. For example, a printer's primary function is to print, but it can also warn the user of a lack of paper, which is an internal secondary function. The other secondary functions, called external secondary functions, are supported by two systems. As depicted in Fig. 3, the model represents certain external secondary functions taken in charge automatically (i.e., without human intervention) by a specific support system called the augmentation system, denoted $\sum^{S_A}$. The most basic $\sum^{S_A}$ must be able to trigger certain useful functions (implemented with tasks) that are needed by $\sum^{T}$. A set of external secondary functions is supported by $\sum^{S_A}$, and the remaining external secondary functions are supported by the external support system, denoted $\sum^{S_{\bar{A}}}$. In this enhanced model, two sub-systems can be considered in $\sum^{S_{\bar{A}}}$ relatively to the customer or the provider of the product. $\sum^{T}$, $\sum^{S_A}$ and $\sum^{S_{\bar{A}}}$ are connected by links (permanent or temporary), which can be informational and/or physical.

The following example illustrates the modelling of a system's monitoring and remote maintenance application (see Table 1) and distinguishes five functions. The primary function f1 deals with the task to perform and depends on the system being

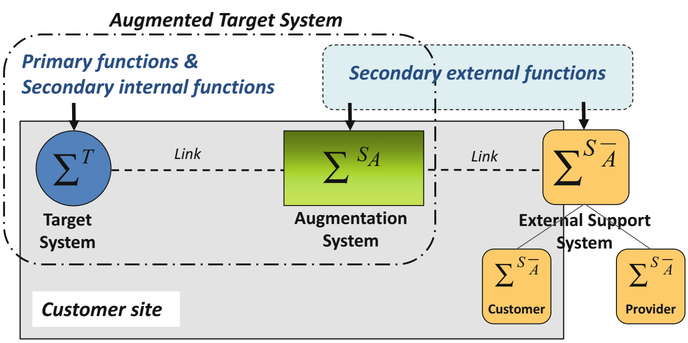


Fig. 3 Functional description with customer and provider support systems

Table 1 Description of the functions

Functions	Description	Type	Augmentation
f1	Perform principal task	Primary	No
f2	Record any failure or operation in the history file	Secondary external	Yes (embedded)
f3	Perform an advanced diagnosis of any failures	Secondary external	Yes (embedded)
f4	Manage maintenance operations of first level	Secondary external	No
f5	Manage remote maintenance operations	Secondary external	No

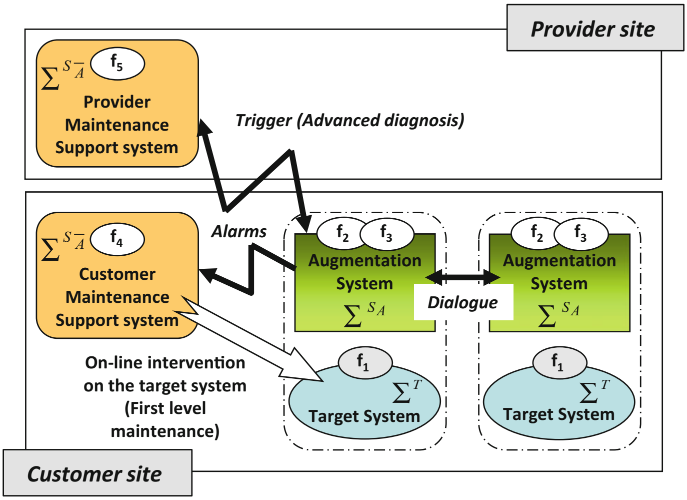


Fig. 4 Example of mapping of the functions

studied. Two secondary functions are responsible for system monitoring and diagnosis (f2 and f3). The maintenance is managed at two levels:

- The secondary function f4 is managed by the customer maintenance service and concerns the first level maintenance operations (f4).
- The f5 secondary function is performed by the provider that manufactures the system and concerns more advanced remote maintenance operations.

Figure 4 illustrates the distribution of these functions and exhibits the different actors. In this example, a "local" augmentation system on each piece of equipment supports the functions f_2 and f_3.

This generic "active" monitoring approach is next detailed in a use case.

4 Use Case

In this section, the focus is held on the use phase via the active monitoring of spatially distributed refuelling pumps in a shop-floor. These refuelling pumps are classically used in the automotive industry to supply fluids (e.g. fuel, oil, coolant) to the vehicles. The car manufacturer can use alternatively or simultaneously a set of pumps in different places of its installation.

The SME that manufactures and provides the pumps is equally responsible to maintain the provided pumps in collaboration with the end user/customer. This last one performs the maintenance operations of first level and the SME intervenes for more complex maintenance operations.

As in the generic example described in Table 1, five functions are considered: for this use case, the primary function f1 deals with the transfer of the different fluids. The four secondary functions are responsible for pump monitoring (f2 and f3) and for maintenance operation management (f4 and f5). More precisely, two of the secondary functions (f2 and f3) are supported by the augmentation system (Fig. 5) (The implementation is out of the scope of this paper. However to give some elements, the IoT and embedded systems provide the major "technological bricks"). The details of the four secondary functions are:

Function f2 (**Traceability**), which is embedded, keeps a history of the events (e.g. failures, usage conditions, dates of the maintenance operations).Two sets of recorded data are considered:

- The first is considered as "private" for the user and regroups different operational data (e.g. type of the refuelled car, type of fluids, users IDs, serial number).

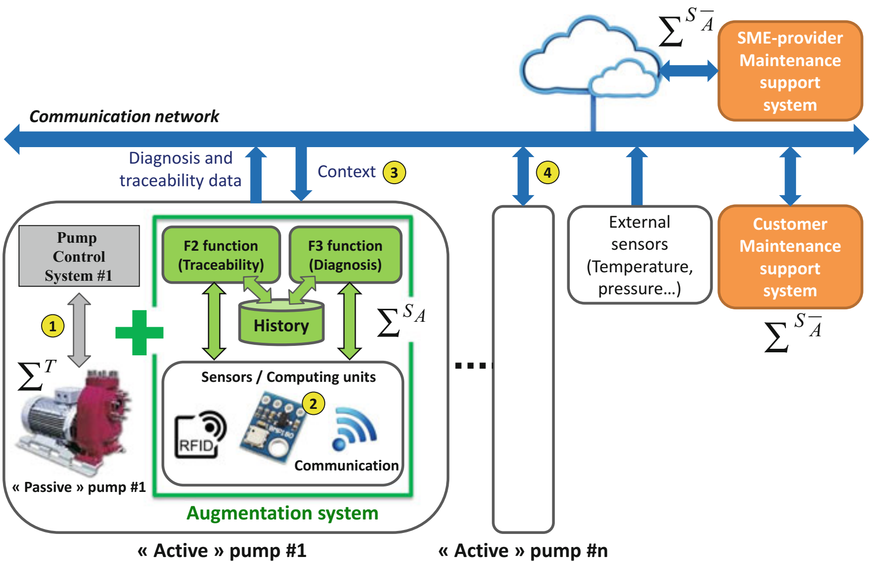


Fig. 5 Illustration showing the f2 and f3 functions

- The second set is considered as "public" and can be exploited by the provider accordingly with a customer-provider agreement. This data set contains "health variables" and information on the usage conditions (e.g. external temperature, shocks detection).

The first data set of this history is exploited by the user/customer maintenance centre in order to optimize the maintenance process and reduce the associated costs.

As outlined in the Sect. 3.1, the second data set constitutes a crucial information flow for the provider. It allows this last one to monitor the usage of its product and can be used to solve, for example, conflicts between the provider and the user/customer. In addition, it constitutes an important source of information for the others phases of the life cycle as highlighted in Fig. 2.

Function f3 (**Diagnosis**) relies on non-intrusive monitoring of the refuelling pump and can exploit several information sources (numbered in Fig. 5):

(1) The pump controller's input/output signals and states,
(2) Pump sensor measurements (e.g., vibration sensor),
(3) The data describing the pump's context or environment obtained via external sensors (e.g., external temperature),
(4) The information exchanged with the other "sister" augmentation systems.

Function f4 (**Maintenance by the user**) and ***Function f5*** (**Remote maintenance by the SME**) exploits the outcomes of the two previous functions helping the user to insure classical maintenance operations. If user needs help, the SME can take over the maintenance; with the advanced diagnosis from f2 and f3 the SME support maintenance system can do remotely some maintenance operation (e.g. set-up, configuration, diagnosis, performance monitoring, analysis…) as described in [16] or [17].

5 Conclusion and Perspectives

The small firms can meet main issues due to the lack of information when the product is out of the SME. After an analysis of the origin of the problem, the paper proposes the active product concept as a key element to solve this problem. A model helping to support this concept is proposed, and introduces a distinction between customers and provider of a product.

With this model as support, a generic "active" monitoring approach (in the use phase of a product) is applied to a set of spatially-distributed refuelling pumps in a shop-floor. This use case has shown that the model can help to make a distinction between functions and permit to represent their mapping embedded on an augmentation system or remotely in a support system. With this model SMEs will keep an "eye" on their product and can remain innovative and productive to comfort or increase their market.

The short term perspective of this preliminary work is to define criteria to help to decide which functions have to be embedded close (i.e. on an augmentation module) to the passive product or have to be supported by external remote systems (i.e. support systems).

The mid-term perspective of this work is to develop tools and a methodology based on the "active product" concept to reduce the problem of "lack of information" identified during the use phase of a (passive) product.

References

1. OECD, Technology, Productivity and Job Creation, Paris (1996)
2. Stark, J.: Product Lifecycle Management: 21st Century Paradigm for Product Realization, 3rd edn. Springer, London (2006). ISBN 1-85233-810-5
3. Terzi, S., et al.: A new point of view on Product Lifecycle Management. In: Proceedings of 5th International Conference on Product Lifecycle Management, Seoul, Korea, 11–13 July 2008
4. Jun, H.B., et al.: Research issues on closed-loop PLM. Comput. Ind. **58**(8–9), 855–868 (2007)
5. PROMISE, 'PROduct lifecycle Management and Information tracking using Smart Embedded systems'. http://www.promise.no (2008)
6. Kiritsis, D.: Closed-loop PLM for intelligent products in the era of the internet of things. Comput. Aided Des. **43**(5), 479–501 (2011)
7. Shin, J.H.: Decision support methods for closed-loop conceptual design. Ph.D. thesis, Ecole Polytechnique Fédérale de Lausanne (2009)
8. Lindwer, M., et al.: Ambient intelligence visions and achievements: linking abstract ideas to real-world concepts. In: Proceedings of Design, Automation and Test in Europe Conference. (DATE 2003), Munich, Germany, Mar 2003
9. Gupta, S.K.S., et al.: An overview of pervasive computing. IEEE Pers. Commun. Mag. **8**, 8–9 (2001)
10. Engels, D.W., Sarma, S.E.: The reader collision problem, AUTO-ID Center White paper. http://cocoa.ethz.ch/downloads/2014/06/None_MIT-AUTOID-WH-007.pdf (2002)
11. FING, Machine to Machine. White paper co-directed by Fing, Syntec Informatique et France Telecom, Paris (2006)
12. McFarlane, D., et al.: The intelligent product in manufacturing control and management. In: 15th IFAC Triennial World Congress, Barcelona, Spain, 21–26 July 2002
13. Sallez, Y., et al.: Service Orientation in Holonic and Multi-Agent Manufacturing and Robotics: Proposition of an Analysis Framework to Describe the "Activeness" of a Product during Its Life Cycle, pp. 257–270. Springer International Publishing (2004) ISBN 978-3-319-04735-5
14. Herterich, M., et al.: The Impact of Cyber-Physical Systems on Industrial Services in Manufacturing. Product-Service Systems Conference—PSS, industry transformation for sustainability and business (2015)
15. Sallez, Y., et al.: The lifecycle of active and intelligent products. The augmentation concept. Int. J. Comput. Integr. Manuf. **23**, 905–924 (2010)
16. Muller, A., Marquez, A.C., Iung, B.: On the concept of e-maintenance: review and current research. Reliab. Eng. Syst. Saf. **93**(8), 1165–1187 (2008)
17. Sangregorio, P., et al.: An integrated system for supporting remote maintenance services. In: IEEE 20th Conference on Emerging Technologies & Factory Automation (ETFA). IEEE (2015)

Part VII
Engineering and Human Integration in Flexible and Reconfigurable Industrial Systems

A Description and Analysis Method for Reconfigurable Production Systems Based on Finite State Automaton

Filippo Boschi, Giacomo Tavola and Marco Taisch

Abstract New production systems are highly reconfigurable and interact with dynamic industrial environments. Their modelling, simulation and analysis of the operations and evaluation of performances are now much more complex than in the past when system had a static and predefined behaviour. This paper proposes a method to describe and analyse complex production systems, based on utilization of FSA (Finite State Automaton). This approach is enabling better understanding and sharing with stakeholders of how a system works, but it is also a good basis for computer based simulation and control. The interaction with external environments is structured in terms of External Events (inputs) and Trigger Outputs. The analysis of the system state evolution in the time domain provides the possibility to calculate KPIs (Key Performance Indicators) in specific conditions or their evolution. In this paper a simplified language syntax describing the automaton including output generation and triggering of external functions of the production environment is proposed. The approach is implemented and demonstrated in a particular industrial domain: industrial machinery fabrication sector.

Keywords Cyber physical systems · Finite state automaton (FSA) · Reconfigurable production systems · Modelling · KPI

F. Boschi (✉) · G. Tavola · M. Taisch
Politecnico di Milano, Via Lambruschini 4/b, 20156 Milan, Italy
e-mail: filippo.boschi@polimi.it

G. Tavola
e-mail: giacomo.tavola@polimi.it

M. Taisch
e-mail: marco.taisch@polimi.it

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_31

1 The Evolution of Production Systems Towards CPS Controlled Environments

An aggressive market competition on a global scale and the increasing frequency of new product introduction and update are forcing companies to continuously upgrade their production capabilities and capacities, leading to a rapid changes of traditional manufacturing paradigm [1].

As a consequence, manufacturing companies, being challenged by volatile markets and uncertain sales forecasts, are challenged in aligning their production systems [1]. For these reasons, the reconfigurability and the changeability (the ability to get early and foresighted adjustments of the factory's structures and processes on all levels to market change) are seen as key aspects that the current industrial production systems have to provide for a strong competitiveness [2].

In order to face these problems, Cyber Physical Systems (CPS) could be the key enabling technology. Thanks to their ability to connect the physical part of each component involved in the production system with its virtual concept, CPS are able to create a unique environment among the data coming from the shop floor and the information concerning the overall aspects of the value chain (i.e. dynamic market demand, products' and equipment's life cycle data). The interaction between the environment and the reactive actions taken through CPS, which translates physical input events into logical ones and logical output events into physical output events, can be mapped using a descriptive automaton-based method [3]. This method provides a description of interactions between physical and cyber parts of CPS, combining finite state-transition which happens in the physical part with a finite set of real-valued variables that evolve in the cyber part as time progresses while the system occupies a state [4].

Thus, this paper proposes a model based on Finite State Automaton (FSA) able to describe the overall state, the sequence of events and evolution of production processes as discrete variables that will be executed in the cyber environment in order to manage in real time the right information, to take the correct decisions and, therefore, to guarantee the correct degree of flexibility and reconfigurability. The proposed approach is mainly oriented to discrete manufacturing production environments.

2 How a FSA Can Describe a Reconfigurable Production System

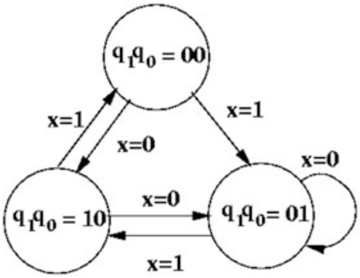
2.1 Automaton Types and Formalization

A generalized sequential logic system that can be described by a number of outputs (n, o) which depend on the present and the past values of the input (n, i) can be formalized as a finite state machine (FSMs) [5]. It is a mathematical abstraction where all states represent all possible situations in which the state machine may ever be. As the number of distinguishable situations for a given state machine is finite, the number of states is finite too. Hence, it is a behaviour model composed of finite number of states, transitions between those states and actions [6].

Such process that provides as a result the set of outputs of the machine starting from a sequence of values as input can be specified as a state machine (SM) by defining a 5-tuple (Σ, Q, q0, F, δ), as described in literature [7], where Σ is the set of symbols representing input to SM, Q = {S1, S2, Sn} is the set of states of SM, q0 ∈ Q is the initial state which is the state at time 0 of M, F ⊆ Q is the set of final states of SM and δ is the transition function. It may be the case that multiple inputs are received at various times, which means the actual transition from the current state to another state cannot be known until the inputs are received (event driven).

There are two types of FSMs that could be used to describe a specific system. They are Moore and Mealy machines respectively named after their authors. In the Moore machine output depends on the current state only. Thus, it adds outputs to each state which is associated and when the machine transition happens into the state, the output corresponding to that state is produced. A Mealy Machine is an FSM whose output depends on the present state as well as the present input and, therefore, it has outputs that are a function of both state and input, leading to have less states but, on the other hand, more complexity. Based on this fact, this paper provides a system description following the Moore Machine model.

These behaviours can be described in graphical and tabular form. The first one, shown in Fig. 1, is a representation that uses as symbols circles and arrows that represent, respectively, the current state of the automaton and the transition from one state to another. Each transition is also described with the incoming input symbol that determines the passage of state. Within the tabular representation, the inputs are listed down on the left side, and the states are reported on the top. The table cell at the intersection of a particular row and column indicates the destination state of the FSM when the row's input is received when the machine is in the column's state.



State / Input	State 1	State 2	State 3	State ..	State n
Input 1					
Input ..					
Input n					

Fig. 1 Automaton behaviour representation, adapted from [7]

2.2 State of a Production Systems

By replicating finite state-machine approach, it is possible to model and simulate a production system by describing each component (machine, line, shop floor or application) as an automaton or an interconnected set of automata. Therefore, it is necessary to identify what states each system can be in, what inputs (or events) can trigger state transitions, and how the production system will behave for each state transition. In this model, the system behaviour is as a sequence of transitions that move the system through its various states [8]. From this, it is needed to identify several key characteristics of the system that can be modelled with a finite state machine. First of all, the system has a particular initial state; it should be possible to describe it by a finite set of states and it must have a finite set of inputs and/or events that can trigger transitions between states. Then, the behaviour of the system at a given point in time, considered as discrete, depends upon the current state and the input or event that occurs at that time. Finally, a system is triggered by external inputs or event, or it can be triggered or it can trigger another system.

A description of simple behaviour of such responsive system is described in Fig. 2 where 3 automata describing three subsystems interact with each other. The system 1 has one initial state q0, one input set Σ, and the output set F. Both parts are built upon a set E of interactions with the environment, called events. The input is a conjunction of events and it describes a condition generated by the environment to which the system reacts (external environment or another automaton). The events can be external or internal (to the global production system) depending on where they come from. During the initial phase, Automaton 1 is in $S_1 1$ state. Once an external event occurs, based on instructions associated to such event, it transitions executing specific actions from $S_1 1$ to a defined state $S_1 2$. The Automaton 1 state change is the

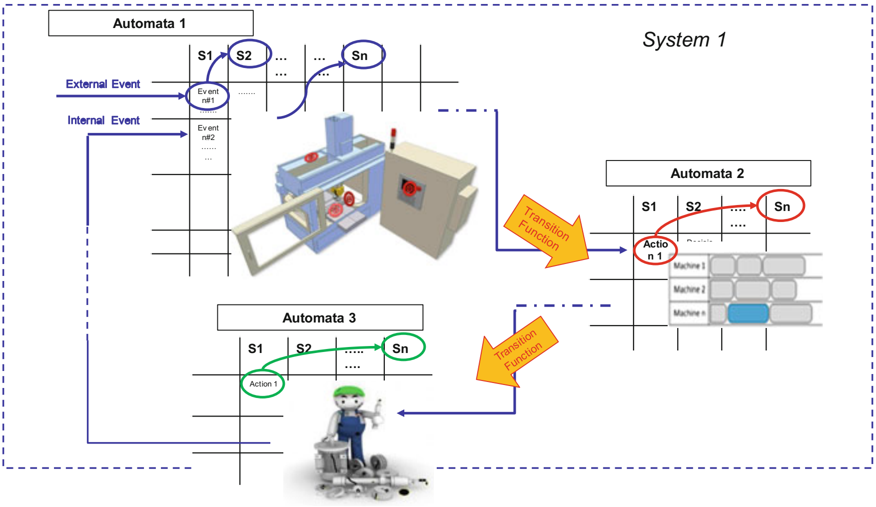


Fig. 2 Multiple automata system example

output of the transition function δ, which represents the Actions that have to be executed by the Automaton 1 and eventually cause its state change. One of the actions of Automaton 1 is the generation of an event that Automaton 2 receives as an input, determining its state change. For this reason, Automaton 2 transitions from state $S_2 1$ to a state defined as $S_2 n$, depending on the specific event that occurs. The same reasoning can be made for Automaton 3, which will react to event coming from Automaton 2 by changing its current state. This behaviour that is based on event-triggered system, ensures the ability to react to asynchronous external events which are not known in advance, allowing in many cases the adaptation to the actual demand without a redesign of the complete system [9]. Development and connection of multiple distinct automata allow the expansion and redefinition of the system configuration to describe the evolving physical production system.

3 KPI Measurements

The proposed model describes the whole set of state in which an automaton can be in the time domain. This ability is ensured as the automaton changes its state every time an event occurs and the model is able to track the temporal sequence of the occurred events and associated state transition. In this way, the model can provide information about the state history of the automaton. Hence, it is possible to estimate time-based performance indicators associated to each state. In fact, knowing that the time is considered discrete, it is possible to describe some machine parameters as an event the machine is subject to, constant over time intervals and memory less. Therefore, considering for example the overall time spent in failure and maintenance state of a specific machine, an estimation of MTTR (Main Time To Repair) indicator for that machine can be provided. Following the same reasoning, the MTBF (Main Time Between Failure) estimation can be obtained. Finally, knowing that machine availability depends on these values, its estimation can be evaluated too. In fact, calculating the time spent in not-productive state, it is possible to describe the real utilization rate of that machine. Therefore, once an appropriate state set is defined, it is possible to estimate different kinds of KPIs (i.e. Overall Equipment Effectiveness-OEE), starting from the analysis of an automaton state evolution in the time domain.

4 FSA Implementation for Industrial Environment on Computer

Implementation of FSA on a computer based system is a well-defined topic, for example in string parsing and regular expression matching. Adoption of FSA is also well known in monitoring and control of real-time systems, where the computer operating system is able to generate (asynchronous) inputs as reaction to external (asynchronous) events collected by I/O devices.

4.1 Architecture and Coding of a FSA

In order to implement a FSA for industrial purposes on a computer, the only requirement is that the operating system is able to manage asynchronous events; most of the programming languages can be adopted to code such application. The Core automaton is implemented by a process in Hibernate state that after a system and environment setup is on hold waiting to manage external inputs. A specific automaton can be instantiated many times, that is it can manage multiple instances of similar physical components (e.g. multiple machines of the same department) working in the same way, utilizing the same automaton, but evolving autonomously. It is required that each instance (each machine) is properly described by a dedicated set of data (a.k.a. Context). The Context contains all the specific information describing the history and the characteristic of the specific instance; it is usually implemented utilizing a static memory area. When an event is generated, the operating system is able to associate the event to the specific indicator of the instance (the physical component) it refers to (this is called Context Pointer). The description of the automaton is usually carried out in a matrix (as described in Fig. 1) where for each state in which the automaton can be, the actions to take for each possible event are described. The actions to take are described by ACTIONS, which are portions of code implementing the operations to execute. The ACTIONS are described by a syntax composed by the following keywords:

- ACTION (ID, Action_Name): The keyword launches the execution of the procedure Action_Name passing the parameter ID (associated to the event triggering the automaton) as argument to identify the instance of the automaton and pointing to the associate Context. The ACTION is a portion of code with the activities required, including the physical output such as printing, displaying or driving an actuator;
- NEXTSTATUS (ID): This keyword is the last of the instruction to be executed in order to manage the event and describes the next state in which the automaton switches;
- OUTPUT (AUTOMATON, ID, event): This keyword generates an event on the instance ID of the automaton.

4.2 Simulation and Monitoring

The utilization of the FSA is a powerful means for simulating the behaviour of the physical system, just assigning an arbitrary set of initial states and generating the desired sequence of events E(t). Moreover, the memorization of the sequence of automaton states in time allows monitoring the evolution and behaviour of the system, assuring the capability to assess and quantify KPIs (see Chap. 3). These abilities, performed by cyber components of CPS (i.e. performing computations, implementing algorithms) depend on variables that only change at the occurrence of discrete events, as such variables take values from discrete sets rather than from a continuum. Due to this unique features, the model of cyber components has typically discrete valued state variables, that are updated by discrete events just as described by FSA [10].

5 Industrial Machinery: Case Study

The industrial real case has been implemented in the EU-project PERFoRM [11] and it concerns the machinery sector. The real production environment is composed of a production line containing different machines, a common scheduling and a maintenance system. In order to model the production system, the real production environment has been simplified considering one block made of only three resources: one machine (lathe, see Fig. 2), the scheduling and the maintenance systems.

Thus, it is possible to model the overall production environment by replying this block every time is needed. These three resources are considered as three different automata, each of them having a finite number of states in which it can find itself, a finite number of possible events which can have an impact on these states, some transition functions and actions it could take. The lathe machine can have 6 different states (as depicted in Fig. 2) and can change from one to another if some external or internal event occurs. It is defined that only 2 external events and 1 internal event can change the state of the lathe machine.

The same automaton can describe multiple physical machines (use multiple Contexts). The work processed by this machine is defined by the scheduling system which has 2 possible states as shown in Fig. 3. It can monitor and control how the lathe machine is performing (S1) or it can schedule the production for the lathe machine (S2). Only 1 event can let its state change: the arrival of a signal from the lathe machine. When this event occurs, it reacts by communicating with the maintenance system, sending it a notification and connecting it to the lathe machine which sent the signal.

Also the maintenance system can find itself in 2 different states (Fig. 3): monitor, and control and execution. It changes from the monitoring state to the execution state when the scheduling system requires its intervention on the lathe machine. Once the maintenance execution has been completed, it turns back to the scheduling system sending it a notification. In this way, thanks to this communication, the scheduling system can provide a new schedule for the lathe machine, which will start working again (Fig. 4).

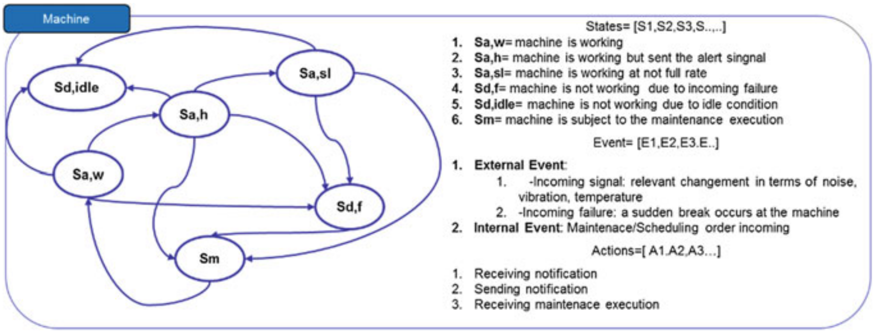


Fig. 3 Lathe machine state, events and actions

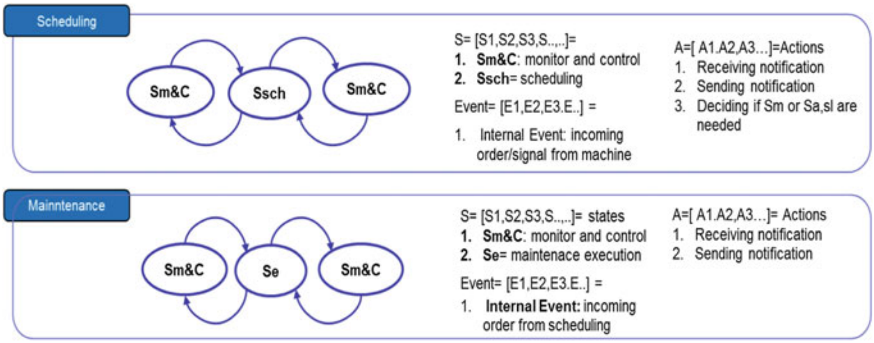


Fig. 4 Scheduling and maintenance system state, events and actions

In order to facilitate the understanding of the method, an example is further provided. Let's assume that at t = 0 the lathe machine is working and so it is in *Sa, w* state. At a certain time, smart sensors embedded on it send an alert to the machine, which leaves the *Sa,w* state and turns to the *Sa,h* one. At this point, the lathe machine sends a notification to the scheduling system in order to let it aware of what is happening in the production flow. The scheduling system, which goes from *Sm&c* state to *Ssch* one, can now decide what kind of actions are needed to determine the lathe machine to resume working. Scheduling can decide to let the machine resume working but at a low rate, by sending it an order and moving to the *Sa,sl* state, or it can decide to stop the production and send an alert to the maintenance system.

In the first case, if the machine has already sent an alert signal to the scheduling system but this system decided to let the machine continue working, at a certain time it can happen that the lathe machine can fail, moving itself from *Sa,sl* state to *Sf* state. Both in the second case mentioned above and in this last case, the scheduling system sends an alert to the maintenance system, which turns its state from *Sm&c* to Se. At the same time, the lathe machine changes its state in *Sm*. Once the maintenance is completed, it goes back to the scheduling system, which can now generate a new schedule for the repaired lathe machine. This narrative description of the automaton can be also described utilizing matrixes as described in Fig. 1b.

6 Conclusions

The proposed approach allows the description of the production system, regardless: its complexity, the number of each components and their actual configuration; the utilization of multiple automata synchronized between them allows expanding the domain we are describing both in terms of additional function and multiple instances of the same components.

Furthermore, it provides a formal modelling to analyse the interaction between the discrete and continuous parts of a cyber-physical system. In fact, as the functional correctness of a CPS is crucially dependent not only on the dynamics of the analogue physical environment, but also on the decisions taken by the discrete control that alters the dynamics of the environment, the FSM provides a finite set of real-valued variables whose dynamics in each state will be governed by a system of ordinary differential equations [4].

The flexibility of the approach associated with the embedded ability to manage asynchronous events makes it very easy to integrate interactions of external systems (e.g. planning, maintenance, logistics, etc.) with human operators. The definition of the admissible state and input event defines the boundary of the systems, and its representation by the state/input matrix allows an easy and understandable analysis and communication of the way the system behaves; this approach can represent the input for implementation of simulation systems on other platforms.

Acknowledgements This project has received funding from the European Union's Horizon 2020 research and innovation program under grant agreement No 680435.

References

1. Spath, D., Gerlach, S., Schlund, S.: Cyber-physical system for self-organised and flexible labour utilisation. In: 22nd International Conference on Production Research (2013)
2. Nyhuis, P., Duffie, N., Brieke, M.: Changeable manufacturing—classification, design and operation. CIRP Ann. Manuf. Technol. Ann. Technol. **56**(2), 783–809 (2007)
3. Maraninchi, F.: The argos language: graphical representation of automata and description of reactive systems. In: IEEE Workshop on Visual Languages, vol. 3 (1991)
4. Shankara Narayanan, K., Trivedi, A.: Hybrid Automata for Formal Modeling and Verification of Cyber-Physical Systems (2015). arXiv preprint arXiv:1503.04928
5. Wakerly, J.F.: Sequential Logic Design Principles, in Digital Design: Principles and Practices, pp. 431–558. Prentice Hall (2005)

6. Thirumaran, M.: Evaluating service business logic using finite state machine for dynamic service integration. Int. J. Comput. Appl. (0975–8887) **22**(7), 33–39 (2011)
7. Hopcroft, J., Ullman, J.: Finite Automata, in Introduction to Automata Theory Languages and Computation, S. Rivas, Ed. Pearson Education, pp. 37–81 (1979)
8. Wright, D.R.: Finite state machines. CSC215 Class Notes. Professor David R. Wright website. N. Carolina State University, pp. 1–28 (2012). Accessed 14 July 2012
9. Albert, A.: Comparison of Event-Triggered and Time-Triggered Concepts with Regard to Distributed Control Systems, in Embedded World, pp. 235–252 (2004)
10. Danda, I.S., Rawat, B., Rodrigues, J.: Analysis and Design of yber Physical System: a Hybrid Control Systems Approach, in Cyber Physical System: from Theory to Practice, pp. 3–31 (2015)
11. PERFoRM—Production harmonizEd Reconfiguration of Flexible Robots and Machinery (2016). http://www.horizon2020-perform.eu

Instantiating the PERFoRM System Architecture for Industrial Case Studies

Paulo Leitão, José Barbosa, Matthias Foehr, Ambra Calà, Pietro Perlo, Gregorio Iuzzolino, Pierluigi Petrali, Johan Vallhagen and Armando W. Colombo

Abstract The PERFoRM project, an innovation action promoted within the scope of the EU Horizon 2020 program, advocates the use of an Industrie 4.0 compliant system architecture for the seamless reconfiguration of robots and machinery. The system architecture re-uses the innovative results from previous successful R&D projects on distributed control systems domain, such as SOCRADES, IMC-AESOP,

P. Leitão · J. Barbosa (✉)
Polytechnic Institute of Bragança, Campus Sta Apolónia, 5300-253 Bragança, Portugal
e-mail: jbarbosa@ipb.pt

P. Leitão
e-mail: pleitao@ipb.pt

P. Leitão
LIACC – Artificial Intelligence and Computer Science Laboratory,
Rua Campo Alegre 1021, 4169-007 Porto, Portugal

M. Foehr · A. Calà
Siemens AG, Corporate Technology, Erlangen, Germany
e-mail: matthias.foehr@siemens.com

A. Calà
e-mail: ambra.cala.ext@siemens.com

P. Perlo · G. Iuzzolino
IFEVs – Interactive Fully Electrical Vehicles SRL, 12048 Sommariva del Bosco CN, Italy
e-mail: pietro.perlo@ifevs.com

G. Iuzzolino
e-mail: gregorio.iuzzolino@ifevs.com

P. Petrali
Whirlpool Europe, Cassinetta di Biandronno, Italy
e-mail: Pierluigi_Petrali@whirlpool.com

J. Vallhagen
GKN Aerospace Sweden AB, 46181 Trollhaettan, Sweden
e-mail: johan.vallhagen@gknaerospace.com

A.W. Colombo
University of Applied Sciences Emden/Leer, Emden, Germany
e-mail: awcolombo@technik-emden.de

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_32

GRACE and IDEAS. This paper, after describing the main pillars of the PERFoRM system architecture, focuses on mapping the system architecture into four industrial use cases aiming to validate the system architecture design before its deployment in the real environments.

Keywords Cyber-physical systems · Seamless re-configuration · Pluggability

1 Introduction

The Industrie 4.0 platform [7] provides a vision to modernize the manufacturing sector towards the smart factories of the future addressing the current requirements of product customization, quality and cost, as well as the rapid and flexible reaction of manufacturing processes to condition changes in terms of product variability and fluctuation and process disturbances.

Aligned with this vision, several distributed control architectures have been developed and proposed in the last years, some of them with European funding. From the literature survey, it is possible to name SOCRADES (Service-oriented cross-layer infrastructure for distributed smart embedded systems) [5], IMC-AESOP (ArchitecturE for Service-Oriented Process—Monitoring and Control) [4], GRACE (InteGration of pRocess and quAlity Control using multi-agEnt technology) [8], PRIME (Plug and produce intelligent multi-agent environment based on standard technology) [12], IDEAS (Instantly Deployable Evolvable Assembly Systems) [11] and ReBORN (Innovative Reuse of modular knowledge Based devices and technologies for Old, Renewed and New factories) [6] (a deep analysis of the results of these R&D projects can be found in [9]), that created the foundations and made the necessary proof-of-concept of these technologies and methods to face the described requirements.

Also aligned with the described challenges, the PERFoRM project was recently launched with the objective of the conceptual transformation of existing production systems towards plug&produce production systems aiming to achieve a flexible manufacturing environment based on the rapid and seamless reconfiguration of machinery and robots as response to operational or business events. PERFoRM will re-use the innovative results of these previous successful R&D projects and establishes an industry-oriented system architecture. The results of the PERFoRM project will be validated into four industrial use cases, which present different reconfigurability requirements and operational scenarios. This paper aims to map the general PERFoRM architecture to each one of these industrial use cases to validate the system architecture design before moving for its deployment in real environments.

The rest of the paper is organized as follows: Sect. 2 overviews the basics of the PERFoRM system architecture. Sections 3–6 describe the mapping of the generic system architecture into four industrial use cases, namely a large compressor producer, a micro-electrical vehicles producer, a home appliances producer and an aerospace components producer. Section 7 presents the compliance with Industrie 4.0, and finally, Sect. 8 rounds up the paper with the conclusions and points out the future work.

2 Generalized PERFoRM Architecture

PERFoRM introduces an innovative approach to handle the seamless production system reconfiguration, combining the plug-and-produce concept and the human role as a flexibility driver in future production systems. The architecture should be as open and generic as possible, allowing to cover as many production domains as possible, and should re-use the best results from the previous successful state-of-the-art R&D projects in the field, instead of developing a new architecture from scratch, which increases the possibility of its industrial adoption. From the existing architectures, PERFoRM derives and sets foundations from the IMC-AESOP [4] and in SOCRADES [5] projects, particularly the use of the Service Oriented Architecture (SOA) approach. In the GRACE project [8] is used for the integration of local and global perspectives and for the integration of quality control in the production process using a multi-agent system infra-structure and the ARUM (Adaptive Production Management) [10] with its integrated agent-based planning and scheduling systems for complex, small-lot products manufacturing, such as aircrafts and ships, which re interconnected using an Enterprise Service Bus (ESB). The ReBORN project [6] is also used as ground-base, particularly due to the developed software tools to retrieve and process maintenance relevant equipment status information. Additionally, the IDEAS [11] and PRIME [12] projects provided useful results related to evolvability and plug-and-produce perspectives.

As result, the system architecture is based on a network of hardware devices and software applications, addressing different levels of the ISA95 standard enterprise architecture, which exposes their functionalities as services following the SOA principles, and are interconnected in a transparent manner by using an industrial middleware, as illustrated in Fig. 1 [9].

The middleware is acting as a common interface between the diverse hardware devices (e.g., robotic cells and Programmable Logic Controllers (PLCs)) and software applications (e.g., MES and SCADA) presented at the PERFoRM ecosystem, ensuring a distributed, transparent, secure and reliable data flow between them. This interconnectivity allows achieving major technological objectives such as modularity, flexibility, pluggability and re-configurability on the shop floor, as well as business goals, such as the improvement of system's performance and efficiency. Additionally, an important innovation of this integration layer is its distributed and cloud approach, instead of the centralized ones that can be mostly found nowadays and can act as a single point of failure as well as a limitation for the system scalability.

An important architectural element to support the interconnectivity of heterogeneous hardware devices and software applications, and consequently supporting the integration of legacy systems, is the use of standard interfaces and technology adapters, enhancing the seamless interoperability and pluggability. In particular, these adapters are responsible to mask the legacy systems by exposing their functionalities according to the PERFoRM standard interfaces.

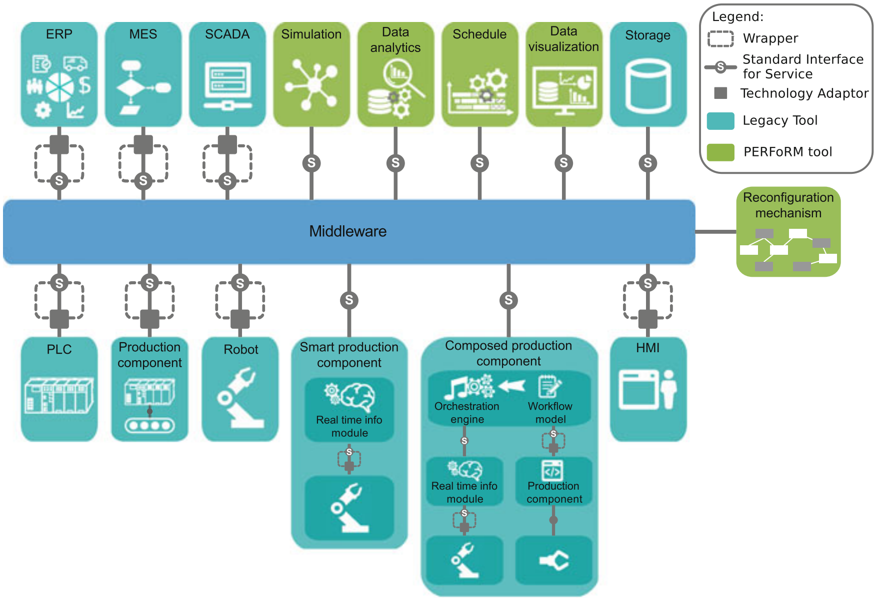


Fig. 1 PERFoRM system architecture

The human role, as in any system, is crucial by promoting a sustained decision making process. PERFoRM architecture has several interaction points with the human stakeholders, providing a bi-directional communication at different decision levels. Data analytics, scheduling and data visualization tools will have human dedicated graphical user interfaces to allow the insertion and display of relevant information. At a lower level, the PERFoRM architecture also allows the interaction with the human by means of Human Machine Interfaces (HMI) platforms, e.g. using the machine dedicated interfaces by empowering the human with mobile devices.

The proposed system also integrates advanced tools to enable the system operationality, namely scheduling, simulation and intelligent decision support, some of them using Multi-agent systems (MAS) technology. MAS [13] is a suitable approach to provide flexibility, robustness and responsiveness by decentralizing the control over distributed, autonomous and cooperative intelligent control nodes. Despite these important benefits, the real time constraints and the emergent behaviour in industrial environments can be pointed out as weaknesses. However, the MAS tools developed in PERFoRM don't face soft or hard real-time restrictions since these tools are placed at strategic and/or tactical planning and control levels without real time restrictions. Additionally, the emergent behaviour should be seen as a potential benefit and not as a problem, since boundaries can be used to ensure stability during the emergency process. In fact, several commercial planning and scheduling solutions are already operating in big companies (see for example the MAS solutions developed by Smart Solutions [1], as well as the Gartner's Strategic Technology Trends for 2016 report that predicts the use of MAS technology as a base for numerous mobile applications by 2020 [2]).

The PERFoRM architecture will be fully developed and instantiated into four industrial use cases, presenting different and specific requirements and covering a wide spectrum of production domains, namely a highly specialized large compressor production facility, a highly customizable producer of small-size electric vehicles, passing by a producer of home appliances and a producer of components to the aerospace industry. The mapping of the generic architecture into these use cases is described in the following sections, aiming to structurally validate the designed system architecture and also identify missing links and misunderstandings during the design phase.

3 Architectural Mapping for a Large Compressor Producer

3.1 Use Case Description

The first use case is related to a factory producing industrial compressors and gas separators. It is characterized by highly complex systems of several tens of thousand components which are typically only produced ones on a customer specific basis. At the same time, single parts can be very heavy and big, requiring special machining stations. As these stations are typically quite expensive, they cannot be set up multiple times within a factory. Together with machining times of several days up to 2 weeks, this produces critical delays and costs in case of machine failures and breakdowns. Currently, maintenance activities are done only reactively and in separated IT systems, namely the: a) maintenance scheduling and b) failure reporting. Thus, maintenance tasks are sometimes recognized too late and cannot be scheduled accordingly.

The objective of this use case process is to integrate the separate systems supporting the early identification of disturbances in production and the delivering to all involved stakeholders (e.g. maintenance, operation, scheduling and logistics) as much information as possible. Basically, three different scenarios can be distinguished when detecting early the disturbances: i) the machine can still operate (maybe with limited capabilities) and a future maintenance should be planned, ii) the machine cannot operate and a repair can be done right away, and iii) the machine cannot operate and maintenance will need to be carried out as soon as all needed material and resources are available. In the first two cases, the work does not need to be rescheduled. This integration, as well as the introduction of a proactive maintenance system, ensures a faster elimination of disturbances, and a reduction of delays in production and machine downtimes.

3.2 Architectural Mapping

Considering the particularities of the described use case, the general system architecture is mapped as illustrated in Fig. 2, where the generic blocks have been replaced by the legacy hardware devices and software applications installed at the plant or to be developed throughout the project. The maintenance tools and machines are the source of failure reporting and machine condition monitoring being their information stored in the Order Equipment Efficiency (OEE) database for further elaboration. Additionally, human operators can open up maintenance tickets, informing the maintenance staff about disturbances on the shop floor within the maintenance database.

Since they are legacy systems, proper technological adapters are required to transform their proprietary interfaces in the standard interfaces defined by PERFoRM. In addition, several new tools are considered to provide advanced features related to the analysis of the data gathered from the existing systems and now integrated. In particular, the data analysis tool aims at analyzing the collected data and at identifying in advance possible disturbances, generating warnings for the maintenance task list. The simulation tool aims at creating several *What-If?* scenarios which can be compared by KPI's, allowing the selection of the best maintenance schedule.

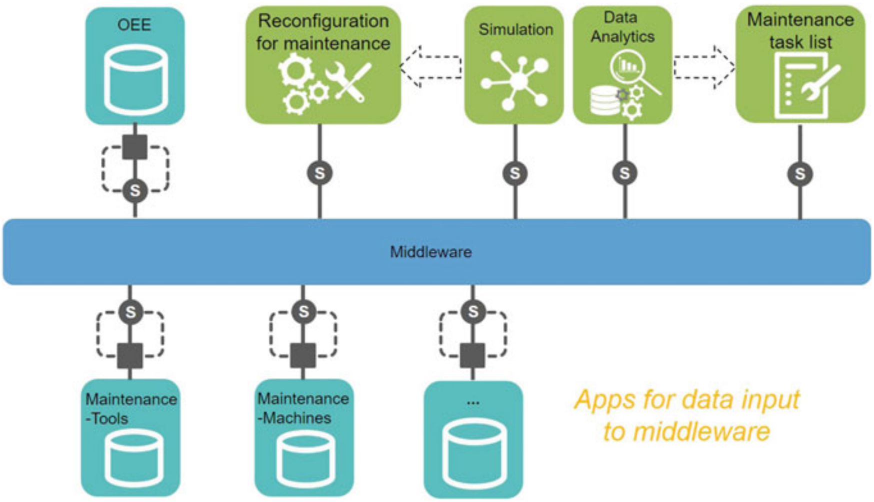


Fig. 2 Architecture mapping for the siemens use case

4 Architectural Mapping for a Micro-Electrical Vehicles Producer

4.1 Use Case Description

The second use case considers a factory plant dedicated to produce micro-electrical vehicles. At the moment, the production line is actually operated completely manually with a welding operator in each island with also multi- skilled competences. The line is being automatized to support the production's efficiency and to permit the necessary flexibility for the production of different type of vehicle configurations (i.e. the easy switch from one vehicle configuration to another one).

This use case aims to enable a high quality production line for micro-electric vehicles, despite the throughput. For this purpose, the seamless integration of modular stations (each one composed by welding robots and Programmable Logic Controllers (PLCs)) is crucial to achieve flexibility and reconfiguration in the production system in order to allow the production of low amounts of micro-cars in an economical manner.

4.2 Architectural Mapping

Considering the particularities of the described use case, the general system architecture is mapped as illustrated in Fig. 3, where generic blocks were updated by the legacy hardware devices ad software applications covered by the use case.

Several welding robotic cells, as well as the powertrain testing station (rolling bench test to check the functionalities of the motorized axle frame) and the chassis testing station (geometrical test of the chassis to check if all the assembly complies with the design), are interconnected to the MES system through the industrial middleware. In addition, agent-based simulation and dynamic scheduling tools are considered to increase the system performance and flexibility. These hardware devices and particularly the software application can access the data stored in the database also by using the middleware. The integration of these hardware equipment and software applications will require the use of proper technological adapters to transform the native data format into the data model defined by PERFoRM.

HMIs are related to the KPIs monitoring and visualization, and also used as graphical interface between the operator and the system for the production's traceability.

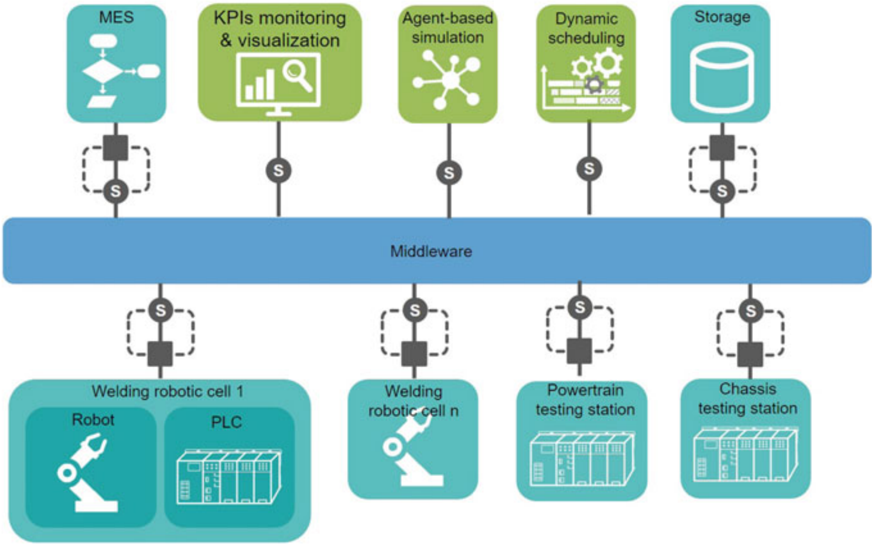


Fig. 3 Architecture mapping for the IFEVs use case

5 Architectural Mapping for a Microwave Ovens Producer

5.1 Use Case Description

The third use case is related to a factory plant dedicated to produce microwave ovens. The current factory continuous adaptation and medium term reconfiguration mechanism is based on a set of processes (Factory Master Plan, Profit Plan, Cost Deployment) which aim at improving KPIs through modification of factory assets and organization described by key business factors. Of course KPI are mainly driven by shop floor data of each single facilities and departments. The behaviour of these facilities is monitored in order to meet middle- and long-term goals to benefit. Currently, the data gathered at shop-floor level lack uniformity (different formats and source of data) and, moreover, correlation (i.e. each data is treated and analyzed without a model or a tool able to describe how each KPI is linked with others KPI or input factors or KBF).

The use case aims at developing a KPI visualization system to help the reconfiguration activities described above and ensuring a more data driven decision process. For this purpose, the data collected from the shop floor should be analyzed and correlated to extract in advance the KPIs and also to detect earlier possible disturbances or performance degradation. This analysis has to be complemented with simulation enabling a prediction mechanism to allow virtual reconfiguration before the actual one.

5.2 Architectural Mapping

Considering the particularities of the described use case, the general system architecture is mapped as illustrated in Fig. 4, where generic blocks were updated by the legacy hardware devices and software applications covered by the use case.

As shown, the data acquired from the shop floor and currently collected by several databases will be integrated in the PERFoRM ecosystem by the middleware and in a PERFoRM database. The integration of these legacy databases will require the use of proper technological adapters to transform the native data format into the data model defined by PERFoRM. Several new tools are considered to provide the required advanced features, namely: the monitoring and visualization system to support the on-line visualization of KPIs, the KPIs optimization tool that uses two different models (MPFQ-K model and Value Stream Map) to identify strategies to improve KPIs and a simulation tool having what-if game functionality to support the analysis of the impact of several degrees of freedom in these KPIs.

The human-machine interaction is mainly reserved for key decision-makers (e.g., production and industrial engineering managers) that will use the monitoring and visualization tool, as well as the proper designed user interfaces for the simulation tool, to understand the current system performance and study how KPIs can be optimized.

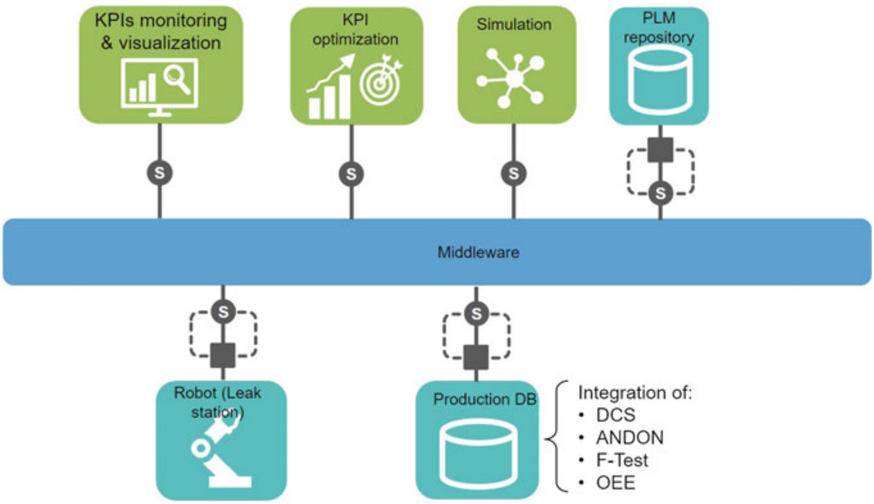


Fig. 4 Architecture mapping for the whirlpool use case

6 Architectural Mapping for an Aerospace Components Producer

6.1 Use Case Description

The fourth use case considers a factory plant that manufactures complex, high value jet engine components with very stringent quality characteristics. The production system extends a functional workshop with standalone work centres and a mix of dedicated and common resources. The level of automation is usually rather low and based on separate process automation cells with low level of process flow integration. The production system has to cope with a large variety of different components, low volumes and varying demands. All data (master data) is stored and managed through the SAP ERP system, which provides the functions for production planning, scheduling and MRP and collects different kind of documentation from the processing and inspection in order to guarantee traceability. Some additional data and information is generated and made available in the PLM system (Team Center Engineering and Manufacturing). The information flow in the typical day-to-day process is a straight communication between different IT systems of CNC-machines, robots and other equipment whereas the near term production planning and scheduling is done by shop floor planners based on ERP data for long term order scheduling and customer demands.

The main objective is the improvement of the flexibility to demonstrate more agile and automated production using an integrated system that can complete a short sequence of common operations in the value adding process chain. The approach aims to develop a modular production cell concept that can be reconfigured with different automated or semi-automated processes. The cell and process modules should be easily and quickly changed depending on current production demands aligned with the ideas of plug-and-produce for cyber-physical systems. The business oriented criteria are to reduce lead times and increase the level of automation as well as equipment utilization. For this purpose, mechanisms for the seamless reconfiguration of the production process should be addressed by pluging-in/-out modular processes in robotic stations.

Several technical challenges need to be addressed, namely interfaces for process modules that allow simple and short change over time, methods for production cell planning and scheduling to maximize throughput, and decision support to identify when to reconfigure the cell to have the best impact on cost and production lead time.

6.2 Architectural Mapping

Considering the particularities of the described use case, the general system architecture is mapped as illustrated in Fig. 5.

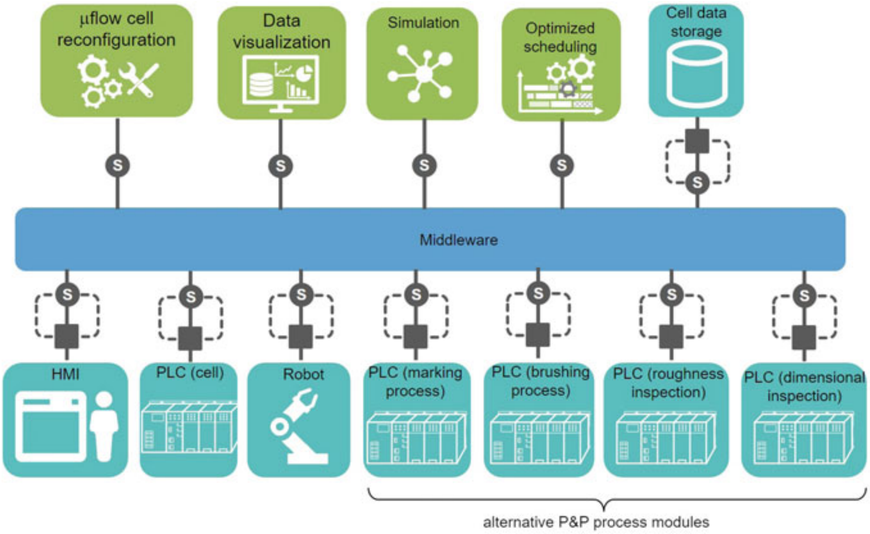


Fig. 5 Architecture mapping for the GKN use case

The modular micro-flow-cell? concept has a base configuration and components to host, control and coordinate the production on the process modules. The cell functions are a PC, PLC for communication and control, a robot system for part handling and processing and the safety system for the cell. Each of the different process modules, that can be replaced with short lead time "plug and produce" have their own PLC running the local control system.

All process related code/process parameters are downloaded from a central database, through the industrial middleware, to guarantee the full control of the configuration and versions of programs. In a similar way, the data generated from the executed processes and/or inspections are uploaded to the ERP system or databases and related systems for analysis and visualization (e.g., OEE/Stop time analysis and statistical process control). The production planning and scheduling is planned to be supported by the optimized scheduling and simulation tools that will ensure the short-term and long-term scheduling, well will trigger the need for use of the cell flexibility and reconfigurability, i.e. when to make the change-over in the cell.

The human interaction will be performed in two manners: i) at strategic level, a data visualization tool will provide to the cell manager a plethora of information related to the current status of the cell, namely current operating processes and tools, as well as KPIs, and ii) at production cell level, a HMI supports the operator during its tasks.

7 Alignment with Industrie 4.0 Platform

The concepts currently being developed within the PERFoRM project are aligned with the current state-of-the-art and road-map trends. Several matches can be devised from "regulatory" documentation [3, 7, 14].

From [7] several scattered key concepts can be seen. CPS are at the cornerstone of the Industrie 4.0, and PERFoRM addresses this by setting its foundation in the CPS concepts, particularly by promoting the symbiotic use of "digital" and "physical" layers of the manufacturing world and also considering its interconnection and interoperability. Optimized decision-making is also refereed in [7] and PERFoRM is addressing this by promoting a set of different tools that will allow the decision-makers, and particularly each of the use cases, to early detect deviations and performance degradation, allowing to take better, more accurate and timely decisions. Integration, either vertical and/or horizontal, is also mentioned to be crucial. PERFoRM also addresses this by promoting the use of a common PERFoRM data model, covering data needs from lower levels into higher levels as also from different domains within the same level (e.g., considering different data needs of devices) and by considering a distributed and interoperable middleware. Finally, and considering only a few key concepts, the "Industrie 4.0 working group" also recommends the development of a reference architecture. This is currently being developed by the Industrie 4.0 Platform, named "Reference Architectural Model Industrie 4.0 (RAMI 4.0)" [14], but from the initial developments, PERFoRM is aligned with what is being considered in RAMI 4.0.

Aligned with what was aforementioned, the International Electrotechnical Commission white paper on Factories of the future also makes several remarks and recommendations [3]. The "connectivity and interoperability" is covered through the development of a distributed and interoperable middleware alongside with the design of a common and cross-layers data model. The *"seamless factory of the future system integration"* is accomplished by the connection of several information data sources as also the consideration of the human as a valuable data source itself. The *"integration of existent systems"* is also managed by the development of hardware and software adapters, adapting the native information language into the PERFoRM ecosystem. Modelling and simulation is also envisioned as crucial building blocks of future systems. Therefore, the PERFoRM architecture considers the use of such tool domains, particularly allowing beforehand to foresee future problems and solutions to these and to allow the optimization of production processes. Other key concepts are located around the human operator and to its role in future production systems. PERFoRM considers this by moving the human to the centre of the architecture and by considering him as a flexibility driver in future systems. Therefore, a special emphasis is being devoted to the study of its integration and interaction to/from the system.

One of the most critical aspects being pointed out by all the reference documents is the use of standards and the promotion of the standardization process. The PERFoRM consortium also considers this as a major road-blocker breaker and enabler for the architecture future adoption and is also promoting this topic by promoting the use of standardized approaches and technologies, e.g., using OPC-UA or AutomationML data models.

Finally, and as also stated in [7], *"The journey towards Industrie 4.0 will be an evolutionary process."*, and also re-inforced in [3], the migration process from legacy systems is also crucial, particularly in the future adoption of such innovative systems. Therefore, the PERFoRM architecture is also complemented and accompanied with a set of migration guidelines that allow the successful deployment into both legacy and new production systems.

8 Conclusions and Future Work

This paper briefly introduces the generic modular system architecture developed under the H2020 R&D PERFoRM project, covering all the different layers in the production process identified by the ISA-95 automation model, being able to respond in a promptly manner to nowadays requirements as aligned with state-of-the-art visions, such as those advocated by the industry 4.0 initiative. The proposed system architecture doesn't set aside legacy equipment and tools allowing the majority of nowadays factories to migrate. This process is accomplished by the use of a common, cross-platform, data model and through the use of technological adapters.

In particular, the paper describes in general terms the instantiation of the generic system architecture into four project's industrial use cases, which allowed to test and validate the PERFoRM system architecture in its full spectrum of applications, ranging from different time constraints (from machinery to backbone levels) and also covering the horizontal (between machine to machine and software application to software application) and vertical integration (from automation hardware machines to planning and simulation software applications).

Future work will be devoted to the implementation, testing and validation of the instantiated use case architectures. Additionally, and in order to promote an effective architectural deployment, future work will be devoted to the study of migration strategies, converting the current existing production systems into PERFoRM compliant solutions. Particularly, the industry best practices in migration strategies within the several domains of ISA-95 are considered and aggregated.

Acknowledgements

This project has received funding from the European Unions Horizon 2020 research and innovation programme under grant agreement No 680435.

References

1. Smart Solutions. http://smartsolutions-123.ru/en/
2. Gartner Identifies the Top 10 Strategic Technology Trends for 2016 (October 2015). http://www.gartner.com/newsroom/id/3143521/
3. White Paper: Factory of the future. International Electrotechnical Commission. www.iec.ch (2015)
4. Colombo, A.W., Bangemann, T., Karnouskos, S., Delsing, J., Stluka, P., Harrison, R., Jammes, F., Martínez Lastra, J.L. (eds.): Industrial Cloud-based Cyber-Physical Systems: The IMC-AESOP Approach. Springer (2014)
5. Colombo, A., Karnouskos, S.: Towards the Factory of the Future: a Service-oriented Cross-layer Infrastructure. ICT Shaping the World: a Scientific View, European Telecommunications Standards Institute (ETSI), pp. 65–81 (2009)
6. Ferreira, P., Doltsinis, S., Lohse, N.: Symbiotic Assembly Systems—A New Paradigm. Procedia CIRP, pp. 26–31 (2014)
7. H. Kagermann and W. Wahlster and J. Helbig: Securing the future of German manufacturing industry: Recommendations for implementing the strategic initiative INDUSTRIE 4.0. Technical report, ACATECH—German National Academy of Science and Engineering (2013)
8. Leitão, P., Rodrigues, N., Turrin, C., Pagani, A.: Multi-agent system integrating process and quality control in a factory producing laundry washing machines. In: IEEE Transactions on Industrial Informatics, pp. 879–886 (2015)
9. Leito, P., Barbosa, J., Pereira, A., Barata, J., Colombo, A.W.: Specification of the PERFoRM architecture for the seamless production system reconfiguration. In: Proceedings of the 42nd Annual Conference of the IEEE Industrial Electronics Society (IECON'16) (2016)
10. Marn, C.A., Mnch, L., Leito, P., Vrba, P., Kazanskaia, D., Chepegin, V., Liu, L., Mehandjiev, N.: A conceptual architecture based on intelligent services for manufacturing support systems. In: Proceedings of the 2013 IEEE International Conference on Systems, Man, and Cybernetics, SMC, pp. 4749–4754 (2013)
11. Onori, M., Lohse, N., Barata, J., Hanisch, C.: The IDEAS project: plug & produce at shop-floor level. Assem. Autom. 124–134 (2013)
12. Rocha, A., Orio, G.D., Barata, J., Antzoulatos, N., Castro, E., Scrimieri, D., Ratchev, S., Ribeiro, L.: An agent based framework to support plug and produce. In: Proceedings of the 12th IEEE International Conference on Industrial Informatics (INDIN'14), pp. 504–510 (2014)
13. Wooldridge, M.: Introduction to Multi-Agent Systems. Wiley (2002)
14. ZVEI: Industrie 4.0: The Reference Architectural Model Industrie 4.0 (RAMI 4.0). http://www.zvei.org/Downloads/Automation/ZVEI-Industrie-40-RAMI-40-English.pdf

A Highly Flexible, Distributed Data Analysis Framework for Industry 4.0 Manufacturing Systems

Ricardo Silva Peres, Andre Dionisio Rocha, Andre Coelho and Jose Barata Oliveira

Abstract In modern manufacturing, high volumes of data are constantly being generated by the manufacturing processes. However, only a small percentage is actually used in a meaningful way. As part of the H2020 PERFoRM project, which follows the Industry 4.0 vision and targets the seamless reconfiguration of robots and machinery, this paper proposes a framework for the implementation of a highly flexible, pluggable and distributed data acquisition and analysis system, which can be used for both supporting run-time decision making and triggering self-adjustment methods, allowing corrections to be made before failures actually occur, therefore reducing the impact of such events in production.

Keywords Data analysis · Industry 4.0 · Manufacturing · Cyber-physical systems

1 Introduction

1.1 Interconnected Manufacturing Systems—Industry 4.0

Nowadays, the world is facing a huge and disruptive paradigm shift based on the IT developments of the last few decades. Several concepts such as Cloud Computing, Internet of Things, Big Data, etc. are creating one of the most challenging and drastic changes in the manufacturing world. The Industry 4.0 [7, 8] concept emerged as an

R.S. Peres (✉) · A.D. Rocha · A. Coelho · J. Barata Oliveira
CTS - UNINOVA, Faculdade de Ciencias e Tecnologia,
Universidade Nova de Lisboa, 2829-516 Caparica, Portugal
e-mail: ricardo.peres@uninova.pt

A.D. Rocha
e-mail: andre.rocha@uninova.pt

A. Coelho
e-mail: am.coelho@campus.fct.unl.pt

J. Barata Oliveira
e-mail: jab@uninova.pt

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_33

idea for the perfect production environment based on the modern interconnected world.

More than the connectivity among the shop floor components and the diverse departments of each company, in Industry 4.0 all of the value chain activities as well as the entire product life cycle intervenients are connected and are capable of producing and sharing information among themselves [16].

Aligned with this vision, the PERFoRM project targets the conceptual transformation of existing industrial production systems towards the plug and produce paradigm, achieving flexible manufacturing environments based on full interoperability and seamless reconfiguration of harmonized machinery and robots as a response to operational or business variations.

Currently, a large amount of manufacturing information is already generated, however most of it is just stored and not processed in order to extract useful knowledge. Hence, with the recent advent of Data Mining it is possible to analyze this data and somehow generate relevant information for the production environment [12, 18].

As part of the PERFoRM project, this paper proposes a framework for the implementation of a data acquisition and processing system, capable of coping with the requirements imposed by the Industry 4.0 vision. The remainder of this paper is organized as follows. Section 1.2 provides an overview of related work on the topic of data analysis in manufacturing. Afterwards, Sect. 2 describes the proposed architecture, followed by Sect. 3 which provides the guidelines for a possible implementation of said architecture. Finally, some closing remarks and details regarding future work are provided in Sect. 4.

1.2 Data Analysis in Manufacturing

Data Analysis can be used to obtain meaningful knowledge or information from raw data, applying algorithms in order to achieve it. Typically, the concept of data mining is often discussed and used due to this purpose.

In a manufacturing context, several approaches and algorithms are used to gather knowledge. Reference [4] reports about the development of an environment providing knowledge based on search, evaluation and generalization, which addresses a clear challenge of suggesting good operating strategies for specific factory conditions at the proper time. Neural Networks (NN) and Genetic Algorithms (GA) were used to identify the data structure, whereas the data extracted was in form of performance obtained. Finally, this type of knowledge can be used to increase the system accuracy. IGem was developed by [10] and is an artificial intelligent tool that uses fuzzy logic, GA algorithms and rule base knowledge which is applied to the diamond industry. The results presented demonstrated the reduction of processing time by 25% when analyzing data. In maintenance, EXPERT-MM was developed to work with historical failure data in order to suggest preventive maintenance schedules [1].

Regarding knowledge extraction, algorithms are applied based on what kind of knowledge is aimed to retrieve. Reference [5] makes an extensive review on data mining functions and applicability, which aims to identify several kinds of knowledge to be mined such as job shop scheduling, quality control, fault diagnostics, manufacturing process, maintenance, defect analysis, manufacturing systems, condition based monitoring, supply chain and others. For most of them, one could apply decision tree algorithms, regression functions, fuzzy clustering techniques, entropy based analysis methods, neural networks, genetic algorithms, sampling methods, among others.

One interesting point to rethink is suggested by [6] which refers to how data is being organized during processing. As clearly stated, the processing model and the data model are often disjoint areas resulting in distributed computing systems to focus only on ordering tasks instead of including abstract models for data processing. This type of models act as a schema for data processing which logically optimizes the process.

The previously discussed techniques can be described as an aggregation of complex algorithms and methods that have been object of studies and improvements. However, [14] presents a different solution for the problem of data extraction and processing. In this case one specific type of raw data (data related to transitions) is processed to assert if a device needs maintenance based on the computation of moving averages and trends of key values. These types of methods are simpler but, nonetheless, provide recognition of abnormal behaviours or irreversible problems with the resources, which is a valuable asset for manufacturers. The example used was a clamp opening or closing time, providing real time information of important resources within a manufacturing process.

2 Layered Data Analysis Architecture

In this section an overview of the proposed real-time data analysis architecture is provided, along with its main goals, requirements and a description of its individual structural elements.

Being a critical part of the PERFoRM ecosystem, the proposed solution is responsible for not only performing the context-aware data analysis, thus generating predictive data that can be used to trigger the system's self-adjustment mechanisms (e.g. reconfiguration), but also for the acquisition of the data itself at both the manufacturing cell and component levels.

Additionally, a given number of requirements are imposed on the architecture's design. First and foremost, in line with PERFoRM's vision the architecture should be generic enough to be applicable to various different scenarios, being open so as to not depend on the existence of a single communication protocol or standard on the shop floor, thus facilitating its industrial integration and adoption. Moreover, it needs to be capable of adapting to changes to the process or its components at run-time, for instance in terms of both pluggability and changes to the Key Performance Indicators (KPI) to be analyzed. Furthermore, data and context representation should follow PERFoRM's common data model in order to enable the seamless interoperability and

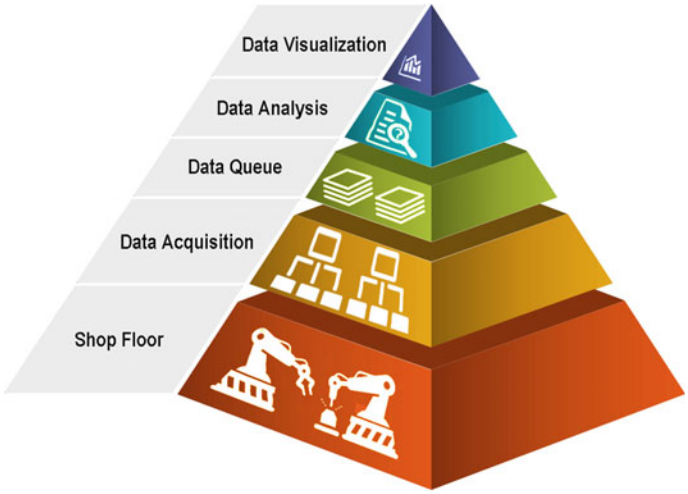


Fig. 1 Architecture layered view

data exchange between the data analysis architecture and the remaining PERFoRM system elements and tools.

Another point to take into account is the aspect of scalability. In order to ensure that the approach is applicable to a varied number of different use cases, it needs to be capable of scaling according to each use case requirements. However, as a system scales its complexity tends to increase to higher levels as a consequence. Thus, in order to tackle this challenge, a layered architectural structure is proposed. An overview of this approach can be seen in Fig. 1.

As depicted, the proposed architecture is divided into several layers in order to decrease the overall complexity, each operating according to a specific purpose on top of the shop floor, which stands as the base layer. Each of the subsequent layers is described in further detail in the remaining subsections of Sect. 2.

2.1 Data Acquisition Layer (DAL)

Standing directly above the shop floor layer, the DAL is responsible not only for the acquisition of relevant data but also by its pre-processing in terms of the extraction of context-aware information.

In regards to the data acquisition, the DAL needs to be flexible in order to adapt to changes coming directly from its sources in the shop floor, be it in terms of new components being plugged or unplugged, or even changes to the KPIs that need to be collected and analyzed. Also, the communication with the shop floor needs to be specified in a generic way, thus allowing the consideration of different requirements from different potential use case. For instance, a specific case might present time constraints in the order of weeks or days, while a different one might require data to be collected and analyzed in near real-time, therefore requiring different approaches.

To this end, the DAL follows an approach similar to that presented in another successful European project, FP7 PRIME [13–15], in which a Cyber-Physical System (CPS) based approach was used. This approach is centred on a Multi-agent System (MAS) architecture which abstracts both components and subsystems (e.g. cells, workstations) alike. The adoption of MAS technology confers additional flexibility and robustness to the DAL, allowing it to quickly adapt to changes in the shopfloor.

No less important is the existence of generic communication interfaces which allow the agents to interact with the environment in a "black-box" fashion, regardless of the underlying technology or communication standard. In PERFoRM's case, this means that the approach can be implemented in a way that the agents can communicate with the hardware via the harmonization middleware, or if required (e.g. specific time constraints), a different instantiation of these interfaces would allow an approach closer to edge computing. Upon collecting the raw data, the agents can pre-process it in order to extract more meaningful information before passing it on to the upper layers, in this case the Data Queue Layer (DQL), which is described in further detail in Sect. 2.2.

2.2 Data Queue Layer (DQL)

The DQL's main purpose is to serve as a distributed continuous buffer for the data coming from the DAL. It should add another layer of robustness, allowing for high-volume streams of data to be transported from the DAL in order to be consumed by the data analysis network. As such, it should provide reliability in terms of message delivery, which can be achieved through the sequencing and replication of data messages.

More than a simple message queue, the DQL should be capable of not only handling a high throughput of data (in order for it to cope with the aforementioned varied time constraints), but also to enrich and filter or aggregate the buffered data as required in order to facilitate its consumption by the Data Processing Layer (DPL).

2.3 Data Processing Layer (DPL)

The last core layer is the DPL, responsible for the actual data analysis of the inputs coming from the lower layers. In the context of PERFoRM, this analysis is meant to generate predictive data related to the KPIs relevant for each use case, producing forecasts and identifying trends and correlations between these indicators.

As such, this layer enables the early detection of possible disturbances, degradation or KPI deviation from the expected boundaries in the shop floor. Hence, due to this capacity for predictive analysis, the DPL is a key-enabler of condition-based maintenance, allowing manufacturers to schedule maintenance operations before a failure actually occurs, thus diminishing the direct impact on production. Additionally, the DPL is not limited to assisting in run-time decision making (e.g. by interfacing with external data visualization tools, which are however outside the scope of this work), being also capable of triggering self-adjustment methods (e.g. self-reconfiguration) which can promptly perform corrections in order to return the system to a state of normal operation.

3 Data Analysis Framework

This section aims at providing the guidelines for a possible implementation of the architecture described in Sect. 2. Each of the layers and related technologies is addressed in the coming subsections, namely the MAS-based CPS (DAL), the Apache Kafka data queue (DQL) and the Apache Storm stream processing network (DPL), detailed in Sects. 3.1 and 3.2, respectively.

3.1 MAS-Based Cyber-Physical System in the DAL

Following the example set in [14, 15], the DAL's CPS can be implemented following a similar pluggable MAS-based approach. The Java Agent DEvelopment framework (JADE) [3] is indicated as it provides a robust infrastructure supporting the agent's core behavioural logic and communication, as well a wide array of auxiliary tools to further facilitate the development process. A representation of the comprised agents, as well as their respective interactions is shown in Fig. 2.

The DAL implementation consists in a MAS network comprising two main different agent types. The Component Monitoring Agent (CMA) is responsible for abstracting individual components (e.g. clamp, robot) in the shop floor, collecting relevant raw data and pre-processing it according to a given set of rules. The data acquisition is performed through a generic Data Collection Interface (DCI), thus allowing this communication to be executed both through PERFoRM's middleware, or directly through the implementation of required communication protocols (e.g. instantiating an OPC UA connector), depending only on the DCI implementation.

Each CMA can then be associated to a Subsystem Monitoring Agent (SMA), to which it passes on its extracted and pre-process data, enabling the extraction of more complex information at a higher abstraction level. Hence, the SMA performs similarly to the CMA, being however responsible for abstracting a subsystem (e.g. robotic cell, workstation), which can in turn be associated to other SMAs. The agents' interactions are FIPA compliant, following the FIPA Request protocol [2].

Both agent types relay their data to the upper layers through a generic Output Communication Interface (OCI), allowing the CPS to be independent from the technology used to implement the remaining layers.

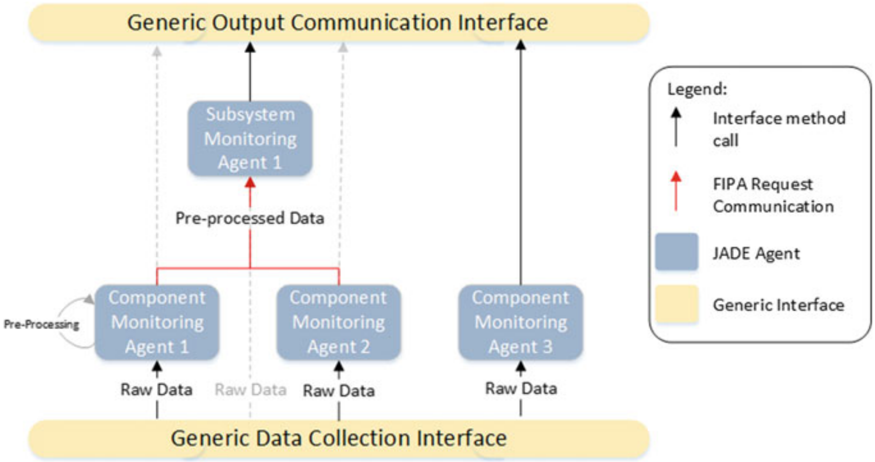


Fig. 2 JADE data acquisition and pre-processing MAS overview

3.2 Data Message Queue and Processing Network

Any implementation of the DQL needs to take into consideration the requirements specified in Sect. 2.2, more specifically in terms of scalability, capacity to handle high volumes of data, low latency and reliability.

With this in mind, Apache Kafka [11, 17] is proposed as a possible framework to implement such a data queue. Kafka is a fault-tolerant, highly scalable, distributed messaging system. In essence, Kafka functions with a publish-subscribe approach, allowing *producers* (data sources, in this case the agents in the DAL) to publish data messages which are maintained in categories called *topics*. These can be subscribed by *consumers* (represented by the DPL nodes), being divided into ordered partitions supporting message persistence and replication. Kafka's message management is optimized for low latency and high throughput, with documented uses for even real-time applications [9].

Finally, for the DPL Apache Storm is considered, being a distributed stream processing system which easily integrates with both databases and queuing technologies (such as the aforementioned Apache Kafka). Storm's processing runs in *topologies*, which are essentially series of nodes each containing certain processing logic, with the associated links specifying the data flow. The framework integrating each of these technologies can be seen in Fig. 3.

With the whole framework integrated, data is collected by the CPS via the DCI, being pre-processed at both the component and subsystems levels. Afterwards, each agent publishes its respective data to the Kafka queue, where it is aggregated in topics according to its specific category (e.g. component origin or data group). Finally, data is continuously consumed by the Storm topology, which can in turn compute the trends and correlations necessary in order to generate meaningful predictive data

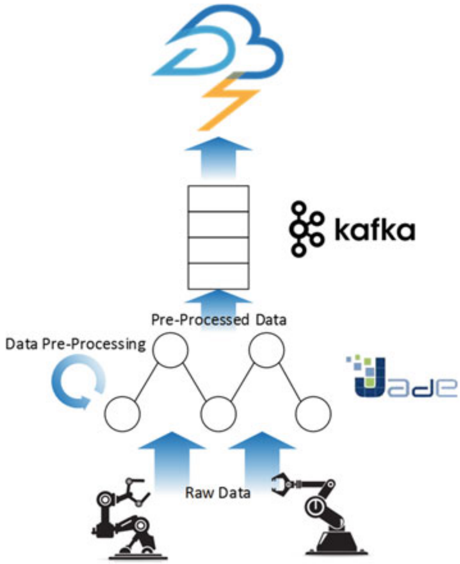


Fig. 3 Framework overview

which can be used for run-time decision making support or as a trigger for self-adjustment methods.

4 Conclusions and Future Work

This paper presents the framework of a highly flexible and distributed system for the acquisition and analysis of manufacturing data. A layered architecture is proposed, which is capable of coping with changes and disturbances in the shop floor, as well as with changing requirements in terms of monitored KPIs or time constraints.

Furthermore, guidelines regarding the implementation details and the related technologies are provided, illustrating a possible implementation of the proposed architecture.

As part of the H2020 PERFoRM project, future efforts will focus on the implementation of each of the proposed architecture's layers according to the described framework. This will be followed by the instantiation of the system in the project's different use cases, allowing for it to be validated across different application domains each with varying time constraints and requirements.

Acknowledgements This project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No. 680435.

References

1. Batanov, D., Nagarur, N., Nitikhunkasem, P.: Expert-mm: a knowledge-based system for maintenance management. Artif. Intell. Eng. 83–291 (1993)
2. Bellifemine, F., Poggi, A., Rimassa, G.: Developing multi-agent systems with a fipa-compliant agent framework. Softw. Pract. Experience 103–128 (2001)
3. Bellifemine, F., Poggi, A., Rimassa, G.: Developing Multi-agent Systems with JADE, pp. 89–103. Springer, Berlin, Heidelberg (2001). http://dx.doi.org/10.1007/3-540-44631-1_7
4. Caskey, K.R.: A manufacturing problem solving environment combining evaluation, search, and generalisation methods. Comput. Ind. 175–187 (2001)
5. Choudhary, A.K., Harding, J.A., Tiwari, M.K.: Data mining in manufacturing: a review based on the kind of knowledge. J. Intell. Manuf. 501–521 (2009)
6. Cushing, R., Belloum, A., Bubak, M., de Laat, C.: Towards a data processing plane: an automata-based distributed dynamic data processing model. Future Gener. Comput. Syst. 21–32 (2016)
7. Drath, R., Horch, A.: Industrie 4.0: hit or hype? [industry forum]. IEEE Ind. Electron. Mag. 56–58 (2014)
8. Gilchrist, A.: Introducing industry 4.0. In: Industry 4.0, pp. 195–215. Springer (2016)
9. Goodhope, K., Koshy, J., Kreps, J., Narkhede, N., Park, R., Rao, J., Ye, V.Y.: Building linkedin's real-time activity data pipeline. IEEE Data Eng. Bull. 33–45 (2012)
10. Holden, T., Serearuno, M.: A hybrid artificial intelligence approach for improving yield in precious stone manufacturing. J. Intell. Manuf. 21–38 (2005)
11. Kreps, J., Narkhede, N., Rao, J., et al.: Kafka: a distributed messaging system for log processing. In: Proceedings of the NetDB, pp. 1–7 (2011)
12. Lee, J., Kao, H.A., Yang, S.: Service innovation and smart analytics for industry 4.0 and big data environment. Procedia CIRP 3–8 (2014)
13. Rocha, A.D., Barata, D., Di Orio, G., Santos, T., Barata, J.: Prime as a generic agent based framework to support pluggability and reconfigurability using different technologies. In: Doctoral Conference on Computing, Electrical and Industrial Systems, pp. 101–110. Springer International Publishing (2015)
14. Rocha, A.D., Peres, R., Barata, J.: An agent based monitoring architecture for plug and produce based manufacturing systems. In: 2015 IEEE 13th International Conference on Industrial Informatics (INDIN), pp. 1318–1323. IEEE (2015)
15. Rocha, A.D., Peres, R.S., Flores, L., Barata, J.: A multiagent based knowledge extraction framework to support plug and produce capabilities in manufacturing monitoring systems. In: 2015 10th International Symposium on Mechatronics and its Applications (ISMA), pp. 1–5. IEEE (2015)
16. Stock, T., Seliger, G.: Opportunities of sustainable manufacturing in industry 4.0. Procedia CIRP 536–541 (2016)
17. Wang, G., Koshy, J., Subramanian, S., Paramasivam, K., Zadeh, M., Narkhede, N., Rao, J., Kreps, J., Stein, J.: Building a replicated logging system with apache kafka. Proc. VLDB Endowment 1654–1655 (2015)
18. Wang, S., Wan, J., Zhang, D., Li, D., Zhang, C.: Towards smart factory for industry 4.0: a self-organized multi-agent system with big data based feedback and coordination. Comput. Netw. 158–168 (2016)

Reconfigurable Stochastic Petri Nets for Reconfigurable Manufacturing Systems

Samir Tigane, Laid Kahloul and Samir Bourekkache

Abstract The use of Petri Nets in the design of Reconfigurable Manufacturing Systems is an attractive field. Petri Nets allow modelling RMSs in an abstract way and to make specification, refinement, verification and validation of these systems. Moreover, the use of Stochastic Petri Nets allows the designer to catch more indispensable aspects (time and stochastic events) of RMSs, thus the designer can make performance evaluation. In order to study explicitly the reconfigurability, in Petri nets, several works proposed the extension of PN called reconfigurable Petri nets. However, these extensions do not deal yet with stochastic Petri nets. The objective of this paper is to present an extension of Stochastic Petri nets to deal with reconfigurability in manufacturing.

Keywords Reconfigurable Petri Net · Reconfigurable manufacturing system · Improved net rewriting system

1 Introduction

A Reconfigurable Manufacturing System (RMS) has the ability to update itself, in order to answer dynamic requirements or unpredictable failures. The reconfigurability of an RMS is done at runtime; hence the system must not stop working but it must keep working and reconfigure itself. This reconfiguration activity requires time and cost. The use of formal methods (Petri nets, automata, logics, etcs) in the design of RMSs brings lots of advantages: high level specification, simulation, properties verification, performance analysis, etc.

S. Tigane (✉) · L. Kahloul · S. Bourekkache
LINFI Laboratory, Computer Science Department, Biskra University, Biskra, Algeria
e-mail: tigane.samir@gmail.com; samir.tigane@yahoo.fr

L. Kahloul
e-mail: kahloul2006@yahoo.fr

S. Bourekkache
e-mail: samir.bourekkache@yahoo.fr

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_34

A Petri net (PN) [9] is a bipartite graph (composed of place nodes, transition nodes, and arcs connecting these nodes) proposed to model/analyse discrete event systems. PNs have been extended to several high level formalisms to enhance their expressiveness (as an example, Generalized Stochastic Petri nets (GSPNs) [3] are the extension proposed to deal with stochastic times in modelled systems). PNs have been massively exploited in the study of manufacturing systems; thus researchers were oriented to propose several extension for the standard PNs to deal with reconfigurability. However, the most proposed high level Petri nets dedicated for RMSs do not consider time and stochastic events (inherent to RMSs). Hence, the objective of the current work is to define an extension of GSPN that handles reconfigurability in RMSs. The remainder of this paper is organised as follows: Sect. 2 presents related work, Sect. 3 introduces the concept of Generalized Stochastic Petri nets and presents our proposed extension, Sect. 3.3 explains the use of the proposed extension using a case study. Finally, Sect. 4 concludes the paper.

2 Related Works

In order to study RMSs, researchers have used either basic Petri nets or high level Petri nets, and they have considered several aspects such as: time, stochastic events, modularity, and reconfigurability. Stochastic Petri nets are a popular extension of Petri nets [8]; they were widely used in the literature to model and evaluate systems with stochastic nature. In [5] different logistic models were analysed in order to study their effects on system performance. A variant of SPN, stochastic Petri nets with fuzzy parameters are used in [15] to model and analyse an FMC (Flexible Manufacturing Cell). The fuzzy parameters are introduced in order to model the uncertainty about some information like the firing rate of a transition. In [10, 11, 13] the authors used a specific type of SPN called stochastic reward nets, where the firing of a transition is associated to an extra condition called guard. In [14], the authors combine an artificial intelligence technique called learning automata (*LA*) with adaptive SPN where each token has its own *LA* that allows adapting the token flow on the net in order to define the optimal firing sequence.

Considering the reconfigurability aspect, many extended PNs were proposed to model and analyse runtime reconfigurability in RMSs. Using graph grammars, PNs transformations rules are defined and used in the refinement of Manufacturing Systems [4, 12]. Besides graph transformation based techniques, rewriting systems were used in Badouel's reconfigurable Petri nets [1, 2] which were extended by [6, 7] in the INRSs (Improved Net Rewriting Systems) formalism. However, even the reconfigurable based approach offers explicit and intuitive modelling approaches where the formalism structure maps the reconfigurable structure of the system; these formalisms keep lacking of analysis techniques and the absence of time and stochastic aspects. In our work, we are interested in defining an approach to reconfigure Generalized Stochastic Petri Nets (GSPNs) without losing the behavioural properties.

3 Improved Net Rewriting System for GSPN

In a Petri Net, transitions model actions that are realised by the system. A ready action is modelled by an enabled transition. The execution of an action in the system is modelled through the firing of the suitable enabled transition in the model. The novelty of Stochastic Petri Net (SPN) is to associate a firing *rate* to each transition. Generalized Stochastic Petri Nets (GSPNs) represent a generalisation of SPNs by distinguishing two types of transitions: a set T_1 of *timed* transitions (represented graphically by unfilled bars) and a set T_2 of *immediate* transitions (represented graphically by filled bars). Hence, in GSPN, the firing delay of an enabled timed transition is exponentially distributed, and the firing of an enabled immediate transition is done in zero time and according to a firing priority. A typical GSPN G is defined as $G = (P, T, F, T_1, T_2, M_0, \Lambda, W)$, where (P, T, F, M_0) is the underlying marked PN, Λ is the delay function, and W is the priority function.

The Improved Net Rewriting System (IRNS) based approach presented in [6, 7] is a net rewriting system used to "*rapidly*" reconfigure PN models (representing reconfigurable manufacturing systems). By "*rapidly*", we mean that there is no need to verify the behavioural properties (i.e. liveness, boundedness, and reversibility) of the reconfigured models, since it is proved that they will not lose these properties after applying a rewriting rule using the IRNS-based approach. The main idea of this approach is to have a net block class library (well-formed net blocks) and then substitute a well-formed subnet of any live bounded reversible (*LBR*) *PN* by another well-formed net block of the same interface type. This transformation always leads to another *LBR PN*.

However, the direct applying of this approach to a GSPN model does not guarantee that the reconfigured GSPN preserves its behavioural properties; an example illustrating this statement is shown in Fig. 1. In this example, Fig. 1a represents a GSPN G to be reconfigured. The reconfiguration is done by replacing the image of

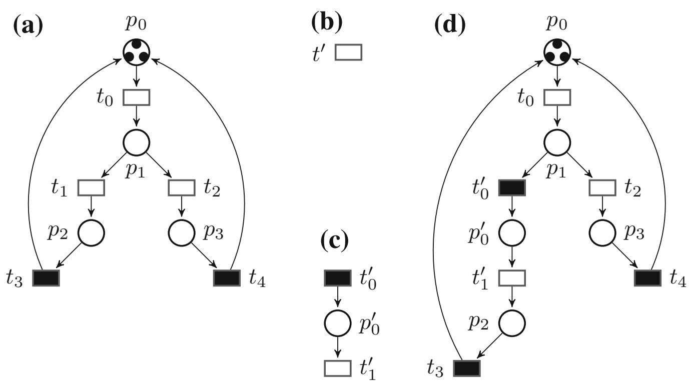


Fig. 1 Example of applying IRNS on a GSPN

the well-formed net block shown in Fig. 1b, by the well-formed net block shown in Fig. 1c. Thus, Fig. 1b, c are the rule left and right hand side, respectively. When applying the rule on the GSPN of Fig. 1a, the left hand side of the rule is mapped to the transition t_1.

The resulting GSPN is the graph G' shown in Fig. 1d. It is obvious that G' is not live, since whenever a token is present in place p_0, it will be consumed directly by the immediate transition t'_0, so the transition t_2 will never be fired (i.e. the transition t_2 is not live). This transformation is done according to the approach described in [6]. To overcome this kind of limitations, we extend this approach to deal with the GSPN. In the following subsections, we use/modify some definitions and net block classes specifications used in IRNS for ordinary PN. These modifications lead to a new approach that allows reconfiguring an LBR-GSPN (live, Bound and Reversible) to another one.

3.1 IRNS for GSPN

We define a GSPN-IRNS as a couple: $\mathcal{N} = (\mathcal{R}, G)$, where:

- $G = (P, T, F, M_0, T_1, T_2, \Lambda, W)$ is an LBR marked GSPN.
- $\mathcal{R} = \{r_i = \{L_i, R_i, f_i, {}^\bullet\tau_i, \tau_i^\bullet\}\}_{i=1,n}$ is a finite set of rewriting rules, where:
  - L_i and R_i (left and right hand-side, respectively, of rule r_i) are two marked GSPNs,
  - ${}^\bullet\tau = (I_L, I_R)$ (respectively $\tau^\bullet = (O_L, O_R)$) represents either two sets of places or two sets of transitions belonging to the two nets L and R, and calls the input interface relation (resp. output interface relation) of the two nets L and R. The input interface relation of a net L_i is denoted as: $I_{L_i} = \{x^i_{in} | i = 1, n\}$. The output interface relation of a net L_i is denoted as: $O_{L_i} = \{y^i_{out} | i = 1, m\}$;
  - $f_i : L_i \rightarrow g_i$ is an isomorphism, where g_i is a subnet of G and $\forall x \in g$ (*where* : $g = g_i - \{f_i(x_i) | x_i \in I_L \cup O_L\}$) and $\forall y \in G - g_i$, we have: $F(x, y) = F(y, x) = 0$.
- For each rewriting rule r: L and R must belong to one of the well-formed net blocks classes which are described in the following subsection.

Applying a rewriting rule $\{L, R, f, {}^\bullet\tau, \tau^\bullet\}$ on a GSPN G leads to a new one G'. Lets F' be the flow relation of G'. For all two nodes v, w in G', $F'(v, w)$ is defined by the following equations:

$$F'(v, w) = \begin{cases} F(v, w) & \text{if } v \notin R \wedge w \notin R \\ F_R(v, w) & \text{if } v \in R \wedge w \in R \\ 1 & \text{if } v \notin R \wedge w \in I_R \wedge \exists x \in I_L, F(v, x) = 1 \\ 1 & \text{if } v \in O_R \wedge w \notin R \wedge \exists y \in O_L, F(y, w) = 1 \\ 0 & \text{otherwise.} \end{cases} \quad (1)$$

Lets P' be the places in G'. The marking of place $p \in P'$ is given by,

$$M'_0(p) = \begin{cases} M_0(p) & \text{if } p \notin R \\ M_{0_R}(p) & \text{if } p \in R. \end{cases} \quad (2)$$

3.2 Well-Formed Net Blocks

In this subsection, we present a set of well-formed net blocks that establish the library of net blocks to be used in GSPN-rewriting. The proposed net blocks cope with the nature of GSPN. We follow the same notations presented in [6]. Hence, we consider two types of places $P^o = \{p|M_0(p) = 0\}$ called *operation places subset*, and $P^r = \{p|M_0(p) \geq 1\}$ called *resource places subset*.

Definition 1 A GSPN G is called *single place net* (*SP*), if $P = P^o = \{p\}, T = F = \emptyset$. In this case, the interface is place-type which is $p_{in} = p_{out} = p$.

Definition 2 A GSPN G is called *single transition net* (*ST*), if $T = \{t\}, P = F = \emptyset$. In this case, the interface is $t_{in} = t_{out} = t$.

Definition 3 A GSPN G is called *Open State Machine* (*OSM*), if the following conditions are satisfied:

1. $P = P^o, |P| > 1, |T| \geq 1$, and no directed circuit exists in the graph of G,
2. $\exists p_i, p_j \in P^o (i \neq j)$, such that ${}^\bullet p_i = p_j^\bullet = \emptyset$, (for a node x in G, we denote by ${}^\bullet x$ (resp. $x^\bullet$) the preset of the node x (resp. the postset of x))
3. $|{}^\bullet t| = |t^\bullet| = 1, \forall t \in T$,
4. $p^\bullet \subseteq T1$ or $p^\bullet \subseteq T2, \forall p \in P^o$.

In this case, the interface is place-type composed of the two places: $p_{in} = p_i$and $p_{out} = p_j$. The presence of two or more concurrent transitions that belong to different kinds (i.e. immediate and timed) in a GSPN makes it not live [3]. Thus, we add the $4th$ condition.

Definition 4 A GSPN G is called an *Open Marked Graph* (*OMG*), if the following conditions are satisfied:

- $P = P^o, |P| \geq 1, |T| > 1$, and no directed circuit exists,
- $\exists t_i, t_j \in T(i \neq j)$, such that ${}^\bullet t_i = t_j^\bullet = \emptyset$
- $|{}^\bullet p| = |p^\bullet| = 1, \forall p \in T$.

In this case, the interface is composed of two transitions: $t_{in} = t_i$ and $t_{out} = t_j$.

Definition 5 A marked GSPN G is called a *Close Marked Graph* (*CMG*), if the following conditions are satisfied:

- $P^r = \{\tilde{p}\}, |P^o| \geq 1$,
- $\exists t_i, t_j \in T(i \neq j)$, such that ${}^\bullet t_i = \{\tilde{p}\}$ and $t_j^\bullet = \{\tilde{p}\}$,

- $\tilde{p}^\bullet = \{t_i\}$ and ${}^\bullet\tilde{p} = \{t_j\}$,
- It exists an *OMG* class block $G' = (P^o, T, F'; M'_0)$ in G, such that: $F' = F - \{F(\tilde{p}, t_i)\} - \{F(t_j, \tilde{p})\}$, and $M'_0(p) = M_0(p), \forall p \in P^o$.

In this case, the interface is composed of two transitions: $t_{in} = t_i$ and $t_{out} = t_j$.

Definition 6 A marked GSPN G is called a *Sequential Close Marked Graph* (*SCMG*), if it can be generated from a *CMS* by rewriting, eventually several times, one of its well-formed subnet block of the class *OMG* or *ST* by another well-formed subnet block of the class *CMG*, such that the input/output nodes of the two subnet blocks belong to the same type (i.e. all of theme are either an immediate transitions or timed transition). The reason of this restriction is already illustrated in the former example (see Fig. 1).

Using the above well formed net-blocks, the designer can specify RMSs using GSPN and specify the reconfigurability in RMSs using GSPN-INRS.

3.3 The Case Study

In order to illustrate the application of the proposed extension, we use a simple RMS example inspired from the one presented in [5]. This RMS is composed of a single reconfigurable machine M which has two possible configurations. In its first configuration $C1$, it produces two types of products A and B. In its second configuration $C2$, the machine produces a third product C besides the two former ones. The RMS uses a transport device TL to bring raw materials $RawM$ to the machine, and to pick up the products. Figure 3a (respectively Fig. 3b) shows the RMS in its first configuration (respectively its second configuration). The interpretation of the nodes in the two graphs is illustrated in Table 1. The first configuration in Fig. 3a can be generated from the (*SCMG*) net block in Fig. 2 by rewriting its place p_1 by the $(OSM)b$ net block in Fig. 2. From this first configuration, we want to reach another configuration with two machines M' and PM. M' products three type of products A, B and C. PM packs the products. The new RMS model is shown in Fig. 3b.

This model is generated from the original model by applying three rewriting rules. The first one is done by replacing its subnet (from the class *OSM*) formed by $sn_1 = \{p_2, c_B, p_b, pB, p_3\}$, by the $(OSM)b$ net in Fig. 2. In fact, the left hand-side of the rule is the $(OSM)a$ net, the right hand-side is the $(OSM)b$ shown in Fig. 2. The left hand-side is mapped to the subnet sn_1. The second one is done by replacing the place p_4 by the $(OSM)c$ net in Fig. 2. Finally, the last new added transition is replaced by the (*CMG*) net block in Fig. 2. The result of applying the $2nd$ and the $3rd$ rules is then substituting the place p_4 depicted in Fig. 3a by the subnet formed by $sn_2 = \{p'4, ld', p_5, pk, p_6, uld', p_7, p_f\}$ depicted in Fig. 3b. The resulting new RMS model is a LBR-GSPN.

Table 1 Meaning of places and transitions

Place	Meaning	Transition	Meaning
t_f	Number of free TLs	tr_{in}	RawM is transported to *M*
p_1	RawM to be loaded into *M*	*ld*	Loading RawM into *M*
p_2	RawM is loaded into *M*	*cA*(*cB*)	RMS choose producing *A*,*B* or *C*
$p_a(p_b)$	Product *A*(*B*) is selected	*cC*	RMS selects product *C*
p_c	Product *C* is selected	*pA*(*pB*)	*M* is processing *A*(*B*)
p_3	*M* has finished the processing	*uld*	Piking up the product by the TL
p_4, p_7	# of finished products	tr_{out}	Product is transported to next cell
p'_4	# of finished products waiting for packing machine *PM*	*uld''*	Moving the product to *PM*
		pC	*M* is processing *C*
p_5	Product is loaded into *PM*	*ld'*	Loading product into *PM*
p_6	*PM* finished packing product	*pk*	*PM* is packing the product
$m_f(p_f)$	*M* (*PM*) is idle	*uld'*	Piking up the product by the TL

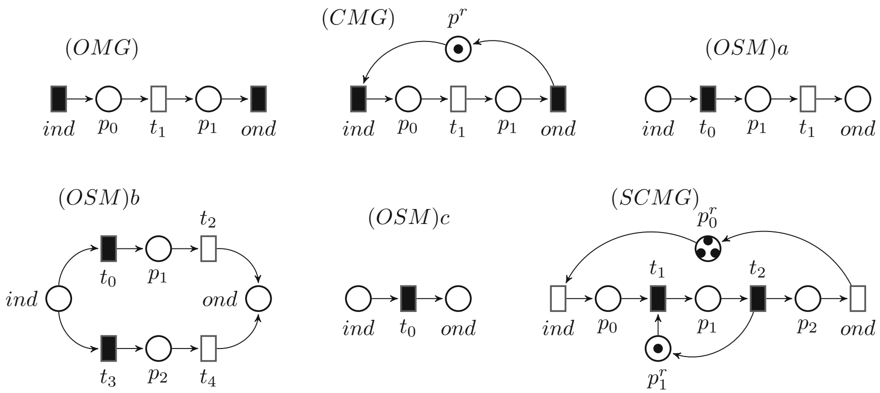
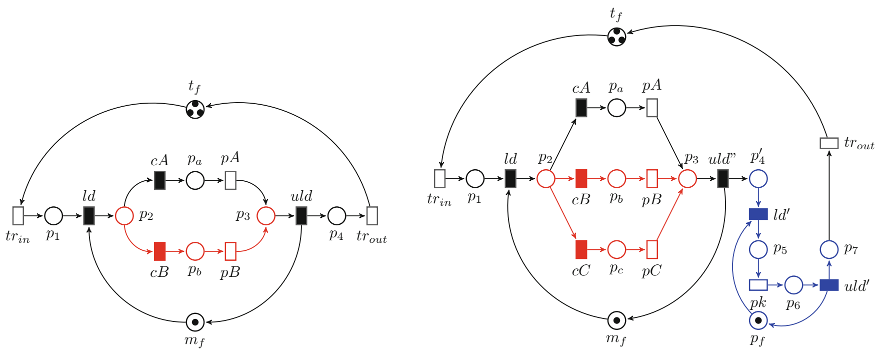


Fig. 2 Examples of well-formed net blocks, where *ind*/*ond* stand for input node/output node

4 Conclusion

The apparition of reconfigurable manufacturing systems enhanced the power of manufacturing systems by several abilities. RMSs have the ability to change their structure to adapt themselves to either to new market requirements and unpredictable damages, or to augment their throughput. RMSs are also critical systems where safety and reliability are indispensable; thus the designer of such systems has a high



(a) RMS model (b) New RMS model

Fig. 3 Reconfiguration by using IRNS for GSPN

responsibility. The use of Petri nets as a formal framework has demonstrated several advantages in the design of RMSs, their verification, validation and evaluation. Researchers in the field of Petri nets proposed several extensions of Petri nets dedicated to RMSs. However, at our knowledge there is not yet extension for Stochastic Petri nets neither generalised stochastic Petri nets. These two formalism are important due to their expensiveness power allowing the modelling of timed and stochastic systems (the case of RMSs).

In this paper, we have proposed an extension of GSPN allowing the reconfiguration of the models. These reconfiguration is formulated using the theory of net rewriting systems which was applied formerly to basic Petri nets. The proposed extension defines a set of well formed blocks which can be used as space part by the designer of RMSs. The proposed reconfiguration is assumed to preserve the three properties of Petri nets (liveness, boundedness, and reversibility). Due to space reason, the paper has presented only some of the proposed net-blocks and applied them on a simple case study. In fact, the manual reconfiguration of an LBR (Live, Bounded and Reversible) GSPN to another one is not always an easy task. As an example, the modification of the structure or the behaviour of an RMS involves considering how the RMS will detect, recover or prevent the deadlock. So the designer will pay an extra effort to model the side effects of this modification. Our approach helps the designer to focus only on reconfiguring the RMS without considering how to maintain its behavioural properties, since they are preserved systematically by our approach. In a future work, we will define the whole set of net blocks, the necessary proofs and a demonstration on a real case study.

References

1. Badouel, E., Llorens, M., Oliver, J.: Modeling concurrent systems: Reconfigurable nets. In: PDPTA, pp. 1568–1574 (2003)
2. Badouel, E., Oliver, J.: Reconfigurable nets, a class of high level Petri nets supporting dynamic changes within workflow systems. Ph.D. thesis, INRIA (1998)
3. Bause, F., Kritzinger, P.S.: Stochastic petri nets (2002)
4. Kahloul, L., Bourekkache, S., Djouani, K.: Designing reconfigurable manufacturing systems using reconfigurable object petri nets. Int. J. Comput. Integr. Manuf. 1–18 (2016)
5. Li, G., Mitrouchev, P., Wang, Y., Brissaud, D., Lu, L.: Evaluation of the logistic model of the reconfigurable manufacturing system based on generalised stochastic petri nets. Int. J. Prod. Res. 6249–6258 (2012)
6. Li, J., Dai, X., Meng, Z.: Improved net rewriting system-based approach to model reconfiguration of reconfigurable manufacturing systems. Int. J. Adv. Manuf. Technol. 1168–1189 (2008)
7. Li, J., Dai, X., Meng, Z., Dou, J., Guan, X.: Rapid design and reconfiguration of petri net models for reconfigurable manufacturing cells with improved net rewriting systems and activity diagrams. Comput. Ind. Eng. 1431–1451 (2009)
8. Marsan, M.A., Balbo, G., Conte, G., Donatelli, S., Franceschinis, G.: Modelling with Generalized Stochastic Petri Nets. Wiley, Inc. (1994)
9. Murata, T.: Petri nets: properties, analysis and applications. Proc. IEEE 541–580 (1989)
10. Philip, A., Sharma, R.K.: A stochastic reward net approach for reliability analysis of a flexible manufacturing module. Int. J. Syst. Assur. Eng. Manage. 293–302 (2013)
11. Postema, B.F., Haverkort, B.R.: Stochastic petri net models for the analysis of trade-offs in data centres with power management. In: International Workshop on Energy Efficient Data Centers, pp. 52–67. Springer (2014)
12. Prange, U., Ehrig, H., Hoffmann, K., Padberg, J.: Transformations in reconfigurable place/transition systems. In: Concurrency, Graphs and Models, pp. 96–113. Springer (2008)
13. Sachdeva, A., Kumar, P., Kumar, D.: Behavioral and performance analysis of feeding system using stochastic reward nets. Int. J. Adv. Manuf. Technol. 156–169 (2009)
14. Shojafar, M., Pooranian, Z., Meybodi, M.R., Singhal, M.: Alato: an efficient intelligent algorithm for time optimization in an economic grid based on adaptive stochastic petri net. J. Intell. Manuf. 641–658 (2015)
15. Tüysüz, F., Kahraman, C.: Modeling a flexible manufacturing cell using stochastic petri nets with fuzzy parameters. Expert Syst. Appl. 3910–3920 (2010)

Part VIII
Virtualization and Simulation in Computing-Oriented Industry and Service

Simulation Platform for Virtual Manufacturing Systems

Radu Dobrescu and Daniel Merezeanu

Abstract The paper presents the design of a cloud simulation platform and the associated services and the computing resources for hybrid simulation of Virtual Manufacturing Systems. At the core of the research is the design of a framework defined as Virtual Development Environment (VDE). Associated to VDE is a set of block functions and algorithms defined as software assets, stored in a large repository implemented within an on-line cloud supported library. The platform offers flexibility concerning the continuous testing and updates regarding the algorithms in order to improve the existing ones.

Keywords Virtual manufacturing systems · Hybrid process simulation · Context-aware systems · Cloud computing · Hardware in-loop · Sensor-cloud infrastructure · Software reusability

1 Introduction

The field of Virtual Manufacturing Systems (VMS) and the associated Virtual Manufacturing Technologies allows the designers to virtually conceive new (individually optimised) machines, robots, logistic installations, sensors, drives and control components. It also allows the achievement of matched combinations of individual components in order to be implemented into a real manufacturing line (RML). Outstanding results in VMS design can be obtained keeping up with the faster progress of information technology and communications. This issue is interdisciplinary and also complex, and constantly VMS is integrating new concepts and technologies such as embedded systems, grid computing, model-driven, cloud

R. Dobrescu (✉) · D. Merezeanu
Department of Automation and Informatics, University Politehnica of Bucharest, Bucharest, Romania
e-mail: rd_dobrescu@yahoo.com

D. Merezeanu
e-mail: danmerezeanu@gmail.com

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_35

computing or sensor networks. Considering the available IT technologies, the main problem in developing VMS is the precision of the models and of the simulation, more exactly the design of a simulation platform as support for an adaptive simulation framework which allows elaborating complex models and optimizing software architectures by dynamically composing reusable modules in response to specific requirements of the users.

In order to satisfy such requirements, the authors develop a platform (named in the following HYPROVIR) which offers a framework for swapping a virtual device with its real counterpart. The virtual model is developed using simulation software obtained from the vendor of the actual device, so that the model offers the same functionality as the actual device. The proposed framework (named Virtual Development Environment—VDE) meets the integrative approach which is nowadays a key aspect in mathematical modelling of industrial systems. It represents also a proof-of-concept for Hybrid Process Simulation (HPS) applications in the design and reconfiguration of a manufacturing line.

2 State of the Art in the Field of Virtual Manufacturing Systems

There are many simulation tools available to support the testing and validation needs of a process planner for a manufacturing system. However, for a complex system the multitude of connections between components may be difficult to replicate in a single simulation environment. Furthermore, simulation models of the entire system may be inaccurate or unavailable, although accurate models may still exist for individual components within the process. For these reasons, a distributed simulation approach is desirable. Integrated simulation with real components for control testing is more accurate, since it includes the same components that will be used in the final process. For these reasons, we hold for the state-of-the-art analysis only two directions of development: one that aims at creating a virtual simulation framework (in short Virtual Reality—VR), the other which performs simulation in a mixed (hybrid) framework (in short Hybrid Process Simulation—HPS) which involves combining actual (real) and virtual components.

2.1 Virtual Reality

New advances in computerized modelling, visualization, simulation, and product data management are making virtual reality (VR) a viable alternative to traditional product realization and manufacturing. In [1] are discussed the reasons why VR can be a powerful tool for applications in manufacturing. The authors of [2] present a novel general-purpose simulation analysis application that combines concurrent operations simulation with the advanced data interrogation and user interaction capabilities of immersive virtual reality systems. The architecture of a Virtual Manufacturing System is presented in [3]. The VMS includes three basic functional modules: virtual manufacture environment module, machining process simulation module and detection analysis optimum module.

2.2 Hybrid Process Simulation

Several works have focused on building virtual models to simulate part or all of a manufacturing line. Among the first, Park [4] develops a methodology for creating a virtual model for pure planning and predictive purposes, but without any real time communication or feedback from actual devices. Ng et al. [5] take this a step further by using I/O synchronization with a 3D graphical simulation to model and study manufacturing resources (e.g., robots). It involves a manufacturing simulation of a machine service support system (MSSS) for remote press line monitoring and diagnostics using discrete event simulations (DES) and computer-aided robotics (CAR). Later, Bohlmann et al. [6] propose a hybrid process net simulator framework which allows specifying complex models, typically represented as a system of differential equations mixed with continuous and discrete-event subsystems, based on a bipartite graph structure. For many authors, the concept of HPS is closely related to model-driven software engineering, where the goal is to develop a framework for reuse of and interoperability between a variety of passive and active information models [7]. The Hardware-in-Loop (HIL) approach was developed to address this type of iterative process by allowing the creation of a test setup using simulations that are connected to real hardware/software. There are however few drawbacks associated with this approach, including the lack of a widely accepted formalized methodology and typical applications being limited to only one region of simulation [8]. In [9], the authors advance the idea of an ontology which provides a conceptual architecture developed for an HPS, such that a general interpretation of a manufacturing system's implementation is possible.

2.3 Combining HIL and VR

HIL and VR methodologies have two important common features that come from the similarities in their key objectives. About VR, it is said that it serves to better understand psychological and physiological processes as they occur in non-mediated settings. HIL is also used by engineers or process planers to better understand a process or system during its non-mediated (normal) operation. Different authors who noticed the similarity between VR and HIL sought to find similarity between their formalisms of representation. For example, the authors of [10] address fidelity by specifically looking at the interoperability of different software applications. Fidelity approaches for testing actual and simulated components in manufacturing systems were considered also by [11]. In [12] the authors designed the architecture from the characteristics and requirements of what they call Reality-in-the-loop (RIL) and Soft Commissioning. Recent achievements bring to the fore issues of emergent simulation technologies such as data analytics [13] and holonic modelling [14].

3 Description of the System Architecture

3.1 Main Components

The main components of HYPROVIR architecture are:

1. The *Virtual Development Environment* (VDE) for VM capable of emulating multiple simulation structures. VME allows integrating various new concepts and technologies such as embedded systems, hardware-in-loop, model-driven architectures, cloud computing or sensor networks.
2. The *Sensor-Cloud interface* (SCI) which allows the integration of sensor network-based technologies (wired or wireless) with virtual manufacturing systems. This includes the interconnection of a physical sensor node with the VDE and also the transfer of real data to cloud.
3. The *Context-aware adaptation framework* (ADFRAM) and its corresponding supporting service-oriented middleware, which provides a systematic way to support adaptation behaviour evolution and offers support for adaptation with multiple concerns. ADFRAM constructs its system global adaptation module by composing multiple basic adaptation modules during run-time, according to the context to date.

a. VDE Role and Objectives

Two activities are at the main core of VDE:

- The modelling activity, which includes determining what the model shall do and which degree of abstraction is needed to this purpose;
- The ability to represent the model in a computer-based environment and to correlate it with to the response of the real system with a high degree of accuracy of description and the obtained simulation.

When using models and simulations in the design and reconfiguration of manufacturing systems, it is difficult to gage the fidelity of the model, especially if the system being modelled doesn't yet exist. The model cannot typically be validated until the system is in place. To overcome this impediment, the concept of Hybrid Process Simulation (HPS), an extension of traditional Hardware-in-Loop

(HIL) technology will be implemented, as a bridge between pure simulation and the final physical system.

b. SCI Role and Objectives

SCI virtualizes multiple physical sensors as virtual sensor. Dynamically grouped virtual sensors are provisioned automatically in response to the requests from users. SCI also provides a user interface for: registering or deleting physical sensors, requesting for provisioning or destroying virtual sensors, controlling and monitoring virtual sensors, and registering or deleting users. Figure 1 shows the system architecture of SCI.

The roles of the main components are as follows:

1. *Portal server*. It offers the menus of the available operations for the end user and for the sensor owners.
2. *Provisioning Server*. It provisions the virtual sensor groups for the requests from the portal server.
3. *Virtual Sensor Group*. Each group is automatically provisioned on a virtual server by the provisioning server. The virtual sensor groups are controlled directly or form a Web browser.
4. *Monitoring Server*. It receives the data about virtual sensors from the agents in the virtual servers and the servers and stores the received data in a database.

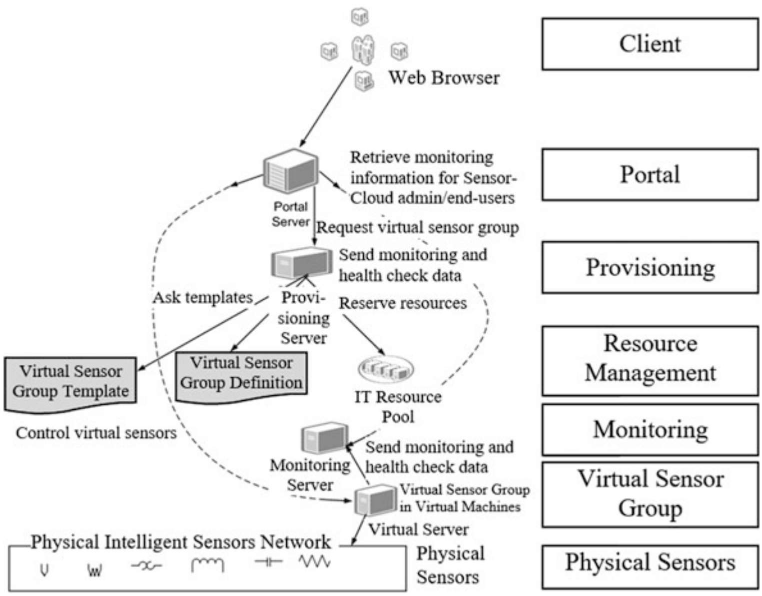


Fig. 1 Architecture of SCI

c. ADFRAM Role and Objectives

ADFRAM provides context-specific fusion of adaptation behaviour. Depending on the context, a set of candidate strategies can be selected and composed into a custom-generated global adaptation module matching the current environmental conditions. Additionally, it ensures the development of mechanisms which allow the global adaptation module to evolve and re-optimize by dynamically composing multiple reusable adaptation modules in response to context changes.

Figure 2 shows the software architecture of ADFRAM.

There are five main components of the framework:

- *Service Manager*. It will entertain requests for services and put the requests into a Request Queue.

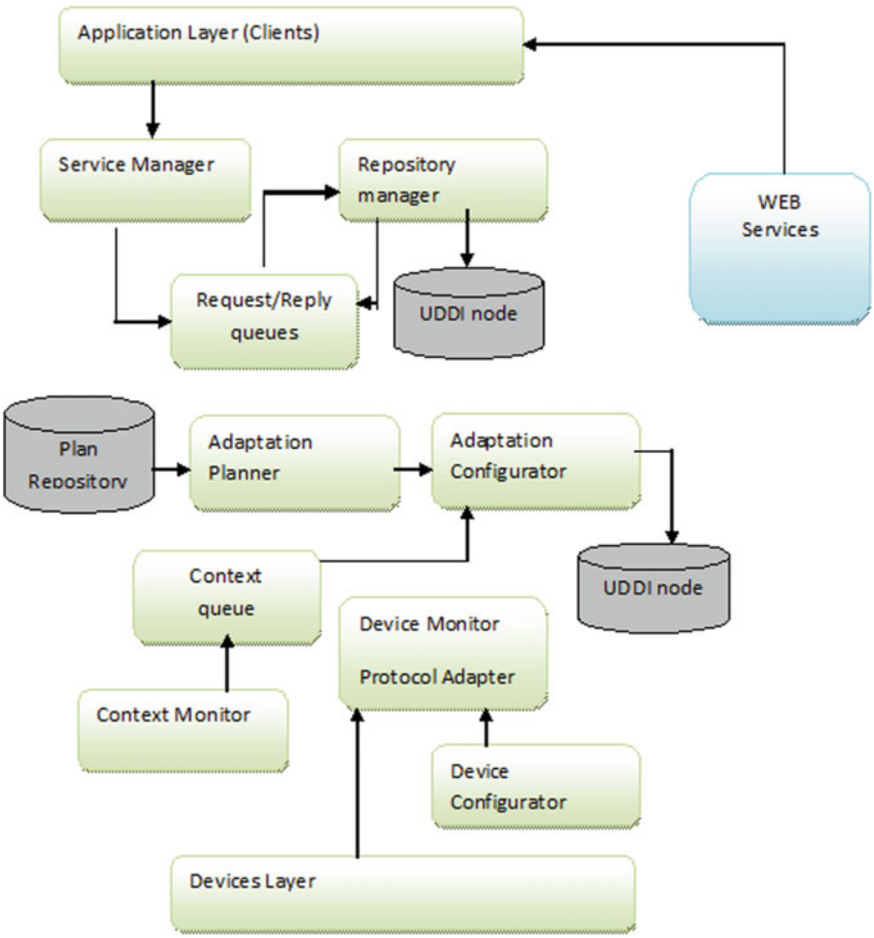


Fig. 2 Architecture of the adaptation framework

- *Repository Manager*. It searches the information on the required web service and places the result into the Reply Queue.
- *Adaptation Manager*. The Adaptation Manager consists of two main components: Adaptation Planner (which acquires information regarding the changing environment) and Adaptation Configurator (which receives the current plans for all devices and stores them in the Active Plan repository).
- *Context Monitor*. The role of the Context Monitor is to monitor the operating environment, check for malfunction device or addition of new devices.
- *Device Adapter*. This component is the actual interface to the API and will be dynamically loaded when required.

3.2 New Software Support

a. Hybrid Process Simulation Support (HPSS)

HPPS is an extension of the Hardware-in-Loop (HIL) technology to address the creation of a test setup using simulations that are connected to real hardware/software. The simulation runs in real-time, and is used for validation analysis and design decisions.

b. Requirements Reuse for Software Development

Software reuse is aimed at significantly incrementing the quality and productivity of software artefacts. Software quality is favoured by using proven and validated components. Software productivity is improved by reducing time-to-market for software applications. An efficient approach is the systematic use of experience and previously developed systems as assets. The asset term refers to any reusable product in the software life cycle (models and domain architectures, requirements, designs, code, data base components, documentation, tests).

c. Variability mechanisms for applying the Software Product Line Engineering (SPLE) concept

SPLE consists of two processes known as domain-engineering and application-engineering. The process of presenting the software assets (in domain-engineering) is called variability modelling. The main objective of application-engineering is to configure a software product from the domain-engineering process by managing SPLE assets using variability modelling technique.

d. Knowledge-Based Engineering System (KBES)

The continually progressing development of computer aided design and management of software-systems demands a constant learning process of the users to ensure that they are capable to exhaust the full potential. KBES will include the necessary product- and process-knowledge, which can be accessed by the user according to each specific situation. On the other hand, KBES will also be an educational tool.

4 Expected HYPROVIR Performance

The most ambitious objective of the project is to design HYPROVIR as a *cloud platform with cloud specific associated services.* This will provide the computing resources and services for access to algorithms for advanced control and optimization of no matter how complex manufacturing systems and processes and even of large-scale industrial plants. These services will allow the user to perform on line risk analysis and hazard prevention using generic control, optimization, fault detection and diagnosis, fault accommodation and risk analysis algorithms. It makes possible to utilise any new achievement provided by Information Technology evolution, i.e. use of the most advanced techniques in the fields of cloud computing engineering, internet-based services, standardized reusable function blocks, complex systems optimization, fault tolerant control and hazard analysis, open communication object for process data and functions in order to create an innovative, effective and reliable Hybrid Process Simulation tool that will respond to the current control demands of the industry.

The fulfilment of the above mentioned objectives will lead to significant achievements in three main directions:

1. Offering a new logistic, operating procedure as open platform for modular object-oriented modelling, simulation, identification and optimization.
2. Providing a generalization of the innovative hardware and software architectures, with a focus on hybrid systems.
3. Sustaining efficient applications in various domains, being useful not only in design and management of virtual manufacturing systems, but also in education, scientific research, technological development and innovation.

The HYPROVIR platform will support the development and testing of software assets used in modelling and simulation of distributed systems in heterogeneous environments, addressing issues such as data consistency, reliability, scalability and the efficient use of underlying resources. This performance is due to the implementation of the On-line web-based assets library.

One crucial aspect of the application consists in receiving and analyzing the process feedback. Users will be encouraged to download the function blocks and test them for different industrial application. Based on the feedback received from the end-users, a selection process will be carried out so that only the most effective assets are kept in the library while less performing ones are either eliminated or optimized in order to meet the users' criteria.

5 Conclusions

HYPROVIR gives a holistic approach to ensure complementarity of Requirements Engineering (RE)—which offers the means to discover, model, and manage the requirements of the products that comprise a product line and Software Product Line Engineering (SPLE) which offers the means of realizing the products' requirements from a common base of software assets. In practice, scalability and traceability of production line features are key challenges for RE. Feature modelling, by contrast, has been widely accepted as a way of modelling commonality and variability of products of a manufacturing line that may be very complex. A holistic approach is a goal-driven feature modelling approach separating the feature space in terms of problem space and solution space features. It also establishes explicit mappings between them and contributes to reducing the inherent complexity of a mixed-view feature model.

To summarize, with HYPROVIR, developing and procuring a software asset for the purposes mentioned above, should be very easy and fast. Using all the resources that Cloud Computing offers, the platform can be seen as an Integration Framework that provides a service-oriented integration approach with standardized interfaces and supporting tools. Each module of the platform will be accessed throughout a Cloud interface. This way the Cloud API will enable the possibility to add new applications with their own interfaces to the framework, applications that are designed for different purposes other than the algorithm library.

By developing HYPROVIR as a PaaS (Platform as a Service), the beneficiaries of the services would obtain access to the optimized and elastic hosting of Internet-scale multi-tier applications. HYPROVIR embeds all the necessary features, easing programming of rich applications and enabling the creation of an environment where complex process control algorithms can be developed and the possibility to host applications coming from different providers that can be tailored to different users. The project also comes to support the research activities in the field by creation of a solid knowledge database in the field of manufacturing industry.

References

1. Hamid, N.S.S., Aziz, F.A., Azizi, A.: Virtual Reality Applications in Manufacturing System. In: Science and Information Conference, pp. 1034–1037 (2014)
2. Dorozhkin, D.V., Vance, J.M., Rehn, D.G., Lemessi, M.: Coupling of interactive manufacturing operations simulation and immersive virtual reality. Virtual Reality **16**, 15–23 (2012)
3. Li, X., Li, C.: Development and application of virtual manufacturing technology. Adv. Mater. Res. **765–767**, 151–155 (2013)
4. Park, S.C.: A methodology for creating a virtual model for a flexible manufacturing system. Comput. Ind. **56**(7), 734–746 (2005)

5. Ng, A.H., Adolfsson, J., Sundberg, M., Vin, L.J.: Virtual manufacturing for press line monitoring and diagnostics. **48**(5), 565–575 (2008)
6. Bohlmann, S., Klinger, V., Szczerbicka, H.: HPNS—a hybrid process net simulation environment executing online dynamic models of industrial manufacturing systems. In: Proceedings of the 2009 Winter Simulation Conference, pp. 874–885 (2009)
7. Weißenberger, B., Flad, S., Chen, X., Rösch, S., Voigt, T., Vogel-Heuser, B.: Model driven engineering of manufacturing execution systems using a formal specification extension of the MES-ML for the generation of MES code. In: IEEE 20th Conference on Emerging Technologies and Factory Automation (ETFA), pp. 1–8 (2015)
8. Viswanathan, J., Harrison, W.S., Tilbury, D. M., Gu, F.: Using hybrid process simulation to evaluate manufacturing system component choices: Integrating a virtual robot with the physical system. In: Proceedings of the 2011 Winter Simulation Conference, pp. 2827–2838 (2011)
9. Harrison, W.S., Tilbury, D.M., Yuan, C.: From hardware-in-the-loop to hybrid process simulation: an ontology for the implementation phase of a manufacturing system. IEEE Trans. Autom. Sci. Eng. **9**(1), 96–104 (2012)
10. Shao, G., Leong, S., McLean, C.: Simulation-based manufacturing interoperability standards and testing. In: Proceedings of 9th International Conference on Progress of Machine Technology, pp. 283–286 (2009)
11. Hibino, H., Fukuda, Y.: Emulation in manufacturing engineering processes. In: Proceedings of 40th Conference Winter Simulation, WSC'08, pp. 1785–1793 (2008)
12. Taixiong, Z., Yage, Z.: Development of hardware-in-loop and virtual reality co-simulation platform for automotive anti-lock braking system. In: Proceedings of the International Conference on Information Science and Control Engineering, pp. 1–6 (2012)
13. Vodencarevic, A., Fett, T.: Data analytics for manufacturing systems. experiences and challenges. In: Proceedings of 20th IEEE Conference on Emerging Technologies and Factory Automation, pp. 1064–1067 (2015)
14. Barbosa, J., Leitão, P., Adam, E., Trentesaux, D.: Dynamic self-organization in holonic multi-agent manufacturing systems: the ADACOR evolution. Comput. Ind. **66**, 99–111 (2015)

Environment to Simulate Distributed Agent Based Manufacturing Systems

Andre Dionisio Rocha, Pedro Barroca, Giovanni Dal Maso and Jose Barata Oliveira

Abstract With the constant evolution of markets, new paradigms have emerged. Those new paradigms require solutions more versatile and able to cope with the unexpected changes. The world of manufacturing involves many risks and in order to reduce initial investments it is frequent to use virtual environments like simulators. This document presents a functional architecture of integration between a generic simulator and a multi-agent system, in order to meet the requirements of evolutions in manufacturing and the demand for customized products. One implementation case and results are presented as well.

Keywords Multi-agent system • Simulation • Distributed systems • Industry 4.0 • Plug and produce

1 Introduction

Due to the market developments and the emergence of new requirements regarding the customization of products, the guidelines of production technologies have changed. The old sequential lines based on mass production are no longer the most

A.D. Rocha (✉) · P. Barroca · J. Barata Oliveira
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa, Caparica, Portugal
e-mail: andre.rocha@uninova.pt

P. Barroca
e-mail: p.barroca@campus.fct.unl.pt

J. Barata Oliveira
e-mail: jab@uninova.pt

A.D. Rocha · J. Barata Oliveira
UNINOVA Institute, Centre for Technology and Systems, Caparica, Portugal

G.D. Maso
Technology Transfer System s.r.l., Milan, Italy
e-mail: dalmaso@ttsnetwork.com

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_36

appropriate solutions and companies need new ones. The lack of capacity of centralized solutions in dealing with the complexity, the high level of education in some economies and the demand for quality and diversity of the products were some of the decisive factors for the use of adaptive capabilities to adverse situations in manufacturing systems.

Production and industrial development are themes that require a large initial investment due to the scope of the involved areas. In order to reduce costs and increase process efficiency there is a constant research into new methodologies and production techniques. One of the alternatives used to reduce costs are the simulation environments. In order to improve the offers available in terms of simulation of industrial environments, this paper suggests an architecture that aims, without regard to the used technology, to integrate an industrial simulator and a multi-agent system, allowing simulating architectures that can handle new paradigms of manufacturing, and demonstrate their efficiency before being implemented in real systems.

This document is composed by five more chapters. The second chapter presents a brief description of the work undertaken in related areas. In the third chapter, the architecture is presented for systems integration. The fourth chapter offers a detailed description of the implementation. Finally, in the sixth chapter some conclusions are drawn.

2 Related Work

Originally the production systems have been designed for mass production, where a production without variety of products and as fast as possible is implied. As the range of specification wasn't comprehensive, a system based on simple rules was ideal, allowing one to create systems with a large production flow but with poor flexibility and variety. With the globalization of markets, new solutions were required; it is imperative to customize products instead of mass production. So, changes were needed in production lines and in the solutions applied to them, towards the dynamic adaptation to customer needs.

2.1 Emergent Production Paradigms

This evolution has brought new production paradigms such as evolvable systems, and reconfigurable distributed processing, among others; in short, adaptive and reconfigurable systems, using optimization strategies to deal with unexpected events and adversities. The strongest branch of research lies on the multi-agent systems, MAS, with a brief description of some of the paradigms.

2.1.1 Reconfigurable Manufacturing Systems

The main idea of reconfigurable manufacturing system is the modular manufacturing composition, where the use of a relation between the working tool and the machine enables the reconfiguration of the system through a reconfiguration of the master controller, obtaining adaptability and scalability [1].

With the easiness of changing the tool on the same machine, by reconfiguration, the reuse of existing lines is possible, which enables to the manufacturing lines to produce various types of products, without major remodelling, and still have some flexibility as required in personalization specifications [2].

Some studies tend to improve the reconfiguration techniques used in these systems, for example in [3] where Petri nets are used to control the logic execution, making it possible to model the execution in order to automatically reconfigure the system based on results.

2.1.2 Holonic Manufacturing Systems

Other architectures proposed are based on the concept of Holon, where the system is seen as a whole composed by parts. In other words, complex systems are formed by several simple systems allowing them to meet more challenging tasks, with heavier requirements [4].

In [5] a system is proposed where the existence of an intermediate type of agent, the Middle-agent, is responsible for creating teams among them so it can provide higher level tasks, who are composed by some atomic skills.

Two other examples: [6] suggesting the ADACOR and [7] PROSA for the integration of the respective architectures into holonic systems due to their modular composition, in which the various actors are considered holons.

2.2 Simulation in Manufacturing

The main role of simulation is to allow a real system to be studied in a virtual environment, abstracting the system and exploiting its behaviours safely and without the costs associated with physical tests, as described in [8]. Typically, with the capacity of accelerating the simulated production time, it allows to get faster and obtain detailed results. Typically based on mathematical models, simulators intend to create a relationship between the input variables and the output ones. There are two models in the literature, exploratory and predictive approaches. On one hand, the exploratory approach creates hypotheses based on past observations, for example in [9] managed by a colony of ants. On the other hand the predictive approach is used to predict future states, for example in [10] which was used to predict the spread of dengue fever in Brazil.

2.3 Related Scientific Topics

There are several scientific developments and researches based on emergent paradigms and simulation environments. However, they all tend to implement a new system, without taking advantage from the specialized work already implemented. Some literature like [11, 12] create an agent-based simulation from scratch, ending with weak visual components. However, the integration with an existing simulator was developed by [13], but instead of developing a general architecture as in [14], the authors implemented a specific solution. The present paper offers a different perspective, by suggesting an architecture regardless of the technology, that proves the integration of two components; these components bring advantages to the scientific and business evolution through its advanced technologies.

3 Simulation Environment Architecture

The proposed architecture aims at integrating a simulator of industrial platforms and a multi-agent system, thus allowing new protocols, new architectures, new ideas and concepts, in relation to the industrial sector and its new paradigms, to be tested. Using a simulator based on modular working tasks, and concepts of the multi-agent systems (MAS), we intend to design a solution that is transversal to the used technology; in other words, it is intended that this architecture is generalized in order to prove that any MAS and any simulator with modular characteristics can be integrated and create a new system compatible with the established specifications.

3.1 Architecture

To maintain horizontality, it is essential that the modules have total abstraction of the control estate in the simulator, being limited to the physical characteristics such as moving a part or perform some actions.

On the other hand, the MAS must provide support for multiple behaviours such as communication and interoperation capabilities. As in the simulator, the MAS agents must be independent from each other and be able to satisfy all the requests that arrive as easily and efficiently as possible.

Figure 1 represents is a global scheme of the architecture. As you can see the interactions between these two systems are made directly. Those interactions can be of many kinds: data, task execution or wakeup events are some of the cases.

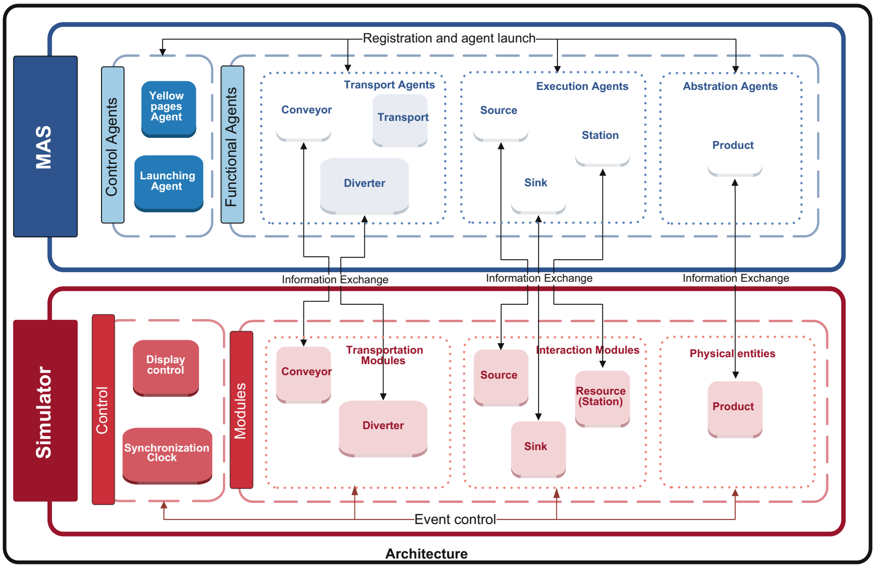


Fig. 1 Global simulation architecture

3.2 Agents

Despite the lack of consensus in the agent definition, the community seems to converge in some essential features: autonomy, evolution and ability to perform tasks, [15].

In this architecture, the agents are divided in two groups, the group of control agents and the group of functional agents. The control agents are composed by only two types: the yellow-pages agent, whose functionality is to keep a list of all the skills and agents that are in the system, and the launching agent which functionality is to launch the other agents into the system's main container.

On the other hand, the functional agents group is divided in three more sub-groups, the transport, the execution and the abstraction agents. The abstraction agents' purpose is the logical representation, only monitoring the status of the product. Another subgroup is composed of the execution agents: source agent, sink agent and resource agent. They are called execution agents because their purpose is to interact directly with the product, i.e., creating, removing or altering them. Lastly, the transport agents: the conveyor, the diverter and the transport agent as their name suggests are responsible for the product transportation on the entire route.

3.3 Transport

In order to increase the versatility of the system, it is necessary to change the traditional transport methods. A logic based on routing tables can be applied. Figure 2 shows the transport network where some knowledge from other areas was applied. One example of telecommunication protocols applied to MAS is given in [16] where the Dijkstra algorithm was used as best route calculator.

It is important to note that in this architecture only the execution modules are placed in routing tables because the transportation is handled by the transport agent and for this reason the products only know their destiny.

3.4 Product

As shown in Fig. 3, the product agent is ruled by an operating cycle. Initially the agent is registered in the system and loads essential information corresponding to the product type.

Then it searches the yellow-pages system for whoever performs and wherever the first task can be performed. At last, the product asks the workstation for the

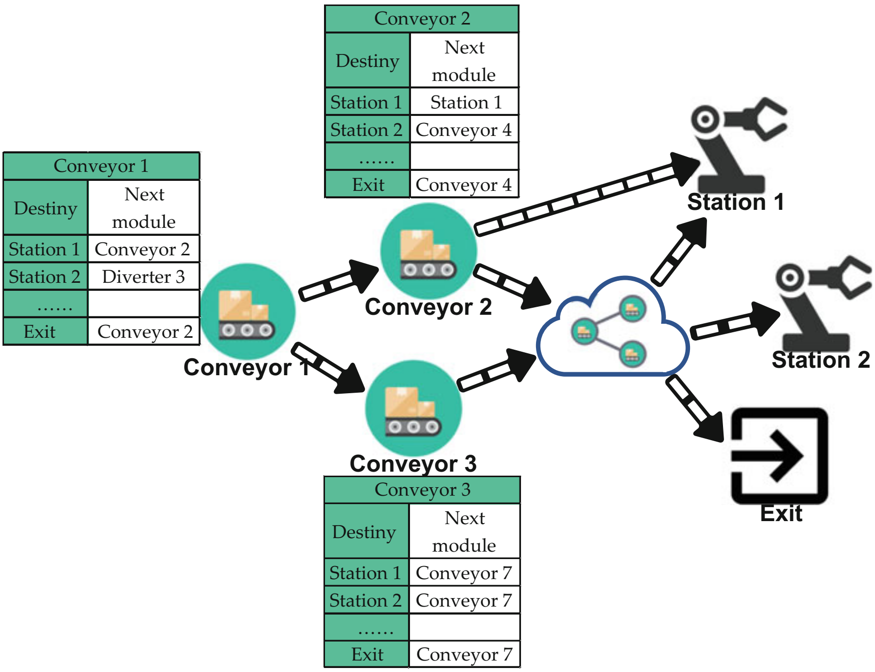
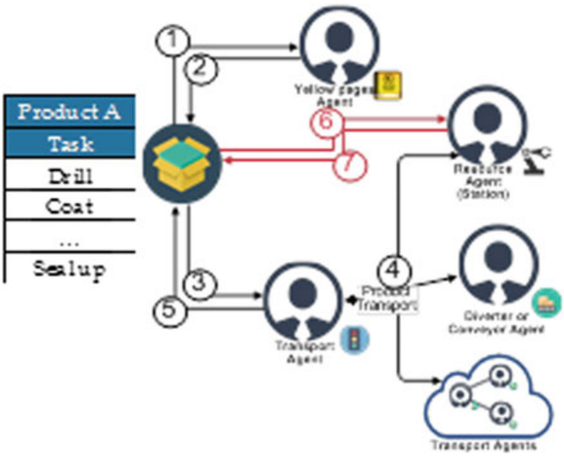


Fig. 2 Transport interaction

Fig. 3 Product interaction



transportation service, and it is up to the transport agent to move the product until its destination, taking care of all aspects and communication protocols between the conveyors and diverters. The last stage is the task execution, which is asked directly to the work station. This cyclic behaviour repeats until all tasks have been performed.

4 Proposed Framework

The Java language was chosen i.e., JADE described for the first time in [17], because the platform and the DDD Simulator from TTS Company [18] operate in a MAS environment.

4.1 Simulator

The simulator differs from many products in the market due to its capability of integration with a 3D environment. This feature makes it much more attractive and enlightening to study possibilities and new settings. The DDD Simulator is based on two major components, a visual environment and a control component responsible for the simulation itself. In order to synchronize the two entities, the simulator works based on the execution of events recorded in a temporal scale, and on the usage of a simulation clock. In the simulator, the two components work in separate layers preventing the logic modules to be directly coupled to graphics. However, it can still read and write to the state of any entity during the simulation; entities are abstracted by agents.

The codification of the modules in the DDD Simulator involves two steps. The first step consists in defining a default sequential code in Java which expresses the behaviour of a module type. In the second stage, the instantiation, based on XML (eXtensible Markup Language), is where each entity is defined, using the standard classes created earlier. As already described, the simulator uses a graphical representation. To create models, an external CAD program is used, that can export in the format ".wrl".

The simulator operation is based on coordinates and directions, so that a setup phase is required to define the referential axis. As expected, the behaviour of the modules must be handled by robotic rules; some expertise in kinematics is needed.

4.2 Multi-agent Systems

The product has a cyclic behaviour; the cycle is formed by four states: contextualize, search, transportation and execution. The informations about the product type are loaded; then the product searches for executing agents, requests transportation and finally asks for task execution. When all the tasks have been accomplished, the product asks for one more transportation service—this time to exit the system, the sink module. Figure 4 shows how the transport protocols work.

Four steps are included, coordinated by the transport agent. In the first stage, the transport agent finds out which is the next module. Based on its destination, it asks the present module where the product should go. The second stage is the allocation request in the next module. The purpose of the third stage is to guarantee that the send position of the product is correct and to free the current position. Lastly, the transport stage is where the transportation occurs.

The transport agent keeps a list with all the products and their stages in the simulation in order to ensure the system's integrity and the parallelism of processes.

5 NOVAFLEX Simulation

As a study case to implement the architecture, a real production cell was considered. The NOVAFLEX has some complexity both in number of modules and intersections, allowing the implementation of some dynamics in the system.

Two different types of product were created: product A with only one task and product B (darker in Fig. 5) with two tasks to perform. In the cell there are two workstations, and to each one a different skill was assigned.

The information about the links between transport modules, the input and output configurations and the station abilities are loaded from multiple files. The communication protocols between agents are based on FIPARequest and for the 3D models the SolidWorks (commercial 3D modelling software) was used.

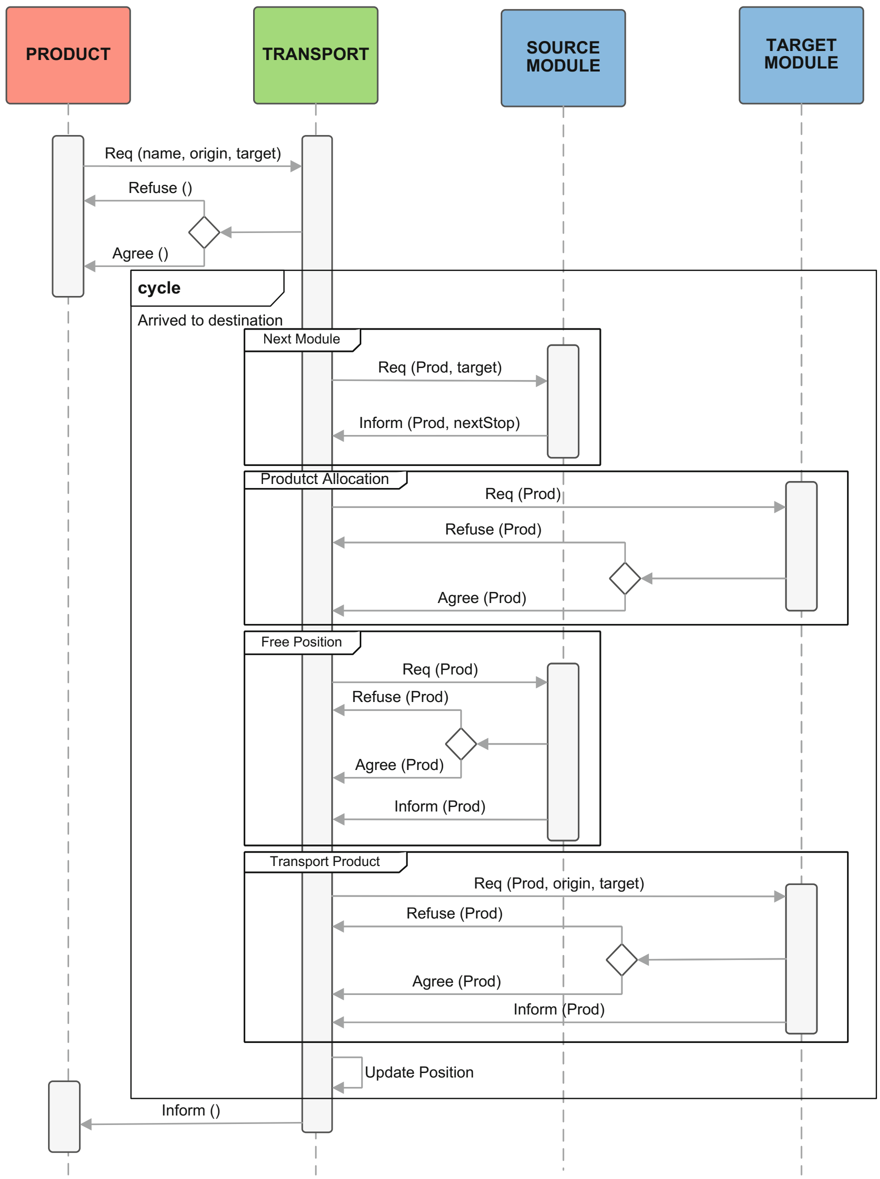


Fig. 4 Product routing execution

Figure 5 displays a scheme of the NOVAFLEX production cell and a small view of the simulation environment. With this scheme we can perceive the system arrangement and where the line components are located. In the small window showing the system's structure, one can observe the system at work and get an idea of the production cell's layout.

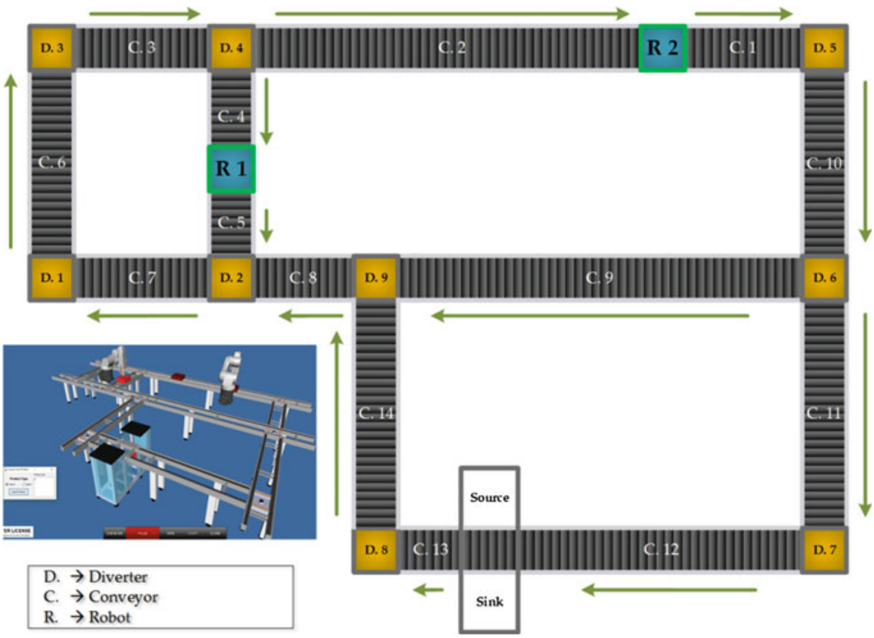


Fig. 5 NOVAFLEX representation and view of the simulation environment

Table 1 Instantiated agents

Agent type	Number
Yellow pages agent	1
Launching agent	1
Entry agent	1
Exit agent	1
Execution agent	2
Conveyor agent	14
Diverter agent	9
Transport agent	1
Entry and exit agent	1
Total	31

Table 1 defines the agent types and quantities. There is an agent that not yet presented—the "entry and exit agent". This agent was developed specifically for this layout, because the source and the sink are both in the same module and share the same access point in the system. This agent is very similar to the diverter one.

As the last feature implemented, a user interface was developed to endow the system with the capability of launch products in real time, allowing to test a product's entry after editing its features file, and check if it has the new imposed behaviours.

6 Conclusions

With the development and implementation of this architecture, the ability to interconnect existing simulation systems with adaptive architectures, like MAS, is proven. The development of new paradigms in the industry and the increasing need of simulate instead of experiments stimulates the creation of alternatives. This work proved the possibility of adopting some of the existing technologies for future solutions.

Despite the delays that message exchanges can add to production, the situation is better as compared to the interruption of production in the cell when a breakdown or error occurs, saving valuable time and money. Another advantage of these new paradigms is the facility to change the style of production or even the products without reconstructing the line, but just by adding or removing components to/from the system whenever it is necessary.

References

1. Mendes, J.M., Leitão, P., Colombo, A.W., Restivo, F. (2008). Service-oriented control architecture for reconfigurable production systems. IEEE International Conference on Industrial Informatics, pp. 744–749. doi:10.1109/INDIN.2008.4618201
2. Koren, Y., Heisel, U., Jovane, F., et al.: Reconfigurable manufacturing systems. CIRP Ann. Manuf. Technol. **48**, 527–540 (1999). doi:10.1016/S0007-8506(07)63232-6
3. Li, J., Dai, X., Meng, Z., et al.: Rapid design and reconfiguration of Petri net models for reconfigurable manufacturing cells with improved net rewriting systems and activity diagrams. Comput. Ind. Eng. **57**, 1431–1451 (2009). doi:10.1016/j.cie.2009.07.013
4. Cavalcante, A.L.D., Pereira, C.E., Barata, J.: Component-based approach to the development of self-X automation systems. IFAC Proc. **43**, 222–227 (2010). doi:10.3182/20100701-2-PT-4011.00039
5. Vrba, P., Tichy, P., Mařík, V.: Rockwell automation's holonic and multiagent control systems compendium, syst man, cybern., Part C. IEEE Trans. Appl. Rev. **41**, 14–30 (2011)
6. Leitão, P., Restivo, F.: ADACOR: a holonic architecture for agile and adaptive manufacturing control. Comput. Ind. **57**, 121–130 (2006). doi:10.1016/j.compind.2005.05.005
7. Van Brussel, H., Wyns, J., Valckenaers, P., et al.: Reference architecture for holonic manufacturing systems: PROSA. Comput. Ind. **37**, 255–274 (1998). doi:10.1016/S0166-3615(98)00102-X
8. Law, M., Kelton, W.D.: Simulation Modeling and Analysis. ACM Trans. Model Comput. Simul. **2**, 1370–1375 (2000). doi:10.1145/1667072.1667074
9. Drogoul, A., Ferber, J.: Multi-agent Simulation as a Tool for Modeling Societies: Application to Social Differentiation in Ant Colonies, pp. 2–23. Springer, Berlin, Heidelberg (1994)
10. Jacintho, L.F.O., Batista, A.F.M., Ruas, T.L. et al. (2010). An agent-based model for the spread of the Dengue fever. In: Proceedings of the 2010 Spring Simulation Multiconference SpringSim '10, p. 1. ACM Press, New York, USA
11. Vrba, P. (2003). MAST: manufacturing agent simulation tool. In: EFTA 2003, IEEE Conference on Emerging Technologies and Factory Automation Proceedings (Cat. No. 03TH8696), pp. 282–287. IEEE

12. Deng, Y., Cao, J., Wang, X. (2012). A distributed multi-agent simulation platform. In: The 15th IEEE International Conference on Computer Supported Cooperative Work in Design, pp. 64–71. IEEE
13. Thomas, A., Castagna, P., Pannequin, R. et al. (2008). Simulation for Product Driven Systems, pp. 221–255
14. Pannequin, R., Thomas, A.: Another interpretation of stigmergy for product-driven systems architecture. J. Intell. Manuf. **23**, 2587–2599 (2012). doi:10.1007/s10845-011-0588-3
15. Monostori, L., Vancza, J., Kumara, S.R.T.: Agent-based systems for manufacturing. CIRP Ann. Manuf. Technol. **55**, 697–720 (2006). doi:10.1016/j.cirp.2006.10.004
16. Rocha, A. (2013). An agent based architecture for material handling systems. M.Sc. Thesis, UNINOVA, Lisbon
17. Bellifemine, F., Poggi, A., Rimassa, G.: JADE—a FIPA-compliant agent framework. Conf. Pract. Appl. Intell. Agents Multi-agent Technol. **99**, 97–108 (1999). doi:10.1145/375735.376120
18. Technology Transfer System s.r.l. http://www.ttsnetwork.net/en/. Accessed 5 June 2016

An Evolvable and Adaptable Agent Based Smart Grid Management—A Simulation Environment

Andre Dionisio Rocha, Miguel Rodrigues and Jose Barata Oliveira

Abstract The liberalization of energy markets and shift towards intensive distributed use has provoked a change in the operating paradigm of electricity distribution networks. The continuity and the reliability of the distribution networks in the context of these new paradigms require structural and functional changes. The concept of Smart Grid allows the adaptation of distribution networks to the new context. This paper proposes an architecture organized to support the power transmission system, which auto management and control. The architecture is a multi-agent one, which uses the flow algorithm for a more efficient routing. Their main aim is stocking the entire electrical grid and distributing the energy effectively and efficiently between the various producers and consumers.

Keywords Smart grid · Multi-agent system · Energy transportation · Intelligent network · Edmonds-Karp · Auto-configuration

1 Introduction

Nowadays the production and distribution of energy are carried out from a central power producer (such as the hydroelectric or thermoelectric, among others). Energy is produced in these plants and delivered to consumers through distribution stations.

A.D. Rocha (✉) · M. Rodrigues · J. Barata Oliveira
Faculdade de Ciências e Tecnologia, Universidade Nova de Lisboa,
Costa da Caparica, Portugal
e-mail: andre.rocha@uninova.pt

M. Rodrigues
e-mail: ma.rodrigues@campus.fct.unl.pt

J. Barata Oliveira
e-mail: jab@uninova.pt

A.D. Rocha · J. Barata Oliveira
Centre for Technology and Systems, UNINOVA Institute,
Caparica, Portugal

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_37

From this fact, we clearly detect a manual process of energy distribution that, in the event of a failure or breakdown of a distribution depot, will cause a power outage to consumers for a shorter or longer period of time, affecting them in consequence.

These limitations led to the creation of a smart energy system called Smart Grid. The main advantage of this system is to automate the entire electrical network and enable communication between the components of the entire network.

A conventional grid presents problems particularly regarding the loss of energy, high investment infrastructures and a clear risk of major disruptions [1]. The smart grid can solve these problems by distributing the generated energy in a way that responds to the demand of energy and by the implementation sensors allowing the communication among consumers and producers which will reduce the number of interruptions [1]. The cost associated to this infrastructure is not as high as the one of electrical networks one.

2 Related Work

2.1 Power Grid

The grid is divided in two major areas: production and distribution. Nowadays, the largest percentage of energy production is carried out by hydroelectric, thermoelectric and nuclear centrals, while the remaining percentage is held by the conventional energy centrals. However, they all share the same objective: to produce enough energy to customers even though in different ways.

The distribution uses the network for taking the energy to final consumers (household, business or industrial).

2.1.1 Production of Energy Problem/Solution

Over the years, it became obvious that the conventional central producers of energy above mentioned are not sufficient and, therefore, imposed the necessity to explore clean energy such as renewable energies. The available renewable energies are: wind, hydro, biomass, solar or photovoltaic, waves and tides, and geothermal.

2.1.2 Smart Grid Case Study

14 pilot projects of smart grids were created and implemented in India to really assess the benefits of smart grids. They also allowed identifying the appropriate technologies and models for the smart grid. All these pilot projects have different features; however, the main objectives are indigenization of technology and development of scalable, regulated and replicable models.

The development of these pilot projects was successful. Preliminary studies show improvements in efficiency and reliability of the distribution network. The distributed computing and the communication components provide real time information and enable the near-instantaneous balance of supply and demand at the device level that was incorporated. By providing the smart grid with awareness and training, it can assure the restriction of consumption depending on load and the display of the consumption data. Consumers also benefit by monitoring and adjusting charges in addition to saving energy through demand response [2].

2.1.3 Conventional Power Grid Versus Smart Grid

There is an increasing need for energy production, forcing the management, optimization and improved efficiency of the entire production and distribution process.

The problem of energy production is solved with new energies since the conventional power generation is not sustainable at present.

The main problems of the traditional power grid are [3]:

- Long operating time, since in most cases present power generation stations have more than 35 years of functioning;
- Obsolete power stations, as more than half of the production of energy accrue from coal;
- Inefficiency, as only 35% of energy efficiency reaches consumers;
- Vulnerability, considering the susceptibility to interruptions in energy delivery, as demonstrated by the largest blackout in history affecting 620 million people.

A Smart Grid is defined as the combination of communication devices and systems integrated in the grid in order to make it intelligent. Creating a Smart Grid is imperative in our days, as a medium-long term solution to the problems that conventional networks are facing.

As noted earlier, the shortage of resources and the growing need for more energy demand the use of an intelligent system capable of checking whether more energy it is really needed. To make this possible, the Smart Grid has sensors and systems, communicating with each other, installed across it. Sensors indicate to the producer if the consumer needs more energy. The producer, after receiving the message from the consumer, will send more energy. The possibility of using any kind of energy makes the smart grid a reliable and sustainable network. Notice that the smart grid that supports this work was conceptualized considering the final consumer's welfare, offering him a maximum capacity power system with different and varied energy options.

Thus, the electric power industry is already in progress, which will revolutionize, we think, the way we deliver electricity to individual consumers. The Smart Grid component is a smart network that constantly analyses the energy consumption, offering convenient prices in real time in response to the hourly energy demand and

Table 1 Comparison between conventional power grid and smart grid [5]

Conventional power grid	Smart grid
Electromechanical	Digital
Unidirectional communication	Two-way communication
Centralized generation	Distributed generation
Few sensors	Sensors throughout the network
Manual monitoring	Automatic monitoring
Manual reset	Auto restoration
Failures and blackouts	Adaptable
Limited control	Full control
Few choices given to the customer	Several choices given to the customer

to the expansion of energy choices offered to customers in order to promote energy efficiency [4].

As can be seen in Table 1, there are important differences between the traditional electric grid and the Smart Grid.

3 Smart Grid System Architecture

The architecture has a central core responsible for the performance of the system. Besides the core, architecture components allow the interaction between the human operator and the system. The research carried out and reported in this paper focused on all the components of this core. The overall architecture is presented in Fig. 1.

In the initial phase, we just have to enter the available information about the electrical grid, the existing nodes, the links that are present and the energy values of each node.

The user can see in real time what is being done by the system, who is providing power (the Producer Agent), and who is receiving it (the Consumer Agent), and compare this information with the requirements (Table 2).

3.1 Smart Grid Optimization Algorithm

Two of the criteria for the choice of the algorithm to be used in this study are the importance of the energy handled and the capacity of energy delivery to the consumer regardless of the point of origin, so that consumers never run out of power.

These criteria have led to the choice of the Edmonds-Karp Maximum Flow algorithm which is a flow algorithm that can recalculate the entire power system as cited before. The Edmonds-Karp Maximum Flow algorithm was designed to optimize the power supplied to the consumer. The use of this algorithm in the architecture further optimizes the energy supply, providing correct management of

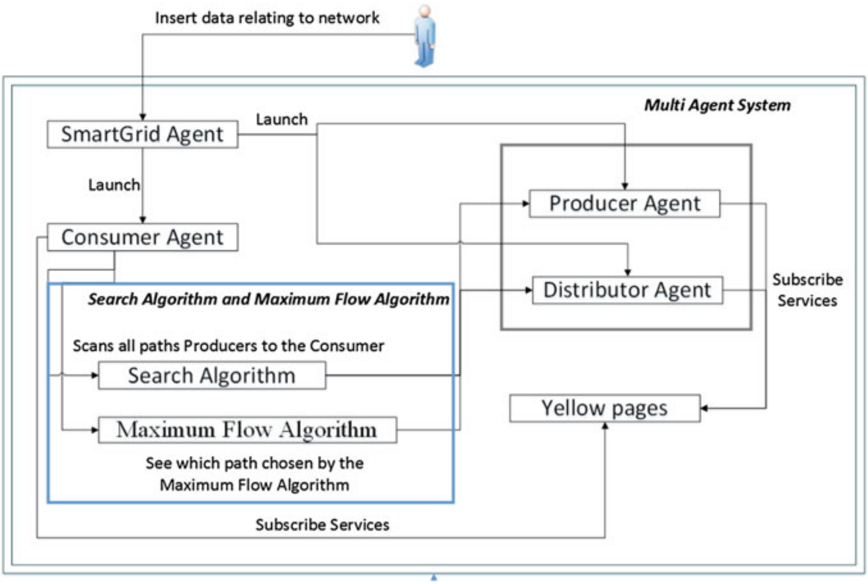


Fig. 1 System architecture

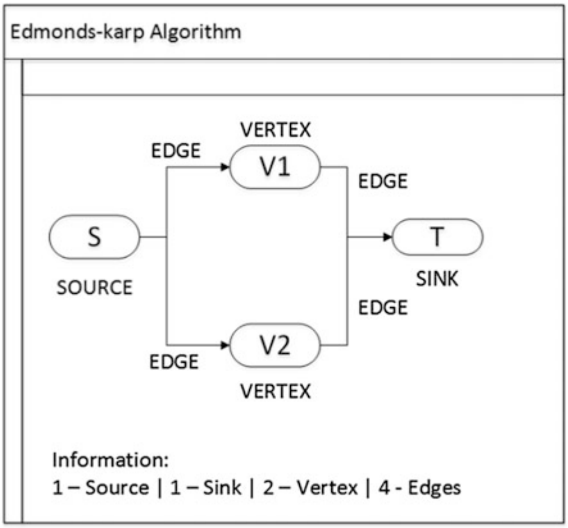
Table 2 Agents' descriptions and functionalities

Name of the agent	Function
Producer agent	Agent responsible for the producer, whose characteristics are the maximum energy and the nodes to which it is connected. Its function is the production of energy for consumers
Distributor agent	Agent responsible for the power transmission station, whose characteristics are the maximum energy and the nodes to which it is connected. Its function is to distribute energy to the consumer
Consumer agent	Agent responsible for the consumer of energy, whose characteristics are the maximum energy and the nodes to which it is connected. This agent only receives power

the sites to which the energy is sent. The algorithm is identical to the Ford–Fulkerson algorithm, except for the search system, because it uses a breadth-first search system [6]. The Edmonds-Karp Algorithm can be used in: (i) Graphs with multiple sources and/or multiple destinations; (ii) Maximum flows of water or electricity paths where it is intended to maximize the number of users [7]. The Edmonds-Karp Algorithm is used to:

- Find the minimum cut that divides the graph into two halves;
- Find the number of pathways that do not use the same edges;
- Find the greatest pairing in a bipartite graph.

Fig. 2 Edmonds-Karp algorithm



If the system has multiple origins and multiple destinations, the algorithm creates a super node "S" that connects all sources and a super "T" node that connects all destinations. Thus, we are able to know the amount of energy required to go from "S" to "T".

Now let us focus on the super nodes and on their meaning. The super node "S" means source and the super node "T" means sink as described in Fig. 2.

In the beginning we filled our network with producers, distributors and end consumers and their respective connections and energy values. After running the whole architecture, the algorithm will always examine whether there is a change in the network or not. In case of a change, the whole self-system recalculates and updates all the connections.

The architecture supports that, at any time of the implementation, a node may be removed or added to the system with different connections and with different energies. The system should have the ability to organize and update itself in order to maximize all network energy flows.

Nodes can be removed or added at a time prior to the system's discharge or during its execution. Because there are sensors monitoring each node, the system is easily capable of knowing whether there are any new nodes or if any node was removed.

3.2 Producer Agent and Distributor Agent

As the search and flows algorithms are operating in real time with respect to the behaviour of the consumer; the consumer agent is aware of all the possible paths of every producer and distributor to itself due to the search algorithm.

So when the consumer requests energy, it sends a message that includes a field that reflects the result of the Smart Grid Optimization Algorithm, knowing from that moment on the chosen path. To identify the path that was chosen, it is only necessary to verify if there is the AID of the producer or distributor in the message. If the answer is affirmative, the producer or the distributor sends an "agree" to the consumer. If the answer is negative, the producer or distributor sends a "refuse" to the consumer.

3.3 Search Algorithm

One of the advantages of implementing this algorithm allowing additional paths, besides the compatibility with our architecture, is the fact of being a lightweight algorithm—because of its constant use—and therefore not taking the system to use unnecessary resources. This algorithm is called Depth-First Search.

The paths are added at the time the user places all nodes with their respective connections in the algorithm. From that moment and when it already points at the Ticket Behaviour of the consumer, it begins the search.

4 Implementation and Execution

For the implementation of the proposed architecture, we used the Java programming language in conjunction with JADE [8]. The protocol used for the exchange of messages between agents was FIPA Request [8].

Figure 3 indicates that there are four important points. At the point marked as 1, the network is updated, by the use of the Edmonds-Karp Maximum Flow algorithm as a decision method. At the point marked as 2, the consumer communicates with producers and distributors, conveying a message that will contain the amount of energy that it needs. At the point 3, there are the producers and distributors. This is where the flow algorithm decides who will provide and who will distribute energy. Some agents respond "agree" and the remaining answer "refuse". Finally, at the last point of the sequence diagram, producers and distributors send a message that informs the consumer that they are supplying power.

There is a very important feature to keep in mind during the implementation of the behaviour that supports the flow calculation: due to the high number of communications and requests to upgrade the network, which often happen, this update should not block the remaining processing.

In order to avoid the blocking of the agent by this behaviour, a *TickerBehaviour* with a low range of tickers was used. To determine whether the implementation of the Smart Grid Optimization Algorithm was in progress, we chose to use the run variable.

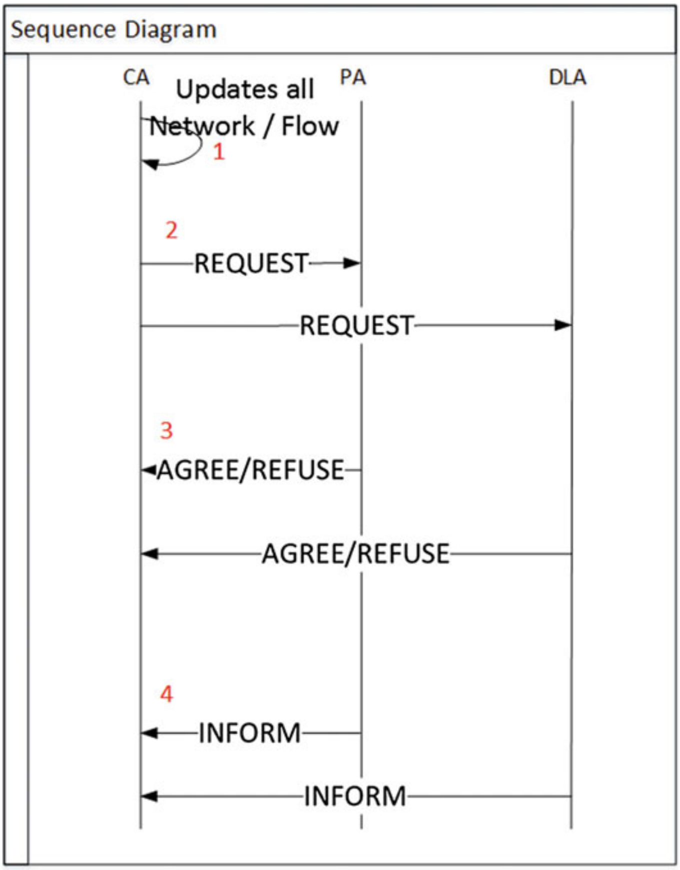


Fig. 3 Execution sequence diagram

These behaviours are triggered as soon as the agent begins its execution, with a distance of 1000 ms between triggers. This value was adjusted to be short enough for a quick update of the network without impairing the performance of the agent's execution. In order to make possible the network energy flow during the execution of the Edmonds-Karp Algorithm, this algorithm is only updated at the end of the calculation, leaving the previous version as the basis for decision making. In the event of a blockage, which is called intermediate state, the update will be based on the previous version.

Shortly, when a consumer is added to the network, it immediately proceeds to further communications with other nodes. Then, every producer and distributor will analyse the results of the Smart Grid Optimization Algorithm to know if they will provide and distribute energy.

5 Results

In order to be able to test all possible situations we used a virtual environment. We performed three simulations as tests, each having more nodes than the former. The objective of holding three simulations is to verify if the system is scalable.

All changes made in the three simulations were performed while the system was running, which allowed us to analyse their changes and the behaviours of the architecture. All simulations to which were added or removed nodes were tested thirty times.

In Fig. 4, the vertical axis is the average time in milliseconds (ms) it takes the system to update the entire network. The horizontal axis represents the simulations. The simulations are divided into two charts. In the first chart it is possible to observe the simulations from which nodes were removed and in the second one the simulation to which nodes were added.

Summing up, there were performed enough simulations and tests in order to find out how the whole system reacts to the changes introduced in the update times of the entire electric grid. When analysing the simulations, we found out that the updates last milliseconds, which is a good response to the simulations.

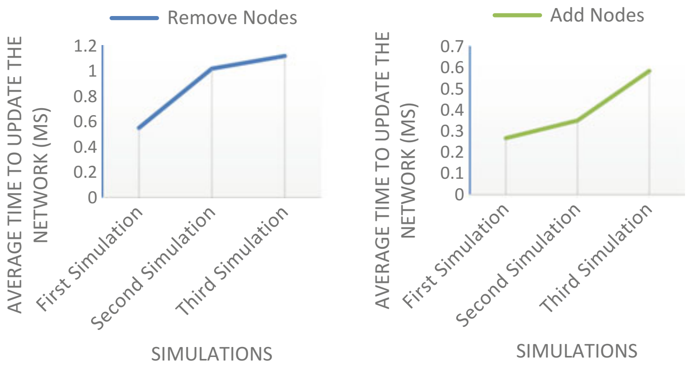


Fig. 4 Final data corresponding to the three simulations

6 Conclusion

With the work done it was proven that a self-organizing multi-agent architecture can be used in power distribution context.

The results are expected and it was concluded that they get a good response to simulated problems, being very efficient on the management of the entire power grid. This solution challenges and makes the concept of Smart Grid as a key one, because it was proved that it is quite advantageous and has a huge potential for management.

Although they were performed in a virtual environment, we conducted sufficient tests both in terms of simulations and of testing, which allowed us to see the whole system reacting to the changes introduced in their respective times of updating the entire electrical network. While analysing testing and simulations, it was concluded that the system updates in milliseconds, with a good response to the performed simulations.

References

1. Peng, Q., and Low, S. H.: Distributed algorithm for optimal power flow on a radial network, 53rd IEEE Conference on decision and control, pp. 167–172 (2014)
2. Kappagantu, R., Senn, S., Mahesh, M., Daniel, S.A.: Smart grid implementation in India—a case study of Puducherry pilot. Int. J. Eng. Sci. Technol. **7**, 94–101 (2015)
3. Joy, J., Jasmin, D.E.A., John, V.R.: Challenges of smart grid. Int. J. Adv. Res. Electr. Electron. Instrum. Eng. **2**, 976–981 (2013)
4. Keyhani, A., Marwali, M.: Smart Power Grids. Springer (2011)
5. Fang, X., et al.: Smart grid—the new and improved power grid: a survey, IEEE communications surveys & tutorials. **14**(4), 944–980 (2012)
6. Kempe, D.: Lecture Notes on the Edmonds/Karp algorithm—CS 570 (2004)
7. Ribeiro, P.: Fluxo Máximo, Faculdade de Ciências da Universidade do Porto (2015)
8. Bellifemine, F., Caire, G., Greenwood, D.: Developing Multi-agent Systems with JADE. Wiley (2004)

Validation of a Holonic Controller for a Modular Conveyor System Using an Object-Oriented Simulation Framework

Karel Kruger and Anton Basson

Abstract This paper presents the use of a commercial, object-oriented simulation framework to facilitate the validation process of a holonic controller. The case study involves a holonic controller for a modular conveyor system. The holonic control is implemented using Erlang and thus exploits the scalability and concurrency benefits it has to offer—the simulation framework and necessary interfacing was then done to accommodate the nature of the implementation. The simulation model interface, incorporating TCP communication and Windows Communication Foundation services, was designed to mirror that of the conveyor hardware to allow for the seamless interchange between emulated and real operation.

Keywords Emulation · Holonic manufacturing system · Reconfigurable manufacturing system · Manufacturing execution system

1 Introduction

Modern markets have enforced a new set of requirements on the manufacturing industry: increased adaptability to accommodate market trends and fluctuations, shorter lead times, increased product variation and customizability [1, 2]. This was already anticipated more than two decades ago [3] and, since then, research has been done on many aspects concerning the transformation of modern manufacturing systems.

A popular approach used in several studies and implementations is that of holonic systems. This idea, originally presented by Koestler [4], can be understood within the manufacturing system environment as the division of a system into autonomous, cooperating entities which work together to accomplish the system functions [5].

K. Kruger (✉) · A. Basson
Department of Mechanical and Mechatronic Engineering, Stellenbosch University, Stellenbosch 7600, South Africa
e-mail: kkruger@sun.ac.za

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9_38

The holonic approach to manufacturing systems have provided many benefits—enhanced system scalability, customizability and fault-tolerance, which lead to increased system reconfigurability and reliability, and reduced complexity [6]. As can be expected, holonic systems have encountered some challenges, of which the most relevant to this paper is that of *system validation*.

The validation of manufacturing systems can be understood as the means to test the system and obtain assurance that the system functions as desired. The validation of holonic systems can be difficult since the system functions are distributed over several processes and/or controllers. Since holonic systems are based on inter-holon cooperation and are often distributed, it becomes harder to validate a control application [7].

Regarding the validation of holonic systems, there has been some research done into tools to aid in the endeavour. One example is that of the Multi-agent Simulation Tool (MAST) presented in [8]. MAST uses a multi-agent system to control a graphical simulation. A discussion of the use of simulation in holonic systems is presented in [9].

There are several simulation software packages available which have been used in many different fields and applications. One such package is Simio, a modelling framework based on object-oriented principles [10]. The inherent architecture of Simio fits well with the idea of holonic systems and is discussed further in Sect. 2.

This paper describes the use of the Simio simulation framework to validate a holonic control implementation through hardware emulation. The hardware emulation is configured to extend the holonic principles of the higher level control architecture by facilitating control distribution and modularity within the simulation framework.

For a case study implementation the control and emulation of a modular, palletized conveyor system is used. Conveyor systems entail complex interactions and logic, and require significant programming and testing efforts during commissioning and reconfiguration activities. These challenges motivate the need for a simulation tool to validate the routing and control logic—especially for systems large in size and complexity—to decrease lead and ramp-up times.

This paper presents a discussion of the holonic architecture, with focus on the inclusion and use of the Simio emulation model to facilitate the control validation. The validation process and the useful tools provided by Simio are also discussed.

2 Simio Modelling Framework

[11] presents Simio as a graphical modelling framework which implements object-oriented principles in both the simulation logic programming and the construction of simulation models. Simio provides the developer with the infrastructure to build up a simulation model with customizable objects. The behaviour of the Simio objects can be customized by adding processes that define the execution logic. Processes are sequences of steps that are executed in a thread of execution.

Steps perform some specified function, such as handling or triggering events that influence the state of the object.

Furthermore, Simio is programmed in Microsoft Visual C#; this opens the framework for incorporation with powerful tools like .NET and Windows Communication Foundation (WCF). Simio also explicitly provides an API for C#, which provides several useful functions for the construction and running of Simio models.

3 Holonic Cell Control

At cell control level, a holonic architecture was implemented in accordance with the PROSA [12] reference architecture. As is clear from Fig. 1, the modular conveyor system is represented as a Resource holon at cell control level. The conveyor holon is comprised of three components: High Level Control (HLC), Low Level Control (LLC) and the physical hardware. The HLC component represents the holon in the virtual cell control environment. This component handles all communication with the holons in the cell, such as service bookings, service cancellations, etc. The HLC

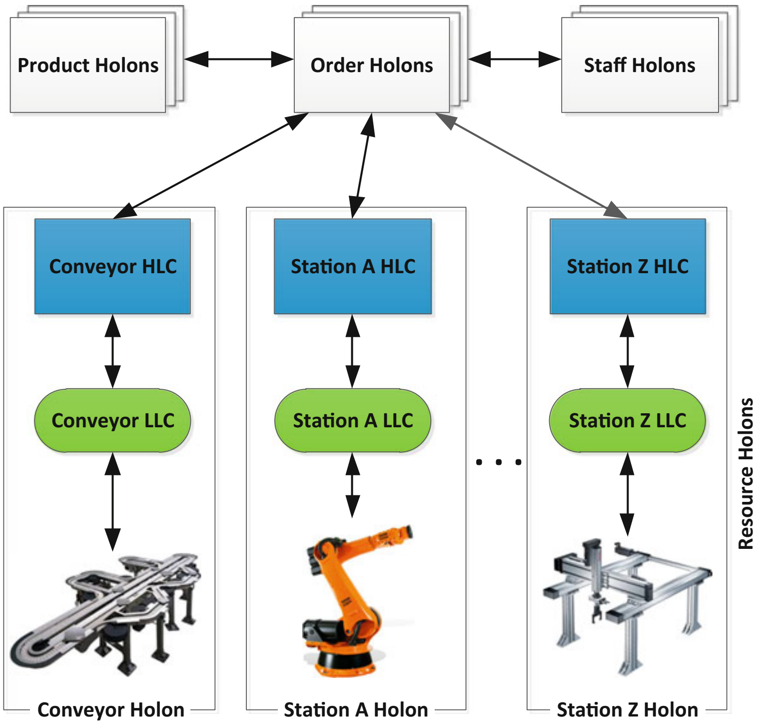


Fig. 1 Manufacturing cell control architecture

activates execution of a desired service through communication with the LLC component. The LLC has interfaces with the physical actuators and sensors of the hardware and can thus coordinate the sequence of hardware actions required to perform a desired service.

4 Conveyor Holon

4.1 Holonic Controller

The Conveyor holon component which forms part of the PROSA cell control application is implemented using Erlang. Erlang is a functional programming language with inherently strong scalability, concurrency and fault-tolerance characteristics.

The Conveyor holon HLC implementation was aimed at exploiting the modularity and scalability advantages that Erlang offers. The HLC component is itself implemented as a collection of holons which encapsulate, and through cooperation, constitute the Conveyor holon functionality. A detailed description is given in [13].

4.2 Interpreter

The Interpreter program provides a link between the holonic controller and the emulation model. The Interpreter maintains an interface to the holonic controller that is similar to that of the low level PLCs of the conveyor (shown in Fig. 2); this interface facilitates TCP communication over multiple ports (the same number as the number of PLCs used in real operation). The Interpreter program creates a link to the emulation model by making use of the Windows Communication Foundation (WCF) services. The two mentioned interfaces are discussed in the following sections.

TCP Communication with HLC: As mentioned, the Interpreter program facilitates TCP communication which emulates the communication to the PLCs that control the conveyor hardware. For the Erlang-based holonic controller programs there is no difference in the communication whether real operation or emulation is performed.

The Interpreter program maintains a port for every PLC that is installed on the conveyor. To communicate the information received from the holonic controller to the emulation model, the Interpreter program parses XML encoded strings received over the TCP ports. The parsing extracts the critical information that must be communicated to the emulation model. In the same way the PLCs will provide notifications based on the feedback of their connected sensors, the emulation model provides feedback based on events in the emulation model. This feedback

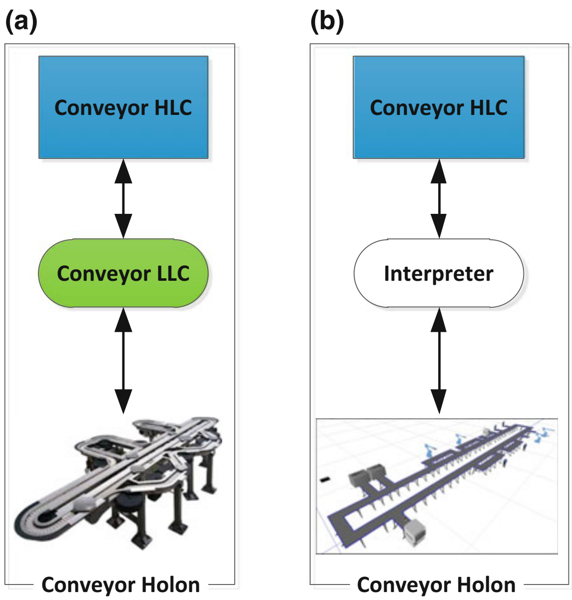


Fig. 2 The Conveyor holon architecture for **a** real and **b** simulated operation

information is then encoded into an XML string and sent via the TCP port to the holonic controller.

WCF Interface with Emulation Objects: In order to interface the Interpreter C# program with the Simio objects during runtime, WCF was chosen to provide the infrastructure for communication. WCF is a software development kit for implementing services on the Windows operating system. Services, in this case, refer to units of functionality and coincide with those used in service-orientation principles [14].

In the Interpreter program, WCF is used to host a service that exposes both events and event handlers. The service can be accessed by clients through bindings, which are configured in a service contract. The service contract allows the various EventInterface step object instances in Simio (discussed in Sect. 4.3), which form part of the model processes, to trigger an event that will be handled by the Interpreter program. The contract also allows for the Interpreter program to trigger an event which is handled by the EventInterface objects.

Using the WCF service, the process step of each transfer node in the Simio model triggers a "notification" event that is handled by the Interpreter. This event is triggered whenever a carrier arrives at a transfer node and the carrier name is supplied as an event data parameter. This notification is forwarded as an XML string to the HLC.

With the notification received, the HLC must determine to which transfer node the carrier in the model must be directed next. This information is then sent to the Interpreter program, where an event is triggered (the name of the next transfer node is specified in the event information). This event is then handled by the process step of the relevant transfer node and the extracted information is used to direct the carrier in the desired direction.

4.3 Conveyor Model

The Simio emulation model for the conveyor is shown in Fig. 3. As will be explained in the following sections, the model is constructed using standard Simio objects and the logic is implemented through Simio processes with customized process steps.

Simio Model: The conveyor system is modelled as a network of nodes linked by transitions (also referred to as paths or links). Nodes are points on the conveyor where two or more transitions meet; on the physical system, nodes are implemented by stop gates (usually in combination with lifting stations or transverse conveyors, and are equipped with RFID readers), as is shown in Fig. 4. These physical entities can be modelled in Simio by transfer node objects for node entities and either

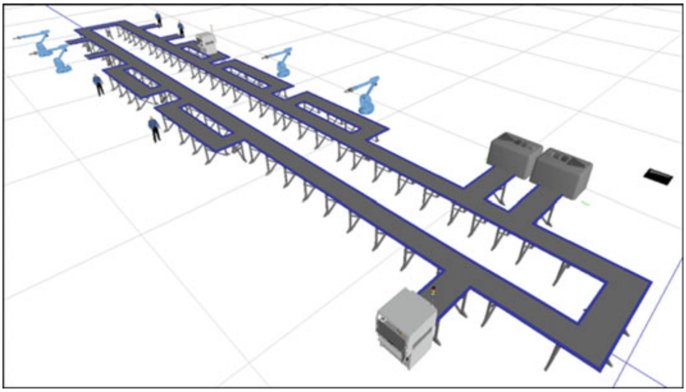


Fig. 3 Conveyor emulation model

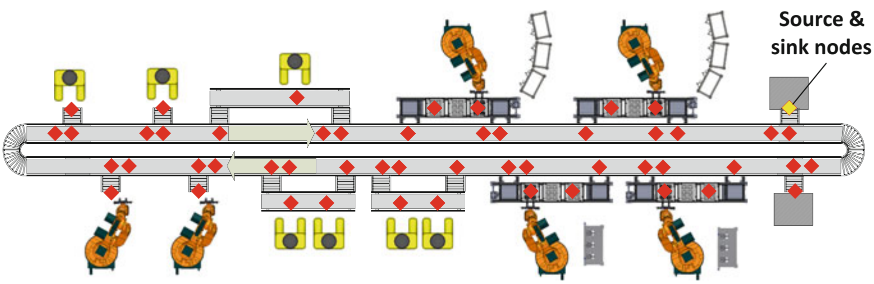


Fig. 4 Schematic of the conveyor with all nodes indicated

conveyor objects (for one-directional transitions) or path objects (for bi-directional transitions).

The model of the conveyor also includes means of carrier storage (i.e. a mechanism to unload or store carriers). For the conveyor used in this case study, this function is performed by an automated carrier magazine. The same functionality can be achieved in the emulation model by using the source and sink standard Simio objects. The source object unloads carriers for the conveyor and the sink model stores carriers.

Simio Processes: The behaviour of the standard Simio transfer node objects can be customized by adding processes to the object instance. Processes are constructed through a specified sequential execution of functional steps. The processes are executed when specific events occur—in the transfer node case, when an entity (carrier) enters the transfer node and the "entered" event is triggered. The process executed when the "entered" event is triggered is shown in Fig. 5.

Figure 5 shows the process which is executed by transfer node objects when they are entered by an entity object. When an entity enters a transfer node, the first step executed in the process is NotifyReady. During this step, the notification event is triggered (this is handled by the Interpreter). The step subscribes then to the event the Interpreter will trigger upon receiving a message from HLC specifying the next transfer node. When the event is triggered and handled by the NotifyReady step, the process enters an Execute step; this step then calls the SetNode process, which uses the obtained event information to specify the node to where the entity must be directed.

Conveyor Emulation: During operation, the first task for the conveyor will be to unload a carrier from storage onto the conveyor; this unload task will be initiated by the controller. This unloaded carrier will be moved to some location, as controlled by a corresponding process in the holonic controller. After unloading, the carrier will arrive at the first transfer node and a notification will be sent to the Interpreter program, where it will be encoded into an XML string and forwarded to

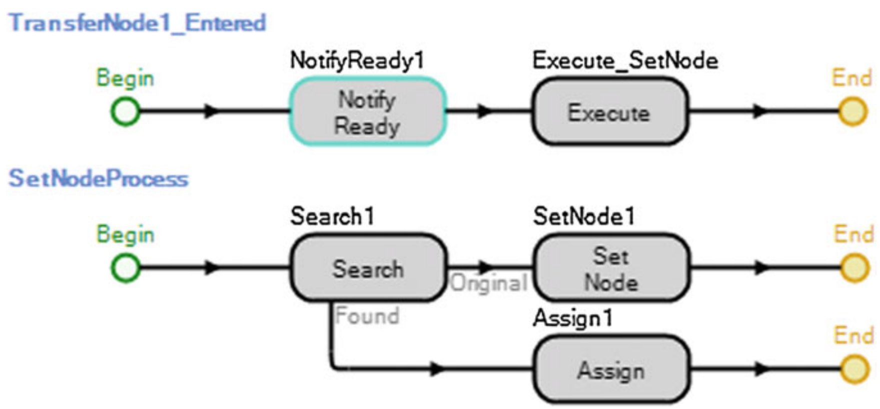


Fig. 5 Simio processes for conveyor node objects

the holonic controller. The controlling process can then react to this notification and send an XML string to the Interpreter which specifies the next transfer node to where the carrier must be moved. The transfer node currently occupied waits for this command to be received from the Interpreter and subsequently directs the carrier on the desired path towards the desired next transfer node.

5 Control Validation

Validation, in this context, refers to the assurance that the holonic control application is performing the system functions as desired. The emulation of the conveyor system using a Simio model offers several advantages for the validation of the control logic.

An important advantage is the ability to perform long-running emulations in short times, as the execution speed of the emulation model can be controlled. Also, the use of Simio emulation allows for testing of specific production scenarios; this is especially useful at the event of reconfiguration. The initial conditions of the emulation environment can be customized to adhere to some HLC scenario, i.e. the conveyor emulation can begin with a “clean” startup or with carriers in predefined locations. The combination of Erlang and Simio simplifies this process—the stateless nature of Erlang programs allows for the various system holons to be launched with specific state data, while the “open” nature of Simio, together with the C# API functions, provide the infrastructure to create custom scenarios.

Even though the research community is still striving toward standardized benchmarks for the performance of holonic systems, the collection of performance data is critical for the validation process. Simio incorporates the functionality to record and process diagnostic data from a performed emulation; this can include information on travelled carrier paths and times, collision detection and time-out errors.

Two quantitative measurements that are easy to obtain through this presented Erlang-Simio application are throughput and resource utilization. For the conveyor, Simio reports data on each conveyor segment—e.g. the throughput, maximum and minimum carriers present at a given time and average time spent by carriers in that segment. When the other resource holons are integrated in the Simio emulation, Simio can report the time a specific resource was used during the total emulation time.

6 Conclusion and Future Work

This paper presented the use of simulation software in the validation of a holonic control implementation. The case study is focused on the validation of an Erlang based holonic controller for a modular conveyor system, where Simio is used to provide a hardware emulation model.

To create an interface between the holonic controller and the emulation model an Interpreter program was developed. The Interpreter program maintains an interface which emulates that of the physical conveyor system by handling TCP communication on multiple network sockets. The Interpreter also provides the means for communication with the emulation model using several instances of WCF services.

With further enhancement, the use of the Simio emulation model could prove to be valuable in the control validation process. The emulation of customized production scenarios is a great advantage in the context of reconfigurable manufacturing systems. The object-oriented nature of Simio also strongly resembles the principles of holonic systems and it thus interfaces well with higher level holonic control implementations.

Future work will entail the enhancement of the Simio emulation, with particular focus on enriched information flow between the control and emulation levels, and also the incorporation of measurement tools within Simio to capture and interpret diagnostic information from emulation experiments. Further work will be done on the construction of Simio models to accurately represent the real system components.

References

1. Bi, Z.M., Lang, S.Y.T., Shen, W., Wang, L.: Reconfigurable manufacturing systems: the state of the art. Int. J. Prod. Res. **46**(4), 967–992 (2008)
2. Koren, Y., Shpitalni, M.: Design of reconfigurable manufacturing systems. J. Manuf. Syst. **29**, 130–141 (2010)
3. Koren, Y., Heisel, U., Jovane, F., Moriwaki, T., Pritschow, G., Ulsoy, G., Van Brussel, H.: Reconfigurable manufacturing systems. Ann. CIRP **48**(2), 527–540 (1999)
4. Koestler, A.: The Ghost in the Machine. Arkana Books, London (1967)
5. Paolucci, M., Sacile, R.: Agent-Based Manufacturing and Control Systems. CRC Press, London (2005)
6. Kotak, D., Wu, S., Fleetwood, M., Tamoto, H.: Agent-based holonic design and operations environment for distributed manufacturing. Comput. Ind. **52**, 95–108 (2003)
7. Leitao, P., Restivo, F.J.: ADACOR: a holonic architecture for agile and adaptive manufacturing control. Comput. Ind. **57**(2), 121–130 (2006)
8. Vrba, P.: MAST: manufacturing agent simulation tool. In: Proceedings of the IEEE Conference on Emergent Technology for Factory Automation, vol. 1 (2003)
9. Valckenaers, P., van Brussel, H.: Design for the Unexpected, 1st edition, Butterworth-Heinemann, ISBN: 9780128036624 (2015)
10. Pegden, C.D.: Simio: A new simulation system based on intelligent objects. In: Proceedings of the 2007 Winter Simulation Conference, pp. 2294–2300 (2007)
11. Pegden, C.D.: Introduction to Simio, Proceedings of the 2008 Winter Simulation Conference, pp. 229–235 (2008)
12. Van Brussel, H., Wyns, J., Valckenaers, P., Bongaerts, L., Peeters, P.: Reference architecture for holonic manufacturing systems: PROSA. Comp. Ind. **37**, 255–274 (1998)
13. Kruger, K., Basson, A.H.: Erlang-based Holonic Controller for a Modular Conveyor System, SOHOMA'16 Workshop, accepted for publication in Springer SCI series
14. Lowy, J., Montgomery, M.: Programming WCF Services, 4th edition, O'Reilly Media (2015)

Author Index

T. Borangiu et al. (eds.), *Service Orientation in Holonic and Multi-Agent Manufacturing*, Studies in Computational Intelligence 694,
DOI 10.1007/978-3-319-51100-9