Hindawi Publishing Corporation
Advances in Decision Sciences
Volume 2012, Article ID 428060, 11 pages
doi:10.1155/2012/428060

Research Article

Virtual Commissioning of an Assembly Cell with Cooperating Robots

S. Makris, G. Michalos, and G. Chryssolouris

Laboratory for Manufacturing Systems and Automation,
Department of Mechanical Engineering and Aeronautics, University of Patras, 26500 Rio, Greece

Correspondence should be addressed to G. Chryssolouris, xrisol@lms.mech.upatras.gr

Received 1 March 2012; Revised 8 July 2012; Accepted 25 August 2012

Academic Editor: Fumihiko Kimura

The Virtual Commissioning (VC) technology is the latest trend in automotive assembly which, among other benefits, promises a more efficient handling of the complexity in assembly systems, a great reduction in the system's ramp-up time, and a resulting shortening of the product's time to market. This paper presents the application of VC techniques to the case of an industrial robotic cell, involving cooperating robots. The complete workflow of the virtual validation of the cell is presented, and the implementation requirements are discussed. Based on the findings, the outlook and challenges for the wide-range adoption of VC technologies in large-scale assembly systems are provided.

1. Introduction

As product life cycles are reduced in the continuously changing marketplace, modern manufacturing systems should have sufficient responsiveness to adapt their behaviours efficiently to a wide range of circumstances [1]. In this context, one of the main challenges that modern assembly systems are faced with, is the cost-driven demand for faster and more secure ramp-up processes. This goal is however underpinned by the constantly rising number of rampups, due to enhanced innovations and the increasing market launches of new products and product variants. The current trend followed by the automotive original equipment manufacturers (OEMs), as highlighted by Bär [2], is the adoption of product, equipment, and process standardization. Nevertheless, this standardization is not by itself capable of guaranteeing that the designed assembly and production systems will be fully operational after their physical deployment. The complexity and diversity of the different line components, in terms of control systems and communication channels, requires a great amount of time for onsite setup, testing, and validation of the assembly equipment. This in turn, is translated into production system downtime and the respective opportunity costs that follow it. Digital simulation of the assembly process has emerged over the last decades as a means of partially handling the validation of such systems prior to their installation. IT systems have been over the past years an evolutionary technology, forwarding the concepts of digital manufacturing. These systems are based on the digital factory/manufacturing concept, according to which production data management systems and simulation technologies are jointly used for optimizing manufacturing before starting the production and supporting the ramp-up phases [2, 3].

However, the current situation in the "digital factory" concept seems to be inefficient in a key element—the integrating factor between product design and production (assembly) planning [4]. Virtual commissioning (VC) on the other hand, goes a step further by including more validation capabilities by means of considering the mechatronic behaviour of the resources. The VC methodology provides a solution to the verification of mechanical behaviour of an assembly line and a cell, in conjunction with PLCs (programmable logical controllers) in loop with a virtual environment. The application of such a methodology may lead to reducing the errors detected during the ramp-up phase that necessitate reworks in upstream processes, since it enables the verification of real PLC engineering with virtual line and cell in the early production design phases [5]. As a result, VC allows for the reduction in the commissioning time, the advance of the start of the production time as well as the reduction in the shutdown times [6].

The area of VC has undergone an extensive investigation so far. Optimization and assessment of automation and production systems can now be accomplished under the use of digital products, resource data, as well as control data. Simulation models involving kinematics, geometric, electric, and control-technical aspects are capable of a 1 : 1 mapping and representation of the virtual commissioning project with the real system.

VC can be a valuable tool for assembly-line design engineers in the sense that it can provide decision-making support to a plethora of decisions, such as the type and number of resources to be used within an assembly line or the selection of communication and interfacing protocols between resources. As a result, more simple, cost- and time-efficient production setups can be achieved. The objective of this paper is to present and analyse all the steps required for the assembly systems' virtual commissioning to be carried out. The main focus is that areas requiring improvements in terms of both hardware and software equipment be identified in order for the gap between theory and practise to be bridged. As it will be shown, although there are many software tools available, the realization of VC is far from being fully automated. In parallel, the benefits from the application of VC techniques are discussed in order for the importance of the techniques to be highlighted with respect to cost and time savings of the investors of this approach.

To provide a better insight on the real-life implication of applying VC, a case study involving a vehicle floor assembly cell is presented. The cell uses cooperating robots, the latest trend in automotive industries. This case however, provides several challenges mainly due to the higher complexity in the control systems as well as in the elimination of the PLC from the top of the control hierarchy. Nevertheless, the continuous emphasis of the assembly industries on multirobot cooperation platforms requires that the particularities of such novel systems be investigated when applying state-of-the-art VC techniques. Towards this direction, a complete method of VC, complemented by a selection of capable software tools, is being discussed in this paper.

2. Literature Review

This section includes an overview of the concepts underlying the VC technologies. Data requirements for VC are also discussed, and existing methods and tools are presented.

2.1. Virtual Commissioning Concepts

An early approach to VC was presented in [7] and was referred to as “soft commissioning”, allowing the coupling of simulation models to real-world entities and enabling the analyst to precommission and test a system’s behaviour, before it was built in reality. The method however, did not consider the entire life cycle of a technical system including requirements engineering and “classical” simulation analysis. The work in [8, 9] has identified that the VC project involves three distinct but interconnected subsystems: (a) the mechanical design including actuators, sensors, and behavioural description of a systemrelated (functional model), (b) the machine control, including its input and output signals and (c) the signal connections between sensors/actuators and the control. Currently, there are two approaches to building a VC project. Under the Software in the Loop (SIL) method, the control programs for the resource controllers (PLC or other) are downloaded to virtual controllers and IP/TCP connection is established between the mechatronic object and the software-emulating controllers. It is obvious that the main advantage of the SIL approach is that no hardware is required during the designing and validation of a control software, while standard desktop PCs can be used for its implementation. On the other hand, there has been identified a low availability of up-to-date control simulation packages for a particular control version and therefore, the control software cannot provide an exact reproduction of the control behaviour [10, 11]. The second method, known as hardware in the loop (HIL),involves the simulation of production peripheral equipment in real time, connected to the real control hardware via fieldbus protocol. Under this setup, commissioning and testing of complex control and automation scenarios, under laboratory conditions, can be carried out for different plant levels (field, line, or plant) [12]. Hybrid commissioning combines an HIL-commissioning and real-commissioning phases, which interact with each other thus, achieving a lower cost and more efficiency of the real commissioning [13]. Finally, two more VC types can be distinguished under the concepts of synchronous and forward simulation. The first one, is used for the comparison of the output of the real system versus the output of the simulation (HIL) model [12]. On the other hand, the forward simulation focuses on the prediction of the control systems’ influence by examining the process parameters against a set of optimisation criteria [14].

2.2. Data Requirements

To realize a VC project, the data requirements involve:

(i) Extended 3D simulation model of the resources to be commissioned, involving geometries, kinematics, electrics, electronics, and controller programs.

(ii) Detailed layout of the production cell, involving exact placing of resources and all relevant equipment.

(iii) Material flow in the shop floor, involving sequence of operations and interdependencies between the production processes.

(iv) Real hardware/software control systems. Either the actual control systems (such as PLCs) or the emulation software can be used for the validation of the virtual prototype.

(v) Detailed definition of the control system’s I/O signals and the respective mapping on the resource components.

(vi) Detailed definition of extra functionalities and signals to be included in the commissioning process (e.g., safety systems, etc.).

(vii) IT infrastructure, such as software drivers and communication protocols (usually TCP/IP) for the networking between the control system and the simulation model.

Once the VC project has been setup, further information regarding the validation of the project is required so as for the target goals to be achieved by the system under examination.

2.3. VC and Simulation Tools

Over the last years, a plethora of commercial packages that can be used for the implementation of a VC project, are available on the market. Delmia by dessault Systemes allows the virtual prototyping of PLC control systems for cells, machines and production lines which uses object linking and embedding for process control (OPC) communication for the coupling of the real control system with the simulated resource. Similarly, the process simulate commissioning package by Tecnomatix, enables users to simulate real PLC code with the actual hardware by using OPC and the actual robot programs, thus enabling the most realistic virtual commissioning environment [15]. The WINMOD software utilizes the Macro file concept, and can be coupled with the control software/hardware in a way similar to that of the real system. It offers accurate representation of the real system and allows for close observation of the input and output signals [16]. Finally, INVISION is an innovative simulation system that allows the planning and visualisation of production operations in real-time simulation. Through its coupling with WINMOD, a real-time HIL simulation can be achieved. For the purposes of our study, we have selected the last two packages due to the following reasons [17]:

(1) The tools that exist in the market do not provide the required performance, in terms of the number of signals they can simulate and the speed of simulation execution. Additionally, the existing tools in the market have integrated support of the different robot manufacturers. These modules use kinematics algorithms that represent the real controller behavior for the simulation of the robot motion in a virtual environment. However, these modules were developed for an offline programming of robots, and only a limited set of program commands is available (mainly motion instructions). The INVISION tool selected in support of this study translates the specific manufacturer's robot programs into a unified language and uses a software robot-integrated controller. This enables the simultaneous processing of the kinematics and the I/O signal processing.

(2) WINMOD is very versatile since it enables the connection of devices through (a) direct coupling via OPC, (b) direct coupling via fieldbus, and (c) fieldbus emulation.

3. VC Workflow

This section is dedicated to providing the description of the workflow for the realization of the VC project. For the validation process, a setup with two PCs is required. The first PC, contains the 3D simulation model (Figure 1) including the spatial layout of the cell, the resources geometry the kinematic constraints that represent the resource behavior. The robot

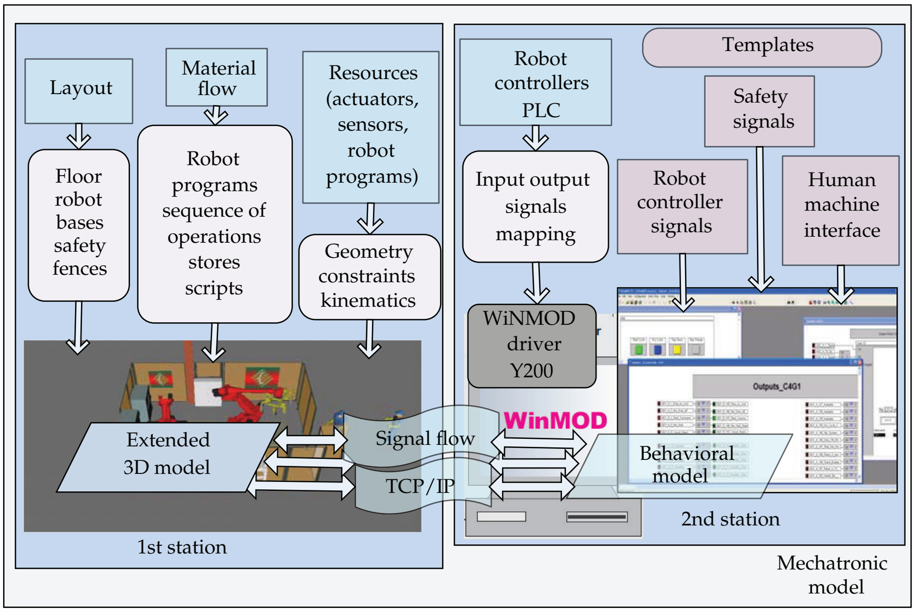


Figure 1: VC project setup.

controller is simulated within this PC as well. This means that the robot's programs (motion commands, signal triggering commands, etc.), the sequence of operations, and any scripts for data/signal exchange among the devices are also included in this model.

The second PC is used for the emulation of the control signals and the signal exchange networks, which are used within the cell, either by the robot or between any other device such as safety equipment, human-machine interfaces, and so forth. This means that all the signals from the PLCs and the robot controller (upper left corner of the 2nd station in Figure 1) need to be documented and mapped via a driver. The driver allows the software (in our case WINMOD) to receive, in real time, the signals either from the actual devices/controllers or the simulated elements. In any case, the simulated signals are transferred via the TCP/IP protocol to the first PC, where the process is visualized. To provide an example of the interaction, one can imagine the operator selecting the start button on the user interface, which is hosted on the second PC. The button triggers the respective I/O signal, transferred by the driver to the simulated robot controller in the first PC. The controller reads the signal and during the execution of its program, motion commands are sent to the robot, according to the programs saved. Throughout the simulation, the virtual robot controllers on the first PC use the second PC so as to exchange signals between themselves thus, eliminating the need for the physical presence of either the robots or the controllers.

3.1. Mechatronic Model Development

The first step of the method is the development of the simulation model upon which the control system will be validated. Existing simulation tools can be used for the detailing of the resources, followed by their conversion into data formats, supported by the package that has been selected for the final simulation. Positioning and modelling of the sensors, definition

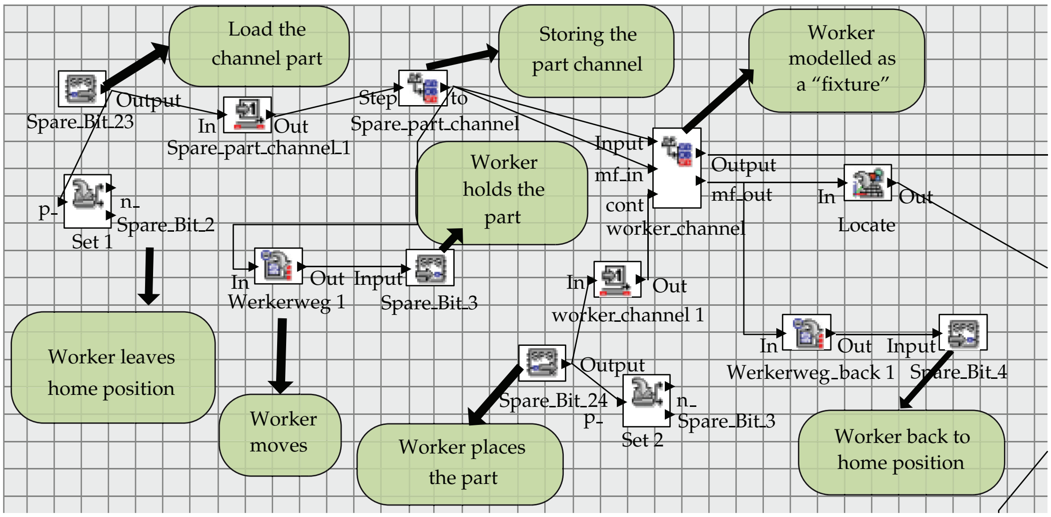


Figure 2: Material flow example diagram.

of the degrees of freedom, constraints definition, and moving axes configurations need to be applied to each resource individually. In the case of robots, the reference coordinate tool frames and the tool centre points (TCP), for the welding guns and grippers are attached to the robots.

3.2. *I/O Signal Definition*

Before the detailing process of the material flow inside the cell, the import of the I/O signals list used by the PLC or any other control system and all resources needs to be defined. Based on the software configuration, the software driver for the emulation of these signals is setup in terms of required memory for data exchange and the signals are generated and logically grouped either as input or output signals. The signal list is then imported to the control emulation software so that each signal has the same functionality as that in the real system.

3.3. *Material Flow Definition*

The next step in the workflow is the determination of the sequence of operations during the operation of the cell. The signal assignment (both input and output) to the robots, the sensors, the operators, and any other entity in the model are carried out in this step. Depending on the nature of the modelled object, either analogue signals (e.g., the open-close percentage of the door) or digital signals (sensor on/off state) can be assigned to it. Depending on the project it is also possible that "dummy" signals are defined as well, in order for the operation of the cell to be controlled through the human-machine interface. The use of a signal to denote the start of the cycle is such a case. Figure 2 shows an example of the flow diagram for the assignment of a part to an operator.

The behaviour model of the production system is completed at this step with the inclusion of the programs to be executed by each resource (e.g., robot, PLCs, etc.), the definition of all simulation activities (such as operator paths), and the realization of any

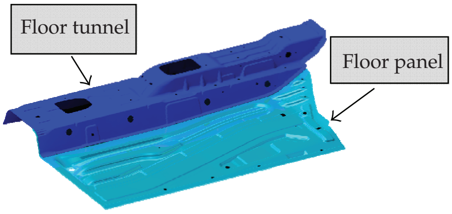


Figure 3: Passenger vehicle floor components.

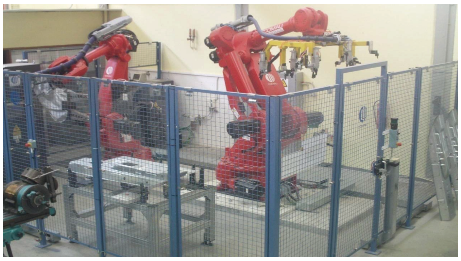


Figure 4: Actual assembly cell.

software interlocks in the control simulation software, for the system's smooth operation. (e.g. simulation cannot start when operators are in the cell, etc.).

3.4. Human-Machine Interface Definition

The human machine interfaces (HMI) such as the control panels of the resources are also simulated in the virtual environment. This is especially in the case of SIL, where no actual hardware is available for the validation procedure. Additional capabilities can be programmed on the HMI in order for the operation of the virtual cell to be controlled. An example is the use of a software button to trigger the loading of the parts by the operators.

4. Automotive Case Study

4.1. Actual Assembly Cell

The VC method has been applied onto a robotic cell that welds the parts of a passenger car floor. These parts are the floor tunnel and the floor panel as shown in Figure 3.

The vehicle's floor parts are assembled by two cooperating robots in the assembly cell as shown in Figure 4.

Cooperating robots, that is, robots communicating with each other for carrying out common tasks, may considerably expand their capabilities. They can be used for reducing the number of required fixtures as well as for shortening the process cycle time, whilst addressing the accessibility constraints introduced by the use of fixtures [18].

The cooperating robots' applications comprise characteristics such as [19]:

(i) workspace sharing: the definition of the critical workspace sections, where only one robot at a time may be present;

(ii) motion synchronisation: the capability of allowing multiple machines in a cell to begin and complete simultaneously a motion command;

(iii) programme synchronisation: a feature allowing robot programmes to remain at certain points until other programmes (controlling other robots, machines or devices) have reached the same ones;

(iv) linked motion: a feature enabling multiple machines to handle a part at the same time.

The robots used in the case study are a Comau Smart NJ 130, carrying a medium-frequency spot-welding gun and a Smart NJ 370, carrying a flexible geogripper that can hold both parts at the same time. Each robot is guided by its own C4Gtype controller. The cooperation of the robots is achieved by their connection to a master/slave coupling, making the use of a central PLC obsolete. Although robot cooperation can also be achieved by using a PLC, the direct connection of the controllers provides far greater real-time capabilities of performing the aforementioned functionalities (motion linking, workspace sharing, etc.).

This particular approach for the implementation of cooperating robots (without using central PLC) is particularly challenging not only due to the advanced functionalities but also to the complexity it contains. The challenges to be met are identified on the issues involving coordination, sequencing, collision, and communication architectures. Real-time motion coordination and communication between the robot controllers require higher computational capabilities from the robot controller's side as well as protocols for high-speed signal exchanging. The programming aspects of such systems are also characterised by higher complexity, since the programmers need to consider the dynamic nature of real-time communication between robots during the generation of a code for the control of robots. The direct, nonsupervised interaction between robot controllers signifies that a very careful mapping and strict determination of the signals being exchanged between the different robots need to be followed [20].

The operation of the cell can be summarized as follows:

(i) The operator loads the floor parts on a loading table inside the cell. The loading table is designed to accommodate the parts of the scenario, and guarantee nominal relevant positions.

(ii) The handling robot uses a modular gripper to pick up both parts from the loading table. It is a geogripper, denoting that it is also used for holding both parts in a specific relative position to each other so that a correct geometry of the final product is achieved. Simple pneumatic clamps are mounted onto the gripper and are used for securing the part.

(iii) Presence sensors that are mounted onto the modular gripper of the handling robot are used for determining the part's length. The sensors are also used for ensuring that the part remains onto the gripper throughout the duration of the assembly process.

(iv) The sensors' signals are used by the cell controller in order to make a decision as to which programs should be executed by the handling and welding robots. Different part lengths signify a different number of spot welds and trajectories of the welding gun and the part itself during handling.

(v) A cooperative motion between the two robots is initiated when the handling robot moves the part in such a way so as for the welding gun to be allowed accessibility

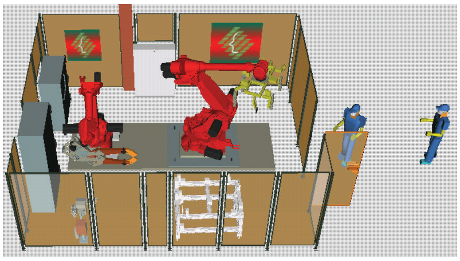


Figure 5: Virtual assembly cell.

to all spot weld locations in a single cycle. While moving, the robots communicate with each other through the DEVICENET, by exchanging signals, denoting the initiation/termination of each motion/operation.

(vi) Once all spot welds have been carried out, the handling robot places the assembled parts on the loading table, and the operators remove them from the cell.

4.2. Virtual Assembly Cell

For the purpose of the project, WINMOD has been used as the control simulation software, running on an Intel Core i5 CPU 650 @3.20 Ghz with Microsoft Windows XP Professional (SP3) and 3 GB of RAM. For the 3D simulation part, a PC running Linux OpenSUSE 11.2, using an Intel Core i7 CPU 650 @2.80 Ghz with 12 GB of RAM, was utilized in combination with the INVISION simulation package.

As it can be verified with Figure 4, the model (Figure 5) is an accurate representation of the real cell in terms of layout, resources, and equipment. The validation successfully justified the functionality of all 1282 control signals and robot programs that were tested in the project. Based on the simulation, the virtual cycle time was 70 seconds, almost identical to the actual cell cycle time.

As it was mentioned in the previous section, the robots have their proprietary controllers, which support specific programming languages. Nevertheless, the INVISION software provides a generic controller that can be used for any type of robot, provided that its geometry and the kinematic relationships between its components are defined within the software. In order for the actual robot programs to be validated within INVISION, they need to be translated into the generic language, and currently there are not tools for automating this process. A further challenge was the transformation of the TCP coordinates for each point from Euler Angles (ZYZ) that are currently used by the robots, into the yaw-roll-pitch (ZXY) convention, supported by INVISION. Another drawback that was identified was the fact that the assigned I/O signals of each robot and PLC need to be manually defined by the PLC/robot programmers and then imported into the WINMOD software. Similarly, the mapping of each resource's signals within the INVISION model with the signals handled by WINMOD is not automated either. Therefore, greater integration between such systems is required, in order for the VC process to be fully automated, so as for more time to be required for value-adding processes, such as the determination of the best robot trajectory, rather than the manual configuration of the signals in the software environments.

5. Conclusions and Outlook

This work has examined the latest advances in VC technologies and presented the complete workflow of applying such techniques to an industrial assembly cell. The results confirm that VC provides a reliable way of validating the operation of a cell prior to each installation. The main benefits of VC involve:

(a) Ramp-up time reduction, affecting the total installation time by 15–25% out of which, a respective 90% accounts for the time required for the development and deployment of the robot programs and a related software at the shop floor [21]. Having validated all the programs through VC, the ramp-up time, where errors in the code might appear, is drastically reduced.

(b) Reduction in investment costs by decreasing to the minimum, the time for the deployment of the new line as well as the time for production tests, prior to having full production volume achieved (ramp up). The cost is reduced even more through the reduction by up to 15% of the human resources, required for troubleshooting during the ramp-up process [10]. The cost of the software tools for the realization of the process (less than 10.000€) cannot be considered significant when referring to large-scale assembly environments such as the automotive.

(c) Enhancement of the reconfigurability of assembly equipment in the sense that all changes are designed in the virtual environment and are not subject to the pressure due to the production stops for onsite testing or troubleshooting. The full potential of the equipment's flexibility can be exploited by carefully designed control strategies, while the technical feasibility of the project is being simultaneously validated.

Future research in the area of VC should focus on enabling the VC tools to test the market demand through the validation of projects. To promote the adoption of VC, manufacturers and their suppliers need to create and provide forward mechatronic simulation models with structured data. A promising path towards this functionality is the use of open-cross functional data that would build simulation models with engineering aspects, representing real plant components, and in this context, the AutomationML [22] seems to be a viable solution.

Acknowledgments

This research was partially funded by the FP6 Project NMP2-CT-2006-026631 MyCar and the FP7/2007–2013 under Grant agreement 285189-AUTORECON.

References

[1] B. K. Choi and B. H. Kim, "New trends in CIM: virtual manufacturing systems for next generation manufacturing," in *Proceedings of the Current Advances in Mechanical Design and Production Seventh Cairo University International MDP Conference*, pp. 425–436, Cairo, Egypt, 2000.

[2] T. Bär, "Flexibility demands on automotive production and their effects on virtual production planning," in *Proceedings of the 2nd CIRP Conference on Assembly Technologies and Systems*, pp. 16–28, 2008.

[3] E. Westkmper, "Strategic development of factories under the influence of emergent technologies," *CIRP Annals*, vol. 56, no. 1, pp. 419–422, 2007.

[4] H. Bley and C. Franke, "Integration of product design and assembly planning in the digital factory," *CIRP Annals*, vol. 53, no. 1, pp. 25–30, 2004.

[5] S. Makris, G. Michalos, K. Efthymiou et al., "Flexible assembly technology for highly customisable vehicles," in *International Conference Competitive and Sustainable Manufacturing, Products and Services (APMS '10)*, Como, Italy, 2010.

[6] R. Drath, P. Weber, and N. Mauser, "An evolutionary approach for the industrial introduction of virtual commissioning," in *IEEE International Conference on Emerging Technologies and Factory Automation (ETFA '08)*, pp. 5–8, 2008.

[7] F. Auinger, M. Vorderwinkler, and G. Buchtela, "Interface driven domain-independent modeling architecture for "soft-commissioning" and "reality in the loop"," in *Proceedings of the Winter Simulation Conference*, vol. 1, pp. 798–805, Phoenix, Ariz, USA, 1999.

[8] J. Kiefer, *Mechatronikorientierte Planung automatisierter Fertigungszellen im Bereich Karosseriebau [Dissertation]*, Universitat des Saarlandes, 2007, Schriftenreihe Produktionstechnik Band 43.

[9] Verein Deutscher Ingenieure (Hrsg), VDI-Richtlinie 3633, Blatt 8 (Entwurf): Simulation von Logistik-, Materialfluss- und Produktionssystemen Maschinennahe Simulation, VDI-Verlag, Düsseldorf, 2005.

[10] G. Reinhart and G. Wnsch, "Economic application of virtual commissioning to mechatronic production systems," *Production Engineering. Research and Development*, vol. 1, no. 4, pp. 371–379, 2007.

[11] M. Spitzweg, *Methode und Konzept fur den Einsatz eines physikalischen Modells in der Entwicklung von Produktionsanlagen (Method and concept for the approach of a physics model to the development of production lines) [Ph.D. thesis]*, Technische Universitat Munchen, 2009.

[12] S. Kain, F. Schiller, and T. Frank, "Monitoring and diagnostics of hybrid automation systems based on synchronous simulation," in *Proceedings of the 8th IEEE International Conference of Industrial Informatics (INDIN '10)*, pp. 260–265, 2010.

[13] S. Dominka, F. Schiller, and S. Kain, "Hybrid commissioning—from Hardware-in-the-Loop simulation to real production plants," in *Proceedings of the 18th IASTED International Conference on Modeling and Simulation (MS '07)*, pp. 544–549, Montreal, Canada, 2007.

[14] S. Kain, S. Dominka, M. Merz, and F. Schiller, "Reuse of hiL simulation Models in the operation phase of production plants," in *Proceedings of the IEEE International Conference on Industrial Technology (ICIT '09)*, pp. 1–6, Victoria, Australia, 2009.

[15] Tecnomatix, http://www.plm.automation.siemens.com/en_gb/products/tecnomatix/assembly_planning/process_simulate/commissioning.shtml.

[16] WINMOD, http://winmod.de/de/index.php?page=virtuelle-maschine-anlage.

[17] M. Bergert, J. Kiefer, S. Hoeme, and C. Fedrowitz, "Einsatz der Virtuellen Inbetriebnahme im automobilen Karosseriebau Ein Erfahrungsbericht," in *Tagungband Der 9, Magdeburger Maschinenbau-Tage*, pp. 388–397, 2009.

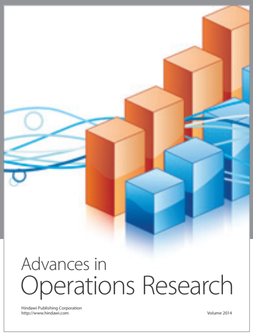
[18] G. P. Ranky, "Collaborative, synchronous robots serving machines and cells," *Industrial Robot*, vol. 30, no. 3, pp. 213–217, 2003.

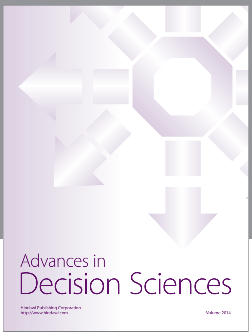
[19] R. Koeppe et al., "Robot robot and human robot cooperation in commercial robotics applications," *Robotics Research*, vol. 15, pp. 202–216, 2005.

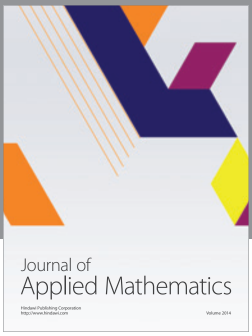
[20] N. Papakostas, G. Michalos, S. Makris, D. Zouzias, and G. Chryssolouris, "Industrial applications with cooperating robots for the flexible assembly," *International Journal of Computer Integrated Manufacturing*, vol. 24, no. 7, pp. 650–660, 2011.

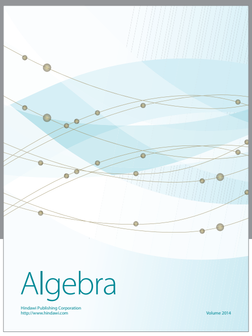
[21] M. F. Zäh, G. Wünsch, C. Pörnbacher, and M. Ehrenstrasser, "Emerging virtual machine Tools," in *proceedings of the 29th Design Automation Conference (ASME DETC '03)*, Chicago, Ill, USA, 2003.

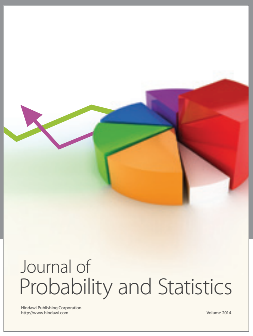
[22] AutomationML, http://www.automationml.org/o.red.c/home.html.


Advances in
Operations Research

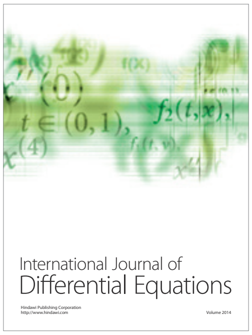

Advances in
Decision Sciences

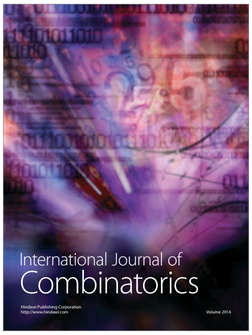

Journal of
Applied Mathematics

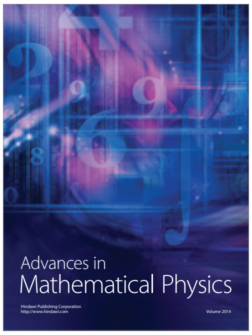

Algebra

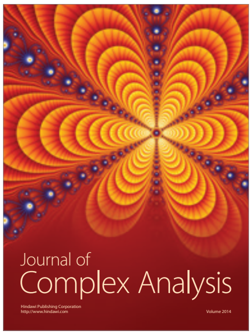

Journal of
Probability and Statistics

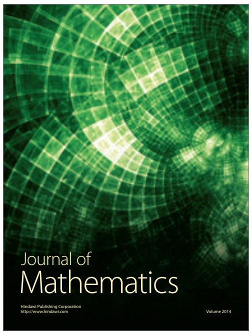
The Scientific
World Journal

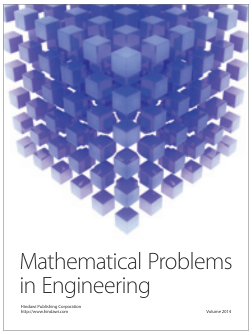

International Journal of
Differential Equations

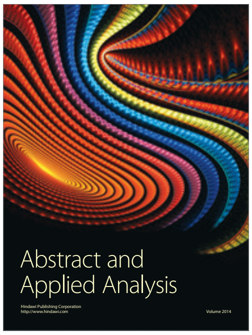

International Journal of
Combinatorics

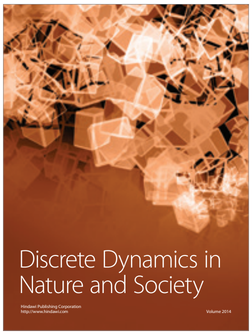

Advances in
Mathematical Physics

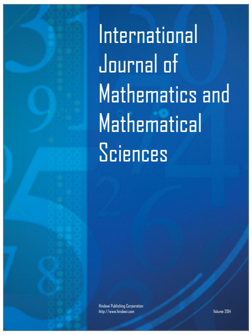

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

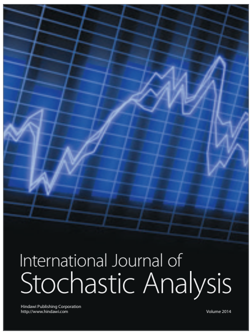

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization