COMMUNICATION MECHANISM OF DISCRETE EVENTS SIMULATION SOFTWARE AND MECHANICAL PROJECTS SOFTWARE FOR INDUSTRY 4.0

Quézia Manuela Gonçalves Laurindo
manuelaglaurindo@gmail.com

Túlio Almeida Peixoto
tulioap@gmail.com

Joao Jose Assis Rangel
joao.rangel@ucam-campos.br

Ítalo Oliveira Matias
italo@ucam-campos.br

Universidade Candido Mendes
100 Anita Pessanha St, Pq. São Caetano
Campos dos Goytacazes, RJ, 28030-335, BRAZIL

ABSTRACT
Advances in information technology have enabled the evolution of production systems beginning a new age of industrialization, the so-called Industry 4.0. This way, companies need to incorporate those advances and introduce new methods and technologies to stay competitive on the market, providing quality products and active services to support customer demand. Within this context, the proposal of this work was to demonstrate the usage of a discrete event simulation model built with the Ururau software and integrated to the Inventor software applied for mechanical designs. The article discusses the relation between discrete event simulation and new processes in Industry 4.0. The results showed the Ururau's capacity of exchanging data online with a mechanical design software, paving the way for its application in advanced industrial production systems, which actively require computational intelligence resources.

1 INTRODUCTION

According to Azimi and Soofi (2017), the technological advances observed in recent decades have influenced on the customer demands, culminating in increasing productivity. Khan and Turowski (2016) stated that those advances have caused great modifications in the industrial production systems, enabling companies to meet customer needs in different ways, applying new business models to add value to products by means of smart and high-performance production systems.

As claimed by Santos et al. (2017), the term Industry 4.0 has been applied in order to refer to the main technological innovations employed to production processes. The Industry 4.0 makes possible for organizations to create intelligent networks throughout the productive chain, by means of the connection between machines and systems, to control processes in an autonomous way. In this respect, as stated by Turner et al. (2016), discrete event simulations (DES) is an important tool to aid decision makings in smart factories, as it functions as a test platform.

The work of Xu et al. (2016) showed the main demands for the discrete simulation systems driven by the age of Industry 4.0. This article approached the necessity of system modeling with

https://proceedings.science/p/85543?lang=pt-br

more computational intelligence resources to support decision making with greater flexibility and in real time.

Lindskog et al. (2016) and Gironimo et al. (2015) assert that strategies should be found to facilitate the implementation of production systems, such as the application of 3D CAD. That tool is used to support the analysis and assessment of changes before constructing the system, in order to prevent and detect design errors. Rekapalli and Martinez (2011) and Woo et al. (2014) said that CAD resources can be integrated to simulation models to facilitate the model validation process in order to gain their credibility.

In view of this, the aim of this work is to display an online communication mechanism between a DES model and a CAD software applied to develop mechanical designs.

Thus, the purpose is to present an online communication mechanism between a DES model and a mechanical design software. Ururau is the DES software, and Inventor® is the mechanical design software. The 3D CAD Inventor® software is commercialized by Autodesk and is applied in complex projects and product simulation. Thus, this study emphasizes the possibility to perform a high-level communication between two softwares used in process projects and industrial equipment to be applied in a decision support of production information systems in smart factories. In general, a smart production mainly includes digital design and co-simulation of a production line, logistics systems, and production information systems.

This article is organized as follows: in Section 2, the connection between discrete event simulation and new processes for the Industry 4.0 is addressed. Section 3 shows the application of an animation tool in discrete event simulation. Section 4 presents the communication mechanism between the Ururau software and the Inventor by computational experiments. Section 5 discusses the results of the mechanism implemented. Lastly, in Section 6 possible extensions of this research are listed together with the conclusion on the proposed approach.

2 DISCRETE EVENT SIMULATION APPLIED TO THE INDUSTRY 4.0

As stated by Xu et al. (2016), since the 1950s, simulation has been employed to improve complex system projects allowing saving time and resources performing tests before building the physical system. Simulation model can be applied to evaluate various alternative scenarios to select the project that achieves the best performance. Therefore, the simulation model provides reliable estimates, which can be used as parameters for decision makers. Researches on simulation and optimization still need further development to become adequate to support and control decisions in the age of Industry 4.0.

Cheng et al. (2016) analyzed the advances that have been taken place in simulation and highlighted the imitation of human actions in models, recognized as a powerful approach to understanding the behaviour of the system investigated. Furthermore, the development of multiplatform tools, interactively grouped and executed on the internet, as well as the progress in building models using conceptual modeling and hybrid modeling, has been making progress. At the same time, a significant improvement has taken place in the hardware design, which included the connection of various processors, in order to remove memory access bottlenecks. The authors asserted that simulation models are attractive as they are tools for decision making in real time.

On the other hand, Penas et al. (2017) stated that the internet has changed the way research and studies have been carried out, and the way companies and services are organized, leading to a new interaction architecture in the company network. Moreover, the customer demands have also adapted. According to a research performed by Aberdeen Group Benchmark about the current industrial needs, customers want products with better quality and viability, more and more smart and innovative products with higher added value.

According to Li et al. (2015), the Industry 4.0 provides tools that allow meeting those customer demands, promoting strategies, such as the integration among different factories usually located in

https://proceedings.science/p/85543?lang=pt-br

different places. Communication networks perform an important role in the connection between companies, in order to increase flexibility, adjustment, and efficiency of communication between manufacturers and customers. Table 1 shows the general layout of Industry 4.0 with the main components of this structure.

Table 1: Main Components of Industry 4.0.

Main Components	Entities
Application	Smart City, Users, Smart Companies, and Smart Services
Cloud	Cloud Big Data, Databases, Servers and Data Mining
Networks	Cellular Networks, Wired Networks and Wireless Networks
Device	Machines, Robots, Mobile Devices and AGV (Automated Guided Vehicle)

Source: Adapted from Li et al. (2015)

According to Harrison et al. (2016), the Fourth Industrial Revolution, called Industry 4.0, includes the Smart Factories. The purpose of these factories is to employ CPS (Cyber-Physical Systems) systems to monitor physical processes by a virtual model that acts like the physical system and take decentralized decisions. Turner et al. (2016) complemented the idea of Smart Factories. They stated that these factories are composed of intelligent machines connected to networks of data replacement and integrated technologies aiming at supplying more flexibility to production lines in the manufacturing process.

Table 2 displays the four industrial revolutions showing the year that marks the beginning of each one, and the main changes occurred over time.

Table 2: Industrial Revolutions.

Industrial Revolutions	Time	Characteristics
First Industrial Revolution	18th century (1784)	• Mechanical (Hydro – power)
Second Industrial Revolution	19th century (1870)	• Electricity – conveyor belt • Mass production (Ford)
Third Industrial Revolution	20th century (1960)	• Programmable units (Mechatronics) • ICT (PCs)
Fourth Industrial Revolution	21st century (20XX)	• Evolution from existing • Cyber Physical Systems (Intelli – Smart)

Source: Adapted from Khan and Turowski (2016)

Liu and Xu (2017) said that the Fourth Industrial Revolution involves the evolution of the already existing systems, as well as the intensification of the cyber systems applied to the industry, as can be seen in Table 2. Information technology is widely used, also providing efficient solutions to face challenges in manufacturing industry. Taking advantages of those technologies to improve industry competitiveness has become a prominent worldwide tendency.

Those advances, according to Grundstein et al. (2017), provide ways to use the database with the help of the network improvement, efficient organization of production data, and control capacity of the factory floor in real time. Due to the development of a CPS, the process automatic control is becoming increasingly important.

Rahmani (2016) said that, because of the rapid expansion of internet services, automated industries benefit from the standards of data recovery and respond to the system anywhere in the world. This way, a global wired network or wireless sharing usually connects different elements of a given system, such as plants, sensors, and controllers.

https://proceedings.science/p/85543?lang=pt-br

3 APPLICATION OF CAD IN DES

In accordance with Zhang et al. (2010), the outputs generated by the simulation are in general shown in tables and statistical graphics, which sometimes make the understanding of the model difficult. Nevertheless, when integrating 3D models with other management tools, the decision makers can access more information. As stated by the authors, the 3D visualization helps detect deficiencies in the system in question before the constructing phase. In the development of the integration mechanism, an Application Programming Interface – API was used to import graphic data.

Xu et al. (2003) also showed the integration between the AutoCAD software and a DES model. They confirmed that the models need less data entry and provide a higher accuracy degree when performing the integration with 3D CAD designs. Besides, during the design phase, resource and time consumption data can be collected and used to continuously update the model, minimizing the use of assumptions and generalized data.

As stated by Kamat and Martinez (2004), concerning an analysis of operation and design using DES, the capability of 3D visualization of a simulated operation allows the developer to ensure there are no errors in the model codification and enables the decision makers to identify the differences between the way they understand the operation and how the developer perceives it. Once verified and validated, the model can be applied as a tool to aid decision making.

Ezzell and Fishwick (2016) add that the tridimensional visualization of simulated scenarios is applied in various academic studies and industrial projects, in different applications. The users create visualizations to analyze and tune simulation models and communicate ideas to third parties. Those visualization models can aggregate resources and help project decisions. The authors presented an interactive interface model, facilitating the understanding of mathematical models by means of a tridimensional design.

4 URURAU-INVENTOR INTEGRATION MECHANISM

The mechanism proposed here performs online data communication between the DES Ururau software and a 3D CAD software applied to mechanical designs. This way, this mechanism allows an element of an industrial plant, which has been designed to improve the performance of a specific system, to be tested directly in a simulation model before being applied to a real process. Therefore, a DES model has now a broader function, in order that a discrete simulation model, besides enabling analyze the dynamics of stochastic systems, can also evaluate the functioning of the dynamics of a new process element, tested together with another software applied to a mechanical analysis.

Thus, the purpose of this study permits a simulation model to be further developed, in which different aspects of a system can be evaluated. Once the communication between softwares is performed, as in Figure 1, a model with both approaches, DES model and mechanical model, can be obtained. Consequently, the model built from this connection allows evaluating both operations and resources of the system and the mechanical effort of a new device to be inserted, for example, in the plant of a process.

https://proceedings.science/p/85543?lang=pt-br

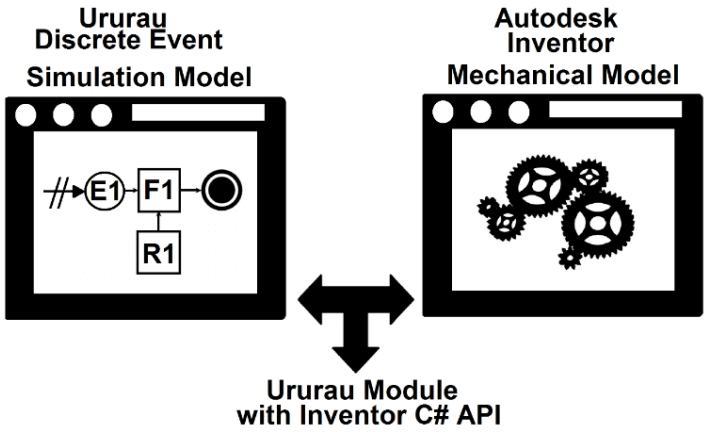


Figure 1: Integration Ururau (DES) - Autodesk Inventor®.
Source: the author

In this work, the discrete simulation environment used was the Ururau. In accordance with Peixoto et al. (2017), Ururau is a Free Open-Source Software (FOSS) of discrete event simulation that acts as a laboratory, in which diverse researchers can use it to test new mechanisms and functions to be applied in discrete simulation. A recent literature research about free softwares for operational research, performed by Dagkakis and Heavey (2015), cited Ururau within a list of forty-four tools with a common purpose. The authors highlighted tutorial videos available on the internet, comparison of results with equivalent commercial tools, and graphic interface easy to operate as positive factors of that software. The environment used to build the mechanical model was the 3D CAD Autodesk Inventor software, which is commercialized by the Autodesk Company; it can be applied to develop complex projects.

A function model was developed in the Ururau to integrate the softwares, as presented in the Appendix A. In this Appendix, the build "DelayStepInventor" (lines from 21 to 26) starts the proxy "jni4net", which translates the calls in java to C# when starting the DES simulation model. When an entity is identified by this command, the "execute" function is run (lines from 46 to 60); that performs the dynamic simulation of the mechanism under study, in the Inventor® environment, and returns the total delay of the steps to the Ururau. Figure 2 presents the integration between the softwares.

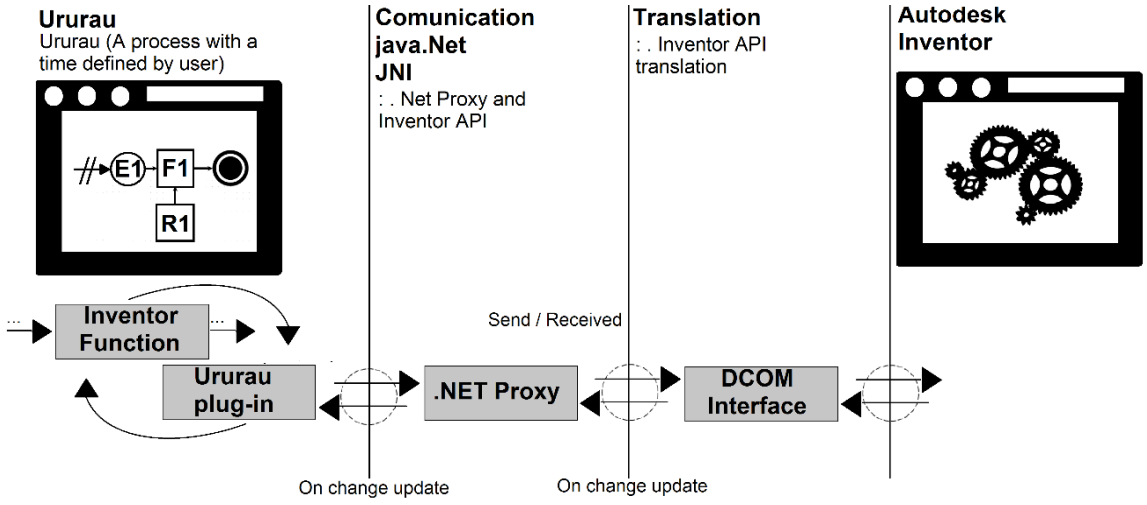


Figure 2: Communication Mechanism of Ururau Software (DES) and Autodesk Inventor®.
Source: the author

To execute the mechanism of integration between them, it is needed to use a layer in java, jni4net (https://jni4net.github.io/), as displayed in Figure 2, to adequate communication, since the API of the Inventor® was programmed using C#, and the Ururau was largely designed in java.

The Ururau Module Inventor Function activates the jni4net library, which functions as an intraprocess bridge between the Java Virtual Machine (JVM) and the Common Language Runtime (CLR); this one is a component of virtual machine of .NET platform to manage the execution of .NET programs, including C#.

The jni4net library calls the function of the API of the Inventor®, responsible for the dynamic simulation of the mechanical design, applying, as parameters, the initial and final steps of the simulation, the simulation time defined in the Ururau by the user, or a probability distribution inserted in the function module of the Ururau. They are steps of movement execution of a machine. When the parameter is per step, the Inventor® calculates the time of dynamic simulation and sends it to the Ururau so that it counts the simulation time for that entity. When the user defines a probability distribution, this time is divided by steps in order that the dynamic simulation of the mechanical model be performed.

The user inserts time values in the distribution that do not exceed the total duration of the dynamic simulation in the Inventor®. If this happens, an exception will be created to interrupt the simulation. The user can see that maximum value in the mechanical model, in the dynamic simulation environment of the software.

Despite the user performs the upload of the mechanical model and puts it in the dynamic simulation environment of the Inventor®, the Ururau is responsible for starting the simulation and running the process when starting the discrete simulation. Figure 3 displays the module implemented in the Ururau software for the integration with the mechanical model.

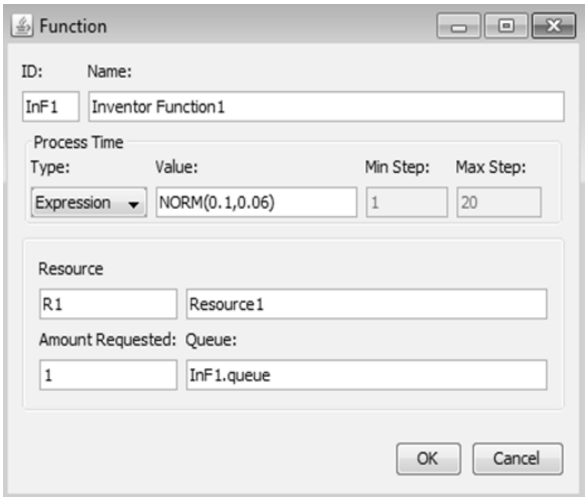


Figure 3: Inventor Function Module.
Source: the author

The parameters that have to be programmed in the model using the communication module can be seen in Figure 4. At the top of the module, the identification of the function is made in the (ID) and (Name) fields. In (Type), the modelist can adjust the desired function and then the data referring to the function in the (Value) field. At the bottom of the module, it can be assigned the resources and queues of the model.

5 RESULTS AND DISCUSSION

An electric saw implemented in the Inventor software was used aiming at testing the communication mechanism between softwares. The mechanical model of the saw is of one of the tutorials of the software itself and is presented in Figure 4. Despite being simplified, this model illustrates the process of the mechanical component that has to interact with the simulation model.

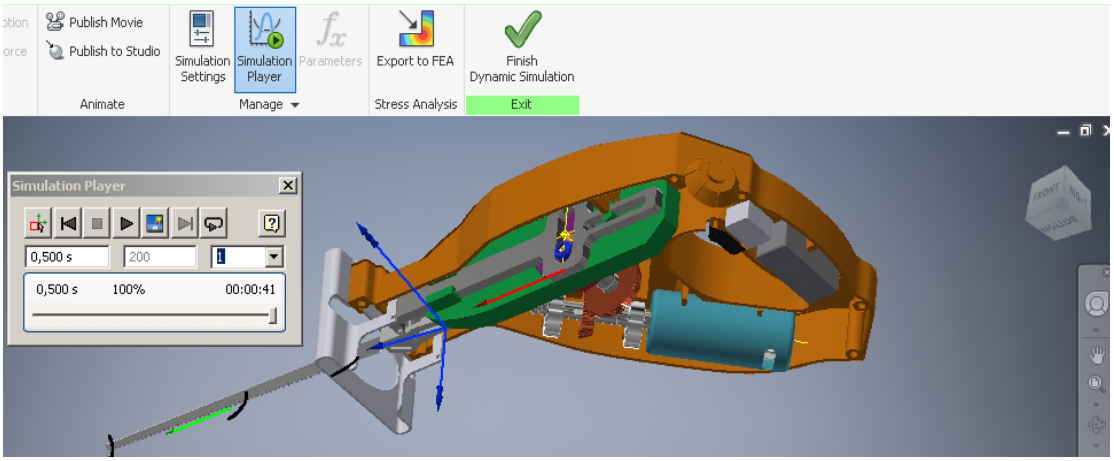


Figure 4: Mechanical model of an electric saw in the dynamic simulation environment of the Inventor.
Source: Autodesk (2017)

Figure 4 shows the dynamic simulation screen of the saw. The arrows in blue, within the saw, are the force components (in newtons) of the sliding joint in the saw. At the bottom of Figure 4, there is the graphic of the respective forces in function of time (in seconds). In the communication window, at the top left of Figure 4, it can be seen that the execution of the model is 100%; that indicates the end of the round in the Inventor software.

The total time of dynamic simulation is shown in the window, “Simulation Player”, in Figure 4. In this case, the time corresponds to 0.500 seconds with 200 steps in total. Therefore, the user cannot define, in the Ururau, the number of steps higher than this one or a distribution that generates time superior to 0.500 seconds.

In turn, the Ururau model has a generator of parts E1 that embraces the time between the arrival of Exponential 0.08 seconds and a part cutting process, which has service time of Normal with 0.09 seconds of mean and 0.03 seconds of standard deviation defined in the InF1 module. This process applies a resource R1 resource (saw) of capacity 1 since there is only one saw.

Figure 5 and Table 3 show, respectively, the simulation model in the Ururau and the system parameters. At the end of the simulation, the Ururau displays the mean time in the queue, the number in the queue, and the occupation rate of the resource. The report referring to the mechanical forces applied to the part can be found in the Inventor by the icon Export to FEA.

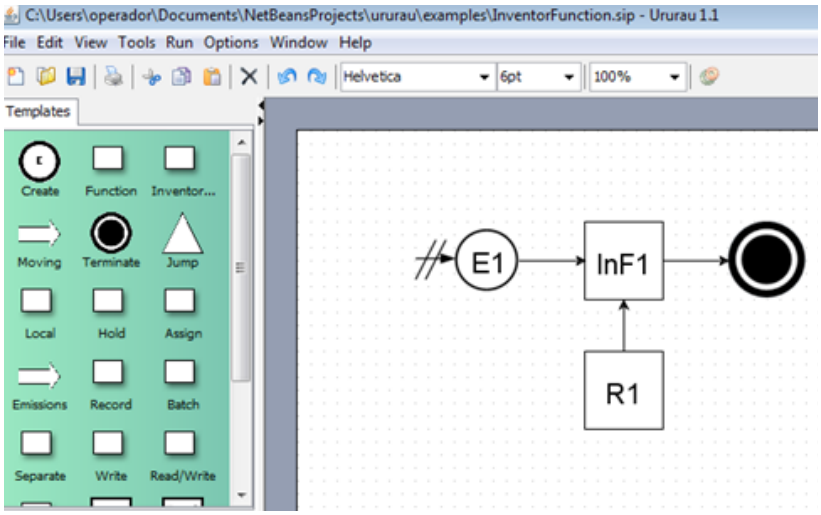


Figure 5: Discrete Simulation Model in Ururau.
Source: the author

Table 3: Element Descriptions for Inventor Discrete Model in Ururau.

	Description	Parameters
E1	Entity: Task type 1	Exponential arrivals (0.08) s.
InF1	Function: Process 1	Normal(0.09, 0.03) s. in queue Q1
R1	Resource: Electric Saw	1 machine

Source: the author

This simulation model was run at a simulation time of 5 minutes using an Intel Core i5® processor with 2.4GB of RAM memory. The total computational time was 2 min. It is important to indicate that, over each round of the simulation model, an animation is executed during the dynamic simulation of the Inventor, that is, it allows visualizing the movement of the electric saw in details. That happens after each entity of the simulation model activates the Inventor Function module of the Ururau.

The animation influences on the execution speed of the simulation, making it slower. In this way, the results presented in Table 4 were only for one replication considering that, in case more replications are run, the total duration of the execution consequently increases.

Table 4: Results of Discrete Event Simulation in Ururau.

Component	Output
F1.Queue Time in Queue	0.095 seconds
F1.Queue Number in Queue	2.264 units
R1 Busy	83,3%

Source: the author

Figure 6 displays the report of the forces from the dynamic simulation in the Inventor® software.

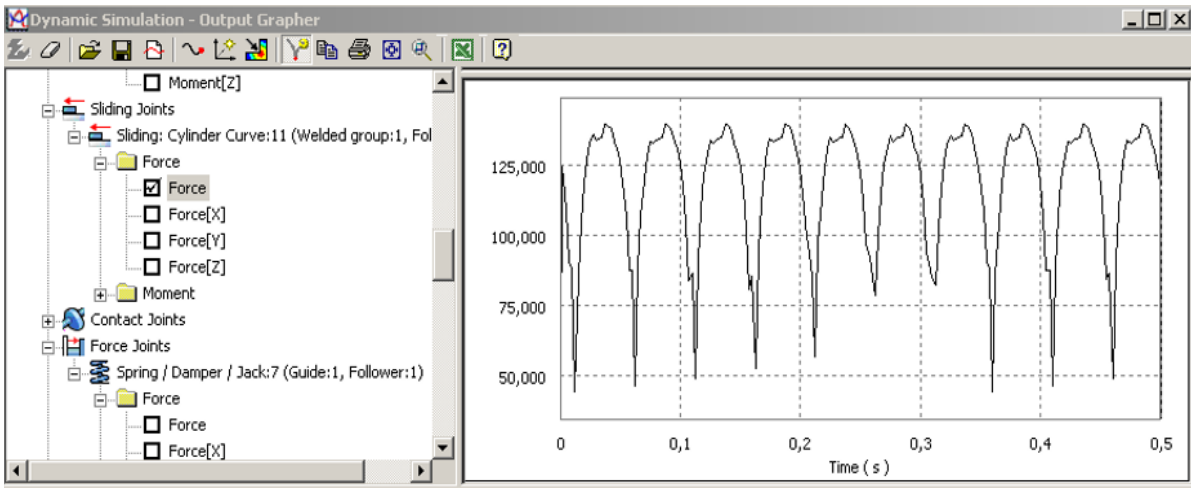


Figure 6: Report of forces (sliding joint) after dynamic simulation in the Inventor.
Source: Autodesk (2017)

The curve referring to the resultant force of the electric saw in function of the time is presented in Figure 6. Other variables of the mechanism can be visualized by means of their selection in the window to the left in Figure 6, for example: Force [X], Force [Y], Force [Z], Moment, Position, Velocity and Acceleration. Notice that only the "Force" is selected in the Figure.

6 CONCLUDING REMARKS

This work presented an online communication mechanism between a discrete event simulation model and a mechanical design software by means of the creation of a particular module for this operation. The use of the communication between interfaces was relatively simple and easily implemented. Thus, the module implemented in the Ururau software allowed exchanging data with the Inventor software enabling the user to build models applying both integrated approaches of discrete event and dynamic systems.

The purpose of this article was to demonstrate the functioning of the communication mechanism between simulation softwares and mechanical design softwares, being a potential tool to be applied to the Industry 4.0. The use of this tool allows reducing the cycle time of the process, minimizing costs and risks for operators and machines.

A suggestion for further research is to evaluate the possibility of using other parameters of the software for mechanical analysis, such as force employed on the element and speed of mechanism functioning in the communication between model data.

ACKNOWLEDGMENTS

The authors are grateful to the Coordination for the Improvement of Higher Education Personnel (CAPES), the National Council for Scientific and Technological Development (CNPq), and the Research Foundation of the State of Rio de Janeiro (FAPERJ) for financial support for this research.

REFERENCES

AUTODESK. **TUTORIAL INVENTOR.** Disponível em: <https://www.autodesk.com.br/products/inventor/overview>.

Azimi, P., and P. Soofi. 2017. "An ANN-Based Optimization Model for Facility Layout Problem Using Simulation Technique". Scientia Iranica 24: 364–377.

Cheng, R., M. Charles, B. Nelson, M. Rabe, C. Currie, J. Fowler. 2016. “Simulation: the past 10 years and the next 10 years”. In Proceedings of the 2016 Winter Simulation Conference, edited by T. M. K. Roeder, P. I. Frazier, R. Szechtman, E. Zhou, T. Huschka, and S. E. Chick, 2180–2192. Arlington, Virginia : Crystal Gateway Marriott.

DAGKAKIS, G.; HEAVEY, C. A review of open source discrete event simulation software for operations research. Journal of Simulation. 2015, v.10, n.3, p.193-206. http://dx.doi.org/10.1057/jos.2015.9

EZZELL, Z; A FISHWICK, P. Building dynamic 3D visualizations through ontology-guided interactions with domain knowledge and simulation models. Journal of Simulation. 2017, v.11, n.2, p.159-173. http://dx.doi.org/10.1057/jos.2016.1.

GIRONIMO, G.; BALSAMO, A.; ESPOSITO, G.; LANZOTTI, A.; MELEMEZ, K.; SPINELLI, R. Simulation of forest harvesting alternative processes and concept design of an innovative skidding winch focused on productivity improvement. Turkish Journal of Agriculture and Forestry. 2015, v.39, p. 350-359. http://dx.doi.org/10.3906/tar-1408-64

Grundstein, S., M. Freitag, and B. Scholz-Reiter. 2017. “A New Method for Autonomous Control of Complex Job Shops – Integrating Order Release, Sequencing and Capacity Control to Meet Due Dates”. Journal of Manufacturing Systems 42. The Society of Manufacturing Engineers: 11–28. doi:10.1016/j.jmsy.2016.10.006.

Kamat, VR, Martinez, JC. General-Purpose 3D Animation with VITASCOPE. Proceedings of the Winter Simulation Conference. 2004, p.1691-1697, http://dx.doi.org/10.1109/wsc.2004.1371518.

Khan, A., and K. Turowski. 2016. “A Perspective on Industry 4.0 : From Challenges to Opportunities in Production Systems”. IoTBD 441–448. doi:10.5220/0005929704410448.

Li, X., D. Li, J. Wan, A. V. Vasilakos, Chin-Feng Lai, and S. Wang. 2015. “A Review of Industrial Wireless Networks in the Context of Industry 4.0”. Wireless Networks 23-41. doi:10.1007/s11276-015-1133-7.

LINDSKOG, E.; VALLHAGEN, J.; JOHANSSON, B. Production system redesign using realistic visualisation. International Journal of Production Research. 2016, v. 55, n. 3, p.858-869. http://dx.doi.org/10.1080/00207543.2016.1218085

Liu, Y., and X. Xu. 2017. “Industry 4.0 and Cloud Manufacturing : A Comparative Analysis”. Journal of Manufacturing Science and Engineering 139: 1–8. doi:10.1115/1.4034667.

Pei, F., T. Yifei, H. Fei, and L. Dongbo. 2017. “Research on Design of the Smart Factory for Forging Enterprise in the Industry 4.0 Environment”. Mechanics 23 (1): 146–152.
PEIXOTO, T. A.; RANGEL, J. J. A.; MATIAS, I. O.; SILVA, F. F.; TAVARES, E. R. Ururau: A Free and Open-Source Discrete Event Simulation Software. Journal of Simulation. 2017, v.11, n.4, p.303–321 doi:10.1057/s41273-016-0038-5

Penas, O., R. Plateaux, S. Patalano, and M. Hammadi. 2017. “Computers in Industry Multi-Scale Approach from Mechatronic to Cyber-Physical Systems for the Design of Manufacturing Systems”.

Computers in Industry 52–69. doi:10.1016/j.compind.2016.12.001.

Rahmani, B.. 2016. “Internet-Based Control of Industrial Automation Systems”. Journal of Intelligent & Robotic Systems. doi:10.1007/s10846-016-0350-3.

REKAPALLI, P. V.; MARTINEZ, J. C. Discrete-Event Simulation-Based Virtual Reality Environments for Construction Operations: Technology Introduction. Journal of Construction Engineering and Management. 2011, v. 137, n. 3, p.214-224. http://dx.doi.org/10.1061/(asce)co.1943-7862.0000270.

SANTOS, Maribel Yasmina, Sá, J. O.; Andrade, C.; Lima, F. V.; Costa, E.; Costa, C.; Martinho, B.; Galvão, J. A Big Data system supporting Bosch Braga Industry 4.0 strategy. International Journal of Information Management, [s.l.], v. 37, n. 6, p.750-760, dez. 2017. Elsevier BV. http://dx.doi.org/10.1016/j.ijinfomgt.2017.07.012.

Harrison, R., D. Vera, and B. Ahmad. 2016. “Engineering the Smart Factory”. Chinese Journal of Mechanical Engineering 1046–1051. doi:10.3901/CJME.2016.0908.109.

Turner, C. J., W. Hutabarat, J. Oyekan, and A. Tiwari. 2016. “Discrete Event Simulation and Virtual Reality Use in Industry : New Opportunities and Future Trends”. IEEE Transactions on Human-machine systems 1–13.

WOO, J. H.; NAM, J. H.; KO, K. H. Development of a simulation method for the subsea production system. Journal of Computational Design and Engineering. 2014, v. 1, n. 3, p.173-186. http://dx.doi.org/10.7315/jcde.2014.017

Xu, J., E. Huang, L. Hsieh, L. H. Lee, Qing-Shan Jia, and Chun-hung Chen. 2016. “Simulation Optimization in the Era of Industrial 4.0 and the Industrial Internet”. Journal of Simulation. Palgrave Macmillan UK. doi:10.1057/s41273-016-0037-6.

XU, J.; ABOURIZK, S. M.; FRASER, C. Integrated three-dimensional computer-aided design and discrete-event simulation models. Canadian Journal of Civil Engineering. 2003, v. 30, n. 2, p.449-459. http://dx.doi.org/10.1139/l02-110.

ZHANG, Y.; MOGHANI, E.; ABOURIZK, S. M. 3D CAD modeling and visualization of the tunnel construction process in a distributed simulation environment. Proceedings of The 2010 Winter Simulation Conference, [s.l.], p.3189-3200, dez. 2010. IEEE. http://dx.doi.org/10.1109/wsc.2010.5679011.

Appendix A: Programming code implemented for the communication between the Ururau software and the Inventor software.

```
/*
 * To change this template, choose Tools | Templates
 * and open the template in the editor.
 */
package org.ururau.commands;
import invapibasics40library.ApiComClass;
import invapibasics40library.IApiComClass;
import java.io.File;
import java.io.IOException;
import net.sf.jni4net.Bridge;
import jsl.modeling.ModelElement;
import jsl.modeling.elements.resource.Entity;
import jsl.modeling.elements.variable.Variable;
/**
 *
 */
public class DelayStepInventor extends Delay {
    private int myMinStep;
    private int myMaxStep;
    private IApiComClass myApi;
    public DelayStepInventor(ModelElement parent, int minStep, int maxStep) throws IOException {
        this(parent, minStep, maxStep, null);
    }
    public DelayStepInventor(ModelElement parent, int minStep, int maxStep, String name) throws IOException {
        super(parent, name);
        File file = new File ("net/lib/jni4net.n.w64.v40-0.8.8.0.dll").getAbsoluteFile();
    Bridge.init(file);
        System.out.println("Load Assembly...");
        myApi = new ApiComClass();
    myApi.AppLoad();
        this.setDelay(minStep, maxStep);
    }
    public int getMinStep() {
        return this.myMinStep;
    }
    public int getMaxStep() {
        return this.myMaxStep;
    }
    public void setDelay(int minStep, int maxStep) {
        if(minStep <= 0 || maxStep <=0)
            throw new IllegalArgumentException("minStep and maxStep must be positive!");
        this.myMinStep = minStep;
        this.myMaxStep = maxStep;
    }
    @Override
    public void execute() {
        Entity entity = getProcessExecutor().getCurrentEntity();
        entity.setProcessExecutor(getProcessExecutor());
        if (!Batch.isVisible(entity))
            return;
     double t = 0;
    if (myApi != null){
        System.out.println("Run Inventor Simulation ...");
        t = myApi.RunSimulation(myMinStep, myMaxStep);
        t = (t<0.0)?0:t;
        System.out.println("t = "+t);
        }
    }
    super.executeCommon(t);
}
```