The use of a Mechatronic Systems Simulator in Engineering Courses

Rafael da Silva Mendonça
DEPEG-UFAM
Federal University of Amazonas
Manaus, Brazil
mendoncarms@ufam.edu.br

André Luiz Duarte Cavalcante
DETEC-UFAM
Federal University of Amazonas
Manaus, Brazil
andrecavalcante@ufam.edu.br

Nasser Jazdi
IAS
University of Stuttgart
Stuttgart, Germany
nasser.jazdi@ias.uni-stuttgart.de

Vicente Ferreira de Lucena Jr.
PPGEE-PPGI-UFAM
Federal University of Amazonas
Manaus, Brazil
vicente@ufam.edu.br

Abstract — **This Innovative Practice Work in Progress presents the proposal and the development of a simulator for an Evolvable Production System that aims to represent a complete mechatronic system, simulate its operation, and support the learning of associated subjects. A mechatronic system is composed of mechanical and electronic modules that in turn may be associated with a software layer that is responsible for the intelligence of the entire system. This intelligence comes through interactions among software agents belonging to the software layer. The mechatronic devices are described through Finite State Machines, whose transitions are triggered by mechatronic software agents. The output of the simulator is a list with the set of skill calls, the time in which such calls are made, the respective system's answer, and the set of communication messages exchanged among the agents' peers. By using this simulator in engineering classrooms, it is possible to construct several proposals of mechatronic systems, represent those systems as complete manufacturing processes, and use intelligent software agents to simulate the complete functionality of the designed system. Afterward, with the results obtained from the simulation, students are prepared to implement real systems. The proposed simulator has been used in engineering courses at the Federal University of Amazonas, and the goal of this paper is describing the characteristics of the simulator and the results of its use in engineering disciplines.**

Keywords — Mechatronic Systems, Evolvable Production System, Manufacturing Training, Multi-Agent Systems.

I. Introduction

Modern production systems are formed by complex structures, mechanical and electronic parts, that together enable the production of specific goods and the control of that production process. A better way to make this work is creating a set of modules that can interoperate. Such modules are known as mechatronic modules because they are formed by mechanical, electronic, software, and network components.

A simulation of a system tries to reproduce the behavior of a real system by using a virtual system, e.g., digital computers running computational models. Therefore, modifications done in a simulator guarantees that all changes and their consequences will occur only in the computational model, making it possible to study the desired system and to improve its production and efficiency in a very safe way.

This work shows a simulator of an Evolvable Production System (EPS) develop at the electrical engineering course of UFAM-Brazil [1, 2, 3]. This EPS gets to shop floor the concept of skills and intelligent agents. Simulators for such systems come with the initial idea of reproducing the behavior of mechatronic agents, that is, intelligent agents, running inside mechatronic modules. More specifically, the simulation tries to reproduce the temporal behavior and the functionalities of the mechatronic agents operating under certain conditions.

This behavior is achieved using the mechatronic agents' description and their skills that represent the functionalities of such mechatronic modules. The presented simulator has an educational purpose, follows all agent theoretical principles, and is inspired by real industrial automation systems. After all validation tests and implementations, the goal is to apply this concept in undergraduate classrooms about mechatronic systems, i.e., it is necessary to apply knowledge in practical context.

The simulator is generic and can reproduce part of any mechatronic system. The authors described technical details of this system in a previous paper that was used as the basis for the construction and formalization of the proposal presented in this current paper [4].

II. Background

A simulation allows verifying the behavior of an object or a system in certain contexts that, although not identical to the real, are the most similar possible. Thus, it is possible to correct failures or errors before the experience takes place in real world.

This simulation study consists of several well-defined steps as conceptual problem formulation, behavioral modeling, input and output data analysis, model translation or implementation, verification, validation, and further experimentation [5, 6].

In this work, the simulator is based on intelligent software agents. Multi-Agent Systems (MAS) are dynamic, self-organized and modularized systems able to make adaptations in the process, such as inserting, changing or removing components while the system is operational [7]. In the simulator, it is possible to create several mechatronic agents

SAMSUNG, CNPq, CAPES, and FAPEAM supported this work.

and to model the hardware temporal behavior of their functionalities.

Simulators are subject of extensive current research. For example, in a more recent work, Neves and his colleagues [8] focus on the design and implementation of a software tool that simulates the self-organizing behavior of a mechatronic systems taking into consideration the EPS manufacturing paradigm. In this work, the basis for the development of their tool was built in the evolvable system itself because it allowed the creation and deployment of all the necessary agents of the system.

Lastly, the simulator presented in this paper is a tool that allows the reproduction of the operation of a mechatronic system through a simple XML file-based description of the software agents. In fact, the simulation of the normal operation, functional tests, and eventually improvement proposals or optimizations, is going to result in a set of data presented to the EPS developer.

III. HOW TO WORK AN EPS

As the EPS is based on MAS, the execution flow of the messages inside the MAS makes the system choose different paths of execution to achieve the intended global behavior.

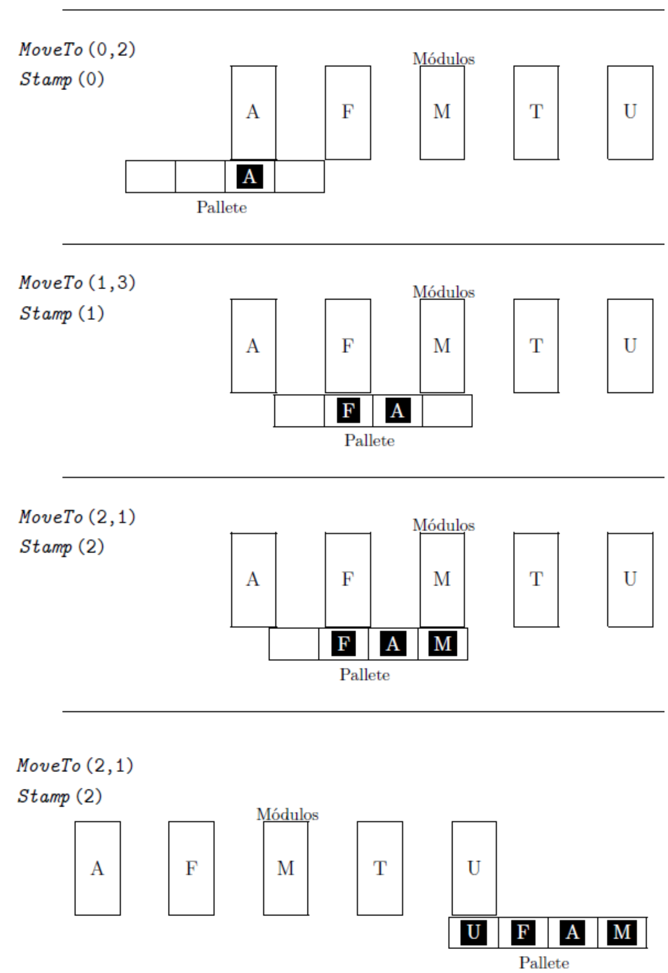

Fig. 1. SIAPE working.

SIAPE is a simple mechatronic system that implements an EPS. In fact, it is the real EPS used in engineering classrooms and was developed at UFAM. It is composed of a set of mechatronic components including one conveyor (treadmill), diverse sensors, and one to five modules that can stamp letters into a sheet of paper loaded on a pallet [9, 10]. In Fig. 1 one can see a diagram of how SIAPE works. The figure is based on the real EPS paradigm, and it will be used to test our simulator.

SIAPE is formed by 1 to 5 *stamp modules*, organized in line, and that can produce anagrams with the combination of these five letters. Such anagrams are the products to be created. Like any other real product, it results in a series of actions of mechatronic modules that execute a production process.

The operation of the systems starts by launching a mechatronic agent called the Anagram agent (Product Agent) that contains the process plan of a single anagram, e.g., the sequence of skill calls that enable the system to stamp each letter of the anagram. When the Anagram agent starts its execution, SIAPE begins to do its skill call sequence described in the process plan.

The process continues until all letters are stamped. In the end, the product is positioned in a specific local to enable the operator to remove the paper stamped, and the operator is free to start another product.

IV. EPS SIMULATOR ARCHITECTURE

The purpose of this simulator is to teach in a direct way the operation of a mechatronic system. Among the learning goals, the students are supposed to practice how to make mechatronic agent's description and their Finite State Machine (FSM) definition. They are also able to analyze the interactions between the mechatronic modules, through their skill calls and the execution of functionalities, all inside only one single platform.

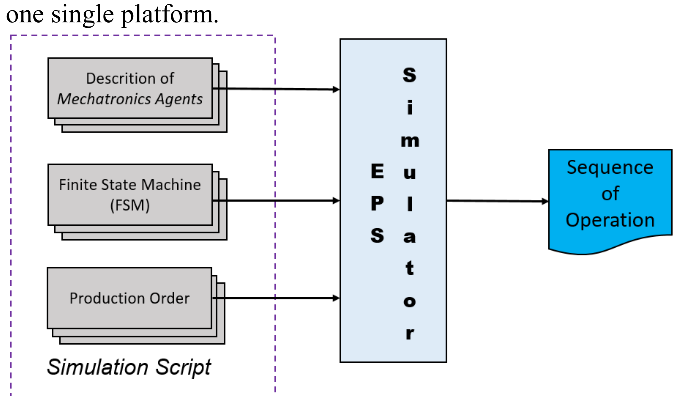

Fig. 2. Simulator Proposal.

The simulator diagram is shown in Fig.2. Its inputs are the description of mechatronic agents, the FSM, and the production order. Its output is the sequence of operations made in the system.

The goal is to reproduce all effects resulting from parameter changes or even by the rise of undesired situations.

The simulator is an interpreter of scripts that can reproduce the behavior of the system given certain conditions [11, 12].

An agent is a piece of software capable of responding to a skill calls or execute them autonomously. A skill is an abstraction of a hardware or software service that the mechatronic system, in which the mechatronic agent runs (or is associated to), can execute.

In the simulator, the model of a mechatronic agent is described by an XML file. By changing the tags contents inside this file, one can modify the specification of the characteristics and functionalities of the agents.

In addition to the set of skill definitions, a mechatronic agent has the real-time hardware behavior, which is defined by an FSM. When a mechatronic agent receives a skill call, then it changes the state inside the FSM; in other words, it enables that the machine evolves in time and executes what it was designed to do.

As the FSM describes the behavior of a mechatronic module and contains information related to its physical operations, then the states are defined according to the description of the real system, in this case, the SIAPE device.

One of the diagrams of the state machine of the system is shown in Fig.3. The initial state of the Conveyor agent is the "conveyor is stopped." When the skill call *MoveTo(x,y)* is received, then the FSM state is changed to execute a sub FSM of this skill. Inside this FSM there is a sub FSM where the state of the conveyor is changed to "conveyor is moving" and it stays there until the "x" module sensor is activated. Then the state of conveyor sub-FSM is changed to "counting position" state. In this state, it counts how many times the sensor is activated; when it achieves "y" the FSM state returns to "conveyor is stopped."

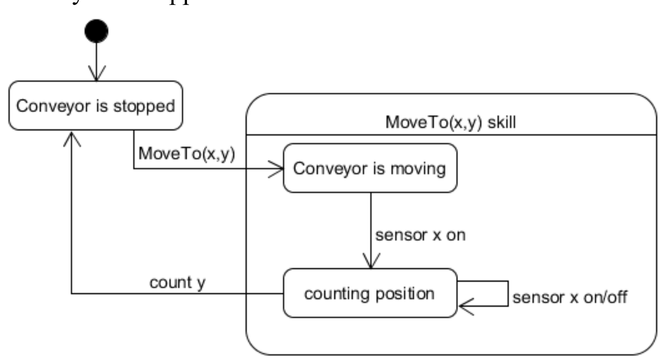

Fig. 3. FSM (Skill Call of *MoveTo(x,y)*).

The production order is registered in XML format files containing the desired sequence of operations, that is, the process plan. The simulator follows this plan with the purpose of representing the operation of the system and its related aspects simply and straightforwardly.

The EPS Simulator reads the input data files with the necessary simulation information; it instantiates the mechatronic agents (MRA) and their respective state machines (FSM). Finally, it instantiates the product agents from the production order information contained in the simulation script.

Each time a new product agent is instantiated, it begins to execute its process plan, and the simulator captures the outputs during execution, that is, it collects and saves the sequence of operation to show the information on the operation of the system. The output is a text with the skill calls among agents, and the state and a time behavior trace of the FSM evolution.

V. Simulator Operation

How the simulated system reacts to the production order is always unpredictable due to the EPS characteristics [12, 13, 14]. The analysis of the output of the simulator enables the production engineer to visualize how the EPS works and how much time it consumes. One can either make changes in the simulated system, simply changing the simulation script, to enable optimization of the used modules (mechatronic agents that run inside the mechatronic tools).

As real mechatronic operations consume time, then this time must represent the real-time behavior of the hardware being simulated. One can associate the FSM with a whole agent or a single skill. The operation of the system reproduced by the simulator follows the steps: once the simulator starts, one Simulator Agent is created, which is responsible for reading the XML file, decoding its content, and finally instating the desired agents.

The simulator runs under EPSCore, the EPS engine responsible by communication among mechatronic agents, that is, the definition and execution of their skill calls. The EPSCore is itself developed in the JADE platform, which provides a set of services that makes the agent iterations easier by mean of FIPA communication protocols [10, 15].

A Production Order is a part of the simulation script that contains the amount and the kind of products to be produced. Therefore, the production order simulates a real production order from a production planning and control department for the simulated system. A production order has, at least, the type and quantity of product agents that will be launched and the time planned to start producing.

Fig.4 shows agents which are responsible for controlling the skill calls addressing two possible execution paths: the simulator path (by using the simulator system), and other in the actual system (by using the real system).

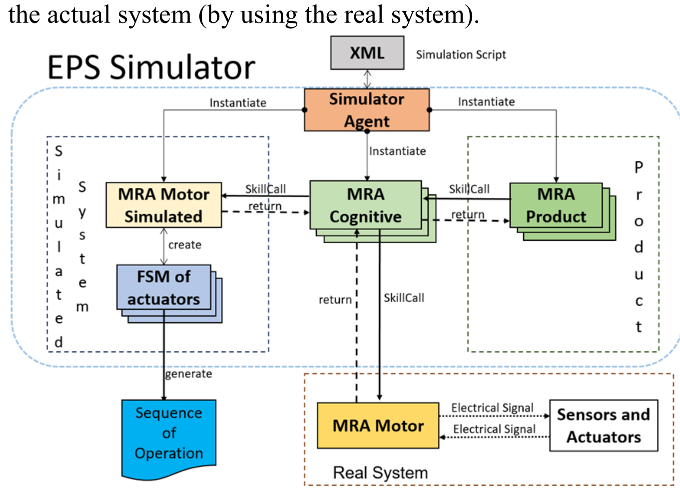

Fig. 4. EPS Simulator.

In the simulator the skill calls, or execution of functionalities, changes the FSM states where time is counted, reproducing real-time aspects of the manufacturing process. In the actual system (real mechatronic system), the skill call occurs directly in hardware triggering actuators or mechatronic modules. Thus, the simulator reproduces the time behavior of the mechatronic agents by following the state machine and the time previously specified for each skill

The MRA Cognitive agent communicating with the MRA Motor Simulated agent or with the MRA Motor agent of the real system has the same procedure since this communication occurs through skills calls.

VI. SIMULATOR USE

The simulator described previously is in use on the Federal University of Amazon, in Amazon State, Brazil, more specifically at Fundamentals of Industrial Automation classes in the Electrical Engineering undergraduate course and in Manufacturing Projects Implementation classes of the Master Program in Electrical Engineering.

The students must develop their theoretical knowledge through practical exercises by using the proposal presented in this paper relating to a real system and its simulation. Lecturers must support new experiences based on the mechatronic system, its model, and description aiming to prepare the students to understand and solve usual problems found in the industrial environment.

The first experience described was planned to use the simulator in the mechatronic system called SIAPE [4]. This system is a didactic platform with all characteristics of EPS with the purpose of producing anagrams according to the letters available in the hardware components.

In the experiments performed, firstly, it was necessary to create an XML file with the FSM description of agents from the real system (the description of the mechatronic agents). The students started the simulation by the reading/parsing of the information they registered on the file. The goal is to teach the students how to operate the mechatronic system and thus how to make a good description. This description aims to represent the time behavior of the mechatronic agents, i.e., it is described the operation of mechatronic modules, its functionalities (services), time execution, characteristics, properties, and extra information about production process.

Given a production order (sequence of production), several anagrams can be produced on the fly. Each anagram type is represented in the system by a mechatronic agent and aims to enable all students to create a complete EPS, using both simulated and real mechatronic modules.

At the classes, after the students had contact with the EPS theory and mechatronic agent programming, they are led to develop some experiments on the real hardware. After that, they are challenged to develop an updated version of the SIAPE with two or three towers, where each tower is a set of 1 to 3 Stamp Modules (in real hardware there is only one tower with 1 to 5 Stamp Modules).

The result was a list with the sequence of manufactured products, time of process, and respective agents' behavior. This process printed the anagrams UFAM with average time 10,264 s and UEA with average time 9,8 s. These times were results of simulation-based in real mechatronic system and confirmed the proper functioning of our proposed simulator.

Initial results showed that there are some difficulties experienced by the students especially related to doing a good description and simulation of the system. Another issue is related to the real-time management because the students do the analysis and try to solve the presented problems without considering important real-time characteristics.

The students analyze, propose improvements, and study the real system with goal to represent the behavior and operation sequence faithfully. Lastly, they may develop other settings, times of execution, agent description, and new hardware architectural proposals, and new systems. In the next steps, the idea is to describe more complex mechatronic systems, improve the obtained results, and propose novel approaches.

VII. CONCLUSION

The development of engineering competences requires the application of technical knowledge in specific contexts linked to the professional practice to support new experiences of active learning aiming to prepare students for solving unusual problems found in an industrial environment

The main objective of this work was to finalize the development of a simulator able to reproduce the behavior of mechatronic systems and to apply their concepts in undergraduate courses. Moreover, the usage of software agents, a new concept in industrial automation, was also planned. These two objectives were accomplished properly, and the resulting system is in use in electrical engineering classes at the Federal University of Amazonas in Manaus, Brazil.

Tests were performed to compare the actual behavior of the system with the simulated one. Thus, we can assert that the developed tool meets the initial expectations for simulating a real mechatronic system.

By using the simulator here presented in our classrooms, we can show the students how to create software agents for real-world applications. It is also possible to describe a complete production process through an XML file, analyze the interactions between agents, and reproduce the functioning of a real system.

The proposed system is functional and is already used in some pilot experiments, in the next months we plan to evaluate the usability of the system. For this experiment, a larger number of students will make use of it during the classes.

ACKNOWLEDGMENT

We would like to thank you all participating students for their valuable suggestions. Thank you to all personnel of the funding agencies and from the Ambient Intelligence Laboratory from CETELI/UFAM.

References

[1] J. Barata, L. Ribeiro, and M. Onori, "Diagnosis on evolvable production systems," pp. 3221-3226, 2007.

[2] M. Onori, and J. Barata, "Evolvable production systems: mechatronic production equipment with process-based distributed control," in 9th IFAC Symposium on Robot Control: SYROCO 2009, pp. 80-85, Gifu, Japan, 2009.

[3] A. Maffei, and M. Onori, "Evolvable Production Systems: Environment for New Business Models," Key Engineering Materials, vol. 467, pp. 1592-1597, 2011.

[4] R. D. S. Mendonça, A. L. D. Cavalcante, and V. F. Lucena Jr., "Proposal of a Simulator for Evolutionary Production Systems (EPS)," in Proc.of the 2017 IEEE 22st International Conference on Emerging Technologies and Factory Automation (ETFA), pp. 1–4, 2017.

[5] W. D. Kelton, and A. M. Law, "Simulation modeling and analysis." McGraw Hill, p. 800, 1999.

[6] IITSEC. Fundamentals of modeling and simulation, interservice /industry training, simulation, and education conference (I/ITSEC) tutorial, Orlando, FL, 2013.

[7] Multi-Agent Systems. In: Objective Coordination in Multi-Agent System Engineering. Lecture Notes in Computer Science, vol 2039. Springer, Berlin, Heidelberg, 2001.

[8] P. Neves, L. Ribeiro, and J.D. Ferreira, "Exploring reconfiguration alternatives in self-organizing evolvable production systems through simulation," in Proceedings of the 2014 12th IEEE International Conference on Industrial Informatics (INDIN), 2014, pp. 511–518.

[9] H. C. C. Amaral, "Sistema Inteligente Ágil de Processo Evolutivo - SIAPE: um protótipo brasileiro de sistemas EPS," Master of Science Thesis, Faculdade de Tecnologia (PPGEE/FT), Universidade Federal do Amazonas (UFAM), p.170, March 2016.

[10] R. D. S. Mendonca, A. L. D. Cavalcante, and V. F. Lucena Jr., "EPSCore: A Didactic Open Architecture of an Evolvable Production System," in Proc.of the 2016 IEEE 21st International Conference on Emerging Technologies and Factory Automation (ETFA), pp. 1–4, 2016.

[11] N. Mustafee, J. Powell, S.C. Brailsford, S. Diallo, J. Padilha, and A. Talk, "Hybrid simulation studies and hybrid simulation systems: Definitions, challenges, and benefits," in Proceedings of the 2015 Winter Simulation Conference (WSC), 2015, pp. 1678–1692.

[12] J. Banks, "The future of simulation." In: Proceedings of the 14th European Simulation Multiconference on Simulation and Modelling - Enablers for a Better Quality of Life. [S.l.]: SCS Europe, pp. 3–9, 2000.

[13] H. Klee, and R. Allen,"Simulation of Dynamic Systems with MATLAB® and Simulink®," Third Edition. Boca Raton: CRC Press. 2018.

[14] R. Merzouki, P. M. Pathak, A. K. Samantaray, and B. O. Boumama, "Intelligent Mechatronic Systems: Modeling, Control, and Diagnosis." Springer – Verlag London, 2013.

[15] A. A. Lüder, and M. Foehr, "Identification and Implementation of Agents for Factory Automation Exploiting Mechatronical Concepts for Production System Structuring," In Industrial Agents, edited by Paulo Leitão and Stamatis Karnouskos, Morgan Kaufmann, Boston, pp. 171-190, 2015.