Proceedings of the 2015 International Conference on Operations Excellence and Service Engineering
Orlando, Florida, USA, September 10-11, 2015

Traceable Petri Nets for the Supply of Multiple Production Lines with Variable Waste

André Mora Barreira, Roberto Truque Salto, Andrés Mazón Arce, Luis Ricardo García Rodríguez, Eldon Caldwell Marín
Industrial Engineering Department
School of Engineering
Industrial Engineering Research Network
University of Costa Rica
San José, Costa Rica
andre.mora@ucr.ac.cr, Roberto.truque@ucr.ac.cr, andres.mazon@ucr.ac.cr,
luis.garciarodriguez@ucr.ac.cr, eldon.caldwell@ucr.ac.cr

Abstract

Petri Nets (PNs) are a modeling tool for the analysis of discrete and concurrent events originally intended for a purely computerized approach. Over time, their use has expanded to include modeling of production lines. An example is that of multiple production lines, where it is possible to run various operations in parallel, and in which there is occasional waste from sources that need to be identifiable. The traditional PNs do not allow the modeling of varying waste, assigning only a fixed portion of tokens to that state. To improve its applicability in a system with varying waste, a model which incorporates transition probabilities to the absorbent waste states with a mean and variance is proposed. For validation a production process using ordinary PNs versus the proposed model was simulated. By comparing their error rates, we found the proposal allowed modeling waste more precisely 66.7% of the time.. It´s concluded that the modeling of variable waste with PNs becomes more accurate when assigning probabilistic transitions. On all occasions it was possible to trace the relevant tokens for waste monitoring. Future research should address transition probabilities according to nonparametric statistics, and continuous flow of events to increase the range of its applicability.

Keywords

Petri Nets, Multiple Production Lines, Modelling, Traceability, Waste

1. Introduction

A fundamental problem for the computer sciences is the proper construction of systems made from information technology (IT) components for automated or organizational environments. These systems are modeled so that customers, manufacturers, users, among others, can, with the help of the model, gain a better understanding of what the system is supposed to do and how it can be implemented, used, modified and improved. In the past 50 years a multitude of modeling techniques have been proposed for these systems, where Petri Nets (hereafter PNs) have managed to maintain their position as one of the most studied and best established modeling techniques (Reisig 2013).

The PNs where created in 1962 by the mathematician and German computer scientist Carl Adam Petri for his doctoral thesis, "Communication with Automata" (Restrepo 2011). It is a tool for the analysis and modeling of processes, done graphically (Van der Aalst 2002).

When the PNs were introduced in the 60´s, the ideas were too advanced to put them into practical use and consequently their attained response was limited. However, by the end of the 60´s they were used in the Project MAC (a research project in the areas of operating systems, artificial intelligence and computation theory) that was taking place at MIT. In the 80's, with the introduction of the colorful PNs, the descriptive ability of the model was enhanced, allowing it to be applicable to larger projects, which stimulated its use. In the 90´s, its use was more common partly due to the rise of interest in graphic modeling techniques, which cemented the PNs as an important contribution to computer science (Reisig 2013).

This mathematical model has been implemented within the industries because of its graphic ease, wealth of information and available simulation software for production lines. However, due to the complexity of the topic, it has mainly been used in the area of automated production lines modeling (Cheng, 2005).

This scientific paper consists of a theoretical framework through which basic terms are explained so that the reader can develop an understanding of the subject and thus understand the following sections. It will review what the supply of multiple production lines consists of, followed by an explanation of the mathematical model of ordinary Petri nets and end with a conceptualization of waste, traceability and they´re importance.

After the theoretical framework an explanation of the proposed model will be presented, explaining why the 5 tuple associated with the ordinary Petri nets will be transformed into a 7 tuple so as to incorporate waste and traceability into the model. It should be noted that the proposals contained within this scientific paper seek the improvement of certain specific aspects within a pre-existing system, given the fact that real data must be utilized for their application.

The proposed model's functionality is demonstrated through a validation, assessing how it might be applied in a real industrial scenario. The Inputs utilized for the validation where extracted from the production line of a company operating with machines working in parallel.

2. Theoretical Framework

2.1. Supply of multiple production lines

Supply, usually, can be defined as the inventory building process given the set objectives in the inventory plan. Production lines have their origin in the Industrial Revolution, where customized production systems gave way to standard production systems. This shift is made possible by production lines, which are defined as a process in which parts or inputs generate a product or are added to a preassemble in order to add value to it. The need to produce more volume comes over the years, and is treated with more complex production systems such as multiple production lines. The latter come to eliminate bottlenecks in a line or reproduce an existing line in order to increase the finished product's output.

Concatenating previously established definitions, we define the theme of this section. Therefore the supply of multiple production lines focuses on the supplement of a multiple production line in order to increase the output of finished products so that the Work in Progress (WIP) decreases, allowing to comply with the provisions in the inventory plan (Medina, 2014).

2.2. Mathematical Foundations of ordinary Petr nets

Petri nets, informally, are a mathematical structure composed of three types of elements: places, transitions and arcs. By convention, places symbolize the passive aspects of the system, the transitions are associated with the dynamic aspects, and arcs represent the causal link between passive and active elements (Restrepo, 2011). An ordinary Petri net is a 5 tuple, where:

- $L = \{l_1, l_2, \dots, l_m\}$ is a finite set of places
- $T = \{t_1, t_2, \dots, t_m\}$ is a finite set of transitions, such that $L \cap T = 0$.
- $F \subseteq (L\ x\ T) \cup (T\ x\ L)$ is a finite set of arcs.
- $W: F \rightarrow \{1,2, \dots\}$ is a function that associates a weight to each arc.
- $M_0: L \rightarrow \{1,2, \dots\}$ is the initial marking of places.

The behavior of systems can be detailed in terms of movements of tokens. These movements occur when a transition enabled to run fires by changing the marking of their places of entry and exit. These changes of state or marking are governed according to the following rules (Murata 1989):

- It is said that a transition t is enabled if each input place l is marked with $m(l)$ tokens and $m(l) \geq w(l,t)$ where $w(l,t)$ is the weight of the arc between l and t.
- An enabled transition may or may not fire depending on whether the event really comes to pass.
- The firing of an enabled transition t removes $w(l,t)$ tokens from each input place l of the t, and adds $w(t,l)$ tokens to each output place l of the t, where $w(t,l)$ is the weight of the arc between t and l.

The dynamic behavior of these systems can be described and validated using matrixes. To perform a firing simulation for the transitions which are intended to be fired Equation 1 is used.

$$M_K = M_{K-1} + A * U_k \quad (1)$$

The incidence matrix (A) for a Petri net with n and m transitions places, is a n x m matrix of integers that typically arises according to Equation 2

(2)

Where (represents the weight of the transition arc *i* from its input place *j* and y is the weight of the transition arc *i* from its output place *j*. , represent respectively , the number of tokens to remove , add , and change from the location *j* when the transition *i* is triggered once . However, the transition *i* is enabled in the marking if:

(3)

The state equation ( ) is a column vector m x 1 that denotes the number of tokens in place *j* immediately after a firing sequence *k* fires. The control vector ( ) is a column vector n x 1 of n - 1 zeros and a non - zero entry (a 1) in position *i* indicating that the transition *i* is shot in the firing *k*. This indicates that there is only one shot transition, however several transitions could be triggered simultaneously so long as they don't all come from the same place *L*. Note that the row *i* of the incidence matrix denotes the marking change as a result of the transitions firing *i* . To illustrate this the following example where the places *L* are represented by *P* (Murata 1989) we provide:

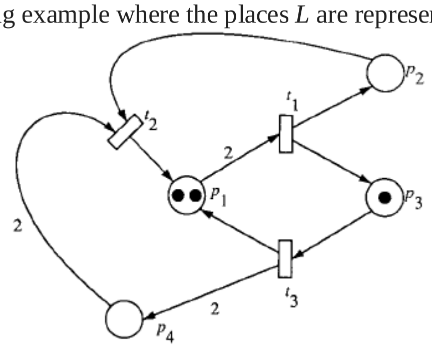

Fig 1. Example of a PNs

In the PNs shown in Fig. 1, we have a transition being shot so as to result in the marking coming from :

$$M_K = M_{K-1} + A * U_k \quad (1)$$

$$\begin{bmatrix}3\\0\\0\\2\end{bmatrix} = \begin{bmatrix}2\\0\\1\\0\end{bmatrix} + \begin{bmatrix}-2 & 1 & 1\\1 & -1 & 0\\1 & 0 & -1\\0 & -2 & 2\end{bmatrix}\begin{bmatrix}0\\0\\1\end{bmatrix}$$

From the first row of incidence matrix ( ) it's evident how the place could lose two token 's to transition . Then, place could win a token from the transition and finally the place could win a token from transition . The same logic applies to each of the following rows of the incidence matrix ( ), eventually reaching the fourth row where the behavior of place is shown.

$$\begin{bmatrix}3\\0\\0\\2\end{bmatrix} = \begin{bmatrix}2\\0\\1\\0\end{bmatrix} + \begin{bmatrix}1\\0\\-1\\2\end{bmatrix}$$

After multiplying the incidence matrix ( ) with the control vector ( ) it is easily appreciated how, if the remaining sum is carried out the equality holds, proving the suitable behavior of the dynamic system in question. The matrix obtained by multiplying tells us how earns a token, has no movements, loses a token, and the place gains 2 token 's as a result of the partition that takes place in transition (which previously was indicated in the incidence matrix).

2.3. The importance of waste control and traceability in a production line

Traceability and waste control are applied in production lines to reduce costs and improve the products and lines quality. In the case of waste, a minimization of such is expected so as to reduce the involved costs and increase the production lines efficiency. Waste is currently defined as all that that is different from the minimum amount of equipment, materials, parts and work time that is absolutely essential for the production. However, this scientific article will refer to waste production as an absorbing state which involves irreparable defects on an article during the production process. Having waste implies costs to get rid of such defective material, as well as costs to order more raw material than that that would be calculates from the Bill of Material (BOM) for the manufacturing of the required units. Moreover, waste increases production times which in turn affects manufacturing even further. Therefore it is always important to identify where waste is generated at each stage of the production line, so as to find ways to decrease it.

Traceability according to AECOC (2008), are those preset and self-sufficient procedures that allow to know the historic, location and trajectory of a product or batch of products throughout the supply chain at any given time, through certain tools.

Traceability can be divided into three components: 1) back traceability 2) intermediate traceability and 3) forward traceability. Backward traceability will allow taking from completed or in-process product important information from it down to its raw materials. Internal traceability will relate entering products into a company with those that will come out of it. Finally, forward traceability will allow knowing where a final product is to be distributed; by taking a specific material sample it can be known from which finished product it came from. (Rojas 2012)

The implementation of a traceability system allows to find where it is most important to make improvements in a process, providing more information about the failures, waste and the times of each thread related to this. (Sanchez 2008) With this, you may notice that traceability enhances the quality of both products and processes of a production line.

3. Proposed model

There are various reasons to model a system using PNs, which may include: to create and evaluate a new system's design, to compare alternative designs of a system and to research possible improvements to a real system. The proposals presented in this paper will aim towards the improvement of the modeling of certain aspects of a real system. When using PNs to model and analyze real systems, it is usually necessary to expand the basic model by which they work. This is done because simple PNs are not suitable for the modeling of many systems in logistics, production lines, communication and manufacturing (Van Lint 1992).

In this section, we will develop a new model which will be referred to as: traceable PNs for the supply of multiple production lines with variable waste. This section will begin by explaining waste in PNs, including the way it is traditionally modeled and how it will be modeled in the proposal. Then, it will continue by explaining the current existing methods of traceability in PNs and the improvements that are proposed in this area. This section will be concluded by presenting the complete, consolidated model based on the proposals for waste and traceability.

The method which can be used to model waste with conventional PNs uses fixed proportions to determine a percentage of tokens which will flow from a current place to the next one. This means that, for a set number of tokens, the amount which flows from one place to another will always be the same. A set proportion of the tokens will pass to the next place while a complementary proportion will be assigned to a different place which will be in conflict with the previous place. This may represent that certain materials which flow through a production line can go to two different destinations: 1) Ending as waste or 2) To continue towards the next step of the production line (Zimmerman 2008). However, by using fixed proportions the model becomes less accurate.

It is emphasized that the modeling of a system allows, after analyzing it, to find improvements that may be applied towards its reality. This means that a robust model will be one that allows a more realistic analysis of a system, which is why this proposal will be aimed at the improvement of waste modeling in a PN. This will be achieved by utilizing probability functions that will be obtained from the analyzed operations which are represented by the model. An example of this will be shown in the next section.

To accomplish this, a new space will be added to the 5-tuple of simple PNs that includes probability functions linked to the control vector, which as explained on the theoretical framework, decides which transitions will be fired. This additional field of the tuple will be denoted as D, and will be defined by:

$$D: T \rightarrow F^{+}$$

This implies that each transition is assigned a value defined by a related probability function, which represents its likelihood of firing. The way in which this new field of the tuple will interact with the control vector is defined as follows:

$$F_x(x) = y \;/\; P(U_{k(i,1)} = 1) = y$$

$$U_{k(j,1)} = \begin{cases} 1 \; si \; U_{k(i,1)} = 0 \\ 0 \; si \; U_{k(i,1)} = 1 \end{cases}$$

F_x represents the probability distribution belonging to F^{+} that will be linked to t_i, which is the transition of a token from an operation (place) to the next in the line of production. The value "x" represents any real data (means, deviations, etc.) which the distribution F_x requires and will be related to the waste in the operation. The result of the distribution is a value "y" that dictates the likelihood of t_i firing, i.e., that its associated value in the control vector is 1. The transition t_j represents the flow of a token from the same operation (input place) from which t_i fires, but towards a different place, which is waste. However, the transition t_j is triggered only when t_i isn't, which means it will only take value of 1 in U_k when its complementary transition t_i has a value of 0 in the same vector.

Once the waste is obtained, it is of value to know its origin and path to make corrections on the line if necessary. For this reason, it is important to have a traceability system that supports the proposed model of waste. Only a couple of analytical methods have been developed to provide a certain amount of traceability using PNs, including Colored Petri Nets (CPNs) proposed by Jensen (2009). In these, data is attached to each token, making it a certain color which can be modified by transitions, which allows to distinguish the resources used in the model. These colors enable certain transitions that require a specific type of token (specific color), corresponding to the expression of its arc. From the model of data records that the structure of the CPNs allows, we pretend to assign records that, by linking together data assigned to each token, will allow to trace the path of the tokens along the model. Therefore, it is proposed to add another space to the 5-tuple of ordinary PNs that saves a record "v" which will register the indicative values of the places l through which the token transits. Whenever two tokens are consumed in a transition t from different entry points (e.g. l_1 and l_2) their values will be linked together in the record "v". Therefore, the new token produced by the transitions will have a record $v = v_1 + v_2$, where v_1 comes from a token with input place l_1 and v_2 from another token with input place l_2.

Identifying each specific token found in the place waste cannot be done only with their trajectory. Of course, the path is known with the trajectory, but with it it's impossible to differentiate, which token came first to the place assigned as waste and which of all the tokens with the same route came last. This means it´s necessary to take time into consideration in addition to the path, allowing traceability to each token by assigning a set of unique and differentiated values. This is done using timed PNs, which include a delay associated with each modeled transition (Murata 1989). A delay is defined as the total time it will take to run the transition. This means another space will be added to 5-tuple of ordinary PNs which includes a function marking the delay associated with each transition. This additional field of the tuple will be denoted as Tm defined by:

$$Tm: T \rightarrow \mathbb{N}$$

This implies that any transition may be assigned a natural number representing its delay.

By consolidating all the previously established additions in this section towards the tuple associated with ordinary PNs, the result is an 8-tuple that mathematically represents the proposed model. Thus, the PNs used in this proposal are defined as follows:

$N = (L, T, F, W, \overline{M}_o, D, C, Tm)$ where:

- $L = \{l_1, l_2, \dots, l_m\}$ is a finite set of places.
- $T = \{t_1, t_2, \dots, t_m\}$ is a finite set of transitions, so that $L \cap T = 0$.
- $F \subseteq (L\ x\ T) \cup (T\ x\ L)$ is a finite set of arcs.
- $W: F \rightarrow \{1,2, \dots\}$ is a function that associates a weight to each arc.
- $M_0: L \rightarrow \{1,2, \dots\}$ is the initial marking of places.
- $D: T \rightarrow F^+$ is a set of probability distributions assigned to any transition
- $C = \{v, tr\}$ is a duo containing the records of value and time that can be assigned to a token
- $Tm: T \rightarrow \mathbb{N}$ is the set of delays associated with any transition

The explanation of the first 5 tuple spaces can be found in the theoretical framework, in the mathematical foundations of ordinary PNs section. The last three tuple spaces were previously explained in this section when the need for these was introduced.

4. Proposal and methodology

In order to support the proposed model in the previous section, a comparative study of the results will take place through regular PN in its current state and the proposed model. For this, will use as reference a real bakery that has multiple production lines, which will be taken as an example for the further validation of this analysis.

For the present purposes, only two operations of the process of the bakery will be modeled: mixing and molding of the dough. Each batch produced in these operations represents a specific amount of breads, which corresponds to the maximum capacity of tolerance that the machines involved can handle. However be understood henceforth that the total amount of mixture used in the first step corresponds to 250 tokens and the same relation will continue in the following stages.

The comparative study was developed through a simulation using the software called CPN Tools and with the additional support of Microsoft Excel. CPN Tool is used for modeling the proposed model due to its popularity in the field of software's of PN and versatility. In addition Excel is used for supplementary functions that were not possible being done in CPN Tools. These functions correspond to the allocation of a normal probability on the transitions or arcs of the network. Because the PN are deterministic, if the model is run by itself, decisions on transitions would take randomly, or could be assigned a fix load for each arc. However the aim with the model is that in a transition that is at a decision point, we can be assigned a stochastic load on it, so that this transition will be probabilistically triggered and will enable the passage of the token at the chosen place. Using Excel, the actual machine´s waste record was analyzed and based on this, a mean and standard deviation associated were calculated. The following section will expand on this issue and the data obtained with a corresponding analysis and justification will be presented.

Mixing and molding of the dough were chosen because they are fully automated, and in order to validate the proposed model, the line of production must contain at least 2 machines working in parallel. A diagram of the processes that will be model is shown in the figure 2.

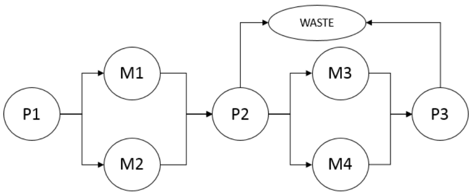

Figure 2. Process diagram of the bakery

In order that the reader fully understands the validation section, a briefly explain of how the process works will be given so that you can familiarized with the modeling. On figure 2 you can see that the process starts in P1, corresponding to the stage where the raw material will be, in this case, the mixture in kilograms of the dough. This will be incorporated into the machine 1 and machine 2 (read by M1 as machine 1, M2 and machine 2, M3 as machine 3 and M4 as machine 4; where M1 and M2 work in parallel and both are mixers, and M3 and M4 also work in parallel and are both molding machines). In this example, of the total of 250 tokens contained in P1, 125 will be given to M1 and the other 125 will be given to M2, as this is the capacity of the mixer. Similarly for molding machines (M3 and M4) will be distributed in equal amounts the amount of mixing that reach P2 without the waste produced in M1 and M2. Note that the amount reaching P2 is referred to as without waste because the aim is to create, in the model, a collection place of waste for each machine (see waste associated P2 and P3 in figure 2), and then carrying out the simulation, to determine of which machine the waste belongs to. The same logic will be developed for molding machines. The purpose is to receive at P3, the amount of bread ready to be baked, represented in kilograms, and be able to determine the amount of waste (in kilograms) generated by each machine and finally, know from which machine it belongs to. This will be done by a simulation that considers historic waste data of the machines and assigns a probability to each batch of product without waste and waste. Noticed that batch will be understood as the amount of tokens that are consumed by machine.

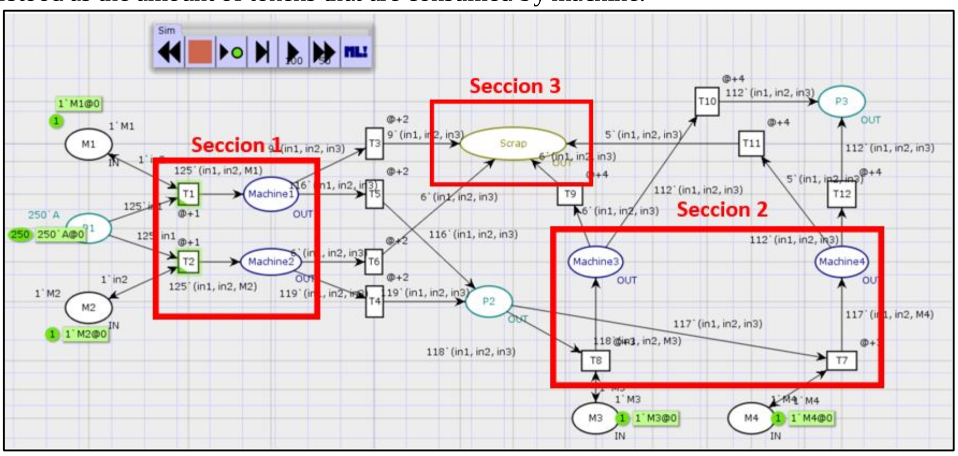

Figure 3. Model Validation where: a) section 1 corresponds to the mixing area, b) section 2 corresponds to the molding are, c) Section 3 is the waste or scrap area.

On Figure 3 it is represented how the model is structured by the use of CPN Tools. Where P1, P2 and P3, as in Figure 2, represents stations which operated the ins and outs of tokens. Also notice that section 1 is conform by the mixing machines, section 2 by molding machines and section 3 is the waste area.

As explained earlier in this document, lead a traceability of the tokens is wanted. That is why in this model must be created places (see M1, M2, M3 and M4) that release action tokens whenever an input token passes through the machine 1, machine 2, machine 3 or machine 4 respectively. Such actions are concatenated to create a traceable input token by assigning a code every time it passes by and specific machine. In addition to this, each transition is associated with a delay established. These delays will make this Petri Net similar to the Colored and Timed Petri Nets (CTPN), where will be printed on every token a delaytime as the tokens pass through transitions, increasing while the global time passes. When this token reaches a final place, in this case the waste area or P3, it will have a delaytime registered to him, along with an action code given as 1` (in1, in2, in3)delaytime, where in1 corresponds to the initial input, in this case bread mixture in kilograms, in2 will be the action token given by the mixer machine (either M1 or M2), in3 will be the action token given by the molding machine (either M3 or M4) and the delaytime will be the time when the token goes to the final stage (either waste area or P3) acquired by all the delaytime associated to the transitions where it passed. In the case of this modeling, the unit of time will not be representative since its objective is to maintain a tracking unit time. The following code is used for concatenation of input and actions:

$$colset\ IN\ =\ with\ M1\ |\ M2\ |A\ |\ M3|M4\ timed;$$
$$var\ in1, in2, in3{:}\ IN;$$
$$colset\ OUT\ =\ product\ IN * IN * IN\ timed;$$

This code had to be used because the CPN Tools does not have a traceability function, therefore it was incorporated to the proposed model. The reason why this function had to be used is because in order to concatenate tokens, they must be previously defined as variables. Once the IN function is executed, in the case at issue in P1, the OUT function will concatenated the data (see Figure 3). For this specific case the concatenated code will be created as 1` (in1, in3, in3) which has 3 positions, but as it is shown in Figure 3, section 1 only has to variables involved, where the third variable would be present till section 2. However, this third variable has to be add in section 1 because CPN Tools doesn´t allow new variables to be incorporated in the concatenated code. Analyzing this with the proposed case, shown in section 1 of Figure 3, the result of the trigger of T1 will be in the form 1` (A, M1, M1), where A = in1, M1 = in2 and M1 = in3 where in3 corresponds to the last variable concatenated, in this case in2 = in3. Later on in section 2 of Figure 3, when T8 is triggered a new value will be added to the concatenate code corresponding to the action token given by M3 or M4. As it was previously shown the color of the token before T8 is 1` (A, M1, M1) but after T8 is triggered, the new action token, in this case M3, will be added to the concatenate code replacing the value of in3 with M3 giving now the final color of the token in form of 1` (A, M1, M3).

The limitation that presents CPN Tools on the issue of traceability, is the inflexibility presented when concatenation of tokens has to be done because it doesn't allow the incorporation of new values to the color or concatenated code, where as shown above, had to be down by replacing action tokens with fake action tokens (case of replacing M1 located at in3 with M3 after T8 was triggered).

5. Validation Results

Once all the parameters for the process model established, proceeds the analysis of data from the production process, only taking into account the two steps in the process, this in order to analyze the behavior of waste for these two operations. For these data you get the following:

Table 1. Waste per Batch Data Summary

Waste per Batch	
Mean	0,09928726
Standard Deviation	0,02356832
Sample Variance	0,00055547
Range	0,09960159
Minimum	0,04780876
Maximum	0,14741036

This table shows that the actual data has a mean of 9.92% waste, with a standard deviation of 2.36%. With that done, we want to confirm that the data presented normal behavior, so that the probability function is adequate to model the process. In conducting this analysis, the following is obtained:

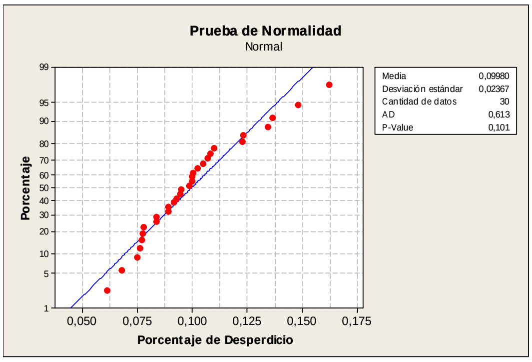

Figure 4. Normality Test

In turn, the same takes place with the characterization of the two machines stage 1 and stage two machines 2. Both machines have waste turn with normal behavior. The mean and standard deviation units waste for these operations per machine is shown in the following table:

Table 2. Mean and deviation per machine

Machine	Mean	Deviation
1	7,36	2,02
2	6,37	1,35
3	6,27	1,52
4	4,13	1,24

This table shows the average measurement units that are waste from the machines 1 and 2 for the operation number 1; and machines 3 and 4 for operation number 2. Note that these data also have a normal waste for processing machines behavior.

5.1. Ordinary Petri Nets

As stated, ordinary Petri Nets are governed only by deterministic values to fire transitions. That's why *triggers* are governed by a single value that does not change over time, and this value in the case of waste is given by the average of the data collected waste. Whereby a modeling is done by ordinary Petri net and the following data was obtained:

Table 3. Modeling of waste by ordinary PNs

	Waste in Operation	Waste per Machine	
Operation 1	13	7M1	6M2
Operation 2	10	6M1	4M2

Therefore the average total waste is preset with a value of 9.92% of waste and therefore always going to waste 25 units of a batch of 250 units. This would be correct if it is assumed that the waste is always constant, however this is not a reality. So now the objective is to know whether the use of probability distributions can improve the model of Petri nets.

5.2. Proposed Petri Nets Implementation

It then proceeds to execute the proposed Petri net with normal transition probabilities assigned to the firing of transitions in the model. This simulation is performed 30 times in order to compare the results with actual data, and to carry out an analysis of it. This is done in the next "Comparison of Results" section. However as a sample, deploying the tool to a production batch, which is 250 units of measurement as described. The following figure shows the configuration of the production process and the simulation done:

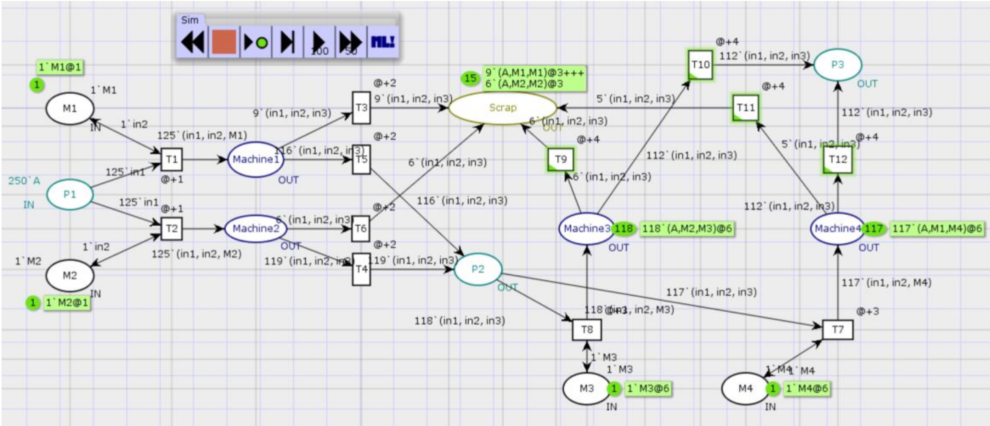

Figure 5. Run of demonstration of the model proposed

With the transition probabilities generated alternately to the software, it is get which transitions should be fired, when they should be fired and its weight in the bow (ratio of 250 mass units of measure). In this case for the firing transitions, we have the following:

Table 4. Modeling waste by the proposed model

Operation	Machine	Waste per Machine
1	M1	9
	M2	6
2	M3	6
	M4	5
	Total Waste	26

The beginning of this simulation, which has already completed stage 1 of the process, is then displayed. As can be seen, on completion of this stage with these probabilities, 15 tokens (which represent units of measure 15 250) go to waste after ending this operation, while 235 continue the process. This operation starts at time 1 and ends at time 3 (lasts 2 units of time), so as you can see in the picture below. On-site is possible to identify the source path of the tokens in the waste place, and its arrival time (which is the time 3).

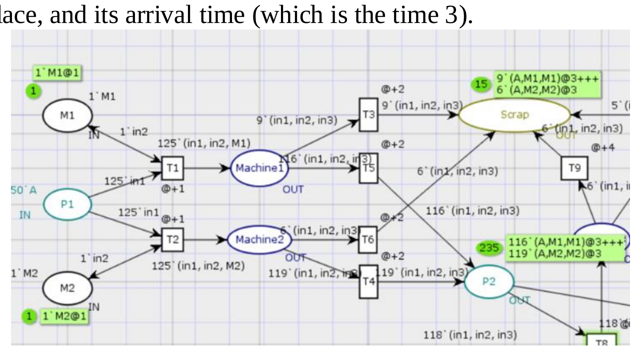

Figure 6. Demonstration of the concatenation of the tokens in the model proposed in stage 1

The same applies to execute the second operation, and the shot of their respective transitions. It is observed in the image that from the 235um who were still in the process: 118um enter the time machine 3 in time 6, and 117 enter machine 4 in the time 6 also. This is because it is allocated a time of 3 between the end of the previous operation and the start of this. Finally with the probabilities previously generated, the results were that machine 3 would generate 7 units of measurement wasted and machine 4, 5 units of measure wasted.

Finally it is shown in the following figure simulation of this lot in the final state:

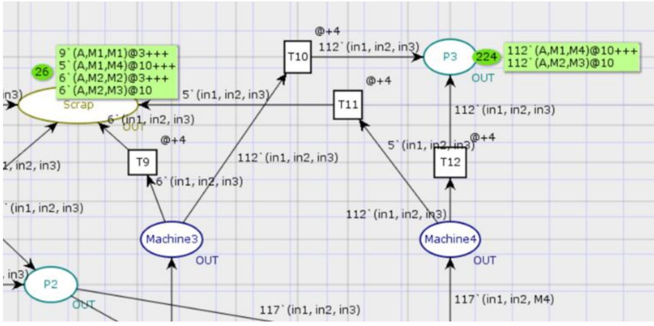

Figure 7. Demonstration of the concatenation of the tokens in the model proposed in stage 2

To understand the figure must be a total of 26 tokens arrive at the place of waste, and 224 follow the process (outside the scope of the model). Meanwhile, of the 26 tokens stored on-site waste, there is information in the tuple that allows in this case generate a traceability. For example, it has that of the 26 tokens, 9 come from the machine 1, 5 come from the machine 4 (which also passed through machine 1), 6 of the machine 2, and finally 10 tokens coming from machine 4 and 2. In turn you have the time each token arrived to this place, which in turn allows distinguishing each batch of the batch as a single actor in the system. To summarize the proceeds from the previous simulation shows:

Table 5. Data table for lot 1

Tokens in waste place	Provenance	Time Income
9 tokens	M1	3 units of time
5 tokens	M1M4	10 units of time
6 tokens	M2	3 units of time
6 tokens	M2M3	10 units of time

Clarification: The concatenation "M1 M4" for example indicates to me that the path of origin includes first and then M1 to M4. Also noteworthy is that the latter is only to show the traceability in waste. Then we proceed to analyze the results according to the results of waste. As shown in this section shows only the simulation of 1 lot for understanding what has been done. This is performed 30 times and the results are shown in the following section.

5.3. Comparison of Results

To illustrate the comparison between the two models, the results for the model shown in the previous section for the batch of 250 units are shown. For this specific example it was obtained as follows:

Table 6. Comparison between models in measurement units per batch

Petri Net	Simulated Waste	Real Waste	Error Percentage
Proposed PN	26	27	3.70%
Ordinary PN	25	27	7.40%

However, this is only a case. This comparison was made for waste actual data 30, and the average error percentage was obtained for each of Petri nets used.

For the results of the Petri net proposal, as there is a probability parameter, the results vary each time the Petri net is running. While for the results of the ordinary Petri net, you have already known what will always be waste. Thus sufficient to compare the average allocated waste, with the actual waste for 30 lots to meet the error rate of this model. It can be verified in Table 7, the percentage of times each type of PN have a lower error rate compared to the other, concerning actual results.

Table 7. Closer Results per PN Model

Closer Results	
Ordinary PN	30%
Proposed PN	66,67%
Draw	3,33%

In turn, complementing, the average error percentage for each Petri net is shown below:

Table 8. Comparison between Petri Nets

Simulated Waste	Average Error Percentage
Proposed Petri Net	10.50%
Simulated Petri Net	19.08%

According to these results shows that on average Petri Net proposed by the group have on average lower error rate than ordinary Petri Nets when compared both with the actual data modeling process. In Figure 8 shows the behavior of both Petri nets modeling approaches, and can be seen as the simulated PN get closer to the real waste behavior, while ordinary maintains an average value which is not necessarily close to real value of waste.

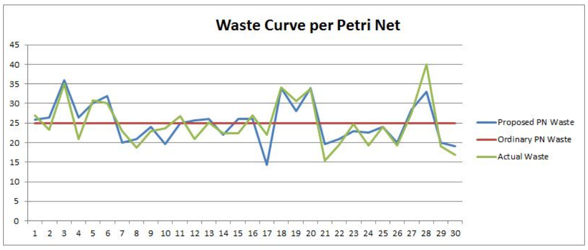

Figure 8. Comparative graph of the behavior of the results against the actual data

With these results, we can mention then that the proposed Petri net with transition probabilities is more adjusted to reality reached in this article, which satisfies the hypothesis of the research group.

6. Conclusions

1. We conclude that although originally Petri nets are used to model discrete models deterministically, after evaluating the model assumptions is confirmed that the application of probabilistic transitions likely maintain their unaltered state equation or any of contrasts their assumptions, and fits better to data due to probabilistic transitions.
2. Using concatenation information relating to transitions triggered in the model is achieved distinguish the tokens as individual actors within the system using an updatable record in each transition during the simulations performed. Thus it concludes that this record of information allows tracing the origin of a token in an absorbing state, since in all cases it was possible to know the origin, route, type and time of arrival of tokens to place waste.
3. While the results indicate that the proposed model appears to improve the simulation of variable waste, for the 67.67% of the time was closer to reality, it is concluded that the results could be improved by testing other probability distributions where behavior data can be adjusted better.

7. Recommendations and future research lines

1. The scope of this research includes the use of parametric statistical modeling Petri Nets. However, in processes where the data used is nonparametric, the proposed model cannot be applied. It is recommended to investigate the use of transitions for nonparametric statistics and thus further extend the range of Petri nets simulation applicable to flexible manufacturing systems.
2. Although the concatenation methodology for modeling this process achieves the generation of relevant traceability information, this methodology in a complex production line with many threads processes imply that the information associated token may be so large it unreadable in a simulator. It is therefore recommended developing a code of threads in the token, which involves creating special functions in the simulator and this would be useful in such processes.
3. Petri nets are used by definition to model discrete event systems, but in continuous flow systems undistributed may not be as appropriate for your application. This is why it is recommended to investigate the use of Petri nets for modeling continuous events where events vary with time in order to expand the applicability of Petri nets.

References

Asociación de Fabricantes y distribuidores (AECOC), Definiciones de trazabilidad, Reporte Técnico AECOC. Comité de trazabilidad alimentaria. Asociación Española de Codificación Comercial. Logística Inversa Trazabilidad p.5, 2008.

Chen H. , Modeling and Performance Evaluation of Supply Chains Using Batch Deterministic and Stochastic Petri Nets, *IEEE Transactions on Automation Science and Engineering* ,Vol. 2 No. 2, pp 132-144, 2005.

Jensen K. & Kristensen L. M., Coloured Petri Nets: Modeling and Validation of Concurrent Systems, Springer-Verlag, Berlin Heidelberg, 2009.

Medina J, Seck J C, Hernández N, Escamilla N. J., Applying an Adaptive Petri Net to Calculate the Makespan in the Job Shop Scheduling Problem, *26th European Modelling and Simulation Symp*. Bordeaux, France pp 463-468, 2014.

Reisig W., Understanding Petri Nets: Modeling Technics, Analysis Methods, Case Studies, *Springer-Verlag* Berlin Heidelberg, 2013.

Restrepo P. Un Método Computacional para la Detección y Caracterización de conflictos en Redes Petri, *Revista Ingenierías Universidad de Medellín* Vol. 10 No. 19, Colombia pp 189-200, 2011.

Rojas O A, Gómez A, Tumbajoy L, Velasco J. M., Modelado Dinámico del Proceso de Trazabilidad de Producto para Redes Petri de WorkFlow, *Revista de la Facultad de Ingeniería de la Universidad de la Salle* No. 19, Colombia pp 117-144, 2012

Murata T., Petri Nets: Properties, Analysis and Applications, *Proceedings of the IEEE* Vol. 77 No. 4, pp 541-580, 1989.

Van der Aalst W, van Hee K., Modeling Workflows, MIT Press, *Workflow Management - Models, Methods, and Systems*, Cambridge, 2012.

Van Lint J.H., Timed coloured petri nets and their applications to logistics, *TNO Institute for Production and Logistics (IPL),* Holland pp 2, 1992.

Zimmerman A., Stochastic Discrete Event Systems: Modeling, Evaluation, Applications, Springer-Verlag Berlin Heidelberg, 2008.

Biography

André Mora Barreira was born in San Jose, Costa Rica in 1992. Attended high school at La Salle high school graduated in 2009. In 2010 enter to the career of Industrial Engineering at the University of Costa Rica where he is pursuing his ninth semester. On the beginnings of 2011, he started working for the company Red Bull, where he was under the charge of SBM for 3 year. Contact: andre.mora@ucr.ac.cr

Roberto Truque Salto was born in San Jose, Costa Rica in 1991. He studied in Lincoln School and graduated in 2010. In 2011 he began studying Industrial Engineering at the University of Costa Rica where he is pursuing his ninth semester. He completed an internship at FIFCO in the logistics area during 2013-2014, and is currently an intern in the business intelligence unite at Ernst & Young. Contact: Roberto.truque@ucr.ac.cr

Andrés Mazón Arce was born in San Jose, Costa Rica in 1993. He culminated primary and secondary courses at Methodist High School of Costa Rica until 2010. He entered the year 2011 at the University of Costa Rica in the career of Industrial Engineering and currently is studying the ninth semester. Contact: andres.mazon@ucr.ac.cr

Luis Ricardo García Rodríguez: was born in San José, Costa Rica in 1993. He attended high school at Saint Joseph High Scool and graduated in 2010. In 2011, he entered the career of Industrial Engineering at the University of Costa Rica where he is pursuing his ninth semester. Contact: luis.garciarodriguez@ucr.ac.cr

Eldon Glen Caldwell Marín, Outstanding Service Award Recipient, IEOM Society, full professor (Cathedraticus), University of Costa Rica with over 20 years of teaching and research experience. He is teacher and mentor of the authors of this article on the topic of Operations Engineering at the University of Costa Rica.Bachelor and Master in Industrial Engineering at University of Costa Rica, also with a Master degree in Operations Engineering at ITESM, Mexico, Financial Analysis and Marketing of Services at Inter-american University of Puerto Rico, Health Services Management at UNED, Costa Rica and finally Ph.D. in Industrial Engineering at the University of Nevada, USA. Currently he is a doctoral researcher at the University of Alicante, Spain, and Academic Excellence Prized (2013 and 2014) researcher at the Ph.D. Program in Education at University of Costa Rica. His research interests include production scheduling, information retrieval, robotics and intelligent development of methodologies for implementing lean systems. Contacts:eldon.caldwell@ucr.ac.cr /egcm@alu.ua.es