Manufacturing Letters 15 (2018) 18–21

Contents lists available at ScienceDirect

Manufacturing Letters

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

A software-defined framework for the integrated management of smart manufacturing systems

Felipe Lopez [a,*], Yuru Shao [b], Z. Morley Mao [b], James Moyne [a], Kira Barton [a], Dawn Tilbury [a]

[a] Department of Mechanical Engineering, University of Michigan, United States
[b] Department of Electrical Engineering and Computer Science, University of Michigan, United States

ARTICLE INFO

Article history:
Received 6 December 2017
Received in revised form 22 December 2017
Accepted 22 December 2017
Available online 24 December 2017

Keywords:
Smart manufacturing
Industry 4.0
Systems integration

ABSTRACT

This paper introduces software-defined control (SDC) as a framework that enables integrated and programmatic management of smart manufacturing systems. SDC consolidates information from the production and enterprise levels in a central controller that monitors performance and detects changing conditions. The integrated view of the system provided by the central controller supports the development of applications, which supply the central controller with new information and reconfiguration recommendations. SDC is designed to be scalable and compatible with current automation technologies. A simulation shows that the incorporation of plant-floor information in management decisions, as supported by SDC, has the potential to improve profitability.

1. Introduction

Smart manufacturing (SM) systems have been defined as “fully-integrated, collaborative manufacturing systems that respond in real time to meet changing demands and conditions in the factory, in the supply network, and in customer needs” [1]. The vision of SM requires a higher degree of integration and agility than is allowed by current industrial automation technologies. ISA-95, the standard for integration in automation, supports only limited sharing of information between process control, operations management, and business planning applications, allowing data silos that prevent integrated management of the system. Moreover, automation equipment (e.g., PLCs, CNCs, robots) used on the plant floor lacks the agility required to respond in real time to changing conditions; programs of typical automation equipment are thoroughly validated before deployment and are expected to operate without major modifications. Integration and agility in industrial automation must improve to make SM a reality.

Significant work has been dedicated to increasing the integration of plant-level and enterprise networks. This approach – referred to as IT/OT integration – seeks to make plant-level data accessible for use across the organization. Once available, plant-level data could be used to increase awareness of plant-floor equipment [2].

In terms of agility, agent-based approaches [3] support fast reactions to disturbances on the factory floor but they have not been fully adopted by industry due to difficulties in their integration with automation technologies and the lack of guarantees of performance towards global objectives. An alternative is found in operations management solutions, such as manufacturing execution systems (MES), that can make minor changes to the configuration of the factory floor through work instructions sent to automation equipment. The benefit of this approach is its compatibility with existing automation technologies. Some drawbacks are that new configurations are limited to predefined modes of operation and that decisions are made only with the information available to the operations management solutions, which have limited information of the plant floor.

There is a gap in the SM literature between IT/OT integration for improved decision making and structured approaches to implement those decisions on the plant floor. As a solution, we propose software-defined control (SDC), an approach that (1) consolidates control and enterprise data to provide a central controller with a global view of the system, and (2) allows the central controller to assist operations management solutions with global information and reconfiguration recommendations that can be rapidly deployed to the plant floor. SDC, illustrated at a high-level in Fig. 1, is inspired by software-defined networking [4], an approach that abstracts the control functionality of switches to simplify the development of management applications for computer networks. In SDC, the functionalities of automation devices are abstracted to support the agile and programmatic management of the plant floor

* Corresponding author.
E-mail address: lopezfe@umich.edu (F. Lopez).

https://doi.org/10.1016/j.mfglet.2017.12.015

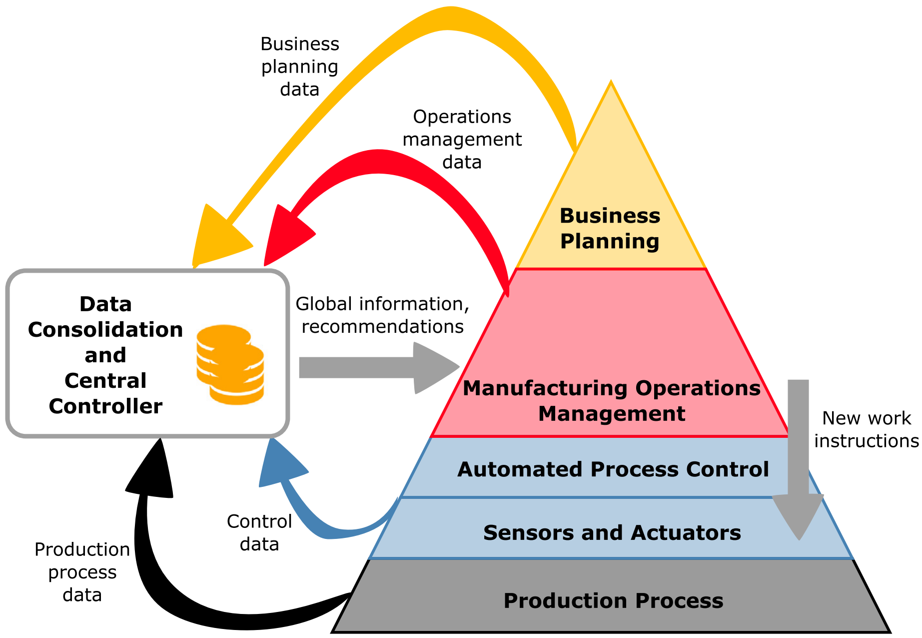

Fig. 1. Consolidation of data and information in a central controller that collaborates with operations management solutions for the rapid reconfiguration of the factory floor.

using information from the entire manufacturing system. Similar approaches have been proposed in manufacturing in recent years [5–7].

2. Software-defined control framework

SDC was designed to meet three requirements:

(a) SDC must coexist with current automation technologies – including the ISA-95 architecture [8], MES and ERP (enterprise resource & planning) software – in a strictly value-add and advisory manner.
(b) The approach must be logically-centralized to support system monitoring and decision making based on the performance of the entire system.
(c) The approach must accommodate various applications (e.g., anomaly detection [9], and dynamic scheduling) that will leverage the global view of the system and provide capabilities to the central controller.

SDC has three main components: a set of centralized databases, a central controller, and a set of applications, shown in Fig. 2. Databases consolidate data from the plant floor, operations management, and business level solutions. Due to the volume of manufacturing data that could be acquired, the databases could be hosted in the cloud and leverage big data technologies. The central controller – the brain of the SDC framework – uses consolidated data to maintain a global view of the SM system. A set of applications query manufacturing data from the central controller to support analyses of the entire system, and return results to the central controller to assist in decision making. The central controller passes information on system performance and changes detected in the system (e.g., machine health is degrading) to the MES to notify operators and plant managers. Additionally, if an alternative configuration of the plant floor has been found to be better suited for the current needs of the system, the central controller passes a recommendation for reconfiguration to the MES, which, if approved, is then deployed to the plant floor as work instructions.

SDC applications perform user-defined tasks that support the operation of the central controller. Communication between applications and the central controller is bidirectional. Applications employ a set of services – designed to be general enough to be reused in several applications – to query and manipulate data from the central controller, and provide it with information to improve its decision making. The separation of the central controller and applications allows the programmatic management of the manufacturing system as new algorithms, models, and optimization techniques may be implemented without having to make changes to the central controller.

The central controller is the software stack that supports the logically-centralized management of automation equipment by suggesting reconfigurations to the MES based on a global view of the system. The central controller is equipped with three components: a catalog of services, a decision maker, and a simulation platform. The decision maker determines the information and recommendations to pass to the MES based on rule-based methods (using predefined rules or discrete-event models) or simulation. The latter requires the use of the simulation platform where a digital twin of the system can be used to support the identification of changing conditions and to predict the effect of alternative plant configurations before their deployment.

The operation of the central controller is supported by the interfaces[1] shown in Fig. 3. The central controller communicates with automation equipment through the southbound interface. Unlike other software-defined approaches where the central controller is allowed to directly reconfigure control devices, the southbound interface in SDC is only used to read and provide context to plant floor data. The southbound interface supports popular communication protocols and encodes data with a unified, standardized protocol (e.g., OPC). Meanwhile, the northbound interface enables communication between SDC applications and the central controller. The northbound interface simplifies the development of applications using REST APIs [10], which allow applications to be dynamically deployed. The central controller also has an eastbound interface that serves three purposes.

[1] The names of the interfaces are consistent with other software-defined frameworks but some functionalities have been altered to comply with ISA-95.