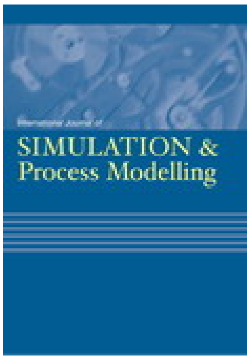


International Journal of Simulation and Process Modelling

ISSN online: 1740-2131 - ISSN print: 1740-2123
https://www.inderscience.com/ijspm

Autonomous entities: a comparison between simulation models

João Paulo Barbieri, Alexandre Ferreira de Pinho, José Arnaldo Barra Montevechi

DOI: 10.1504/IJSPM.2023.10060041

Article History:

Received:	26 November 2021
Last revised:	22 May 2023
Accepted:	08 July 2023
Published online:	26 October 2023

Autonomous entities: a comparison between simulation models

João Paulo Barbieri*

Federal Institute of Education, Science and Technology of
South of Minas Gerais – IFSULDEMINAS,
Av. Vicente Simões, 1111, Pouso Alegre, Minas Gerais, Brazil
Email: joao.barbieri@ifsuldeminas.edu.br
*Corresponding author

Alexandre Ferreira de Pinho

Production Engineering and Management Institute,
Federal University of Itajubá – UNIFEI,
R. Antonio Simao Mauad, 152, apto. 402,
Itajubá, Minas Gerais, Brazil
Email: pinho@unifei.edu.br

José Arnaldo Barra Montevechi

Production Engineering and Management Institute,
Federal University of Itajubá – UNIFEI,
Av. BPS, 565, Itajubá, Minas Gerais, Brazil
Email: montevechi@unifei.edu.br

Abstract: The manufacturing systems, even with technological advances, are affected by factors linked to human decisions, especially in manufacturing systems with a high proportion of manual labour, such as an assembly line. In the simulation and modelling process, attributing greater autonomy to entities generates effects on the model. Such effects need to be considered by the modeller. To explore these effects, this article presents a method, whose name is iDAV, in which it is possible to analyse the elements of a simulation model and the nature of the connections between these elements. The iDAV method was applied to two discrete-event simulation (DES) models. Both models represent the same assembly line. The difference between them lies in the degree of autonomy of the entities. Through the graphs, it was possible to verify the increase in complexity and variability of responses between the two DES models.

Keywords: discrete-event simulation; DES; autonomous entities; simulation and modelling.

Reference to this paper should be made as follows: Barbieri, J.P., de Pinho, A.F. and Montevechi, J.A.B. (2023) 'Autonomous entities: a comparison between simulation models', *Int. J. Simulation and Process Modelling*, Vol. 20, No. 1, pp.60–69.

Biographical notes: João Paulo Barbieri holds a degree in Business Administration at Economic Science College of South of Minas in 2013 and a Master degree in Production Engineering from the Federal University of Itajubá in 2016. He has experience in the area of production engineering, with emphasis on production systems and logistics, working mainly on the following topics: discrete-event simulation, agent-based simulation, and hybrid simulation.

Alexandre Ferreira de Pinho works in the area of operations research, computer simulation, and decision support systems. He holds a degree in Mechanical Engineering from the Federal University of Itajubá (UNIFEI), a Master degree in Production Engineering from the Federal University of Itajubá (UNIFEI), a Specialist degree in Project Analysis and Systems Management from PUC-MINAS, and PhD in Mechanical Engineering from UNESP.

José Arnaldo Barra Montevechi holds a degree in Mechanical Engineering from the Federal University of Itajubá in 1985, a Master's degree in Mechanical Engineering from the Federal University of Santa Catarina in 1989, a PhD in Mechanical Engineering from the University of São Paulo in 1995, and a postdoc degree at University of Texas in the USA in 2008. He has experience in the area of production engineering, working mainly on the following topics: artificial intelligence, economic engineering, investment analysis, discrete-event simulation, virtual reality, optimisation, Monte Carlo simulation, and operational research.

1 Introduction

As one of the most powerful analytical tools, the simulation is a method that assists in the design, planning, analysis, and optimisation of manufacturing systems (Shannon, 1975; Schönemann et al., 2015). There are several types of simulation approach, e.g., Monte Carlo Simulation (MCS), continuous simulation (CS), system dynamics simulation (SDS), agent-based simulation (ABS), and discrete-event simulation (DES) (Shannon, 1975; Macal and North, 2005; Law, 2015; Siebers et al., 2010; Viana et al., 2014). The most widely used approach to analysing and understanding the dynamics of manufacturing systems is the last one, i.e., DES (Brailsford and Hilton, 2001; Negahban and Smith, 2014).

For more than 40 years, in the operational research (OR), the basis of the simulation scientific community has been the DES (Ingalls, 2008; Siebers et al., 2010; Scheidegger et al., 2018). It is not without reason that, for many researchers in OR field, the word simulation is a synonym of the term DES (Brailsford, 2014). In the organisational context there are many applications for DES in areas such as manufacturing.

Manufacturing systems, even with technological advances, are affected by several factors that include uncertainties and variability. One of these factors is in human decisions, especially in manufacturing systems with a high proportion of manual labour, such as an assembly line. It is challenging to include human behaviour, even partially, in a discrete simulation model. Nevertheless, a more detailed modelling of the human element can generate interesting insights for decision makers. The attribution of greater autonomy to the entities of a simulation model allows, e. g., the development of studies involving fatigue work rhythm, production goals, and the limits of a production system.

In terms of modelling, what are the impacts of attributing greater autonomy to the entities present in a simulation model? As a contribution to a better understanding of the effects of assigning greater autonomy to entities in a simulation model, this paper analyses and compares the elements and connections that exist in two simulation models. The only difference between these two models is the degree of autonomy of the entities. Another contribution of this paper is the creation of a method whose name is iDAV (acronym of the terms: *identification*, *definition*, *analyses*, and *verification*). Based on graph theory, the iDAV method makes it possible to understand each element and its connections within the simulation model.

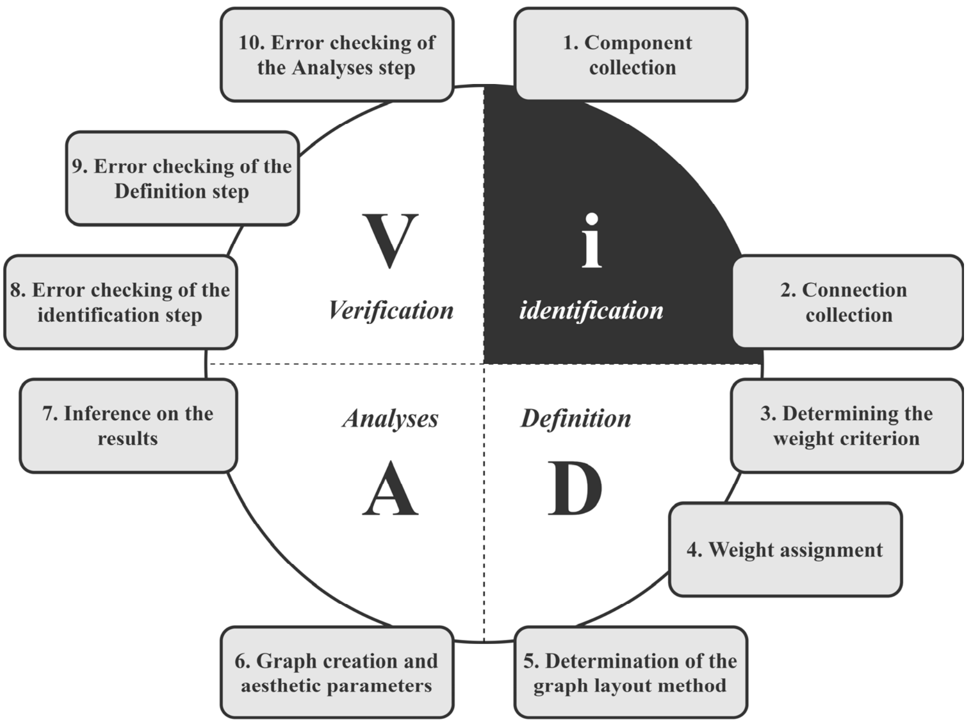
The remainder of this paper is organised as follows. The next section presents the description of the simulation models and the iDAV method. Then, in Section 3 will be the theory and discussions about the results found from the application of the iDAV method. In the last section, the main conclusions and proposals for future research.

2 Material and methods

2.1 The iDAV method

To verify the effects of adding autonomy to the workers, model entities, the iDAV method was developed. The method name is the acronym for the terms: *identification*, *definition*, *analysis*, and *verification*. These terms are the main steps in the method.

Figure 1 The iDAV method



As shown in Figure 1, the *identification* step consists of two actions:

1 component collection

2 connection collection.

In this step, the components in the simulation model are identified and registered. The same is done with the connections of these components; it is essential to document the relationship between each component. The result of this step is materialised in a matrix that can be built using electronic spreadsheets.

In the *definition* step, there are three actions:

3 determining the weight criterion

4 weight assignment

5 determination of the graph layout method.

The adoption of criteria to assign weights to connections is important, because in the simulated environment there may be connections made by *drag-and-drop* or by programming blocks. Thus, adopting a weighting criterion and applying it helps to distinguish the types of connections that exist in a simulation model. At this step, the graph layout method that will portray all components and their connections, i.e., the structure of the simulation model, is also determined.

In the *analyses* step are two actions:

6 graph creation and aesthetic parameters

7 inference on the results.

There are tools that help in creating and defining the aesthetic parameters of graphs. One of these tools is the Gephi® – an open-source software used for analysing and visualising networks. The Gephi® was chosen due to its mechanisms that facilitate communication with electronic spreadsheets. With the graph created, the action of inference about the results begins. This inference is made based on statistical concepts and methods.

The *Verification* step is the last of the iDAV method. This process is composed of three activities that are:

8 error checking of the identification step

9 error checking of the definition step

10 error checking of the analyses step.

Keeping in mind the continuous improvement of the simulation model, the *Verification* step exists to help to identify possible errors that occurred in the previous steps.

The operation of the iDAV method is cyclic, i.e., the *identification*, *definition*, *analyses*, and *verification* steps can be performed multiple times. The letter '*i*', in its lowercase form, indicates the beginning of the process. In the next sections, the iDAV method and all its steps will be performed. Specifically, in this paper, we identify and analyse the structural part of simulation models.

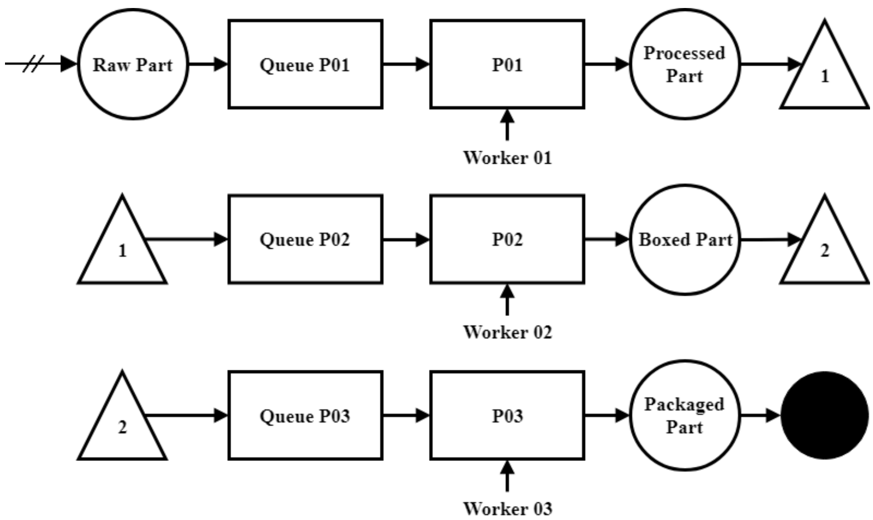
In the next section, there is a description of the elements that make up the simulation models, including the behaviour of the entities present in the manufacturing process.

2.2 *The computational models*

In this research, a manufacturing system with an in-line layout will be adopted. The choice of this type of system is related to the researcher's knowledge of the general particularities of this system. Furthermore, it can be highlighted that an in-line manufacturing system is simple in nature and it allows exploring the levels of human decision within the limitations that the system itself imposes.

Figure 2 represents the conceptual model of an assembly line composed of three manual operations. Each operation contains one worker and one queue – whose capacity is for one product. It is important to say that this assembly line actually exists and it is located in the south of Minas Gerais, Brazil. Using the IDEF-SIM (integrated definition methods – simulation) modelling technique (Montevechi et al., 2010), the conceptual model was developed.

Figure 2 Conceptual model of a manufacturing line



Based on Figure 2, the simulation model DES_Individual was built, as well as the simulation model DES_Autonomous. The construction of the simulation models was made by the multi method simulation modelling tool AnyLogic®. The probability distributions of workstations P01, P02, and P03 are, respectively, normal (61.33, 8.29), normal (70.14, 11.18), and normal (50.97, 5.18). The values of the probability distributions are in seconds. Each workstation has a queue capacity equal to 1.

For Bruzzone et al. (2011), the simulation models become more realistic when intelligent agents reproduce human behaviour. Du and He (2018) developed their research in which they report four individual characteristics: behaviour, emotion, personality and perception. In this paper, two characteristics will be considered: behaviour and perception. In the first simulation model, the behaviour is scripted. In the second simulation model, entities have a behaviour based on the perception of variables belonging to the manufacturing system.

Thus, the human factor is represented individually and its behaviour is scripted in the DES_Individual model. In summary, the simulation model starts with the workers in the *resting* state. At 8:00, the workers go to the *working* state. At 12:00, the workers leave for lunch and return to the *working* state at 13:00. At 17:00, the workers move to the *resting* state.

In the DES_Autonomous model, the workers have autonomous behaviour and are able to act according to the conditions present in the simulated environment. Thus, the workers have autonomous characteristics, since they changed their own work rhythm during the simulation on the model. This change occurs when each worker checks their production and compares whether such production is above, below, or within the previously established production goal. Therefore, in addition to the *resting*, *working*, and *lunching* states, each worker has the *checking*, *normal*, *fast*, and *slow* states. It should be noted that the decision to change the work rhythm occurs in the *checking* state.

Table 1 shows a summary of the main characteristics of the simulation models.

Table 1 The main characteristics of the models

Model	*Behaviour*	*States*
DES_Individual	Scripted	*Resting; working; lunching*
DES_Autonomous	Autonomous	*Resting; working; lunching; checking; normal; fast; slow*

In total, the DES_Individual model has a variable that collects the number of packaged parts, a calendar that defines the occurrence of *resting*, *working*, and *lunching* states, and four parameters linked with the 3D window that define the scripted behaviour of each worker.

In the case of the DES_Autonomous model, an event that changes the work rhythm was added for each worker. Based on simple equations, this change occurs through stochastic data from 24 parameters related to worker effort, worker skill, work consistency and working conditions. These parameters were included in accordance with the studies reported by Barnes (1980). In addition to these parameters, there are two more that define the work goals. Finally, there are seven variables associated with the parameters that generate instant information about workers' work rhythm.

With the two simulation models finished, the 2-sample t test was applied to verify if the models are statistically equal. This step is important, because without it there is no way to make safe comparisons between simulation models. A sample of 100 replications of the total daily production value of the assembly line was taken from each simulation model. The result of the 2-sample t test was p-value equal to 0.65, this means that there is not enough evidence to conclude that the means differ at the 0.05 level of significance.

3 Fundamental knowledge and method

System is a term commonly observed in scientific documents that report studies carried out in different areas such as physics, mathematics, chemistry, biology, letters, philosophy, engineering, among others. The system concept is applied to any 'whole' in which there are interacting 'components' (Bertalanffy, 1968). The union of 'isolated' components, the relationships among the components and the layer structure are fundamental characteristics of a system (Forrest, 2018). The system concept can also be applied to simulation projects. In the case of this paper, each simulation model is seen as a system in which there are a number of components and connections among them.

A simulation project addresses problems involving real systems. In fact, a model fulfils the function of reproducing, in a different way, an object, a system or an idea (Shannon, 1975). Robinson (2008) establishes that a model is a simplified representation of reality built for a specific purpose. Furthermore, in simulation models it is also possible to represent human behaviour (Hlupic and Vreede, 2005). The simulation models used in this scientific research represent a manufacturing production line.

The set of logical and causal relationships that occur in real systems is the heart of a simulation model. In a simulation model, the main characteristics of a real system are captured and reproduced virtually (Sargent, 2014; Law, 2015; Brailsford et al., 2019). According to Shannon (1975), a good simulation model needs to contain the following attributes:

- simple to be understood by the user
- goal or purpose directed
- robust, in that it does not give absurd answers
- easy for de user to control and manipulate, i.e., it should be easy to communicate with
- complete on important issues

- adaptive, with an easy procedure for model modification or updating
- evolutionary, in that it should start simply and become more complex, in conjunction with the user.

In this paper, the evolutionary attribute of a simulation model is explored in order to understand the effects of giving greater autonomy to workers, entities in the model. Starting from the idea that a computational model consists of elements with different functions that are intrinsically connected, the structure of a simulation model is based on three fundamental aspects:

1 the number of components

2 the connection among the components

3 the essence of the connections.

The *identification* step, the first one of the iDAV method, was applied to both simulation models. Thus, 82 components were counted in the DES_Individual model and 184 components in the DES_Autonomous model. Also, in the *identification* step, 132 connections were counted in the DES_Individual model and 276 connections in the DES_Autonomous model. Table 2 shows a summary of the data collected from the simulation models in the *identification* step.

Table 2 The elements and connections of the models

Models	*n*	*e*
DES_Individual	82	132
DES_Autonomous	184	276

The *Definition* step, the second one of the iDAV method, was applied to both simulation models. The graph theory (Godehardt, 1988; Wallis, 2007) served as a conceptual basis for the definition of weights and the layout of graphs. Thus, the cyclomatic complexity (CC) equation, equation (1), was used to analyse the essence of each connection among the components in the computational models.

$$v(G) = e - n + 2(p) \tag{1}$$

The CC equation was developed by McCabe (1976) and it is one of the most used metrics in computational measurement according to Polančič and Cegnar (2017). In practical terms, the CC equation measures the logical decision-making power of a system (McCabe, 1976).

In this paper, the term *v(G)* is the number of logical actions that are made in a connection among the components; this number is the weight adopted in each connection among the components of the simulation model. The term *n* represents the number of components involved in a connection. The term *e* means the number of logical sequences among the components involved in a connection, i.e., the term e depicts the flow of information that occurs when each line of code is triggered. The term *p* represents the number of routines present in a connection.

In the simulation models, the connections have different natures. Thereby, there are connections among components in which there is no need to write lines of code. In such cases, *drag-and-drop* action is commonly adopted. However, there are also connections created from writing lines of code. In order to determine and distinguish the essence of these connections in the simulation models, equation (1) was applied. The result of the *Definition* step is in Table 3.

Table 3 The weights and number of connections

v(G)	*DES_Ind.*	*%*	*DES_Aut.*	*%*
1	110	83.3	110	39.9
2	13	9.9	97	35.1
5	9	6.8	9	3.3
9	0	0.0	60	21.7
Total	*132*	*100.0*	*276*	*100.0*

The *v(G)* is the result of equation (1). For *drag-and-drop* connections, *v(G)* is equal to 1. For connections created by simple lines of code, *v(G)* is equal to 2. For connections made by blocks of lines of code, which include routines as *If/Then/Else*, *v(G)* is greater than 2.

Also, in the second step of the iDAV method, the layout of the graphs is defined. This definition of the graph layout was based on the works of Godehardt (1988), Di Battista et al. (1994), Herman et al. (2000), Kaufmann and Wagner (2001), Koren (2005), Wallis (2007), Jacomy et al. (2014) and Tamassia (2014). There are several layouts for graphs, e.g., circular layout, radial axis layout, Frushterman-Reingold layout, ForceAtlas layout, and Yifan Hu layout.

The construction of the graphs, which represent the simulation models, was based on the Gephi® software – open-source software used for analysis and visualisation of networks. In this paper, the ForceAtlas layout was adopted. This layout was chosen because it highlights the parts of the simulation models and the importance of each component. The graphs produced from this layout help to understand the structural aspects of simulation models.

Before the creation of the graphs that represent the simulation models, Figure 3 and Figure 4, it was necessary to elaborate two matrices.

$$D = \left[d_{ij} \right] \in \Re^{m \times n} \tag{2}$$

Based on equation (2), the matrices were created using the Excel® tool, a spreadsheet editor produced by Microsoft. All existing components and connections in the DES_Individual and DES_Autonomous models are depicted, respectively, by equation (3) and equation (4). The weights of each connection were also added to the matrices. After creating these matrices, they were transferred to the Gephi® software.

$$IND = \begin{bmatrix} ind_{1,1} & \mathrm{B} & ind_{1,82} \\ \mathrm{C} & \mathrm{E} & \mathrm{C} \\ ind_{82,1} & \mathrm{B} & ind_{82,82} \end{bmatrix} \tag{3}$$

$$AUT = \begin{bmatrix} aut_{1,1} & \text{B} & aut_{1,184} \\ \text{C} & \text{E} & \text{C} \\ aut_{184,1} & \text{B} & aut_{184,184} \end{bmatrix} \quad (4)$$

The *Analyses* step, the third one of the iDAV method, was performed and it was applied aesthetic parameters for the graphs. These parameters are in the Gephi® software. Figure 3 and Figure 4 were created based on the matrices, equation (3) and equation (4). The aesthetic parameters such as colours, diameter of circles and thickness of lines were defined in order to faithfully portray the components and their connections. The colours represent parts of the simulation model. The components related to 3D presentation are represented by yellow colour. The red colour depicts the workstations on which the workers are located. The workers and all their parameters and variables are represented in blue. The diameter of each component is defined by the value of connections and their weight. It should also be noted that each component is represented by an identification number.

So far, the six actions of the iDAV method have been presented. The following actions are:

7 inference on the results

8 error checking of the *identification* step

9 error checking of the *definition* step

10 error checking of the *analyses* step.

These actions will be covered in the next section.

4 Results and analysis

The third step of the iDAV method has two actions, and the last one is the inference about the results. The first result achieved is shown in Table 2. The attribution of greater autonomy to workers, who are present in the manufacturing line, generated an increase in the number of components and lines. The number of components and lines grew, respectively, in the order of 124.4% and 109.1%, i.e., the attribution of greater autonomy to entities doubled the size of the simulation model.

As shown in Table 3, there was also a change in regard to the types of connections existing in the simulation model. For example, in the DES_Individual model, 83.3% of the connections are *drag-and-drop*. This percentage drops to 39.9% in the DES_Autonomous model. In addition to this drop, there is an increase in connections with lines of code, especially connections of type *if/then/else*, i.e., 60.1% of the DES_Autonomous model has more complex connections than *drag-and-drop* connections.

The increase of the components and connections, and the proportional decrease of the *drag-and-drop* connections generate impacts on the characteristics of the graphs. The thickness of the connections increased as did the diameter of the components. This can be confirmed by comparing Figure 3 with Figure 4.

Figure 3 The structure graph of the DES_Individual model (see online version for colours)

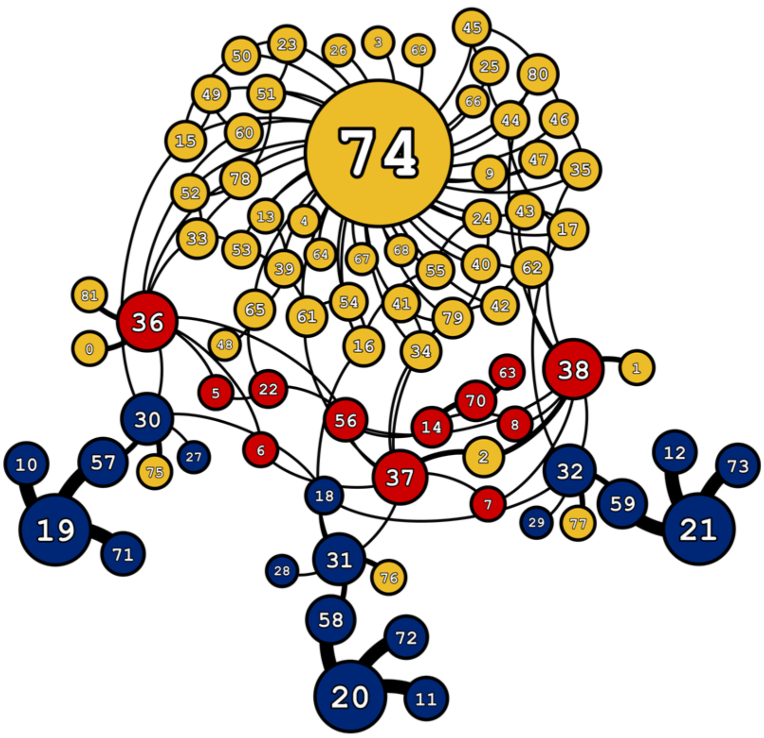
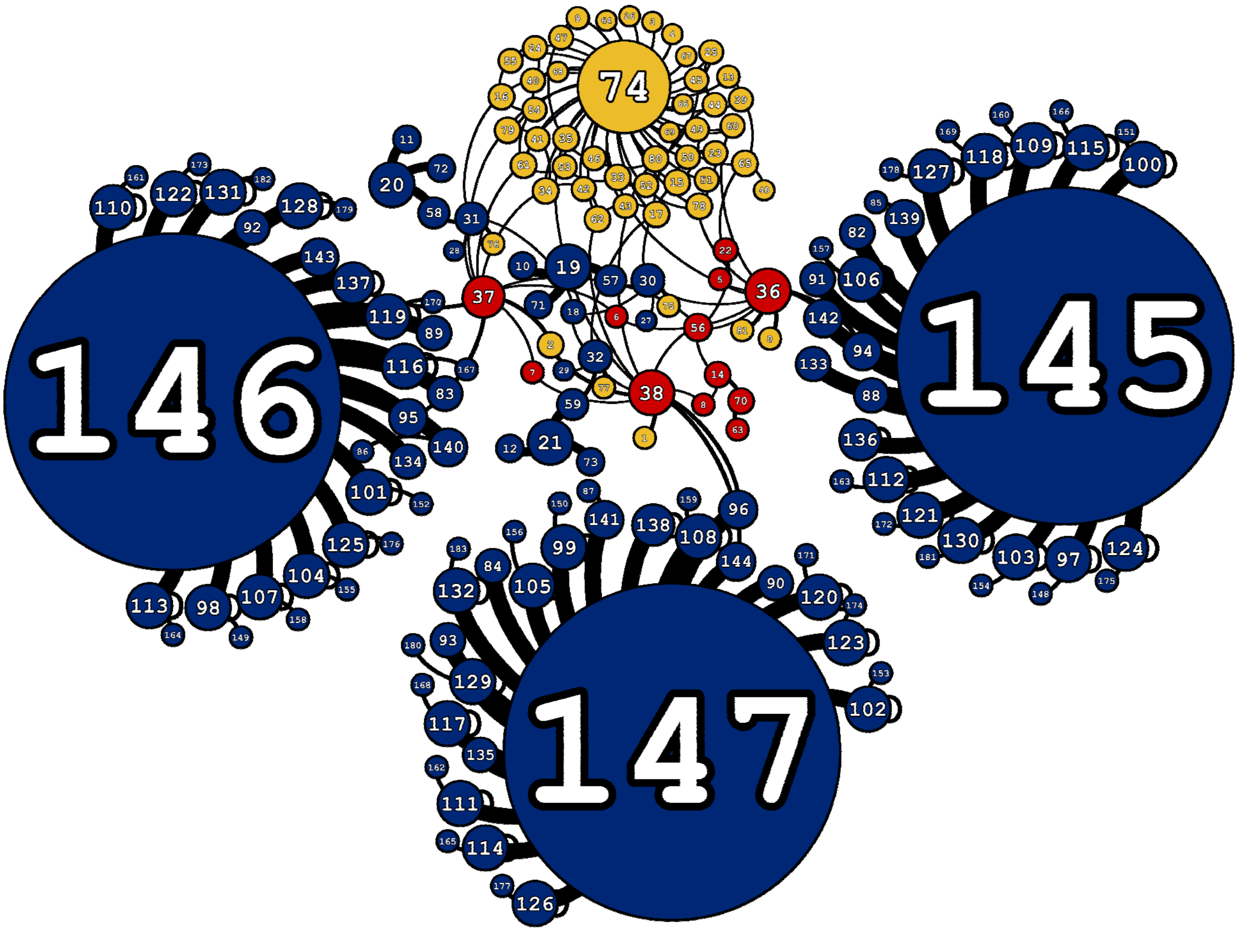


Figure 4 The structure graph of the DES_Autonomous model (see online version for colours)



About the idea of communication, Shannon (1975) establishes that the simulation model must be easy for the user to control and manipulate. By giving greater autonomy to entities, the number of code blocks increased. This fact can make it difficult for stakeholders to control and manipulate the variables present in the model. The lack of knowledge about the logical functioning of the computational language applied in the simulation model can also be a barrier for stakeholders.

Each part of simulation models has more or less influential components. In this paper, the colours yellow, red and blue were adopted. The colours adopted help us to understand which are the most influential components in each part of the simulation model. For example, according to Figure 3, the influence of the *window3d* component whose identification number is 74 is remarkable in the DES_Individual model. This component brings together all components related to 3D presentation; due to it its diameter is larger. In total, there are 42 graphical components connected to the window3d component. These 42 connections are simple and were made through *drag-and-drop* action.

A simulation model must be easy to be understood by the user (Shannon, 1975). In a simulation model, the graphical part composed of 3D components allows the stakeholders to have a clearer understanding of the workflow, actions and behaviour of entities within the simulation model. For this reason, the 3D presentation elements were adopted in this paper. However, the effect of adding 3D components to the simulation model is notable in the graph of the DES_Individual model. If all components related to the 3D presentation were removed, the most influential components would become those related to the workers.

The components 19, 20, and 21 also standout in Figure 3. These components bring together variables responsible for the scripted actions of the workers during the simulation of the DES_Individual model. These actions are related to changes among *resting*, *working*, and *lunching* states. These changes show up in the 3D presentation. Furthermore, the thickness of the lines connecting these components is greater, as such connections were made through a block of lines of code.

Still in Figure 3, the components whose numbers are 36, 37, and 38 are also highlighted. They represent the workstations in which the workers are allocated. In addition to the components related to manufacturing line workers, there are also graphic components associated with the production and packaging of the parts. On average, there are nine components associated with the components 36, 37, and 38.

As shown in Figure 4, the addition of variables and parameters, which allow greater decision-making capacity of entities within the simulation model, generated an increase in complexity. According to Shannon (1975), a good simulation model can be robust but should not produce absurd answers. Thus, as stated in the last paragraph of Section 2.1, the 2-Sample t statistical test for the mean was performed, and the result was p-value equal to 0.65. This means that there is not enough evidence to conclude that the means between the two simulation models are different. Thus, the two simulation models produce similar results. Despite this similarity, when it is added variables and parameters that give greater autonomy to workers, an increase in the variability of the results can be seen, as shown in Figure 5.

In Figure 5, the increase in the variability is the result from the presence of autonomy for each worker to decide the rhythm of work according to the previously stipulated goal. In a manufacturing system where there is a high proportion of manual labour, the human factor generates variability in the process. When human factor-related variables are not included in a simulation model, system variability is limited to the probability distribution of each workstation. Therefore, the DES_Autonomous model became more robust, not generating absurd responses. The effect of this robustness is in the variability of the model's results.

Applying the evolutionary principle stated by Shannon (1975), a principle in which a simulation model should start simple and become more complex, the *checking*, *normal*, *fast*, and *slow* states were added. These states represent the changes in the work rhythm of each worker on the manufacturing line. Thus, the perception is included in entities that already had scripted behaviour. In fact, when it is considering two of the four individual characteristics treated by Du and He (2018), the behaviour of entities becomes more autonomous during the model simulation, i.e., the degree of decision of these entities is greater than that of the entities of the DES_Individual model.

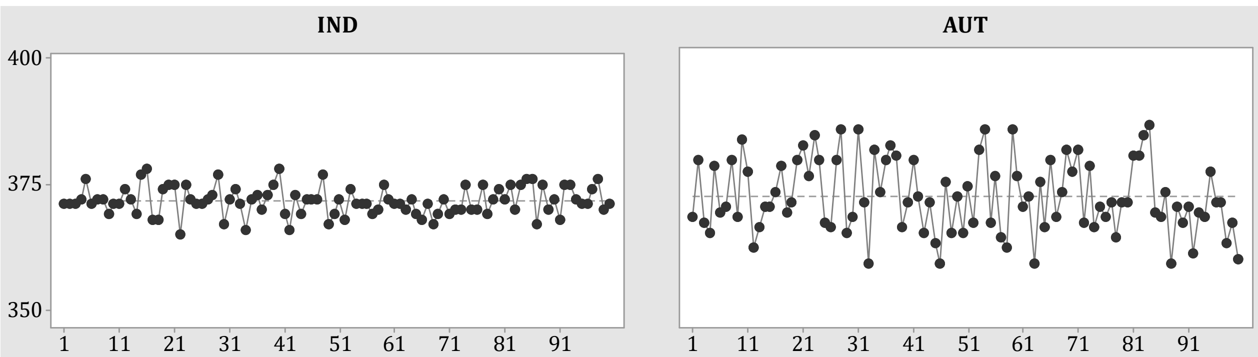
The autonomy of each worker is associated with the addition of 34 components. According to Figure 4, the components 145, 146, and 147 connect the other 33 components through blocks of lines of code, whose logic is *if/then/else*. This type of connection increases the line thickness. Therefore, components 145, 146, and 147 have a larger diameter. With the application of the iDAV method, it is evident the importance of the 145, 146 and 147 components in the DES_Autonomous model. Therefore, the components related to 3D presentation become less important in the model.

In graph theory, making use of the handshaking lemma, each component has a *v(G)* value. This value was generated based on the weight and the number of connections of each component. After creating the graphs, it is possible to extract the average *v(G)* of the simulation models. Thus, the average *v(G)* of the DES_Individual model and the DES_Autonomous model are, respectively, 4.4 and 9.6. This means that, on average, the components of the DES_Individual model have the value of 4.4 and the components of the DES_Autonomous model have the value of 9.6. When adding the 102 components and their connections, there was an increase of the average *v(G)* in the order of 118.8%. This increase was mainly due to the nature of the connections. According to Table 3, around 60% of connections in the DES_Autonomous model have weights equal to or greater than 2.

After reaching the objective proposed by a simulation model, it is possible that the modeler is tempted not to improve, or even simplify, the connections that make up the simulation model. Therefore, the actions present in the *Verification* step, the last one step of the iDAV method, helped to identify, for example, faults and programming excesses.

The errors in the previous steps were identified during the *Verification* step. Such errors were corrected in order to improve the final results generated by the iDAV method. For example, there was a simplification in writing the lines of code that connect the components. Consequently, the weights of these connections were revised. This revision impacted the graphs of each simulation model. Another type of improvement occurred in the aesthetic parameters of the graphs, such as the adjustment of colours to improve visual communication related to the sectors of the simulation models.

Figure 5 Comparison of the variability of the DES_Individual model and the DES_Autonomous model



From the point of view of visual communication, after the *Verification* step, the graphs faithfully represent the simulation models. From this representation, the modeller is able to identify the most critical sectors and simplify the model. The graphs help the modeller to verify that the simulation model has a directed objective and addresses important issues of the real system, as stated by Shannon (1975).

In fact, the *Verification* step was carried out systematically to ensure the quality of the results of the previous steps. This step can be understood as a filter in which the improvement of the model is carried out. Furthermore, after this step, the model can become simpler for the user to understand. This better understanding is in accordance with the concepts defined by Shannon (1975).

In manufacturing systems, the human factor is an important element. Digiesi et al. (2009) emphasise that human labour is an essential element in modern manufacturing systems and, therefore, it must be reliably expressed in the modelling process of such systems.

The human behaviour is a theme of modelling and simulation and it is still a traditional challenge for the simulation area itself and its application in the business areas (Bruzzone et al., 2007). Indeed, identifying and conceptualising the structure of human behaviours and interactions is a common problem in modelling complex systems involving human factor.

In DES, the inclusion of human performance generates an opportunity to know about the impact and importance of the human factor in the system (Baines et al., 2004). But, it is a challenge to perform a more detailed modelling of the human factor through the DES (Baines and Kay, 2002). Despite being a challenge, it is possible to attribute some human characteristics to entities in a DES model. However, such inclusion generates effects as shown through the graphs.

According to Shannon (1975), the simulation model must be easy to be controlled and manipulated by the user. Furthermore, the simulation model must also be adaptive, with easy procedures for modification or update. These characteristics proposed by Shannon (1975) guided the development of simulation models. Although, it is possible that other simulation approaches, e.g., ABS or hybrid simulation, are more suitable. We affirm this, because of the characteristics present in the construction of a simulation model considering the ABS approach. For example, in an ABS approach, entities are represented through state diagrams. Such diagrams, because of their logic, simplify programming.

5 Conclusions

This paper sought to answer the question: what are the impacts of attributing greater autonomy to the entities present in a simulation model? The impacts are:

1 increase in the number of components and connections related to workers

2 increased connections with lines of code

3 increased variability of the results generated by the model.

The increase in the number of components and connections was expected. However, the simulation model was doubled in size, and this was not expected. Adding autonomy to entities generated more *if/then/else* connections. Proportionally, almost 60% of the connections in the model with autonomous entities were made through lines of code. This fact was not expected either. The variability of the results generated by the model with autonomous entities can be seen as something positive, because this is a common feature found in the manufacturing systems composed of human decisions. The autonomous model is closer to reality. About the methodology, the iDAV method was created to analyse the structure of simulation models. Such method makes it easy to identify the importance of components and their connections in a simulation model. Furthermore, the iDAV method also helps to understand the structural part of the model and the areas that can be improved. Finally, the question for future research remains: are the effects of adding greater autonomy to entities the same regardless of the simulation approach used? The iDAV method can help us find answers to this question. Thus, for future papers, we suggest applying the iDAV method to models built with other simulation approaches, e.g., dynamic systems simulation. We also suggest choosing other graph layouts to specifically analyse other aspects such as the nature of connections.

References

Baines, T., Mason, S., Siebers, P.O. and Ladbrook, J. (2004) 'Humans: the missing link in manufacturing simulation?' *Simulation Modelling Practice and Theory*, Vol. 12, Nos. 7–8, pp.515–526.

Baines, T.S. and Kay, J.M. (2002) 'Human performance modelling as an aid in the process of manufacturing system design: a pilot study', *International Journal of Production Research*, Vol. 40, No. 10, pp.2321–2334.

Barnes, R.M. (1980) *Motion and Time Study: Design and Measurement of Work*, 7th ed., John Wiley & Sons, New York.

Bertalanffy, L. (1968) *General Systems Theory*, Braziller, New York.

Brailsford, S.C. (2014) 'Discrete-event simulation is alive and kicking!', *Journal of Simulation*, Vol. 8, No. 1, pp.1–8.

Brailsford, S.C. and Hilton, N.A. (2001) 'A comparison of discrete event simulation and system dynamics for modelling healthcare systems', *Operational Research Applied to Health Services*, Proceedings... Glasgow, UK.

Brailsford, S.C., Eldabi, T., Kunc, M., Mustafee, N. and Osorio, A.F. (2019) 'Hybrid simulation modelling in operational research: a state-of-the-art review', *European Journal of Operational Research*, Vol. 278, No. 3, pp.721–737.

Bruzzone, A.G., Briano, E., Bocca, E. and Massei, M. (2007) 'Evaluation of the impact of different human factor models on industrial and business processes', *Simulation Modelling Practice and Theory*, Vol. 15, No. 2, pp.199–218.

Bruzzone, A.G., Tremori, A., Tarone, F. and Madeo, F. (2011) 'Intelligent agents driving computer generated forces for simulating human behaviour in urban riots', *International Journal of Simulation & Process Modelling*, Vol. 6, No. 4, pp.308–316.

Di Battista, G., Eades, P., Tamassia, R. and Tollis, I.G. (1994) 'Algorithms for drawing graphs: an annotated bibliography', *Computational Geometry*, Vol. 4, No. 5, pp.235–282.

Digiesi, S., Kock, A.A.A., Mummolo, G. and Rooda, J.E. (2009) 'The effect of dynamic worker behavior on flow line performance', *International Journal of Production Economics*, Vol. 120, No. 2, pp.368–377.

Du, X. and He, W. (2018) 'Crowd evacuation simulation based on emotion contagion', *International Journal of Simulation & Process Modelling*, Vol. 13, No. 1, pp.43–56.

Forrest, J.Y. (2018) *General Systems Theory: Foundation, Intuition and Applications in Business Decision Making*, Springer, Switzerland.

Godehardt, E. (1988) *Graphs as Structural Models: The Application of Graphs and Multigraphs in Cluster Analysis*, 4th ed., Vieweg, Wiesbaden.

Herman, I., Melançon, G. and Marshall, M.S. (2000) 'Graph visualization and navigation in information visualization: a survey', *IEEE Transactions on Visualization and Computer Graphics*, Vol. 6, No. 1, pp.24–43.

Hlupic, V. and Vreede, G. (2005) 'Business process modelling using discrete-event simulation: current opportunities and future challenges', *International Journal of Simulation & Process Modelling*, Vol. 1, Nos. 1–2, pp.72–81.

Ingalls, R.G. (2008) 'Introduction to simulation', in *Winter Simulation Conference*, Proceedings... Miami, FL, USA.

Jacomy, M., Venturini, T., Heymann, S. and Bastian, M. (2014) 'ForceAtlas2, a continuous graph layout algorithm for handy network visualization designed for the Gephi software', *PLOS ONE*, Vol. 9, No. 6, pp.1–12.

Kaufmann, M. and Wagner, D. (2001) *Drawing Graphs: Methods and Models*, Springer, Berlin.

Koren, Y. (2005) 'Drawing graphs by eigenvectors: theory and practice', *Computers and Mathematics with Applications*, Vol. 49, Nos. 11–12, pp.1867–1888.

Law, A.M. (2015) *Simulation Modeling & Analysis*, 4th ed., McGraw-Hill, New York.

Macal, C.M. and North, M. J. (2005) 'Tutorial on agent-based modeling and simulation', *Winter Simulation Conference*, Proceedings... Orlando, FL, USA.

McCabe, T.J. (1976) 'A complexity measure', *IEEE Transactions on Software Engineering*, Vol. 2, No. 4, pp.308–320.

Montevechi, J.A.B., Leal, F., Pinho, A.F., Costa, R.F.S., Oliveira, M.L.M. and Silva, A.L.F. (2010) 'Conceptual modeling in simulation projects by mean adapted IDEF: an application in a Brazilian tech company', *Winter Simulation Conference*, Proceedings... Baltimore, MD, USA.

Negahban, A. and Smith, J.S. (2014) 'Simulation for manufacturing system design and operation: literature review and analysis', *Journal of Manufacturing Systems*, Vol. 33, No. 2, pp.241–261.

Polančič, G. and Cegnar, B. (2017) 'Complexity metrics for process models – a systematic literature review', *Computer Standards & Interfaces*, Vol. 51, pp.104–117.

Robinson, S. (2008) 'Conceptual modelling for simulation Part I: definition and requirements', *Journal of the Operational Research Society*, Vol. 59, pp.278–290.

Sargent, R.G. (2014) 'Verifying and validating simulation models', in *Winter Simulation Conference*, Proceedings... Savannah, GA, USA.

Scheidegger, A.P.G., Pereira, T.F., Oliveira, M.L.M., Banerjee, A. and Montevechi, J.A.B. (2018) 'An introductory guide for hybrid simulation modelers on the primary simulation methods in industrial engineering identified through a systematic review of the literature', *Computers & Industrial Engineering*, Vol. 124, pp.474–492.

Schönemann, M., Herrmann, C., Greschke, P. and Thiede, S. (2015) 'Simulation of matrix-structure manufacturing systems', *Journal of Manufacturing Systems*, Vol. 37, Part 1, pp.104–112.

Shannon, R.E. (1975) *Systems Simulation: The Art and Science*, Prentice-Hall, Englewood Cliffs.

Siebers, P.O., Macal, C.M., Garnett, J., Buxton, D. and Pidd, M. (2010) 'Discrete-event simulation is dead, long live agent-based simulation!', *Journal of Simulation*, Vol. 4, No. 3, pp.204–210.

Tamassia, R. (2014) *Handbook of Graph Drawing and Visualization*, CRC Press, Boca Raton.

Viana, J., Brailsford, S.C., Harindra, V. and Harper, P.R. (2014) 'Combining discrete-event simulation and system dynamics in a healthcare setting: a composite model for Chlamydia infection', *European Journal of Operational Research*, Vol. 237, No. 3, pp.196–206.

Wallis, W.D.A. (2007) *A Beginner's Guide to Graph Theory*, 2nd ed., Birkhäuser, Boston.