Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 50 (2016) 129 – 134

www.elsevier.com/locate/procedia

26th CIRP Design Conference

A framework for automatically realizing assembly sequence changes in a virtual manufacturing environment

Mussawar Ahmad[a]*, Bilal Ahmad[a], Robert Harrison[a], Bugra Alkan[a], Daniel Vera[a], James Meredith[b], Axel Bindel[c]

[a]*Automation Systems Group, WMG, University of Warwick, CV4 7AL, Coventry, West Midlands, UK*
[b]*Mechanical Engineering, University of Sheffield, South Yorkshire, S10 2TN*
[c]*HSSMI, Queen Elizabeth Park, London, E20 3BS*

* Corresponding author. Tel.: +44 (0) 2476 573413. *E-mail address:* Mussawar.Ahmad@warwick.ac.uk

Abstract

Global market pressures and the rapid evolution of technologies and materials force manufacturers to constantly design, develop and produce new and varied products to maintain a competitive edge. Although virtual design and engineering tools have been key to supporting this fast rate of change, there remains a lack of seamless integration between and within tools across the domains of product, process, and resource design - especially to accommodate change. This research examines how changes to designs within these three domains can be captured and evaluated within a component based engineering tool (vueOne, developed by the Automation Systems Group at the University of Warwick). This paper describes how and where data within these tools can be mapped to quickly evaluate change (where typically a tedious process of data entry is required) decreasing lead times and cost and increasing productivity. The approach is tested on a sub-assembly of a hydrogen fuel cell, where an assembly system is modelled and changes are made to the sequence which is translated through to control logic. Although full implementation has not yet been realized, the concept has the potential to radically change the way changes are made and the approach can be extended to supporting other change types provided the appropriate rules and mapping.

Keywords: Assembly sequence; assembly process; virtual engineering; hydrogen fuel cell; control logic

1. Introduction

The mass customization paradigm seeks to provide individualized products at near mass production costs [1]. Product variety can be achieved by making changes at different points during a product's life cycle from the design phase through to the use phase, and is further facilitated by product modularity [2]. With certain high value products, in order to ensure that customer or legislative requirements are met, it is necessary to make changes at the design phase *e.g.* reducing the harmful emissions of a combustion engine. This has the impact of affecting all downstream product realization domains such as process design and manufacturing system design [3]. As a result of this impact, the paradigm of reconfigurable assembly systems (RAS) has been proposed as a means of accommodating such design changes efficiently to reduce costs, in a way that is not possible for traditional dedicated manufacturing lines [4]. There has been a significant amount of work on describing the nature of a RAS [4, 5], but to reach the full potential of such systems, the knowledge capture and translation through the product realization domains must be as agile as the vision for the physical manufacturing system. In industry, the lack of seamless integration through the product, process and resource design domains results in clunky knowledge transfer where miscommunication or an entire lack of communication results in delays and errors. This is further exasperated by the global nature of businesses today whereby such domains may exist across multiple organisations spanning several countries and continents [6]. This has been referred to as the *co-evolution* problem [6, 7] but can more generally be described as the need for engineering concurrency [8]. The use of virtual engineering (VE) tools are becoming ever more

doi:10.1016/j.procir.2016.04.178

prevalent in industry and, within their respective domains, facilitate in change realization more cost effectively than has previously been possible *i.e.* changes can be evaluated in a virtual environment and simulations can be carried out to assess performance without needing to invest in physical materials and resources. However, despite the sophistication of such tools, making a change to a product must still be *manually* translated to a change in the manufacturing system via the process domain to assess: accessibility, fixture design changes, assembly sequence feasibility *etc.*[8]. This research proposes a method for translating product design changes through to automation system control logic for deployment automatically using vueOne (a virtual engineering tool developed by the Automation Systems Group at the University of Warwick.)

2. Review of literature and gap analysis

2.1. Approaches for domain integration

Many researchers have looked at how to integrate the Product, Process and Resource (PPR) domains. The definition of what the product realization domains should be called and how they should be defined *i.e.* what factors should be encompassed within them, varies depending on background and experience. An early example is presented in [9] which describes what should be mapped, but not necessarily how such mapping should be achieved in a practical sense. However, it is clear that in order to attain integration, it is necessary to decompose each of the heavy, complex domains into smaller components to a satisfactory level of simplicity, allowing an identification of where mapping between the domains is appropriate [3].

A component based (CB) system was proposed by Rosenman and Wang that compared five collaborative architectures and described a web-based interface to manage component agents [10]. To address communication issues regarding design changes Chao *et al.* used the agent attributes of proactiveness and autonomy to co-ordinate the design activities of multidisciplinary design teams [11]. Ribeiro *et al.* extended their previous works to demonstrate how agent technology could be used to support “plug and produce” in run time [12]. Process plans were embedded within product agents which coordinated with system resources (conveyors, gates, stations) to route pallets through the system based on product requirements. Wang *et al.* argued that realizing agent-based approaches in real-time is difficult as the decision making process is neither deterministic nor event-driven. Instead, they proposed a process oriented function-block (FB) framework, where each FB represents basic assembly operations with embedded algorithms to describe how to fulfill the operation which in turn communicate directly to control systems or operators in the form of instruction sheets [13]. A method to support integrated product and process design was proposed by Mervyn using manufacturing middleware that synchronized applications [14]. Java and XML facilitated portability, and compatibility was achieved with common data structures.

Alternative integrative approaches include the use of knowledge-based systems within ontologies - enriched and supported by semantics. Numerous examples of this approach can be found in the literature, with each researcher choosing different areas to focus on and differing ontological structures to meet the requirements of their case. Lohse presented the ONTOMAS framework to reduce assembly system design effort using domain ontologies and implementing a function-behavior-structure paradigm to capture the characteristics of modular assembly system equipment [3]. A similar abstraction approach was proposed by Hui *et al.* that used semantic objects to retrieve information from documents of various formats and by inference allowing domain specific tools to become better integrated [15]. Lanz used feature based modelling to capture detailed product knowledge, categorizing features into geometric and non-geometric, to provide knowledge for a holonic manufacturing system [16]. Raza and Harrison described a collaborative production line planning approach supported by knowledge management theory [17]. A service-oriented architecture was proposed and supported by semantic web services that allowed automatic discovery and execution of assembly processes by modelling and mapping assembly processes and systems in [18]. An influential architecture for integrating the PPR domains is the Virtual Factory Framework (VFF) which is a data model that links and stores knowledge to support engineering concurrency in the resource domain [19], but does not have the granularity to model system control logic. More recently, knowledge-based mapping has been used to support in the selection of function blocks for manufacturing resource components [20], and Ramis *et al.* [21] showed how product requirements could be translated directly through to dynamically changing programmable controller logic. Chen *et al.* extended EAST-ADL (a language developed to model automotive electronic systems, see [22]) to model production systems using MetaEdit+ [23]. Mapping within and between the concepts of *Equipment, Process* and *Product* were achieved through the EAST-ADL feature links.

2.2. Virtual prototyping manufacturing systems

Virtual prototyping tools (VPTs) for manufacturing systems should provide: 1) a model consistent with real manufacturing systems, 2) effective simulation and prototyping capability and 3) a means for collaborating with key product realization domain stakeholders [24]. These tools facilitate “digital manufacturing” and state-of-the-art examples include: DELMIA Automation by Dassault Systems [25] and Technomatix by Siemens [26]. Both tools provide modules or workstations within common environments to model a large range of mechatronic systems. However, both are “heavyweight” applications requiring high end computing and specialized training, and as they focus on flexible rather than modular, reconfigurable systems, they are not inherently designed to easily identify the impact of making changes [24, 27]. A number of academic groups have also presented VPTs namely: Min *et al.* [28] who integrated real-time machine tool data within a virtual manufacturing environment. Suk-Hwan *et al.* [29] developed Web-based virtual machine tools to interactively operate CNC machine tools, and Dietrich *et al.* [30] presented sample scenarios for how the real and virtual processes could be integrated. While these tools show promise in their respective applications, they do not easily integrate

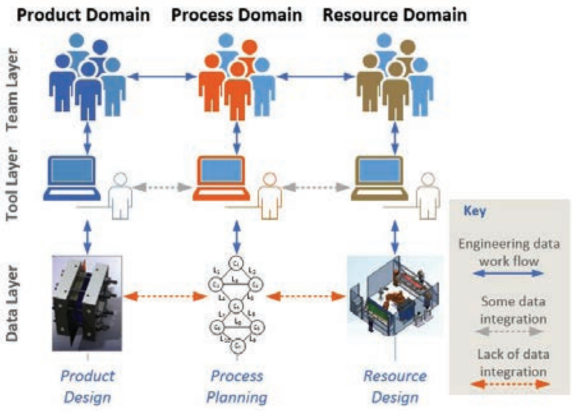

Figure 1 Engineering data workflows identifying where integration gaps exist

with the other domain stakeholders. Just as section *2.1* identifies decomposition of domains as a necessary means of mapping and integration, it is as important that the architecture of VPTs follow the same component orientated data structure. Fig. 1 describes engineering data workflows and as indicated by the dashed arrows in grey and orange, there is a limited, or a lack of, integration at the tool and data layers. As a result, changes cannot easily be made unless these are discussed at the team layer which represents domain experts.

2.3. Summary

The review of literature has touched upon how data and knowledge within the respective domains can be captured and stored, described several methods for integrating the various product realization domains and presented the state-of-the-art VPTs for manufacturing systems. The authors' propose that although all of the elements and interfacing technologies exist for integrating domains, it is necessary to explore how best to do so *i.e.* create and test architectures to identify those that lend themselves best to industrial needs. In this research, a framework is proposed that maps domain elements using semantic technology, supported by ontologies as a backend and a novel set of CB virtual engineering (VE) tools as the user-interface to understand the engineering data workflow from the change in product design through to automation system control logic.

3. Methodological Framework

This research looks at a specific type of product design change, one of assembly sequence, on the assumption that other system components such as jigs and fixtures are not modified significantly. This results in only the control logic requiring substantial rearrangement *i.e.* a new set of pre and post-conditions for each step. This is true for products where variants are designed with modular components so that realizing a family only requires a change to the pick sequence. Typically, in a mixed-model assembly line, variation is achieved using RFID technology or barcodes which communicate the appropriate component variant location to the control system. This is something that is pre-programmed and is not the nature of the change that is being described in this paper. Rather, this research looks at the need to change an assembly sequence as a result the product domain wishing to introduce a product variant and how *this* change is pushed through the product realization process. Furthermore, the focus is only on fully automated or semi-automated systems as manual systems require only that the operator be told to change an operation, this can be achieved with a new set of work instructions with some additional training if necessary.

3.1. Component Based VPT Architecture

As has already been mentioned, this research uses a set of CBVE tools to support in data integration. The architecture of the tools and how this links to the larger data integration structure is presented in Fig. 2. Firstly, the linking of the visual data for the product and resource data is supported by their respective CAD/CAE environments, which is simplified to the VRML format. This data includes information such as geometry and component relationships so that, at a basic level, a fit can be identified between the product components and resource components *i.e.* jigs and fixtures within the CBVE environment. Once within the system design, the VPT can support in layout configuration to determine where buffer and storage systems should be placed, potential areas of unreachability (as in the case of a robot), and whether the system fits into a floor space as described by the end user *etc*.

When a product design lands on the desk of a process planner, the first task is to understand the nature of the relationship of the components, where the precedence constraints are and thus determine a viable assembly sequence. From this information, assembly processes and operations can be identified, further enriched by considering product quality. It is based on this set of requirements that the resource domain designs an assembly system, with continuous communication with the other domains as to the limitations of various approaches, and how they impact cost and functionality, resulting in product design and, in turn, process design changes. Then the CBVE tools, with coordinated input from the process and resource domains, are able to route the product components through the system by setting conditions for resource component kinematic actuation. This facilitates

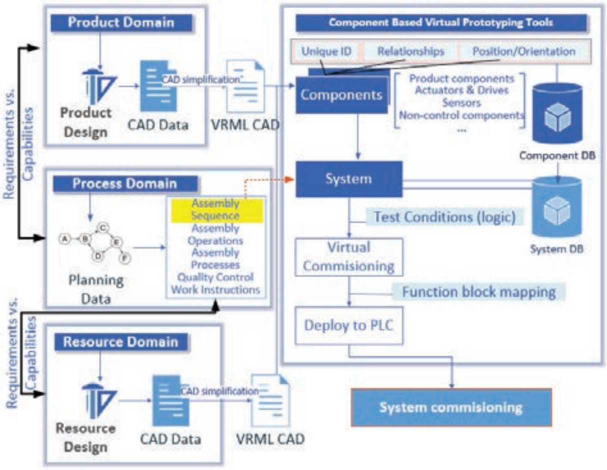

Figure 2 Integration architecture for component-based tools within a PPR framework

virtual commissioning *i.e.* to check that the conditions execute the assembly sequence proposed by the process domain. From this point, control code can be automatically generated and deployed onto a real system.

3.2. Realizing assembly sequence changes

Fig. 3 presents how, in this research, an assembly sequence is mapped to the process sequence and then, in turn, how this logic maps to controlling actuator behaviour. In addition, a 2-axis gantry (with the abstract product components of the product domain and the fixture position) is presented to visualize how the actuator logic links to the assembly system. In this example, the sequence being proposed by the process domain is that the liaison, L_2, is to be achieved before L_1 (note that this is typically represented by a directed graph). Achieving a liaison is composed of picking a product component and then placing it at the relevant position on the adjacent product component. This is translated through to process sequence which describes how the requirements of the process description are to be realized by the assembly system specific components. The conditions for transitioning from one state of the process sequence to the next depends on the conditions of the assembly system *i.e.* whether a sensor has been activated, the position of an actuator *etc.* The process sequence aspect of this framework prevents the logic becoming embedded within the actuators and thus allows changes to be more easily made.

If a change is to be made to swap the assembly sequence *i.e.* L_1, L_2 rather than L_2, L_1, then the process description can be changed to match (note that the fixture may need to be changed and components would need to be flipped to achieve this in a real world setting). Furthermore, by mapping the process description to "components" of process sequence they too can be switched around and in turn the system would respond accordingly. The caveat here is that the pre and post conditions associated with the components of the process sequence would need to be moved as the process description changes. Thus it is necessary to identify the associated pre and post conditions of a step and if and how they remain linked to it.

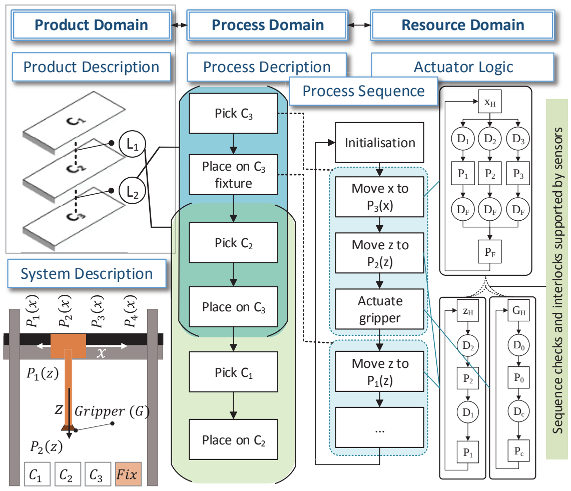

Figure 3 Assembly sequence to process logic and actuator logic mapping and system configuration

3.3. Mapping using ontologies

The structure of the mapping presented in Fig. 3 is represented in an ontology class diagram in Fig. 4. The structure of the ontology is largely based on the architecture of the CBVE tools with the additional, novel addition of *processDesc* class to map the process description to the *processSequence* through the *processElement* class. The *processElement* class is a descrption of a single state of the *processSequence* such as "move X actuator to position 1." The *processSequence* and the *controlComponent* classes are both subclasses of the *component* class which has *componentID*. This ID is used to identify a specific instance of a *component*. The *controlComponents* consist of objects such as sensors and actuators. Although the *processElement* is equivalent to a *state* it is kept separate to more easily discern between pre and post conditions as well as map *processStep*, a subclass of *processDesc*, to the control logic. This mapping is done through the *stateID* class to which *processElement*(s) and *controlComponent*(s) are both linked to through the *hasStateID* object property. In order to move from one *processElement* or *state* to the next, a *condition* needs to be fulfilled. Instances of the *condition* class are mapped to *StateID*(s) from which *state*(s) can be inferred.

The *processDesc* class contains within it each component step of the system to replicate the process description. This data would be populated into the ontology via a spreadsheet in XML format using the spreadsheet plug-in for Protégé [31] (an ontology editor developed by Stanford University and used for constructing and editing ontologies in this study) developed by Kola and Rector [32]. This would be necessary as the engineering tools currently being used do not have the capability to describe the process at this level. It is proposed that the process domain team would access and update this spreadsheet and then the changes would be updated and reflected in the ontology. The updated state of the ontology would then be used to update the database of the CBVE tools resulting in changes to the control logic. This is explored in more detail in the following chapter.

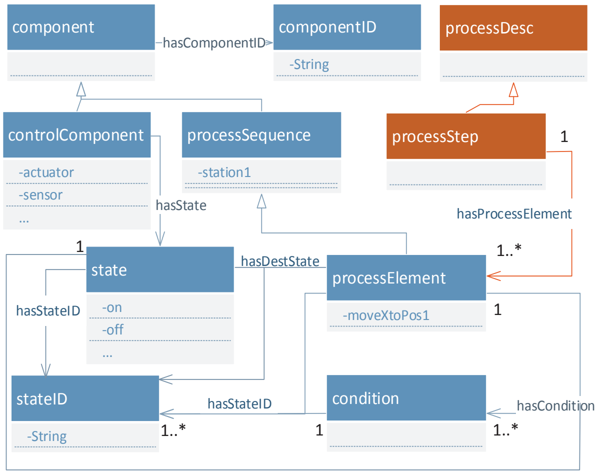

Figure 4 Process sequence and process logic ontology

4. Case Study

This research is driven, in part, by the needs of the fuel cell industry to support in being able to rapidly introduce new design variants into assembly systems. As this is still a maturing industry, with a technology that has the capability to be utilized across a large range of applications, nuanced changes to the design will continue to be made to meet requirements. As such, the case study looks at a sub-assembly which is among one of the more challenging to accomplish [33] the *gas diffusion layer (GDL) – gasket* liaison. Nuances in the design of these fuel cell components, which can significantly influence performance, also have an impact on the assembly sequence.

4.1. Product design and system configuration

An exploded diagram of a fuel cell is presented in Fig. 5a as well as the design variants for the gasket-GDL liaison. The addition of a liner as presented by *Variant 1* allows a sub-assembly of a single fuel cell to be made (often this is referred to as the membrane electrode assembly (MEA)), however this results in a more complex and thus more expensive product component. On the other hand, *Variant 2* provides a reference window within which the GDL is placed, potentially improving the product quality. Fig. 5b describes the system configuration and the associated process sequence. The reason that both components are listed in the first three boxes (highlighted) is that the sequence change needs only to be actioned at this point. The initial sequence of the system is to assemble *Variant 1*, and then the system is reconfigured to assemble assembly *Variant 2*. The case study describes how and where the data is being transformed, translated and stored in the system.

4.2. Proposed implementation

A component-based "state" is described in XML format in Fig. 5c which is an export of the CBVE tools system control database. The appropriate tags within the XML are used to semantically map to the relevant ontology classes (Fig 5c). Semantics are used to initiate the development of a semantic dictionary so that this approach can be applied to other software tools and not limited to vueOne. This XML carries the logic associated with the assembly of *Variant 1*. The process description is generated manually through an identification of the appropriate constituent process element steps. This means that when the process description is changed, it is done so in a format that is recognizable. To achieve *Variant 2* the only changes made are changing the relevant state name in the process element and the associated condition. As this is a relatively simple example of how this approach could be implemented, the authors' need to explore more complex systems to ascertain whether only a change to the state name and the conditions is sufficient to realize system control logic reconfiguration.

4.3. Discussion

The implementation of the approach was tested manually to test the architecture and identify shortcomings. The translation of data from the VE tools database through to the ontology appears to be a relatively simple process as the tags exist within the XML and the use of semantics simply adds meaning to these tags so that the machine can understand them and thus "auto-populate" the ontology with instances. On the other hand, the mapping of the process description to the process elements is laborious and one of the challenges here is to ensure consistency at the transition point. Furthermore, the open nature of a spreadsheet means that the risk of error is greater

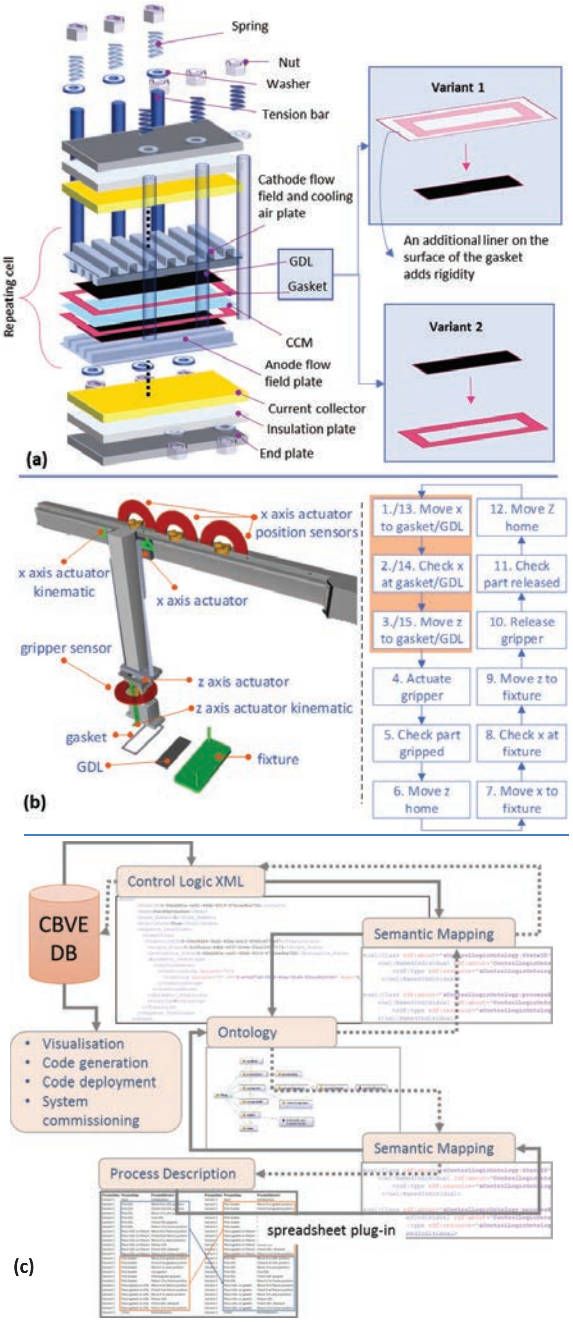

Figure 5 (a) Exploded view of hydrogen fuel cell and gasket-GDL liaison design variants (b) system configuration and process sequence (c) Proposed implementation than a dedicated GUI. Despite this shortcoming, the proposed framework is a promising approach to making sequence changes quickly, and by integrating with a virtual engineering tool that can auto-generate PLC code, these changes can potentially be seen on a real system almost instantly.

5. Conclusions and Further Work

This research has described the importance of integrating the product realisation domains so that changes can be made more easily, with specific focus on the process sequence and control logic. A framework has been presented which is more granular in nature than those available in the literature [3, 19] allowing changes and assessments, to be made on the assembly system at the level of control component logic. The novelty of this work is to show how integration is possible using a set of CBVE tools, and carrying the CB philosophy through the framework to prevent data inconsistency. However, full implementation has not been carried out and further work will fully integrate the heterogeneous software tools using semantic technologies, and ontologies with changes being made through an integrated GUI. The next step of this work extract the process elements, in sequence, and import them into the process description GUI. From this, the user will map the process steps to the elements using an intuitive interface.

Acknowledgements

The authors gratefully acknowledge the support of Innovate UK through the Fuel Cell Manufacturing Project (ref: 101980), UK EPSRC through the KDCM research project, and Arcola Energy for providing valuable insight on industrial challenges.

References

[1] S. Hu, X. Zhu, H. Wang, and Y. Koren, "Product variety and manufacturing complexity in assembly systems and supply chains," *CIRP Annals-Manufacturing Technology,* vol. 57, pp. 45-48, 2008.

[2] S. J. Hu, J. Ko, L. Weyand, H. ElMaraghy, T. Lien, Y. Koren, *et al.*, "Assembly system design and operations for product variety," *CIRP Annals-Manufacturing Technology,* vol. 60, pp. 715-733, 2011.

[3] N. Lohse, "Towards an ontology framework for the integrated design of modular assembly systems," University of Nottingham, 2006.

[4] Y. Koren and M. Shpitalni, "Design of reconfigurable manufacturing systems," *Journal of manufacturing systems,* vol. 29, pp. 130-141, 2010.

[5] M. G. Mehrabi, A. G. Ulsoy, and Y. Koren, "Reconfigurable manufacturing systems: key to future manufacturing," *Journal of Intelligent manufacturing,* vol. 11, pp. 403-419, 2000.

[6] T. Tolio, D. Ceglarek, H. ElMaraghy, A. Fischer, S. Hu, L. Laperrière, *et al.*, "SPECIES—Co-evolution of products, processes and production systems," *CIRP Annals-Manufacturing Technology,* vol. 59, pp. 672-693, 2010.

[7] T. Tolio, M. Sacco, W. Terkaj, and M. Urgo, "Virtual Factory: An Integrated Framework for Manufacturing Systems Design and Analysis," *Procedia CIRP,* vol. 7, pp. 25-30, // 2013.

[8] F. Demoly, X.-T. Yan, B. Eynard, S. Gomes, and D. Kiritsis, "Integrated product relationships management: a model to enable concurrent product design and assembly sequence planning," *Journal of Engineering Design,* vol. 23, pp. 544-561, 2012.

[9] H. K. Rampersad, *Integrated and Simultaneous Design for Robotic Assembly: Product Development, Planning*: John Wiley & Sons, Inc., 1994.

[10] M. Rosenman and F. Wang, "A component agent based open CAD system for collaborative design," *Automation in Construction,* vol. 10, pp. 383-397, 5// 2001.

[11] K.-M. Chao, P. Norman, R. Anane, and A. James, "An agent-based approach to engineering design," *Computers in Industry,* vol. 48, pp. 17-27, 5// 2002.

[12] L. Ribeiro, A. Rocha, A. Veiga, and J. Barata, "Collaborative routing of products using a self-organizing mechatronic agent framework—A simulation study," *Computers in Industry,* vol. 68, pp. 27-39, 2015.

[13] L. Wang, S. Keshavarzmanesh, H.-Y. Feng, and R. O. Buchal, "Assembly process planning and its future in collaborative manufacturing: a review," *The International Journal of Advanced Manufacturing Technology,* vol. 41, pp. 132-144, 2009.

[14] F. Mervyn, A. Senthil Kumar, S. H. Bok, and A. Y. C. Nee, "Developing distributed applications for integrated product and process design," *Computer-Aided Design,* vol. 36, pp. 679-689, 7// 2004.

[15] W. Hui, X. Dong, D. Guanghong, and Z. Linxuan, "Assembly planning based on semantic modeling approach," *Computers in Industry,* vol. 58, pp. 227-239, 2007.

[16] M. Lanz, "Logical and semantic foundations of knowledge representation for assembly and manufacturing processes," *Tampereen teknillinen yliopisto. Julkaisu-Tampere University of Technology. Publication; 903,* 2010.

[17] M. B. Raza and R. Harrison, "Design, development & implementation of ontological knowledge based system for automotive assembly lines," *International Journal of Data Mining & Knowledge Management Process,* vol. 1, 2011.

[18] I. M. Delamer and J. M. Lastra, "Ontology modeling of assembly processes and systems using semantic web services," in *Industrial Informatics, 2006 IEEE International Conference on*, 2006, pp. 611-617.

[19] M. Sacco, P. Pedrazzoli, and W. Terkaj, "VFF: virtual factory framework," in *Proceedings of 16th International Conference on Concurrent Enterprising, Lugano, Switzerland*, 2010, pp. 21-23.

[20] B. Ramis Ferrer, B. Ahmad, A. Lobov, D. Vera, J. L. Martinez Lastra, and R. Harrison, "A knowledge-based solution for automatic mapping in component based automation systems," presented at the 13th IEEE International Conference on Industrial Informatics (INDIN), 2015.

[21] B. Ramis Ferrer, B. Ahmad, A. Lobov, D. Vera, J. L. Martinez Lastra, and R. Harrison, "An approach for knowledge-driven product, process and resource mappings for assembly automation," presented at the 2015 IEEE International Conference on Automation Science and Engineering (CASE), 2015.

[22] V. Debruyne, F. Simonot-Lion, and Y. Trinquet, "EAST-ADL — An Architecture Description Language," in *Architecture Description Languages: IFIP TC-2 Workshop on Architecture Description Languages (WADL), World Computer Congress, Aug. 22–27, 2004, Toulouse, France*, P. Dissaux, M. Filali-Amine, P. Michel, and F. Vernadat, Eds., ed Boston, MA: Springer US, 2005, pp. 181-195.

[23] D. Chen, A. Maffei, J. Ferreirar, H. Akillioglu, M. R. Khabazzi, and X. Zhang, "A Virtual Environment for the Management and Development of Cyber-Physical Manufacturing Systems," *IFAC-PapersOnLine,* vol. 48, pp. 29-36, 2015.

[24] D. Vera, A. West, and R. Harrison, "Innovative virtual prototyping environment for reconfigurable manufacturing system engineering," *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture,* vol. 223, pp. 609-621, 2009.

[25] *Delmia 3D digital manufacturing solution, Dassault Systems*. Available: http://www.3ds.com/products-services/delmia/

[26] "Technomatix digital manufacturing tool, Siemens."

[27] M. B. Raza, "A knowledge based approach to integration of products, processes and reconfigurable automation resources," PhD, Loughborough University, 2012.

[28] B.-K. Min, Z. Huang, Z. J. Pasek, D. Yip-Hoi, F. Husted, and S. Marker, "Integration of real-time control simulation to a virtual manufacturing environment," *Journal of advanced manufacturing systems,* vol. 1, pp. 67-87, 2002.

[29] S.-H. Suh, Y. Seo, S.-M. Lee, T.-H. Choi, G.-S. Jeong, and D.-Y. Kim, "Modelling and Implementation of Internet-Based Virtual Machine Tools," *The International Journal of Advanced Manufacturing Technology,* vol. 21, pp. 516-522.

[30] U. Dietrich, T. Schulz, and N. Yaramanoglu, "Bringing real and virtual worlds together in the manufacturing process," *Journal of Advanced Manufacturing Systems,* vol. 1, pp. 51-65, 2002.

[31] *Protege homepage*. Available: http://protege.stanford.edu

[32] J. Kola and A. Rector, "Importing Spreadsheet data into Protégé: Spreadsheet plug-in."

[33] M. Abe, T. Oku, Y. Numao, S. Takaichi, and M. Yanagisawa, "Low-cost FC stack concept with increased power density and simplified configuration utilizing an advanced MEA," *SAE International Journal of Engines,* vol. 4, pp. 1872-1878, 2011.