Proceedings of the
44th IEEE Conference on Decision and Control, and
the European Control Conference 2005
Seville, Spain, December 12-15, 2005

Formal Specification of ADACOR Holonic Control System: Coordination Models

Paulo Leitão[1], *Member, IEEE*, Armando W. Colombo[2], *SM, IEEE*, Francisco Restivo[3], *Member, IEEE*

[1] Polytechnic Institute of Bragança, Quinta S[ta] Apolónia, Apartado 1134, P-5301-857 Bragança, Portugal
e-mail: pleitao@ipb.pt

[2] Schneider Electric - HUB & Globalization of Technology, Steinheimer Str. 117, 63500 Seligenstadt, Germany
email: armando.colombo@ de.schneider-electric.com

[3] Faculty of Engineering, University of Porto, Rua Dr. Roberto Frias, P-4200-465 Porto, Portugal
e-mail: fjr@fe.up.pt

Abstract— **The formal specification of agent-based and holonic manufacturing control systems assumes a critical role in order to understand and synthesize those complex systems. This paper presents the formal specification of the coordination models for the ADACOR holonic control system. For this purpose, it is used High-Level Petri Nets to model the behavior of individual ADACOR entities, AUML interaction diagrams to represent the interaction between those entities and mailboxes structures to synchronize the evolution of the Petri net models associated to the holons.**

I. Introduction

Flexible manufacturing systems are asynchronous, stochastic and dynamic environments requiring the development of re-configurable manufacturing control systems that support small batches, product diversity, high quality and low costs, imposed by global markets. Holonic and multi-agent approaches seem to be suitable to face requirements due to their decentralization, modularity, autonomy and re-use control software features.

Holonic Manufacturing Systems (see http://hms.ifw.uni-hannover.de/) translates to the manufacturing world the concepts developed by A. Koestler for living organisms and social organizations [1], and is characterized by holarchies of holons (i.e., autonomous and cooperative entities), which represent the entire range of manufacturing entities. A holon is a part of a (manufacturing) system that has a unique identifier, may be made up of sub-ordinate parts and, in turn, can be part of a larger whole.

ADACOR (ADAptive holonic COntrol aRchitecture for Distributed Manufacturing Systems) is a holonic control architecture that aims to improve the performance of control systems in industrial stochastic scenarios, increasing the agility and flexibility of the enterprise [2].

ADACOR holonic control system is build upon a set of autonomous, cooperative and self-organized holons, each one representing a manufacturing component. ADACOR defines four holon classes: product (PH), task (TH), operational (OH) and supervisor (SH). The PH, TH and OH represent respectively, products, production orders and physical resources available in the shop floor. SH presents innovative characteristics, by introducing coordination in decentralized control. The innovative adaptive ADACOR production control approach balances between a more centralized to a more flat approach, passing through other intermediate forms of control [2]. The presence of SHs in decentralized systems and the presence of self-organization capability associated to each ADACOR holon allows the re-configurability of the control system, combining the global production optimization with the agile reaction to unpredictable disturbances at the shop floor level.

A formal modeling methodology is required to formalize, validate and synthesize the structure and the behavior of holonic control systems, and particularity to analyze the co-operation between distributed holons. The formal tool should capture characteristics like concurrency, asynchronous operations, deadlocks, conflicts solutions and resource sharing, which are inherent to holonic control systems.

The Petri net formalism allows modeling the control system structure and behavior, but also to formal analyze and validate the production system and production control system specifications, based in its strong mathematical background (functional analysis and linear algebra). For more information about the Petri net formalism and tool the authors recommend to read [3-6].

In [7] the ADACOR control system is formally specified using High-Level Petri nets, through the formal modeling and validation of the behavior of each individual ADACOR holon. The ADACOR holons, as parts of a distributed real-time system, need to interact in order to achieve global objectives. This paper describes the specification of coordination models between ADACOR holons using the Petri net formalism. One of the designed coordination models will be then formally analyzed and some important specifications of ADACOR validated.

The paper is organized as follows: first, section 2 describes briefly the modeling of the behavior of ADACOR entities using High-Level Petri nets. After highlighting the need to count with a mechanism that will support the interaction between ADACOR entities, section 3 gives the fundamentals about mechanisms for the interaction between Petri net models. Section 4 describes the specification of some coordination models that support the ADACOR control system and the section 5 presents the analysis and validation of one of those designed coordination models. Finally, section 6 rounds up the paper with conclusions.

II. FORMAL MODELING OF ADACOR CONTROL SYSTEM

The formal modeling of holonic control systems provides a comprehensive view of system specifications and functionalities, thus playing a key role in its design and later in its implementation.

This section makes an overview of the formal modeling of three ADACOR entities using High-level Petri nets, namely the PH, TH and OH, which will be used during the designing of the coordination models. A detailed description of the modeling of all ADACOR entities can be found in [7-8].

A. Product Holon Model

A *product holon* represents an available product to be produced by the factory plant and its behavioral model is illustrated in the Petri net model of the Fig. 1 [7].

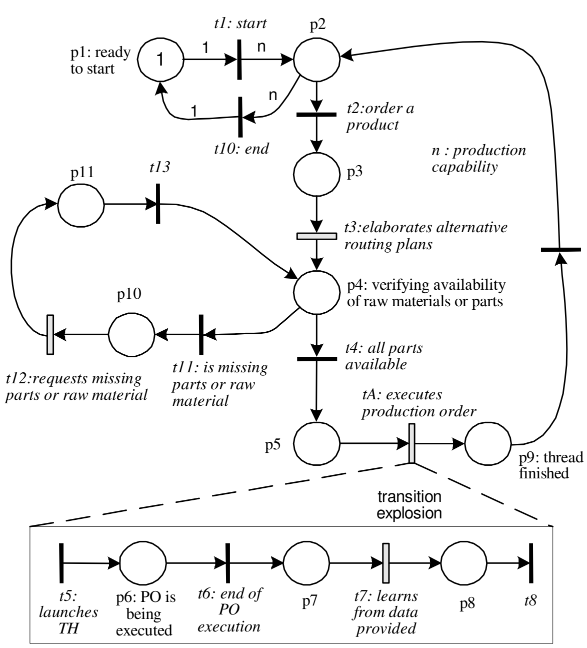

Fig. 1 Petri net model of the product holon

Each new order will generate a new thread to handle its execution, comprising mainly the short-term process planning, management of the sub-parts and management of the production order execution. The PH continues waiting for new orders, being able to process simultaneously several orders, limited by the production capability *n*.

The transition *tA* that represents the execution of the production order (PO), can be exploded in a sub-Petri net model, also illustrated in the Fig. 1. In this activity, the PH launches a TH that will be responsible for the management of the production order execution.

B. Task Holon Model

Each available production order, launched to produce a product, is represented by a *task holon*. The behavior of a TH comprises mainly the order decomposition, resource allocation planning and plan execution activities, as illustrated in the Petri net-model of Fig. 2 [7].

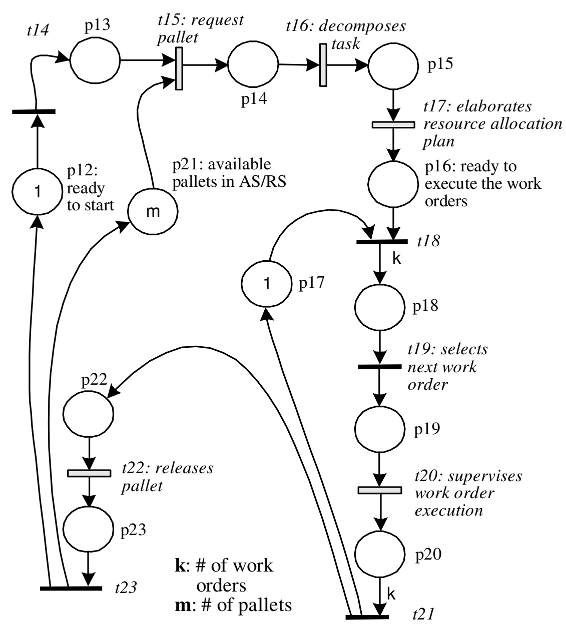

Fig. 2 Petri net model of the task holon

A production order comprises the execution of *k* work orders. After the allocation of all work orders, the TH starts the execution of the plan by selecting the next work order to be executed, and triggering the supervision of the work order execution, modeled by the transition *t20*. Once the work order is started, the control is given to the OH, and the TH waits for the end of the work order execution.

After the execution of all work orders, the TH releases the pallet and transfers the relevant information about the production order execution to the PH.

C. Operational Holon Model

The *operational holons* represent the physical resources, such as operators and robots. The OH model, represented in Fig. 3 [7], contains several sub-behaviors that are handled asynchronously in parallel, so that the execution of one process doesn't block the execution of another process. For example, when executing a work order (modeled by *t19*), the OH can monitorise its execution (transition *t8*) or can handle the announcement of new work orders (transition *t10*).

The work order to be executed is selected according to the local schedule, the availability of the buffer (place *p21*) and the state of the machine (place *p20*), being then physically executed as modeled by the timed transition *t19*.

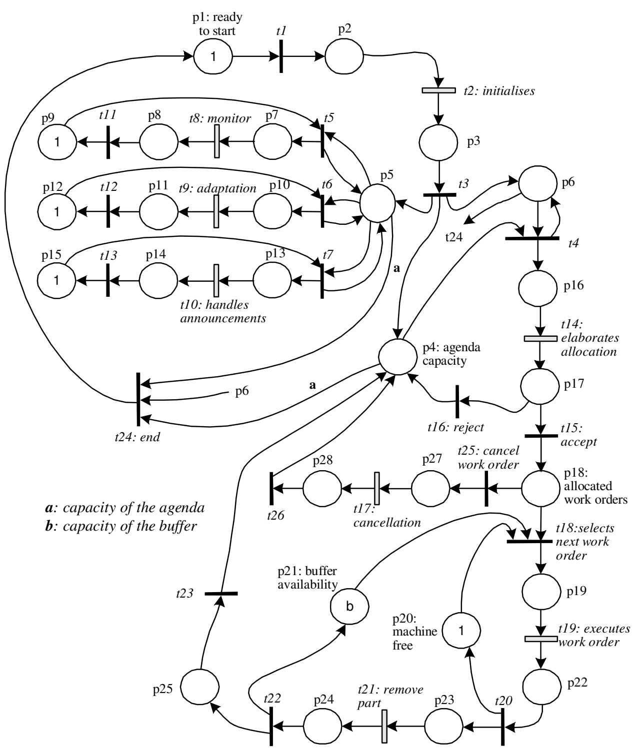

Fig. 3 Petri net model of the operational holon

When its execution finishes, the resource returns to the idle state, being able to initiate the execution of another work order. In this case, the part is transferred from the current machine to the next machine, according to the schedule plan.

III. Basics of Coordination Models in Petri nets

ADACOR holons are real-time systems that need to interact in order to achieve global objectives. The interaction between distributed ADACOR holons is modeled using AUML interaction diagrams [9], which are an extension of the UML's sequence diagrams for the multi-agent systems. Fig. 4 illustrates two interaction diagrams, from those developed for the ADACOR control system [8], which illustrates the interaction between the PH and TH, and the interaction between OHs to synchronize the operation of physical resources.

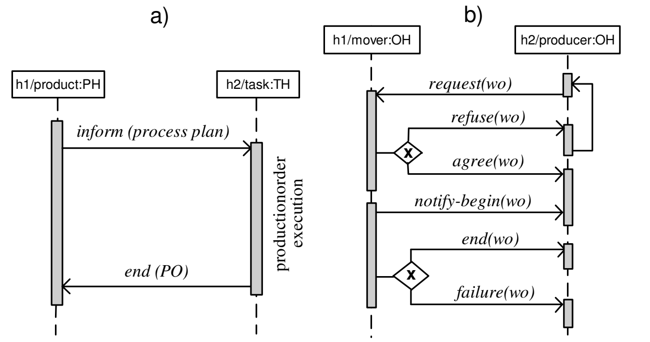

Fig. 4 Interaction diagrams: a) between product and task holons, b) between operational holons (to synchronize the operation of physical resources)

The interaction between ADACOR holons is performed by synchronizing the individual Petri net models. A set of methods have been developed and long time applied for the interactions of real-time systems. Such structures facilitate the synchronization and other kind of interactions like the communication and the mutual exclusion specifications, among others. All these kind of interactions are typical specifications and/or conditions found in a real-time system.

Examples of structures and methods for handling interactions are: MUTEX (Mutual Exclusion Object), Queue Theory, Mailbox, and Rendezvous. All these mechanisms are quite similar. As an example, MUTEX is a concept introduced by [10], whose name comes from the fact that it is useful in coordinating mutually exclusive access to a shared resource. In computer programming domain, a MUTEX object is a program object that allows multiple program threads to share the same resource, but not simultaneously.

Petri net models of real-time systems can use some or all the above addressed methods and structures. Due to the graphic and mathematical form of the Petri net, each of the methods and structures has a particular form of representation, e.g. a place plays the role of a MUTEX or a Mailbox, a sequence of places and transitions plays the role of a FIFO queue, the firing of a transition synchronizes two or more processes modeled by two or more input places to this transition, etc. (see [4]).

In this work, the synchronization of the evolution of each Petri net model uses places playing the role of Mailboxes.

IV. Specification of Coordination Models in ADACOR Control System

The synchronization of the Petri net models can be extracted from the AUML interaction diagrams designed to model the conversations between ADACOR holons. The coordination of Petri net models is illustrated in two different situations: interaction between two distinct ADACOR holon classes and interaction between two holons using the same holon model.

A. Coordination between Product and Task Holons

The first example is related to the interaction between the PH and the TH, as illustrated in the Fig. 5.

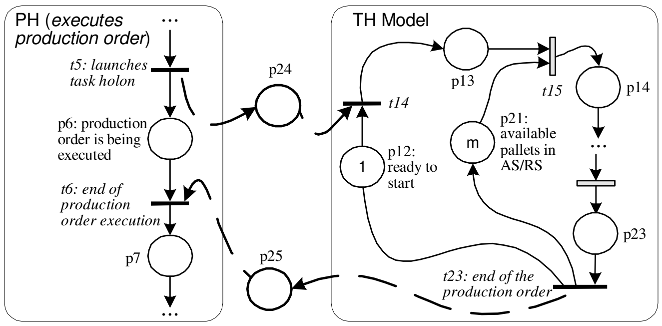

Fig. 5 Coordination of individual models of product and task holons

The transition *tA* of the PH model represents the execution of the production order. In this activity, the PH launches a TH to supervise the production order execution, transferring the control to the TH, and waiting for its conclusion. The notification is logically represented by marking the place *p24* whose output transition is a transition of the TH model.

During the evolution of the net, as soon as the transition *t5* fires, the shared place *p24* as well the place *p6* gets a token. The place *p6* remains busy by a token as long as the TH performs its activities, playing the role of a monitoring place. The existence of a token in this shared place represents the indication that the TH can start its execution.

The operator of the ADACOR system can use the meaning of this place and the meaning of the token that marks the place as source of valuable information for generating model-based monitoring indexes, within a more general supervisory control function.

When the execution of the production order finishes, the TH notifies the PH, providing the relevant information related to its execution, by marking the shared place *p25*. The place *p25* gets a mark only if the TH completes successfully its activity. But, if an error or failure happens, the referenced place will never be marked, and the place *p6* remains marked more time than was planned. Extensions of the model associated with the place *p6*, e.g., the addition of watch-dog structures and sub-nets, allows using these conditions for performing error detection and diagnosis functions. Such kind of structure has been developed and applied to the lower level activity models.

B. Coordination between Operational Holons

The second example is the coordination between a producer and a mover operational holons that interact during the loading/unloading of a machine. These holons use the same operational holon Petri net model, as illustrated in Fig. 6.

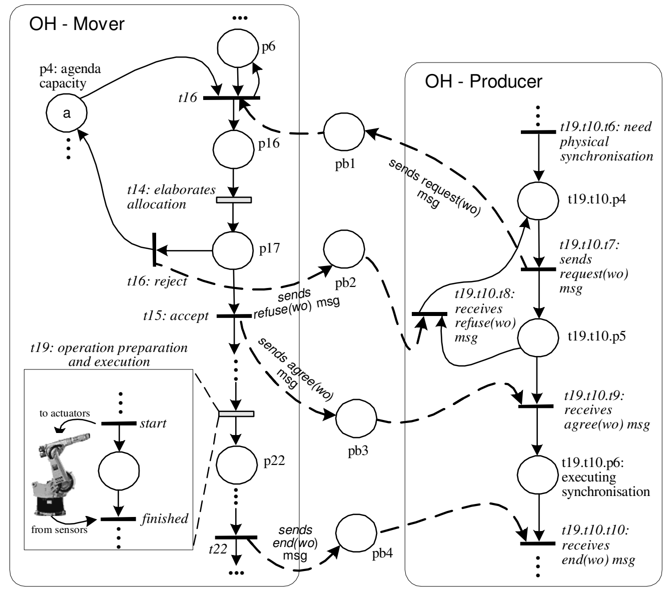

Fig. 6 Coordination of individual models of producer and mover holons

The producer holon requests physical coordination by marking the place *pb1* (which corresponds to the action of sending a *request(wo)* message) that will enable the transition *t16* in the OH model, entering in a state waiting for the response of the mover operational holon (agree or refuse of the work order execution).

The timed transition *t19* models the execution of the mover operation, and can be exploded until the physical control of the machine using stepwise refinement, where the transition "*start"* represents the start of the execution of the operation and the transition *"finished"* represents the end of the execution of the operation. At the end of the auxiliary operation execution, the mover holon notifies it by marking the place *pb4* (which corresponds to the action of sending an *end(wo)* message) that will enable the transition $t_{19}.t_{10}.t_{10}$ in the producer holon model.

The same methodology is applied to all the interactions.

V. Analysis and Validation of Coordination Models

The edition and analysis of the Petri net models for the ADACOR holons, and the elaborated coordination models, have been done using the PASCELL software tool [11].

The qualitative and quantitative analysis of the coordination models in ADACOR control system is exemplified for the interaction between the product and task holons due to the easy understanding that it provides.

A. Qualitative Validation

The qualitative analysis is based on discrete-event simulation and structural analysis of the matrix representation of the net. It allows the verification of the structural and behavioral properties of the model, extracting conclusions about the operation of the system. In this work, the boundedness, reversibility, liveness and conservativeness behavioral properties are analyzed using linear algebra methods, provided by the PASCELL software tool.

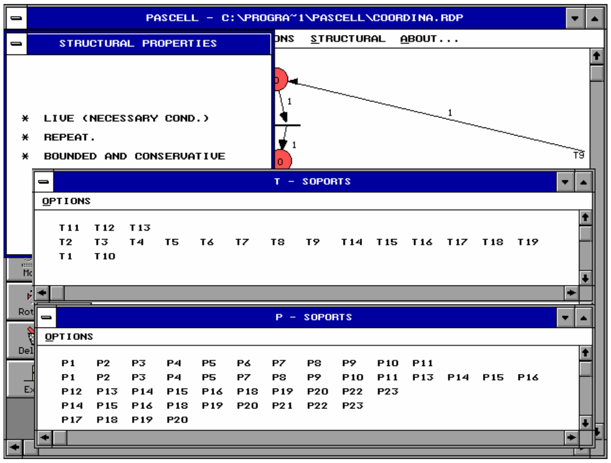

Fig. 7 Qualitative analysis of the coordination model

The structural analysis for the referred coordination model is illustrated in Fig. 7 and shows that the model is deadlock free, i.e. each individual Petri net evolves according to its behavior without stopping in an undetermined intermediate state. It is also possible to verify that the Petri net model is repetitive, which means that the Petri net returns to the initial state, through well defined work cycles in the execution of production orders. The Petri net-model is conservative, which means that the tokens do not disappear neither new tokens are created, during the holon's life-cycle.

The analysis of P-invariants allows confirming mutual exclusion relationships among places and resources involved in the coordination model structure and behavior. The analysis of the 5 P-supports (place invariants), Fig. 7, allows validating, e.g., the following specifications:

- The $\{p_{12}, p_{13}, p_{14}, p_{15}, p_{16}, p_{18}, p_{19}, p_{20}, p_{22}, p_{23}\}$ P-Support represents formally the parallel mutual exclusion relationship representing the process of execution of a production order.
- The $\{p_{17}, p_{18}, p_{19}, p_{20}\}$ P-Support represents formally the parallel mutual exclusion relationship representing the process of the physical execution of the set of operations (execution of the process plan). Moreover, the transition *t18* fires when the set of operations are able to be executed in the resources, modeled by transition *t20*. The end of execution of all operations belonging to the production order is verified by transition *t21*.
- The $\{p_1, p_2, p_3, p_4, p_5, p_6, p_7, p_8, p_9, p_{10}, p_{11}\}$ and $\{p_{14}, p_{15}, p_{16}, p_{18}, p_{19}, p_{20}, p_{21}, p_{22}, p_{23}\}$ P-Supports represents formally, respectively, the PH and TH work cycles.
- The $\{p_1, p_2, p_3, p_4, p_5, p_7, p_8, p_9, p_{10}, p_{11}, p_{13}, p_{14}, p_{15}, p_{16}, p_{18}, p_{19}, p_{20}, p_{22}, p_{23}, p_{24}, p_{25}\}$ P-Supports is an amalgation of the two previous P-Supports and represents the coordination model work cycle, i.e. the complete execution of a product involving the interaction between the PHs and THs.

The structural analysis of the Petri net coordination model of Fig. 7 shows that the coordination model presents 3 T-supports. The analysis of this set of T-supports allows validating, e.g., the following specifications:

- The $\{t_{11}, t_{12}, t_{13}\}$ T-Support is concerned to the sequence of operations related to request the parts, to execute the product, when they are missing, through the invocation of another PH.
- The $\{t_2, t_3, t_4, t_5, t_6, t_7, t_8, t_9, t_{14}, t_{15}, t_{16}, t_{17}, t_{18}, t_{19}, t_{20}, t_{21}, t_{22}, t_{23}\}$ T-Support models the main path of operations, related to the execution of the production order, showing the synchronization between two models: the net evolves from the PH to the TH after the transition *t5* fires, representing the act of a PH launching a TH to supervise the execution of the production order, and evolves from the TH to the PH when the transition *t23* fires, representing the end of the production order execution.
- The $\{t_1, t_{10}\}$ T-Support represents the sequence of operations to launch and remove PHs in the system.

B. *Quantitative Validation*

The quantitative analysis is performed by means of the simulation of the temporized Petri net models, which requires the introduction of the time parameter associated to the transitions (see [11]). For this purpose, deterministic distribution times will be used, since the holons are composed of real hardware/software components with deterministic time behavior.

The timed transitions *t3*, *t7*, *t12*, *t15*, *t16*, *t17* are computational activities being estimated 1 time unit per transition. The transition *t20* represents the activity related to the preparation and the physical execution of the work order. For this study it was considered that this activity takes 10 time units. The transition *t22* includes the transportation of the part to the storage system and the releasing of the pallet, after the execution of all work orders. The time associated to this transition is dependent of the current capacity of the transport system, but for the study is was estimated 4 time units. The other transitions are instantaneous (atomic-firing).

Additionally, it was considered that the production capacity is equal to 3 (i.e. *n* is equal to 3), the number of pallets is 5 (i.e. *m* is equal to 5), and each production order comprises 2 work orders (i.e. *k* is equal to 2).

As in the qualitative analysis, only the quantitative analysis for the interaction between the PHs and THs will be described. The Gantt diagram of Fig. 8 reflects the temporal sequence of the system operation. Moreover, cyclic evolution, existence of bottlenecks, mutual exclusion activities, etc., can be easily discovered and optimization strategies can be proposed and on-line verified.

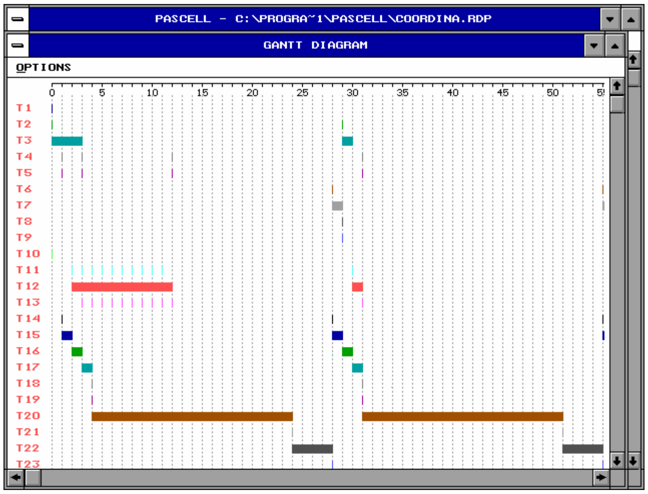

Fig. 8 Gantt Diagram for the Performance Analysis

As an example, Fig. 8 shows clearly the sequence of execution of two production orders. For each production order, a pallet is requested, the task is decomposed and an execution plan is elaborated, each one taking 1 time unit (see respectively the temporal evolution of transitions *t15*, *t16* and *t17*). Then the production order is physically executed during 20 time units (see the evolution of the transition *t20*), because it comprises two work orders and the execution of an individual work order takes 10 time units. After the physical execution, the pallet is released, which takes four time units (see the temporal evolution of the transition *t22*).

Additionally, the simulation of this coordination model allows verifying some important production parameters. As an example, the analysis of the evolution of the number of tokens in the place *t6* of the PH model, allows to determine the number of instances of the same product present simultaneously in the factory plant.

The procedure for qualitative and quantitative analysis described for this coordination model, has been repeated for the other designed coordination models, allowing validating the specifications of the ADACOR system.

VI. Conclusion

The ADACOR holonic control system is build upon a set of autonomous entities, addressing the improvement of agility and flexibility of manufacturing systems. The behavior of individual ADACOR holons are formally modeled with High-level Petri nets taking advantage of the powerful mathematical foundation associated to them.

The ADACOR components have forms of interactions, such as synchronization and communication between holons, and mutual exclusion of resources. This work describes the specification of coordination models between the ADACOR entities, using mailboxes structures to synchronize the evolution of each individual Petri net model. These interactions were developed upon the set of interaction diagrams modeled using AUML tool.

The specification of the coordination models was illustrated with the design of two examples. The formal validation of the coordination models, illustrated using one of designed coordination models, allows a rigorous specification of the ADACOR control system.

The authors are going on with the application of Coloured Petri nets for modeling and validate ADACOR-related control automation systems, in order to solve some gaps associated to High-level Petri nets formalism, such as the increasing of the model in a non-controllable manner when the system presents many instances of the same component. An initial effort to formally model and validate the ADACOR product holon using Coloured Petri nets is discussed in [12].

Acknowledgments

The authors would like to thank the European Commission and the partners of the Innovative Production Machines and Systems (I*PROMS) Network of Excellence for their support.

References

[1] A. Koestler, *The Ghost in the Machine*, Arkana Books, London, 1969.

[2] P. Leitão, A. W. Colombo and F. Restivo, "ADACOR, A Collaborative Production Automation and Control Architecture", *IEEE Intelligent Systems*, vol. 20, no.1, pp. 58-66, 2005.

[3] M. Silva and R. Valette, "Petri Nets and Flexible Manufacturing", LNCS, vol. 424, Springer-Verlag, pp. 374-417, 1989.

[4] T. Murata, "Petri Nets: Properties, Analysis and Applications", *IEEE*, vol. 77, no. 4, pp. 541-580, 1989.

[5] L. Holloway, B. Krogh, and A. Giua, "A Survey of Petri Net Methods for Controlled Discrete-Event Systems", *Discrete-Event Systems: Theory and Applications*, vol. 7, no. 2, KAP, pp. 151-190, 1997.

[6] M. Zhou, *Petri Nets in Flexible and Agile Automation*, Kluwer Academic Publishers, 1995.

[7] P. Leitão, A.W. Colombo and F. Restivo, "An Approach for the Formal Specification of Holonic Control Systems", LNAI, vol. 2744, Springer-Verlag, pp. 59-70, 2003.

[8] P. Leitão, "An Agile and Adaptive Holonic Architecture for Manufacturing Control", PhD Thesis, University of Porto, 2004.

[9] J. Odell, H. V. Parunak and B. Bauer, "Representing Agent Interaction Protocols in UML", LNCS, vol. 1957, Springer-Verlag, 2000.

[10] E. W. Dijkstra, "Solution of a Problem in Concurrent Programming Control", *Comm. ACM*, vol. 8, no. 9, 1965.

[11] A. W. Colombo, "Development and Implementation of Hierarchical Control Structures of Flexible Production Systems Using High-Level Petri Nets", *Manufacturing Automation Series*, K. Feldmann (ed.), MeisenbachVerlag Bamberg, 1998.

[12] P. Leitão, A. W. Colombo, F. Restivo and R. Schoop, "Formal Specification of Holonic Control System ADACOR, using High-Level Petri Nets", in Proceedings of the 1st IEEE International Conference on Industrial Information (INDIN2003), Canada, 21-24 August 2003.