2013 21st Mediterranean Conference on Control & Automation (MED)
Platanias-Chania, Crete, Greece, June 25-28, 2013

OntoMoPS: A Modular Production System Description Ontology

George J. Tsinarakis, *Member, IEEE* and Chrisa J. Tsinaraki, *Member, IEEE*

Abstract— **A modular Petri net based methodology for the modeling, analysis and performance evaluation of production systems has been introduced in our previous works. According to this methodology, production systems are decomposed in a small number of fundamental subsystems and the models of the individual subsystems are connected to synthesize the overall system model. The modeling and further processing of the production systems can be accomplished using standard Semantic Web technologies. This allows the utilization of standard tools and supports the interoperability of different production system designs.**

In this work we introduce an ontology for the semantic description of modular production systems, which is the basis for the development of a semantic-based framework for modular production system simulation and processing.

I. Introduction

In our previous works [1] [2], we have introduced a general methodology for the modeling, analysis and (performance and property) evaluation of production systems using Petri nets and their continuous variations and extensions (Continuous and Hybrid). According to this methodology, a complicated system of any complexity and topology is decomposed in a small number of fundamental subsystems following a well defined procedure. The number and the types of the fundamental subsystems used (called from now on modules) as well as the connections between them are defined according to the topology, the features and the events that take place in the production systems under study. The models of the individual subsystems that have been examined with respect to their properties, their behavior and their complexity, are then appropriately connected to synthesize the model of the overall system. This overall model preserves the main properties and characteristics of the fundamental module models.

In general, the use of fundamental modules allows the robust modeling, processing, simulation, as well as property and performance analysis of complicated production systems. Modular modeling allows the reuse of the fundamental module models in order to represent a significant number of systems, following some well-defined steps that make the modeling and system study procedures safer and more efficient. In addition, a modular modeling approach reduces the overall complexity, as the overall production system modeling and study problem is split in small and simple problems that are easier to handle using existing mathematical tools without facing problems of increased complexity (e.g. state space explosion) [3]. Also, it is easier to remove from the performance of the model unwanted events (e.g. certain conflict types) [4].

In the present work, we propose the use of standard Semantic Web technologies for the modeling and further processing of the production systems. In particular, the Semantic Web languages – i.e. OWL/RDF [12][13][14] – can be used for the production system design specification. Such a description supports the interoperability of different production system designs. In addition, it allows the utilization of standard Semantic Web tools (i.e. reasoners) for the model validation. It also allows to perform reasoning on the production models, which automatically calculates the values of some of the model properties (for example, the tools used and the setup values in a specific machine in order to perform a specific process on a specific raw material). The utilization of domain ontologies in production systems has recently gained focus [5], but there has not yet been developed an ontology that describes in detail the production systems. In the ontology proposed in [6] the production process is described in some depth in the context of the MSE model that allows inter-enterprise collaboration; the ontology of [6], though, is not adequate for the detailed description of the modular production systems. This is due to the fact that the MSE ontology describes the production process as a part of the overall performance of an extended enterprise, consisting of different enterprises that cooperate in order to produce an overall product. In this case, the production by itself is not the main concern, since much attention must be paid on the communication issues between partners as well as on managerial, marketing and sales-related decisions. Our point of view is exclusively related with the production process and its most important features and details that have to be taken in account.

In this paper we introduce *OntoMoPS* (*Ontology for the description of Modular Production Systems)*, an ontology that allows the detailed semantic description of modular production systems. OntoMoPS is the basis for the development of a semantic-based framework for the representation of modular production systems and their further processing. OntoMoPS also allows the semantic-based description of the production system research in Digital Library environments and thus provides advanced search capabilities within the digital library contents.

The rest of this paper is organized as follows: The fundamental concepts of the modular production systems are

G. J. Tsinarakis is with CAM Laboratory, Department of Production Engineering and Management of Technical University of Crete, Greece. (corresponding author to provide phone: 30-28210-37306; e-mail: tsinar@dpem.tuc.gr).

C.J. Tsinaraki is with MUSIC, Department of Electronic & Computer Engineering, Technical University of Crete, Greece (e-mail: chrisa@ced.tuc.gr).

presented in section II, the OntoMoPS ontology is described in section III and the paper concludes in section IV, where our future research directions are also outlined.

II. Modular Production System Fundamentals

Productions systems (or manufacturing systems) are networks of machines and buffers that are appropriately connected (according to a given topology and sequence) in order to perform certain types of processes that make possible the production of final products with given characteristics (quality, geometry, size, material used, surface texture etc.). The parts receive an operation in each machine and between operations they wait in buffers of given finite capacity to be served [7]. This means that the machines are the active components of the system that perform actions, while the buffers are passive components that keep parts between the processes. The machines use different types of tools to perform the processes. The tools are selected with respect to specific factors, such as the material of the parts that will be processed, the initial and final part geometry and the features and capabilities of the machine (such as the speed, the number of iterations, accuracy etc).

The general category *machine* describes different types of equipment that perform added value processes in order to transform initial parts to final products. For example, in machinery the most common types of machines are lathes, mills, drills, grinders that can be CNC (Computer Numerical Control) or traditional ones. For each process to be performed different sets of tools are used (as an example, a commercial catalog for aluminum machining is available in [8]). The machines use different sets of tools with respect to the material of the parts, the processes performed, the desired surface texture etc. In addition, ancillary equipment is used (that is, non productive equipment necessary for the processes to be performed efficiently). Such equipment is used during the production process to create the appropriate conditions (e.g. pressure, temperature, cleaning the environment or the parts etc) or to move parts from the machines to the buffers and vice versa (material handling equipment or transportation equipment such as conveyors, AGVs, monorails and lift trucks).

The state of the machines is generally considered to be discrete, and the state of the overall system is described from the states of the machines and from the levels of the buffers. Different types of events with most important the machine breakdowns, machine starvation due to empty buffers and machine blockage due to full buffers change the state of the machines and of the production system. This is the main reason for which discrete event based techniques such as automata [9] and Petri nets and their variations [10][11] are very popular tools in the relative research literature for the modeling and study of such systems. These events cause disturbances in the production process and are responsible for loss of production capacity, increased machine idleness and reduction of individual machines as well as for the reduction of the overall system productivity. When such an event happens in a part of a production system, the disturbance disperses in the previous and in the following machines, causing bigger and bigger problems as time passes and this is not fixed (for example, a full buffer does not allow to the previous machine to send the processed part and start processing another one. If the previous buffer is also full, the machine before it will face the same problem soon. Similarly, a starved machine leads soon to an empty buffer after it and to more starved machines soon). The disturbances because of machine breakdowns are in fact the main reason for the addition of buffers between machines, as they reduce the effects of these events and increase the autonomy of the individual machines. Other less important events that cause disturbances in a production system may be met, such as the non availability of transportation equipment especially in the case that the same equipment is used in different parts of the factory (resource sharing), when resource allocation problems arise or sometimes because of the absence of raw materials or tools. However, these cases will be ignored here.

In our previous works we have classified the production systems to different categories according to the production strategy concerning the use of the machines. In particular, we have considered and studied analytically the cases of dedicated production systems and multi operational production systems. In the first case, each machine performs one type of process during the whole working period and sends and receives parts in the same input and output buffers (the routing of all the parts in the system is the same). This type of production system has low flexibility and is typical for the cases of systems that produce one product, or a small number of variations of this product (according to customer needs) in big quantities. In this case, the setup times are low and the cost of building such a system is high. The machine setup describes the time and the actions needed to change all the necessary parameters of the machine in order to perform the specific operation (set of tools used, speed of the machine, orientation of the part, description of the tools movements and many more).

In the case of multi operational production systems, the machines divide their production capacity and time for the production of different types of products (not in parallel). The second case is much more complicated as the machines have different combinations of sets of input and output buffers according to the process performed at each time instant. As a consequence, different routings of parts in the system are possible and the use of advanced material handling equipment is necessary. In addition, different machine setups are associated with the performance of different processes from a machine and different sets of tools are necessary for each machine in order to perform the different types of processes.

Except from these two types of production systems that we are mainly interested in this paper, there exist some more. In batch production systems the performance of a process may start only when the whole batch of raw materials is available and at that time all the parts are concurrently transformed to final products. In the Flexible Production systems different types of manufacturing flexibility (routing, operation, machine, product etc.) increase the number of the production alternatives (e.g. sequence of process performance, processes performed in each machine) and of the flexibility of the system under study.

III. THE ONTOMOPS ONTOLOGY

We present here the OntoMoPS ontology that we have developed for the representation of modular production systems in the Semantic Web environment.

The development of the OntoMoPS ontology has been influenced by the *METHONTOLOGY* ontology engineering methodology [15]. In particular, we have performed the METHONTOLOGY *Development-Oriented Activities* (i.e. *specification*, *conceptualization*, *formalization* and *implementation*), which are presented, together with their outcome, in the following paragraphs.

Specification. The specification activity states which are the mission (i.e. why the ontology is developed), the usage and the intended end-users of the ontology [15].

The OntoMoPS ontology has been developed for the representation of modular production systems. It will be used in order to allow further processing of the modular production system models using standard Semantic Web tools as well as the semantic-based description of the production system research in Digital Library environments. Its intended end-users will be production system engineers and researchers that will interact with it either while processing a production system model or while interacting with Digital Libraries.

Conceptualization. The conceptualization activity structures the domain knowledge as meaningful conceptual models [15].

The fundamental entities of the modular production systems are represented by concepts in the conceptual model of the OntoMoPS ontology. The major concept in our conceptual model is the *ProductionSystem*, which represents production systems. A production system takes a set of inputs and produces a set of outputs. A production system input is a raw material part, represented by the *RawMaterial* concept and an output is a product part, represented by the *Product* concept. Both *RawMaterial* and *Product* specialize the *Part* concept, which represents all the parts that may exist in a production system.

The *PartiallyProcessedPart* concept, which also specializes the *Part* concept, represents all the parts that have received some processes (can not be considered anymore as raw materials) but not all the necessary processes that will transform them to final products (they have received a subset of the processes needed). A production system is associated with its input and output parts through the *productionSystemInput* and *productionSystemOutput* attributes, of type *RawMaterial* and *Product* respectively. In addition, a production system has a set of buffers, which are represented by the *Buffer* concept, and is associated with them through the *hasBuffer* attribute. The production system is associated with the buffers that contain the raw materials through the *rawMaterialBuffer* attribute and with the buffers that contain the final products through the *productBuffer* attribute. In Fig. 1, a production system of given topology is presented. In this, Buffers 1 and 2 are raw material buffers (Production System inputs), buffers 7, 8 and 9 are final product buffers (Production System Outputs) and buffers 3, 4, 5 and 6 contain different types of partially processed parts, while there are 5 machines with input and output buffers as shown in Fig. 1.

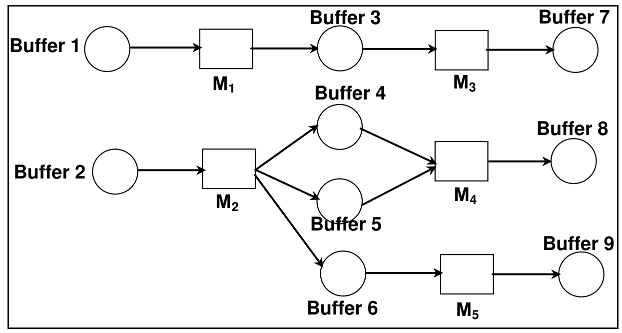

Fig. 1. Production System representation example

The production systems also have a set of resources, which may be machines or ancillary equipment. The resources are represented by the *ProductionSystemResource* concept and the concepts that specialize it; they are associated with the production system through its *hasResource* attribute. There are many types of ancillary equipment, according to the specific needs of the system under study. Some popular types of ancillary equipment met include dryer machines (represented by the *DryerMachine* concept), stoves (represented by the *Stove* concept), air supply, water supply, oil supply and electricity supply equipment (represented, respectively by the *AirSupply*, *WaterSupply*, *OilSupply* and *ElectricitySupply* concepts).

The production systems perform certain processes on their inputs that lead to the produced outputs. A production system is associated to a process, which is represented by the *Process* concept, through the *process* attribute. A production system is also associated with the production strategy it implements, represented by the *ProductionSystemStrategy* concept, through the *productionStrategyModule* attribute. A production strategy consists of modules represented by the *ProductionStrategyModule* concept, which are associated with the production strategy through the *module* attribute. The *ProductionStrategyModule* concept is specialized by the *BatchStrategyModule*, *DedicatedStrategyModule*, *MultiProductiveStrategyModule*, *FMSStrategyModule* and *HybridStrategyModule* concepts that represent, respectively, the modules of the batch, dedicated, multi-productive, FMS and hybrid production systems. There are four types of dedicated strategy modules (*TransferLine*, *Assembly*, *Disassembly* and *ParallelMachine*), and three specializations of multi-productive modules (*MultiProductiveMachine*, *MultiAssembly* and *MultiDissassembly*) that have been decided based on the geometry of the production system as it is defined from the number of inputs and outputs of each machine.

Finally, a production system has the attributes *throughput*, *meanMachineIdleness* and *meanWaitingTime* that specify the system throughput, the mean idleness of the production system machines and the mean waiting time in the system. These attributes describe some of the most popular, according to the literature, production system performance measures.

The production system resources are represented by the the *ProductionSystemResource* concept and its specializations *Machine* and *AncillaryEquipment* which represent, respectively, the machines and the ancillary equipment. A *Machine* has the attributes *machineInput*, *machineOutput*, *previousBuffer*, *nextBuffer*, *productionStrategy*, *tool* and *machineState* that represent, respectively, the machine input, the machine output, the buffer containing the machine input, the buffer that receives the machine output, the production strategy implemented by the machine, the tools that a machine uses to perform the necessary processes in the given raw materials and the current state of the machine (i.e. starved, blocked, under repair and processing). It also has the attributes *CNC*, *idleness*, *meanOutputRate*, *productionCapacity*, *setup* and *setupTime* that specify, respectively, if a machine is CNC, its idleness time percentage, its mean output rate, its production capacity, its setup and its setup time.

A *Buffer* has the attributes *bufferInput*, *bufferOutput*, *previousMachine*, *nextMachine*, *organizationStandard* and *bufferLevel* that represent, respectively, the buffer input, the buffer output, the machine that produces the buffer input, the machine that receives the buffer output, the organization standard followed by the buffer (i.e. one of the LIFO, FIFO, priorities and random strategies) and the buffer level (i.e. one of the Empty, Full and Partially Filled). It also has the attributes *capacity* and *meanQueueLength* that specify, respectively, the buffer capacity and its average queue length.

The events that may occur in a production system are represented by the *Event* concept and its specializations *BufferEmpty*, *BufferFull*, *MachineBreakdown* and *MachineRepaired*. They represent, respectively, the events of a buffer being full, of a buffer being empty, of a machine breakdown and of a repaired machine. Notice that the events that cause disturbances in the production are closely related with the machine state and the buffer level and are not considered here separately.

It has already been mentioned that the processes performed by the production systems are represented by the *Process* concept and its specializations. *Process* specializes the *Service* concept, which represents the services offered in the context of the production system, including, in addition to the processes, the transportation services that are represented by the *Transportation* concept. The most popular types of processes in production systems are cutting, milling or lathing and they are represented, respectively, by the *Cutting*, *Milling* or *Lathing* concept. However, there are also other process types that are specialized according to the production needs of certain product types that have not been taken into account in the current design of the OntoMoPS ontology. A process is performed from the machine using the suitable set of tools, on a raw material that has specific input geometry and results on one or more products from the same material with specific output geometry, size and surface texture. The raw material, the input geometry, the output geometry, the size and the surface texture of the process are represented by the attributes *rawMaterial*, *inputGeometry*, *outputGeometry*, *size* and *surfaceTexture*.

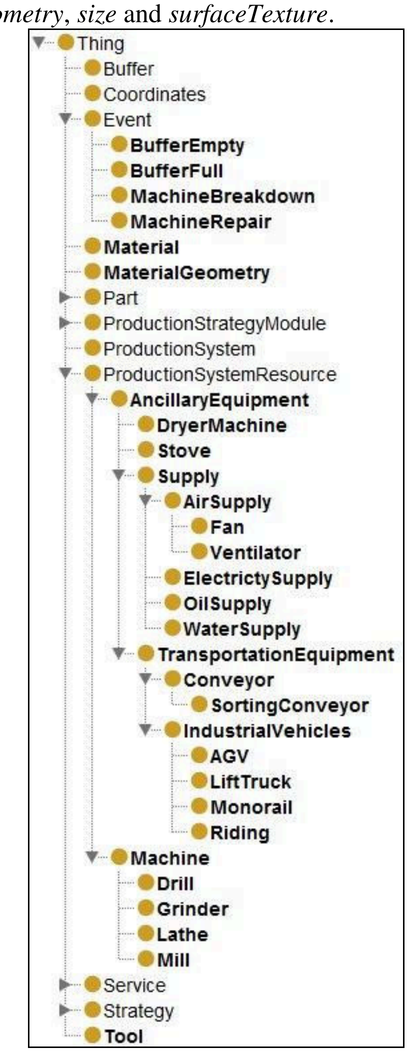

Fig. 2. Overview of the OntoMoPS ontology concept hierarchy. The sub-hierarchies of the ProductionSystemResource and Event concepts are fully expanded.

An overview of the concept hierarchy of the OntoMoPS ontology is presented in Fig. 2, Fig. 3 and Fig. 4.

Formalization. The formalization activity transforms the conceptual model in a formal or semi-computable model [15].

The conceptual model of the OntoMoPS ontology has been formalized using the OWL DL semantics. As a consequence:

a) The concepts of the OntoMoPS ontology conceptual model have been formally described as OWL-DL classes. For example, the *ProductionSystem* concept has been formally described by the homonym OWL-DL class.

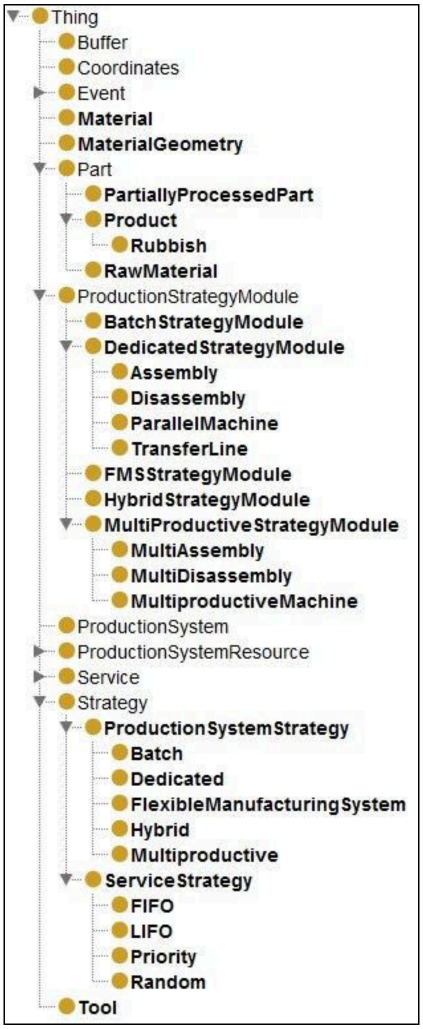

Fig. 3. Overview of the OntoMoPS ontology concept hierarchy. The sub-hierarchies of the Part, ProductionStartegyModule and Strategy concepts are fully expanded

b) The simple type attributes have been formally described as OWL-DL datatype properties. For example, the *capacity* attribute has been formally described by the homonym OWL-DL datatype property.

c) The complex type attributes have been formally described as OWL-DL object properties. For example, the *previousBuffer* attribute has been formally described by the homonym OWL-DL object property.

Implementation. The implementation activity builds computable models in an ontology description language [15].

In order to build a computable model in OWL, we have decided to express the OntoMoPS ontology in the RDF syntax of OWL-DL (OWL-DL is the OWL specie that guarantees decidability).

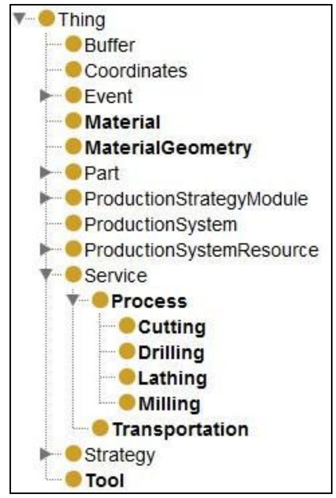

Fig. 4. Overview of the OntoMoPS ontology concept hierarchy. The sub-hierarchy of the Service concepts is fully expanded

As an example of production system description using the OntoMoPS ontology, consider a very simple production system *S* with throughput 10, consisting of an input buffer *IB* of a capacity of 200 items, an output buffer *OB* of a capacity of 100 items and a machine *M*. *S* implements a production strategy *PS* through the process *Pr* for the transformation of a raw material *MT* to the product *P*. The RDF graph that represents the specification of *S* is shown in Fig. 5.

Notice that the production system *S*, the input buffer *IB*, the output buffer *OB*, the machine *M*, the production strategy *PS*, the process *Pr*, the raw material *MT* and the product *P* are represented by homonym individuals of the OntoMoPS ontology classes. These individuals are associated using the OntoMoPS ontology properties (for example, the *Machine* individual *M*, which represents the machine, is associated, through the object properties *previousBuffer* and *nextBuffer*, with the *Buffer* individuals *IB* and *OB*, which represent, respectively, the input and the output buffer). Notice also that the *ProductionSystem* individual *S*, which represents the production system, is associated, through the *throughput* datatype property, with the integer value *10*, which is its throughput value.

The OntoMoPS ontology currently focuses on the two most popular production system categories, namely the dedicated and multi-productive production systems. We plan to extend the OntoMoPS ontology in order to describe in detail the batch, FMS and hybrid production system categories in the future steps of our work.

IV. Conclusions – Future Work

In this paper we have presented OntoMoPS, an ontology that allows the detailed semantic description of modular production systems.

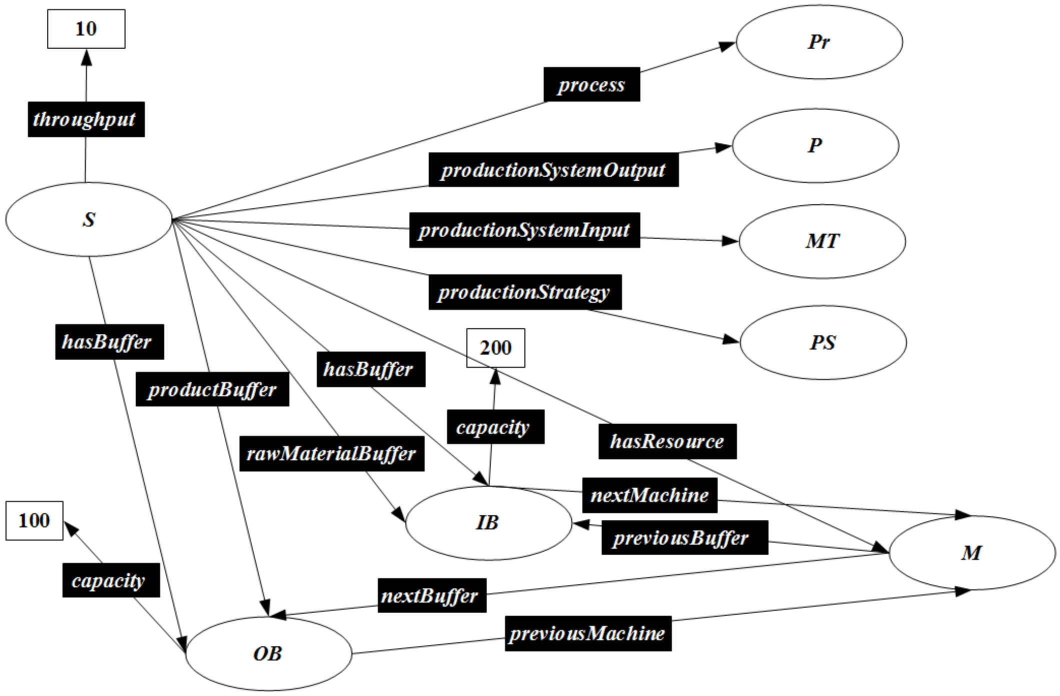

Fig. 5. Production System representation example

In particular, the OntoMoPS ontology focuses on the production process and its most important features and details that have to be taken into account. OntoMoPS allows the representation of modular production systems in the Semantic Web environment and supports their further processing using standard Semantic Web tools (like, for example, reasoners that automatically calculate the values of some of the production system model properties).

OntoMoPS also allows the semantic-based description of the production system research in Digital Library environments, thus providing advanced semantic search capabilities within the digital library contents. The OntoMoPS ontology is the basis of a semantic-based framework that we are developing for the representation of modular production systems and their further processing. We also plan to extend the OntoMoPS ontology in order to describe other types of systems with similar features.

Finally, we plan to provide an OWL 2 [16] version of the OntoMoPS ontology in order to exploit the OWL 2 features for the more accurate representation of the production systems.

References

[1] Tsinarakis G. J., Tsourveloudis N. C. and Valavanis K. P., "Modular Petri Net Based Modeling, Analysis, Synthesis and Performance Evaluation of Random Topology Dedicated Production Systems", Journal of Intelligent Manufacturing, vol. 16, pp. 67–92, 2005.

[2] Tsinarakis G. J., Tsourveloudis N. C. and Valavanis K. P., "Modeling, Analysis, Synthesis and Performance Evaluation of Multi-Operational Production Systems with Hybrid Timed Petri Nets", IEEE Transactions on Automation Science and Engineering, vol. 3, Issue 1, pp. 29 – 46, Jan. 2006.

[3] Katsigiannis, Y.A., Georgilakis, P.S. and Tsinarakis, G.J., "A Novel Colored Fluid Stochastic Petri Net Simulation Model for Reliability Evaluation of Wind/PV/Diesel Small Isolated Power Systems", IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, vol. 40, issue 6, pp. 1296 – 1309, 2010.

[4] Thevenon,L. and Flaus, J.M., "Modular representation of complex hybrid systems: application to the simulation of batch processes", Simulation Practice and Theory, 8(5), 283–306, 2000.

[5] Martinez L. J., Delamer I., Ubis F., "Domain Ontologies for Reasoning Machines in Factory Automation", ISBN 1936007010, International Society of Automation, 2010.

[6] Lin H.K. and Harding J.A., "A manufacturing system engineering ontology model on the semantic web for inter-enterprise collaboration", Computers in Industry, vol. 58, no. 5, pp. 428-437, 2007.

[7] Tsourveloudis N., Dretoulakis E. and Ioannidis S., "Fuzzy work-in-process inventory control of unreliable manufacturing systems", Information Sciences, vol. 127, no 1-2, pp. 69 – 83, 2000.

[8] Plansee-Tizit Co., "Tools for aluminium machining".

[9] Ramirez-Serrano, A., Sriskandarajah, C. and Benhabib, B., "Automata-based modeling and control synthesis for manufacturing workcells with part-routing flexibility", Robotics and Automation, IEEE Transactions on , vol.16, no.6, pp.807-823, Dec 2000.

[10] Zurawski, R. and Zhou M. C., "Petri nets and industrial applications: A tutorial," IEEE Transactions on Industrial Electronics, vol.41, no.6, pp.567 – 583, 1994.

[11] Čopík M. and Jadlovský J., "Utilization of Petri Nets for the Analysis of Production Systems", Procedia Engineering, vol. 48, 2012, pp. 56-64, 2012.

[12] Brickley D. and Guha R. V. (eds.), "RDF Vocabulary Description Language 1.0: RDF Schema", W3C Recommendation, Feb. 2004. http://www.w3.org/TR/rdf-schema.

[13] Manola F. and Milles E. (eds.), "RDF Primer", W3C Recommendation, Feb. 2004. http://www.w3.org/TR/rdf-primer

[14] McGuinness D. L. and van Harmelen F. (eds.), "OWL Web Ontology Language: Overview", W3C Recommendation, 10 Feb. 2004. http://www.w3.org/TR/owl-features

[15] Fernández, M.; Gómez-Pérez, A.; Juristo, N., METHONTOLOGY: From Ontological Art Towards Ontological Engineering. Symposium on Ontological Engineering of AAAI. Stanford (California). March 1997.

[16] W3C Consortium, "OWL 2 Web Ontology Language Document Overview (Second Edition)", W3C Recommendation, 11 Dec. 2012, http://www.w3.org/TR/owl2-overview/.