An innovative proposal for systemic modeling, analysis and simulation in a continuous production process

René Tapia[a*], Pablo Viveros[b,c], Adolfo Crespo[c]

[a]RelPro S.A, Santiago, Chile

[b]Universidad Técnica Federico Santa María, Department of Industrial Engineering, Valparaíso, Chile

[c]Department of Industrial Management, University of Seville, Spain

Abstract:

This research aims to develop an innovative proposal for systemic modeling, analysis and simulation. There are different techniques to estimate availability and production in continuous production plants, being one of the most used the Reliability Block Diagram technique [1-3] because of its simplicity and probability framework, but it has significant approximations. Another widely used technique to model and analyze are Markov-Chains (discrete time) [4]. Both techniques have been specially designed to analyze for a limited number of functional settings and disposition of elements, like parallel, stand-by, serial, and so on [5-7]. Additionally, the last mentioned techniques do not allow studying the variability of variables which is commonly demanded.

So this paper is focused to develop a technique based on specific algorithms which allows engineers to have a broad perspective of the system with a flexible layout framework, simulating the production, availability and runtime of the plant considering the impact of each random event over each one of all elements of the process. The presented algorithms have the advantage that are made based on the occurrence of each event, so the time duration of processing is based on the number of random events [8, 9], reducing dramatically the simulation times.

Keywords: Availability Analysis, Reliability, Simulation of Maintenance, Continuous Production Plants, RelPro

1. INTRODUCTION

The study of the behavior and performance of production systems have been always a constant need, in order to get a maximum return on the basis of costs and expenses to be taken. To estimate the production based on maintenance activities, i.e. repair or other kind of events, formerly was not a task to be executed in design stages, but nowadays is an essential process towards competitiveness.

Diverse methodologies have been developed to measure the performance of the plants or systems, like Markov Chains, Petri-Nets and the extension of Reliability Block Diagrams applied to estimate availability. And more recently has been incorporated the computer simulation of the process. Each methodology has its own advantages and disadvantages, which will be studied further.

The main disadvantage of the mentioned methodologies is that most of them do not consider the dynamic impact of the events that occurs in any equipment to each one of the rest of the equipment of the plant. Markov Chains and Petri-Nets are technically impossible to implement at a level of the entire plant, because of the large number of equipment, and even in computers the layout of the states becomes technically infeasible. Reliability Block Diagrams constitute just an approach, which can lead to significant errors. And most of the simulation environments are not designed to model and represent the impact of the events of each equipment to each one located in the plant.

2. PROBLEM STATEMENT

Current methodologies presents several disadvantages in modelling and representation of production systems, the most important disadvantages are:

*rene.tapia@relpro.pro

- Most the methodologies are approaches.
- Most of the methodologies do not consider risk
- There is no methodology that considers the Life Cicle of the assets in the estimation.

2.1. Approaches to estimate

There are some important approaches in the estimation and calculus of availability in productive systems. One of the main approaches, with higher impact from the theoretical vision to a real vision is related mainly with the use of Markov Chains, assuming that the transition rates are constant, and during all the calculated period. Many authors in this specific field of availability and production calculation uses Markov Chains [4, 5, 6], but indeed they only use constant transition rates, which means only exponential distributed times [10]. However, processes of interest like reliability, availability, maintainability, and safety (RAMS) do not necessarily suggest a behavior described by exponential times. In this matter Petri-Nets have is advantaged because this methodology, in spite of being a formalization of transitions and occurrence of events in a single graphical modelling, is used in simulation mainly in Generalized Stochastic Petri-Nets (GSPNs) [11], in which allows to establish customized time probability distributions, (this will be studied further in next sections). In the case of RBD, there is also possible to use custom probability distributions of time [1]. But PNs (GSPNs incl.), Markov-Chains and RBD are also not allowed to do analysis in production systems in which there are machines that work by default with idle capacities, unless significant changes are made in the modeling, that in the case of RBD and Markov-Chains are necessarily approximations. Other approach is that those methodologies does not consider the different production levels that the system could have, are simplified to be used with a single production rate.
In the special case of RBD, it has several approaches, especially in the calculation of availability of serial systems, not considering the propagation or impact of events of a specific machine in the rest of the machines, this is clear when is known that the availability of serial systems is the product from the availability of each of the machines, calculated previously independently of the behavior of the others.

2.2. Risk consideration

The evaluation of the average performance of manufacturing systems has been widely investigated in the manufacturing system engineering literature. However, there is industrial evidence that production variability due to random disturbances cause the observed production rate to be different from its average value [14]. Analyzing the variability is possible to find optimal designs, maintenance policies, and so on that meet desired service levels. None of the most used methodologies has been deeply developed, or have been expanded with the focus on the study of the risk and variability of the indicators of interest, production and availability. The mentioned methodologies, PNs, Markov-Chains and RBD are completely focused on the study of the mean and expected values of the key performance indicators. There are not total complete research in risk using those methodologies, there are just simplified mechanisms and as results of numerical examples, we can highlight the work of Xin-Feng et al [12], were discrete production is studied, and Manitz et. al [13] and Li S. et al[18] in the continuous case. One the few works that studies the variability is Li C. et al [15], but using discrete simulation. Colledani et al. [14] also presents a methodology of decomposition to obtain the total variability of a line with buffers, but is only applicable for series systems.

2.3. Life Cicle consideration

Until recent time the calculus of reliability was only considering the models described by Perfect Renewal Process, it means, modeling that consider the situation “as good as new” once the maintenance activity over the asset has been done. During the last years other modelling considerations have been adopted with more presence in reliability research, like Non-Homogeneus Poison Process (NHHP) [20] and the Generalized Renewal Process (GRP) [16], in example [20, 21, 22, 23]. But these research works are not focused in evaluate the trend of availability and production of an equipment, this means that are far away to evaluate the trend of availability and production of a system.

None of these methodologies currently allow to implement a long-term analysis unless some special module is developed, or discrete calculations are carried out under different system conditions. While considering the above citations is possible to calculate the expected availability over time to an equipment, it is difficult to do for a complete system. Finally it is concluded that there is no methodology that appropriately consider the equipment degradation over time in a systemic environment to estimate the availability and production.

3. STATE OF THE ART

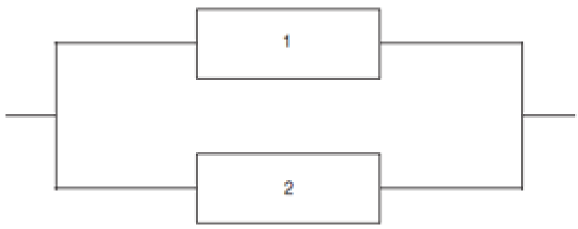
Currently there are some techniques and methodologies to estimate availability and production in continuous production systems. The most used ones are Markov Chains, PetriNets, Reliability Block Diagram and Simulation Softwares. We proceed to describe these methodologies below:

3.1. Markov Chains

There are discrete time and continuous Markov chains, for purposes of analysis of continuous production systems we consider homogeneous continuous time Markov chain (CTMC) [11]. CTMC is a state space model, in which each state represents various conditions of the system. In homogeneous CTMCs, transitions from one state to another occur after a time that is exponentially distributed. The arcs representing a transition from one state to another are labeled by the constant rate corresponding to the exponentially distributed time of the transition. If a state in the CTMC has no transitions leaving it, then that state is called an absorbing state, and a CTMC with one or more such states is said to be an absorbing CTMC. For the multiprocessor example, we now illustrate how a Markov chain can be developed in a simple way for 2 machines.

Below is developed an example of Markov chains analysis implementation, a productive subsystem comprising two identical devices in parallel is developed.

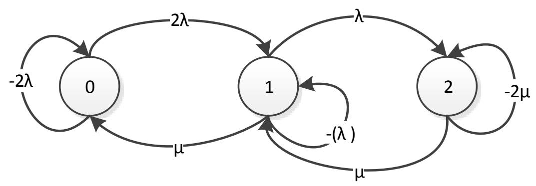
Figure 1. RBD diagram for Markov chain analysis example.



There are 3 cases only, given that the equipment are identical. It is defined $P_i(t)$ as the probability that the systems is in the state i at the time t. In addition has the boundary condition that at the beginning both units are operating normally, this implies that:

$$\begin{aligned} P_0(0) &= 1 \\ P_1(0) &= 0 \\ P_2(0) &= 0 \end{aligned} \qquad (1)$$

Figure 1. Markov chain diagram for 2 devices in parallel.



It is possible to omit the differential analysis and move immediately to the steady state analysis, when probabilities $P_i(t)$ are invariant in time. Now the equations that remain are, including the boundary conditions:

$$\begin{aligned} -2\lambda P_0 + \mu P_1 &= 0 \\ 2\lambda P_0 - (\lambda + \mu)P_1 + \mu P_2 &= 0 \\ \lambda P_1 - \mu P_2 &= 0 \\ P_0(0) = 1;\ P_1(0) = 0;\ P_2(0) &= 0 \end{aligned} \quad (3)$$

Based on the presented system solving for P_0+P_1 corresponds to the availability of the system. In this case is:

$$A = P_0 + P_1 = \frac{\mu^2 + 2\lambda\mu}{2\lambda^2 + \mu^2 + 2\lambda\mu} \quad (4)$$

Based on the example the difficulty level that may exist when it comes to the analysis of a more complete system is seen clearly. Therefore, it becomes practically an infeasible methodology to more complex systems. Also, consider constant rates of occurrence of events transforms Markov chains in an unwise choice [6, 10].

3.2. PetriNets

A Petri net [9] is a more concise and intuitive way of representing a situation to be modeled. It is also useful to automate the generation of large state spaces. A Petri net consists of places, transitions, arcs and tokens. Tokens move from one place to another along arcs through transitions. The number of tokens in the places represents the marking of a Petrinet. If the transition ring times are stochastically timed, the Petri net is called a Stochastic Petri Net (SPN). If the transition ring times are exponentially distributed, the underlying reachability graph, representing transitions from one marking to another gives the underlying homogeneous CTMC for the situation being modeled.

For the paralell system, we are interested in finding the probability that an incoming task is turned away because all *n* processors are tied up by other tasks being processed. The parameters associated with this pure performance model are, arrival rate of tasks, service rate of tasks, the number of task response times. The performance model assumes that the arriving task forms a Poisson process of rate and the service requirements of tasks are independent, identically distributed with the exponential distribution. A deadline d is associated with each task. Let us also take the number of buyers available for storing incoming tasks as b. Place process contains the number of processors available. Initially there are n tokens here representing n processors. In general PetriNets are a graphical method to analyze the process, but is not a mathematical method or an algorithm to solve the problems of thorughtput calculation.

3.3. Reliability Block Diagrams

In Reliability Block Diagrams each component of the system is represented as a block [1, 9]. The blocks are then connected in series, parallel, stand-by, and so on, based on the operational dependency between the components. If for the system to be up all the components need to be operational, blocks in a RBD are connected in series. On the other hand, if the system can survive with at least one component then blocks are connected in parallel, this logics is extended to other logics like partial redundancy, stand-by and even buffer systems. An RBD can be used to model availability if the repair times (and failure times) are all independent. Fig 3 shows availability model with *n* machines where all machines are required for the system to be up. From this we conclude that the RBD represents a simple series system. Given a failure rate λ_i and repair rate μ_i, the availability of each machine is given by:

$$A_i = \left(\frac{\mu_i}{\lambda_i + \mu_i} \right) \quad (5)$$

So the availability of the system is:

$$A_s = \prod_{i=1}^{4} A_i \qquad (6)$$

Figure 3. Series system.

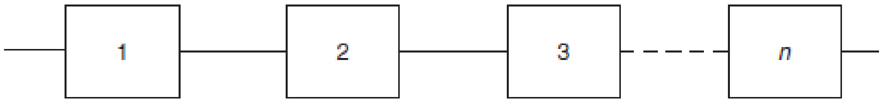
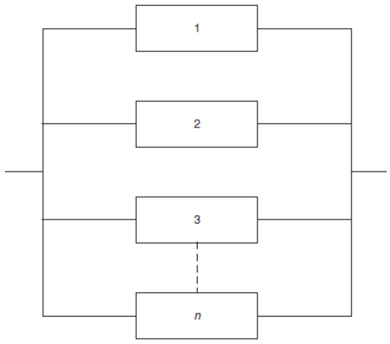


Figure 3 shows a multiprocessor availability model with *n* processors where at least one processor is required for the system to be up. From this we conclude that the RBD represents a simple parallel system. Given a failure rate λ and repair rate μ, the availability of the system is:

$$A_s = 1 - \prod_{i=1}^{n} \left(1 - \frac{\mu}{\lambda + \mu} \right) \qquad (7)$$

Figure 4. Multiprocessor system.

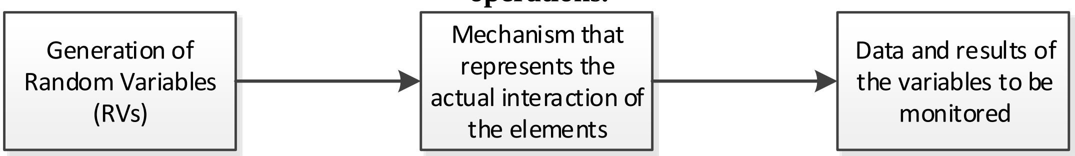


Reliability Block Diagram allows to mix the systems and the n, for example, the multiprocessor system will be now just a block, which could be a part of a series system. Thus the net is reduced to have a single availability for the entire system.
To involve production estimation trough RBDs is only necessary to do a multiplication between the availability of the system with the nominal production rate of the plant and the evaluated time.
The result of the calculus is then an expected total availability or production, which is useful when is needed to do a forecast of the total production in a very long-term analysis.

4. PROPOSAL ABOUT THE PROBLEM

By itself, the simulation corresponds to the process of designing a model in which different input variables, with the purpose to study the output variables generated by the model and evaluate and make decisions. In terms applied to the current context, the simulation corresponds to a non-deterministic method based on the statistics used to analyze complex expressions that could be extensive in terms of time and cost to assess accurately. Simulation allows to study how the items interact, analyze their compatibility, criticity and effects that might occur for which no consideration was.

Figure 5. Basic diagram for representation of the operation of a simulation in the area of operations.



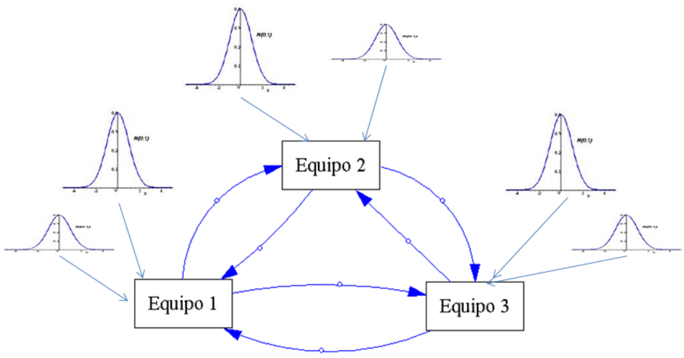
4.1. Reliability and Maintainability Modelling

Considering what reliability engineering theory raises, there are mainly two random processes involved: the maintainability function and the reliability function. How has been mentioned in the state of the art, the reliability, denoted as $R(t)$, corresponds to the probability that an item could work without failures during a time t, since in t=0 it has the condition as good as new or other state, given by the specific modelling method (PRP, NHPP, GRP), and the maintainability $m(t)$ corresponds to the probability to repair an item before a time t, since t=0 is when the repair task was initiated. Both expressions are defined from probability distribution functions, and represents the random variables Time To Fail (TTF) and Time To Repair (TTR), related to reliability and maintainability respectively.
To run a simulation of a system is necessary to take as input the information indicating how the equipment interacts, it means the logic-configuration in which are related. So the input variables are:

1. Behavior of times to fail of each equipment. (Random Variable)
2. Behavior of times to repair of each equipment. (Random Variable)
3. How an item failure or an item repair impacts in the others items. In other words, how the items are related in terms of logic-configuration under which they are interacting. (Deterministic information).

We have to remark that the reliability functions *R(t)* and maintainability *m(t)*, are defined by a fitting method to the historic data, if there is. The most suitable method is maximizing the likelihood to each distribution. Once the parameters to each distribution that maximizes the likelihood are determined a rank is generated with the Kolmogorov-Smirnov test. The process of choosing the best distribution once the parameteres are estimated it may even visually, but there are also tests like the Anderson-Darling and Shapiro-Wilk, when is more related to normal distributions.

Figure 6. Representation of interactions of an operating system of 3 the items.



For this methodology have been considered various types of events occurring in the production environment, and certain types of interactions that the equipment have, which corresponds to existing logic-configurations between them.
Within the behavior of the equipment, the items have the following attributes:

- Times between random failures, associated to a certain pdf.
- Random repair times, associated to a certain pdf.
- Time between operational stoppages, associated to a certain pdf.
- Duration of operational stoppages, associated to a certain pdf.
- Some certain productive capacity of the item.

By the side of the subsystems we have considered the following types of logic configurations:

- Series configuration
- Parallel configuration.
- Partial redundancy configuration, aka k-out-of-n

- Stand-by configuration
- Load-Share configuration
- Stockpile, aka buffer

Probability distributions have been considered for use in the model correspond to: Exponential, Log-Normal, Normal, Dirac delta, Uniform and Weibull of 2 and 3 parameters. Also NHPP and GRP modelling is allowed. To the random number generation the inverse distribution method is used.

4.2. Modelling Elements

4.2.1 Equipment Tree

The Equipment Tree corresponds to the hierarchy and detailed component-level constitution of all the equipment that form the system, given through a hierarchical table at which level each subsystem and each component/equipment belongs.
The Equipment Tree list all components in the system. Each component has a unique code, a probability distribution of failure time and a unique probability distribution of repair time. The *components* then correspond to the elements that can fail and could be repaired.

4.2.2 Flowcharts

The methodology aims to, through a practical and intuitive insight, accelerate the process of modeling the most, so that practitioners should only have to identify the operating logic based on the material flow in the system, for this purpose is needed to be developed, after obtained the Equipment Tree, the flow diagram of system to model. The diagrams are composed of two types of elements, *nodes* and *arcs*, the first can be of systems, equipment or changes in flow. The *arcs* are easier however, only indicate where to which the material is transported.

4.2.2.1 Nodes

Within the several types of layout nodes and these can be classified into three:

4.2.2.1.1 Processable Nodes

In this type are the nodes representing systems, subsystems and equipment. Systems and subsystems are used as nodes that exclusive group other diagrams, in contrast the equipment nodes are the last elements in the Equipment-Tree.

a) Equipment: are those elements that fail and are repaired, so are the elements on which the data is loaded into:

- Probability distribution of time between failures
- Probability distribution of time for repairs
- Preventive Maintenance Policy
- Maximum production capacity
- Other optionals

b) Subsystems: Are groups of equipment or other subsystems. Don’t have specific probability distribution functions for failures or repair. In contrast, the behavior of the subsystem is given by the interaction of the failures and repairs of the elements that compose the subsystem.

4.1.2.1.2 Node of Logic Configuration

These nodes correspond to the method by which the flow could be divided, cumulated, etc. being able to represent different logical configurations. Through these nodes is possible to establish different types of redundancy subjected to different areas of the system.
Each node has its own logical configuration algorithm, based on state changes in each item (equipment) or subsystem that is defined in your environment flux levels being processed and communicated immediately to the other elements the impact of the last event in itself.

4.2.2.1.3 In and Out Nodes

They are the last type of node, but no less important, since its purpose is to represent where the mass flow is delivered and where it comes from, so these nodes open and close the flow. These are nodes in ideal conditions, since they do not experience failures.
The In and Out nodes are within any system or subsystem diagram that can be decomposed.

4.2.2.1 Arcs

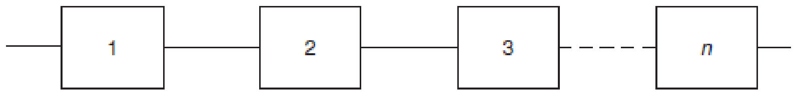
Graphically correspond to arrows, from where to where there is a transfer of mass flow. Indicates the start and destination. Not correspond to elements that may experience failure or a stoppage, only serve to connect the existing mass flow between elements.

4.3 Types nodes of configuration logic

4.3.1 Series

This type of configuration is when the dependency between certain equipment operations requires that to the existence of system availability requires that all equipment is in a state of availability. Any unavailability, independent of which item (equipment) will generate an overall unavailability of the system or subsystem.
The layout of a subsystem in this configuration corresponds to:

Figure 7. Layout for equipment under series configuration.



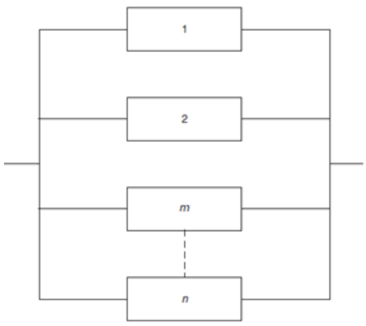
4.3.2 Load-Share

This configuration represents the case where a group of machines (equipment/subsystem) share the mass (flow) passing through a production line. This distribution may or may not be equal to each machine. This depends exclusively on the characteristics of capacity available to each machine.
The benefit of this type of functional logic is that it gives the possibility to operate at lower production rates than the maximum of the line, ie the failure of a machine only generates a proportional or partial loss in production, but not the unavailability entire system. Each machine (equipment/subsystem) has a level of impact (in percentage) on the production line to which it corresponds. Can also be configured the availability in excess of capacity, in this way is the case when the sum of the impacts that the machines have on the line is above 100%.

4.3.3 Partial redundancy and Parallel (k-out-of-n / 1-out-of-n)

The parallel configuration or full redundancy occurs when there is a group or sub-system consisting of N equipment, where the unavailability of one or more components, up to N-1, generates no impact on the overall operation, always requiring either conforming equipment cannot meet the performance of the entire subsystem. Thus, the unavailability of this system is defined if and only if all equipment are unavailable.

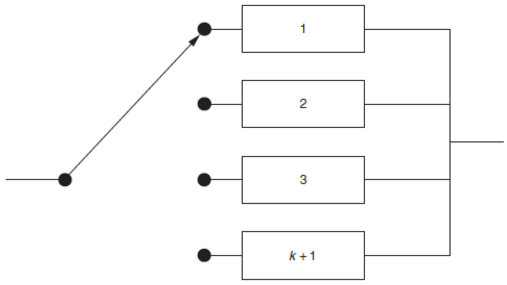
Figure 8. Diagram for partial redundancy configuration.



4.3.4 Stand-by

This type of functional dependency can represent the operation of a group of machines where one works at the same time, and if it fails instantly goes into operation other machine which is as a backup. There may be more than one machine as a backup, as outlined in the following diagram:

Figure 9. Diagram for stand-by configuration.



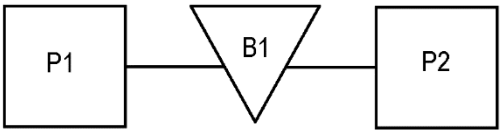
4.3.5 Conditioned Flow

Is also possible to use another configuration, the conditional flow, where a priority is assigned to different machines to be able to deliver the material. This router node sets the order of preference you want to give to a list of equipment or subsystems to processes the material flow.

4.3.6 Stockpile / Buffer

A stockpile is characterized by allowing the accumulation of material in the form of batch or continuous within itself, getting these as a result of a previous process and arranging to be taken as input in a subsequent process. This method is limited by the maximum capacity of the stockpile and dynamic flows that are in the subsystem upstream and downstream.

Figure 10. Representation of accumulator system between 2 processes.



Thus, depending on: inflows, outflows, capacity, and level of accumulation that possess at a given time, the stockpile can be be in different states of accumulation. The fundamental operation of the stockpile is to provide material to the downstream subsystem while the upstream system is in a state of unavailability, thus achieving greater advantage of the available hours of the downstream system, which logically should be part of the bottleneck production system in terms of capacity, since otherwise the need to implement a storage system loses relevance and prominence, since in this way the system upstream would require constant production in their hours of operational readiness given to the system downstream may prosecute higher the output rate of the system upstream. The variables that define the stockpile are: Maximum Capacity [$Q_{máx}$], Initial Capacity [Q_{in}], Downstream Rate [F_{ad}], Upstream Rate [Fu_a].

4.4 Representation and calculation on the system.

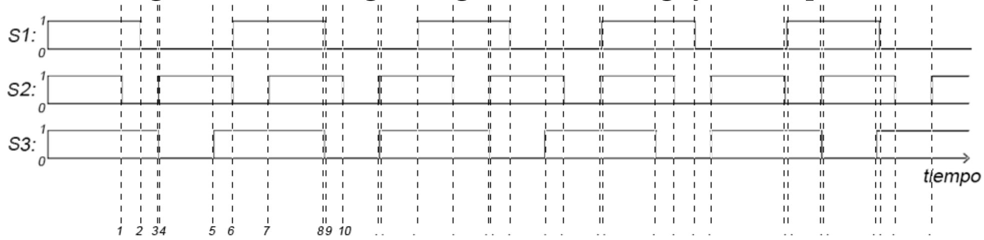
The method of processing subsystems aims to identify all the interactions generated by analyzing what happens as a result of each event in the life line of them and finally get a final life-line for each of the subsystems, and this life-line could be used again in case if another subsystem contains the previous one.

Processing subsystems depends purely on dependencies configuration that the elements have, for this purpose various algorithms dedicated to each configuration were prepared. We have to note that each subsystem is linked to a dependencies configuration, and in the row for that subsystem with unique parameters related to the configuration that has been associated , for example in the case of partial redundancy variables are given: how many are required for the subsystem is available, in Load-Share the production rate, etc.

4.4.1 Calculation focused to the state change.

The aim is that the processing methodology and calculation of indicators would be the fastest way possible. Most simulators for continuous processes perform the calculation of indicators and identification of states through monitoring a certain interval of time, this process is extremely inefficient when compared with oriented view state changes. It means that monitoring is performed only when something in the system changes state. For this study, all occurrences of fault level machine are independent therefore a simulation oriented state change.

Figure 11. Outlining the algorithm scanning system in parallel.



This algorithm works asking the state each time an item will change the state, then the impact of this change of state is analyzed in each of the system elements. Now the way that the event impacts on other elements depends on the established functional logic. Which have been defined in 4.3

These logical relationships are embodied in a multidimensional array *R* where recursively runs the impact on each element.

$$R = \begin{bmatrix} R_{1,1} & \cdots & R_{1,j} & \cdots & R_{1,n} \\ \vdots & \ddots & & & \vdots \\ R_{i,1} & & R_{i,j} & & R_{i,n} \\ \vdots & & & \ddots & \vdots \\ R_{n,1} & \cdots & R_{n,j} & \cdots & R_{n,n} \end{bmatrix} \quad (8)$$

n: total number of processables nodes

i: id of the element that suffered a change of state.

j: id of the element on which there is an impact on its state because of the change on element *i*.

At the same time each vector is defined as the following:

$R_{i,j} = \{v_{i,j}(\mathrm{f_i}), \mathrm{d_{i,k}}, \mathrm{t_{i,k}}\}$ (9)

Where:

$v_{i,j}(f_i)$: instant mass flow on element *j* given that the mass flow on *i* now is f_i

$d_{i,k}$: type of stoppage on the *k*-th stoppage of element *i*.

$t_{i,k}$: time of stoppage of the *k*-th stoppage of element *i*.

$p(d_{i,j,k})$: Binary variable which indicates if the change of state of *i* generated a stoppage in the element *j* (ie. Change in mass flow)

Once analyzed the impact and collected the instantly generated variables other algorithm is run, that studies which will be the next event to occur. That could be: Failure of element, End of repairing of an element, Change of mass flow in an element, Empty stockpile, Maximum level of a stockpile has been reached. The algorithm processes and determines what the next event to occur in the system, through the use of a procedure that calculates the remaining life time of each element. The time at which this event will occur is determined and is used again in the matrix *R*. For each subsystem a life-line is generated, which is saved in a matrix:

$S_i = \left\{P_1, P_2, P_3, ..., P_{l_i}\right\}$ (10) ,where:

P: Vector which has the information of mass flow at a given time.

l_i: Total number of change of stated of the element *i*

$P_x = \{m, t, g\}$ (11), where:

m: mass flow
t: time
g: new state of the element (working, ie. suffering degradation, or in stoppage)

4.4.2 Implementation of the calculation

The presented method can be used to calculate:

- Total number failures, per element, subsystem, and system
- Total number of system detention
- Number of failures dumped by a stockpile system
- Total mass flow for an established simulated period of time

Given the method is possible to use the overall algorithm to analyze the risk based a large number of simulations in short times. In this way, the methodology is useful when is about risk measuring. For example, with the method is possible to calculate with accuracy the probability of meeting a required amount of production for an established period.

5. CONCLUSION

A simple, reliable and fast method to simulate systems focused in the occurrence of failures has been presented. This methodology presents the advantage that is not necessary to define stochastic Petri-Nets states or States of the whole system like in Markov Chain modelling, is only necessary to establish the dependencies, from whom we have considered the most common seem in the industry. The work of establish the state-based models by hand is a task that took long time, and is highly vulnerable to mistakes in the process of establishing these states.

To use the proposed modelling no previous knowledge about RBD, Markov Chain or Petri-Net is required. The requirement to the user is just to identify the functional relationship between the machines/elements. The methodology, as a state change based simulation method, allows to run a large number of simulations, and as a difference with other state-change based simulation environments, the present methodology only focuses only limited states, so the speed is increased dramatically. This allows to study a larger number of cases and a large number of combinations and possibilities.

Lastly, this methodology represent disadvantages in terms of events that are not related to faults and its implications. The method also oriented calculations only consider long term or transient states. The calculations do not consider production states of calibration, but indicated by average production machine. So, further work could be related to fix those approaches.

Acknowledgements

Thanks for RelPro® for allowing the funding for this research.

References

[1] Introducing buffer inventories in the RBD analysis of process production systems, Marco Macchi, Fredy Kristjanpoller, Marco Garetti, Adolfo Arata, Luca Fumagalli Reliability Engineering & System Safety, Volume 104, August 2012, Pages 84-95

[2] Reliability and availability analysis of low power portable direct methanol fuel cells N.S. Sisworahardjo, M.S. Alam, G. Aydinli Journal of Power Sources, Volume 177, Issue 2, 1 March 2008, Pages 412-418

[3] Chapter 4 - Reliability, Availability, and Maintainability Analysis, Eduardo Calixto, Gas and Oil Reliability Engineering, 2013, Pages 169-347

[4] Make your Petri nets understandable: Reliability block diagrams driven Petri nets, Jean-Pierre Signoret, Yves Dutuit, Pierre-Joseph Cacheux, Cyrille Folleau, Stéphane Collas, Philippe Thomas, Reliability Engineering & System Safety, Volume 113, May 2013, Pages 61-75
[5] Probabilistic analysis of a series–parallel repairable system with three units and vacation, Linmin Hu, Dequan Yue, Jiandong Li Applied Mathematical Modelling, Volume 34, Issue 10, October 2010, Pages 2711-2721
[6] A novel optimal preventive maintenance policy for a cold standby system based on semi-Markov theory, Chongquan Zhong, Haibo Jin, European Journal of Operational Research, Volume 232, Issue 2, 16 January 2014, Pages 405-411
[7] Availability/reliability engineering analysis of three, four, and five-stage unreliable series transfer lines, Thomas F. Hassett, Duane L. Dietrich, Journal of Manufacturing Systems, Volume 14, Issue 6, 1995, Pages 427-438
[8] A simulation study on maintainer resource utilization of a fast jet aircraft maintenance line under availability contract, Partha Priya Datta, Anupam Srivastava, Rajkumar Roy, Computers in Industry, Volume 64, Issue 5, June 2013, Pages 543-555
[9] Simulation inferences for an availability system with general repair distribution and imperfect fault coverage, Jau-Chuan Ke, Zheng-Long Su, Kuo-Hsiung Wang, Ying-Lin Hsu ,Simulation Modelling Practice and Theory, Volume 18, Issue 3, March 2010, Pages 338-347
[10] Fuqua, Norman B., "Markov Analysis," The Journal of the RAC, Third Quarter 2003.
[11] Bause, F., Kritzinger, P.: Stochastic Petri Nets – An Introduction to the Theory (2nd edition), 2002,
[12] Xin-Feng He, Su Wu, Quan-Lin Li, Production variability of production lines, International Journal of Production Economics, Volume 107, Issue 1, May 2007, Pages 78-87
[13] M. Manitz, Tempelmeier H., The variance of inter-departure times of the output of an assembly line with finite buffers, converging flow of material, and general service times. OR Spectrum, January 2012, Volume 34, Issue 1, pp 273-291
[14] Colledani, M., Matta, A., and Tolio, T. (2010). Analysis of the production variability in multistage manufacturing systems. CIRP Annals - Manufacturing Technology, 59(1), 449-452.
[15] Congbo Li, Ying Tang, Chengchuan Li, Lingling Li: A Modeling Approach to Analyze Variability of Remanufacturing Process Routing. IEEE T. Automation Science and Engineering 10(1): 86-98 (2013)
[16] V. Krivtsov, O. Yevkin, Estimation of G-renewal process parameters as an ill-posed inverse problem, Reliability Engineering & System Safety, Volume 115, July 2013, Pages 10-18
[17] Safety and Reliability Society, SARS, Applied R&M Manual for Defence Systems, Part D - Supporting Theory, GR-77 Issue 2011, 2011.
[18] Jingshan Li and Semyon M. Meerkov. Production variability in manufacturing systems: Bernoulli reliability case. Annals of Operations Research 93 (2000) 299–324.
[19] Monika Tanwar, Rajiv N. Rai, Nomesh Bolia, Imperfect repair modeling using Kijima type generalized renewal process, Reliability Engineering & System Safety, Volume 124, April 2014, Pages 24-31
[20] Hoang Pham, Xuemei Zhang, NHPP software reliability and cost models with testing coverage, European Journal of Operational Research, Volume 145, Issue 2, 1 March 2003, Pages 443-454
[21] S. Zarezadeh, M. Asadi, N. Balakrishnan, Dynamic network reliability modeling under nonhomogeneous Poisson processes, European Journal of Operational Research, Volume 232, Issue 3, 1 February 2014, Pages 561-571
[22] Yeu-Shiang Huang, Chi-Chang Chang, A study of defuzzification with experts' knowledge for deteriorating repairable systems, European Journal of Operational Research, Volume 157, Issue 3, 16 September 2004, Pages 658-670
[23] Marco Scarsini, Moshe Shaked, On the value of an item subject to general repair or maintenance, European Journal of Operational Research, Volume 122, Issue 3, 1 May 2000, Pages 625-637, ISSN 0377-2217
[24] Tim Bedford, Isha Dewan, Isaac Meilijson, Athena Zitrou, The signal model: A model for competing risks of opportunistic maintenance, European Journal of Operational Research, Volume 214, Issue 3, 1 November 2011, Pages 665-673