Proceedings of the 2021 Winter Simulation Conference
S. Kim, B. Feng, K. Smith, S. Masoud, Z. Zheng, C. Szabo and M. Loper, eds.

AVAILABILITY AND STOCK RUPTURE ESTIMATION BY USING CONTINUOUS AND DISCRETE SIMULATION MODELS

Edson Ursini
Henry Santos

University of Campinas
Technology School
R. Paschoal Marmo, 1888, Jd. Nova Itália
Limeira, SP, 13484-332, BRAZIL

Marcelo Okano

CEETEPS and
University of Campinas
R. dos Bandeirantes, 169, Bom Retiro
São Paulo, SP, 01124-010, BRAZIL

ABSTRACT

We propose a method to solve an Availability and Stock Rupture problem considering static and dynamic conditions. Information about failure and repair rates, dependence or independence of the blocks are considered, as well as knowledge about spare stock and Maintainability. The analysis is done, initially, using exponential processes, considering the transient solution by Matlab-Simulink. This analysis allows to evaluate the steady state solution and the main aspects that must be addressed through Discrete Event Simulation. We consider a situation in which it is possible to use two blocks, one in operation and one in the spare stock, with a single repairman. SLA (Service Level Agreement) contracts are common, which the parties involved sign and failure to comply with these contracts may result in a fine. An approach to improve the dimensioning of such systems in terms of their availability and Stock Rupture as close as possible to the real ones.

1 INTRODUCTION

The design of the book Dependability (Avizienis et al. 2001): Basic Concepts and Terminology in 1992, after several studies have accelerated a set of concepts and terminology for Dependability. A set of threats, attributes and means by which reliability is achieved to generate the ability to deliver a service that can be justifiably trusted can be defined as Dependability. The most important aspects related to Dependability are (ITU-T 2012):

- **Availability**: operability of the system to deliver applications to users, taking into account system stops, programmed or not (failures), interrupting their functionality totally or partially;
- **Reliability**: guarantee of delivery of application data packets, in an ICT context, in an integrated manner, between systems and/or users;
- **Maintainability**: The ability of an item under stated conditions of use, to be retained in, or restored to, a state in which it can perform a required function.

This article proposes to extend the analysis for dimensioning of spare parts stock. At work (Ursini et al. 2020) an analysis was made in relation to a System that has spare parts functioning as hot-standby (or warm standby), that is, the devices work in redundancy, but when one failure does not affect the functioning of the System because the other , which was already active, goes into operation. The purpose of this article is the analysis of a System in which spare parts are stocked and, therefore, when there is a failure the device in stock has to assume the function of the equipment that failed (or cool standby). It can be said that this device is in standby mode, but it requires some action in order to become the active device. There are several situations in which it is necessary to evaluate the Availability of a System as well as its Out of Stock point. Often the correct value of Availability (Unavailability) is associated with heavy fines for violating

978-1-6654-3311-2/21/$31.00 ©2021 IEEE

the SLA (Service Level Agreement), which is the practical implementation of QoS (Quality of Service), especially for critical systems that have to be out of service the minimum possible time. Availability is usually described as a number of nines which means that, depending on the case, the system can be out of service for very few minutes (or seconds) per year, (Chumash 2019). According the work by (Uptime Institute 2021), with six nines, or 99.9999%, an average customer in a six nines portfolio would experience about 2.6 seconds of downtime per month, or less than 32 seconds per year. In this article we propose to carry out the analyzes considering the three variants for the state of the system, F = in Failure, O = in Operation, and S = in Stock. For this, the systems will be evaluated in a transient and stationary regime. In addition, it is important to make use of other distributions other than the exponential, which is what occurs in real cases. For this, the use of both Continuous Simulation and Discrete Event Simulation (DES) is essential. The work proposal includes both Continuous Simulation (Matlab-Simulink) and DES (Arena). Continuous Simulation, is useful due to the fact that it uses exponential distributions and allows an explicit solution of the system of differential equations. These equations allow knowing the transient state and the steady state (this one with closed equations). DES also shows the evolution of the system, from transient to steady state, but its equations are implicit (there is no formulation of the system of equations). In most cases it is not possible to use Continuous Simulation because real-life distributions do not have properties such as the lack of memory of the exponential distribution. When it is not possible to use Continuous Simulation, Simulation by DES can be used. Thus, for calculation purposes, with closed equations, at any instant of time, the Continuous Simulation, whenever the system allows it, is preferable because it is easier and faster to calculate and has a deterministic and defined value and not a confidence interval like DES. In addition, DES generally requires a high processing time. In Section 2 some related works are presented, in Section 3 the implementation in Matlab-Simulink is described considering the exponential distributions, and there is also an extension of this analysis considering some peculiarities of the system. In Section 4 the approach is through Simulation by DES, mainly to contemplate other probability distributions not covered in the Markovian model, in Section 5 an approach of a case study with real data is made, in Section 6, there is a discussion of the proposed approach, and finally, in Section 7 , conclusions of the work are listed.

2 RELATED WORKS

In (Lenz and Rhodin 2011), the authors show the main types of series-parallel configurations of the different types of RBDs (Reliability Block Diagram). The work (Blokus and Dziula 2020) highlights that excessive development of the system with more back-up subsystems is not an appropriate direction, as it does not result in a respective increase of reliability parameters. It seems more useful to concentrate on improving the reliability of particular subsystems, rather than adding more back-up subsystems with lower reliability. In (Souza-Franco et al. 2019), the authors analyze dynamic positioning systems using Monte Carlo simulation to address uncertainties in the calculation of reliability. The article (Kanso et al. 2014) studies the management to provide the best redundant configuration that allows an increase in Availability, including the reboot of the systems when necessary, using Petri nets. The authors of (Barabadi et al. 2011) recognize that knowledge of the mean time to repair (MTTR) is not enough, but discuss other issues related to the maintainability of the systems. For that, they propose the study of time dependent and independent variations for the variable linked to the repair time. In (Lenz and Rhodin 2011) the authors work with reliability calculations of complex systems using the Mathematica software addressing series-parallel and mixed systems. In (Barabadi 2012) the author recognizes the importance of considering other factors so that the availability assessment is done in the best possible way. In this case, they address the Operating Environment. Table 1 shows the comparison of our article with some related works.

Table 1: Comparison with related works.

Related work	Continuous + Discrete Simulation
(Reliawiki 2021) model works with RBDs blocks.	We consider several different parameters and distributions.
(Blokus and Dziula 2020), focuses on putting more systems for back-up.	We propose a detailed evaluation of each system.
(Souza-Franco et al. 2019), apply Monte Carlo simulation.	We expand the analysis by using Continuous and DES (it implicitly uses the concepts of Monte Carlo's method).
(Kanso et al. 2014), uses Petri nets.	We can get all the results of the Petri nets and still observe the transient states reasonably simply.
(Barabadi et al. 2011), recognizes that just knowing MTTR and MTBF may not be enough, but addressing the general concepts of maintainability.	We also recognize that it is necessary to broaden the scope of the approach and consider the concept of Dependability to be fundamental.
(Lenz and Rhodin 2011), performs calculations using Mathematica software.	We use scientific (Matlab) and DES (Arena) softwares.
(Barabadi 2012), considers Operating Environment.	We also consider the importance of the Operating Environment (Dependability).

Figure 1 shows the state diagram of a situation where there is initially a block in Operation, a block in Stock and none in failure (state F, O, S = 0,1,1). The repair rate is ρ and the failure rate is λ.

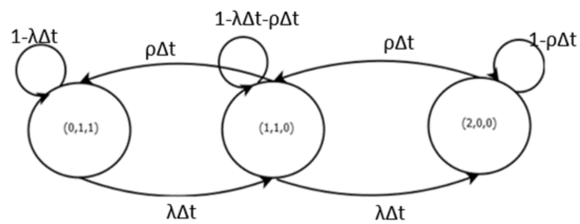

Figure 1: State diagram of the proposed system.

Availability means having at least one unit in Operation. As only one can be effectively in Operation, the other will be in Stock (S) or under Repair. Therefore, if one unit is in Failure (under Repair), there is a Stock Rupture (but not yet Unavailability). System will be Unavailable when both units are in Fault (F).

Our approach takes into account complex systems of high cost (as in a Data Center) in which detailed analysis of the Failure, Repair and Stock, including transient states, must be carried out. The state called "Stock" can mean a device in "cool standby" or, in fact, a device in the Spare Stock.

3 IMPLEMENTATION IN MATLAB-SIMULINK

Considering, as usual, that the transitions occur according to the exponential distributions (Markovian system), we can model the differential equations that compose the system.

3.1 Implementation of the basic model

Initially, the most important transitions as described in Fig. 1 will be considered. Thus, considering the state transitions of the system in Fig. 1, disregarding the higher order terms as usual, $(\Delta t)^2$, $(\Delta t)^3$, .. ., arranging the terms and dividing by Δt, we have the following equations:

$$\lim_{\Delta t \to 0} \frac{P_{011}(t+\Delta t)-P_{011}(t)}{\Delta t} = \dot{P}_{011}(t) = -\lambda P_{011}(t) + \rho P_{110}(t) \quad (1)$$

$$\lim_{\Delta t \to 0} \frac{P_{110}(t+\Delta t) - P_{110}(t)}{\Delta t} = P_{110}^{\cdot}(t) = -(\lambda + \rho)P_{110}(t) + \lambda P_{011}(t) + \rho P_{200}(t) \quad (2)$$

$$\lim_{\Delta t \to 0} \frac{P_{200}(t+\Delta t) - P_{200}(t)}{\Delta t} = P_{200}^{\cdot}(t) = -\rho P_{200}(t) + \lambda P_{110}(t) \quad (3)$$

This implementation was made in Matlab-Simulink with three integrators.

In the steady state, doing P_011 + P_110 + P_200 = 1, we get:

$$P_{011} = \frac{\rho^2}{\lambda^2+\lambda\rho+\rho^2}, P_{110} = \frac{\lambda\rho}{\lambda^2+\lambda\rho+\rho^2}, P_{200} = \frac{\lambda^2}{\lambda^2+\lambda\rho+\rho^2} \quad (4)$$

For the purpose of better evaluate and know the system, we will initially assume that $\lambda = 0.2$ failures/h and $\rho = 0.3$ repairs/h. Thus, we obtain the values for the steady state (P1 = Stock Rupture, SR and P2 = Unavailability, U), according to Table 2.

Table 2: Steady state values.

P0	**P1 (SR)**	**P2 (U)**
0.4737	0.3158	0.2105

Figure 2 shows the evolution of the states, starting from the initial state P0 = P011 = 1.

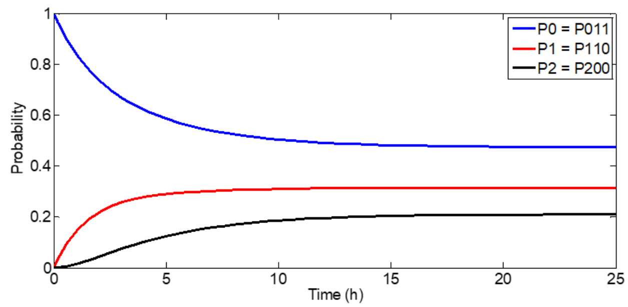

Figure 2: Transient solution for state values.

We can notice that the steady state values are the same as described in Table 2. The idea of Continuous Simulation is to foresee possible bottlenecks and/or points that may be critical for the evaluation of the system. In this way, we will move on to a more detailed analysis like in Fig. 3.

3.2 Extending the analysis

Considering Fig. 3, be the following configuration in which the state is represented by F, O, S, that is: in Failure, in Operation or in Stock. Note that we can consider a device time in Stock before it is activated, that is, it could take a while ($1/\mu$) before going into Operation (we will have the state (1,0,1) before the state (1,1,0)). On the other hand, we can consider that a failure in the state (1,0,1) would have a rate lower than the rate of the states (1,0,1) or (1,1,0) because it would imply a failure in the device in Stock. In the same way, in the state (1,1,0) the repair rate would be lower than in the states (2,0,0) or (1,0,1) because we would be contemplating only the repair and not the activation of the blocks (repair for placement in Stock).

Thus, with these considerations we are emphasizing the importance of planning maintenance and predictive maintenance, especially when it comes to high-cost items, including the need for these types of maintenance in relation to the new industrial revolution, Industry 4.0, (Bukhsh et al. 2019; Hitechnectar 2021).

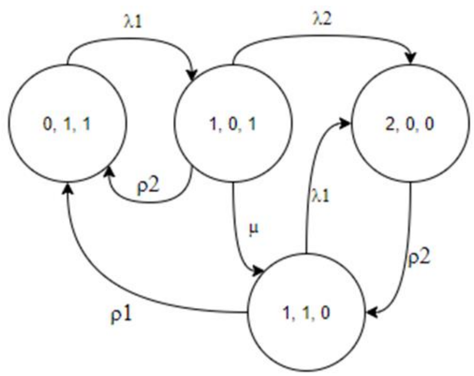

Figure 3: State diagram of the extended model.

This is the configuration of a practical case. There are only two units, one in Operation and one in Stock. We want to estimate the SR probability of out of stock and the U probability, the unavailability of the system.

In the extended model, we can then study three additional characteristics:

- The μ rate at which the plate (or device) in stock (E) is exchanged, making the plate in operation (O). The assumption is that the μ rate is quite high (that is, the time for it to occur is relatively small);
- The rate λ_2 would occur on the plate (or device) in the stock before it goes into operation. This rate must be less than the rate of the device in operation, λ_1;
- In state (1,1,0) the repair is $\rho_1 < \rho_2$ because only the repair time is being counted and not putting the device into operation.

Doing the equations, we will have:

$$\lim_{\Delta t \to 0} \frac{P_{011}(t+\Delta t)-P_{011}(t)}{\Delta t} = \dot{P}_{011}(t) = -\lambda_1 P_{011}(t) + \rho_2 P_{101}(t) + \rho_1 P_{110}(t) \tag{5}$$

$$\lim_{\Delta t \to 0} \frac{P_{101}(t+\Delta t)-P_{101}(t)}{\Delta t} = \dot{P}_{101}(t) = -(\lambda_2 + \mu + \rho_2) P_{101}(t) + \lambda_1 P_{011}(t) \tag{6}$$

$$\lim_{\Delta t \to 0} \frac{P_{110}(t+\Delta t)-P_{110}(t)}{\Delta t} = \dot{P}_{110}(t) = -(\lambda_1 + \rho_1) P_{110}(t) + \rho_2 P_{200}(t) + \mu P_{101}(t) \tag{7}$$

$$\lim_{\Delta t \to 0} \frac{P_{200}(t+\Delta t)-P_{200}(t)}{\Delta t} = \dot{P}_{200}(t) = -\rho_2 P_{200}(t) + \lambda_2 P_{101}(t) + \lambda_1 P_{110}(t) \tag{8}$$

The implementation in Matlab-Simulink is seen in Fig. 4 and the visualization of one of the equations, the one that provides P110 (the other three blocks are similar), as in Fig. 5:

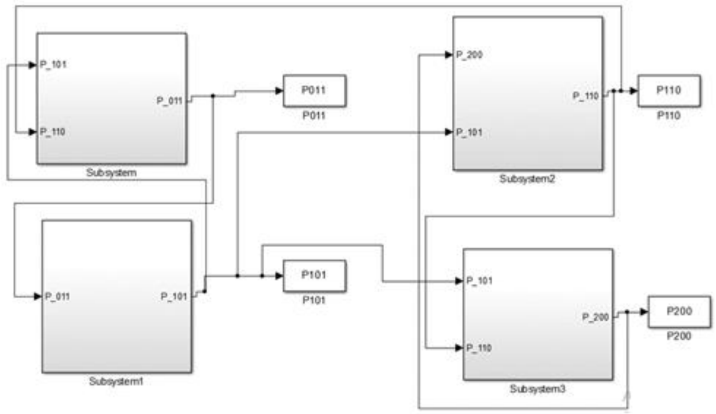

Figure 4: Implementation in Matlab-Simulink.

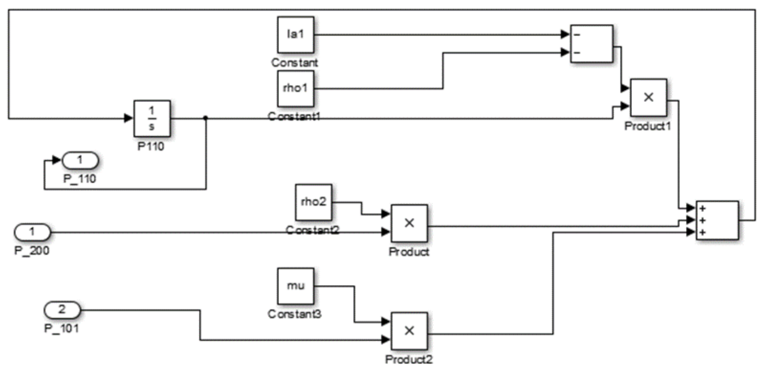

Figure 5: Implementation of equation (E6) in Matlab-Simulink.

In the steady state, we have $P_{011} + P_{101} + P_{110} + P_{200} = 1$. Thus,

$$P_{011} = \frac{\rho_1\rho_2^2+\lambda_2\rho_1\rho_2+\mu\rho_1\rho_2}{\lambda_1^2\lambda_2+\lambda_1^2\mu+\rho_1\rho_2^2+\lambda_1\lambda_2\rho_1+\lambda_1\lambda_2\rho_2+\lambda_1\mu\rho_2+\lambda_1\rho_1\rho_2+\lambda_2\rho_1\rho_2+\mu\rho_1\rho_2} \tag{9}$$

$$P_{101} = \frac{\lambda_1\rho_1\rho_2}{\lambda_1^2\lambda_2+\lambda_1^2\mu+\rho_1\rho_2^2+\lambda_1\lambda_2\rho_1+\lambda_1\lambda_2\rho_2+\lambda_1\mu\rho_2+\lambda_1\rho_1\rho_2+\lambda_2\rho_1\rho_2+\mu\rho_1\rho_2} \tag{10}$$

$$P_{110} = \frac{\lambda_1(\lambda_2\rho_2+\mu\rho_2)}{\lambda_1^2\lambda_2+\lambda_1^2\mu+\rho_1\rho_2^2+\lambda_1\lambda_2\rho_1+\lambda_1\lambda_2\rho_2+\lambda_1\mu\rho_2+\lambda_1\rho_1\rho_2+\lambda_2\rho_1\rho_2+\mu\rho_1\rho_2} \tag{11}$$

$$P_{200} = \frac{\lambda_1(\lambda_1\lambda_2+\lambda_1\mu+\lambda_2\rho_1)}{\lambda_1^2\lambda_2+\lambda_1^2\mu+\rho_1\rho_2^2+\lambda_1\lambda_2\rho_1+\lambda_1\lambda_2\rho_2+\lambda_1\mu\rho_2+\lambda_1\rho_1\rho_2+\lambda_2\rho_1\rho_2+\mu\rho_1\rho_2} \tag{12}$$

For the purpose of validating this model, we make, as before, $\rho_2 = \rho_1 = 0.3$, $\lambda_2 = \lambda_1 = 0.2$, and $\mu = 0$, remembering that time is the inverse of the rate and that if $\mu = 0$, therefore time will be infinite (no transition will occur). Table 3 shows that with $\mu = 0$ we relapse in the previous case, so the model can be considered validated. Note that $P_1 = P_{101} + P_{110}$.

Table 3: Values of stationary states in the extended model.

P0	P1		P2
P011	**P101**	**P110**	**P200**
0.4737	0.1895	0.1263	0.2105

The extended model can be used to evaluate some aspects in relation to the simplest model. The first analysis is in relation to the μ transition rate. In practice, the value of μ must be high since its time must be low (time = 1/rate). Making μ = 20 actuations /h and maintaining the value of the other parameters, we have (Fig. 6),where the value of P101 is displayed.

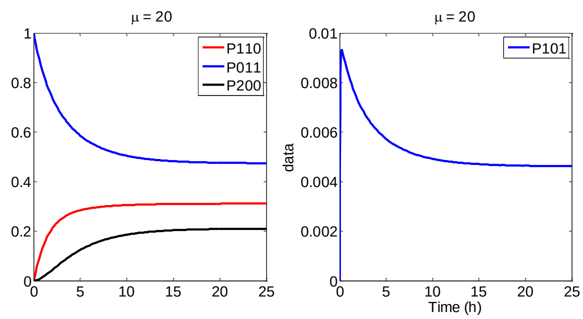

Figure 6: The transient values in the extended model to μ = 20, and highlighting the value of P101.

For μ = 20 we see that the value of P101 is less than 0.006 (Fig. 7). For μ = 2 the value of P101 is less than 0.04 and for μ = 1 the value of P101 is less than 0.06 (these last two values of μ are very high compared to the values of ρ_1 or ρ_2). This indicates that the value of P101 should not be significant even for values of μ even shorter (or, conversely, longer times but still much shorter than the repair or failure times) than usual for the situation evaluated. Our investigation intends to evaluate what happens when the distributions linked to repair (or failure) no longer have exponential characteristics (in addition to the exponential, we use Constant and Lognormal for repair and Weibull for failure). For this, it is necessary to use DES models, since the distributions will no longer have Markovian characteristics. Thus, the DES model is shown in Fig. 7, considering the significant states of Fig. 1.

4 DISCRETE EVENT SIMULATION

After a more in-depth analysis with Matlab-Simulink, in the proposed model we started to evaluate other characteristics that cannot be contemplated with the Markovian model. Fig. 7 shows two identical closed rows. At the top dashed rectangle (A) there is a device, which keeps switching indefinitely (Failure-Repair-Stock). The assumption is that this device is initially in Operation (O). At the bottom dashed rectangle (B) there is another device generated, which also alternates states indefinitely (Failure-Repair-Stock). It is initially in the Stock, and it is retained in the Hold block. The two dashed rectangles are exactly the same, but they are always in different states (unless both are in failure). While the upper block starts in the Operation (O) state, the lower block starts in the Stock (S) state. The block that is in state (O) will only leave that state in case of a failure, going to state (F) and then goes into repair. When this occurs, the failure block sends a signal to the block that was in Stock (S), which immediately goes to state (O). This will not happen when a double failure occurs (both blocks are in failure and both go into repair). Then, the processing is reversed and the lower dashed rectangle (B) has the block in Operation and the upper rectangle (A) has the block in Stock (or under repair). The two blocks communicate via two global variables, Warning 1 and Warning 2. Block A, held in Hold A (and it is not in repair), is activated by the occurrence of a condition, with the variable "Warning 1 == 1", what is done when there is a fault in block B, after a failure has occurred. The same is true on the other side with respect to block B (Warning 2 == 1), when held in

Hold B (and it is not in repair). We can see that both parts have two Processes, one to cover the failure time and the other to cover the repair time. There are also two decision points (one in each rectangle), where the question is whether the state is Operation (O) or not.

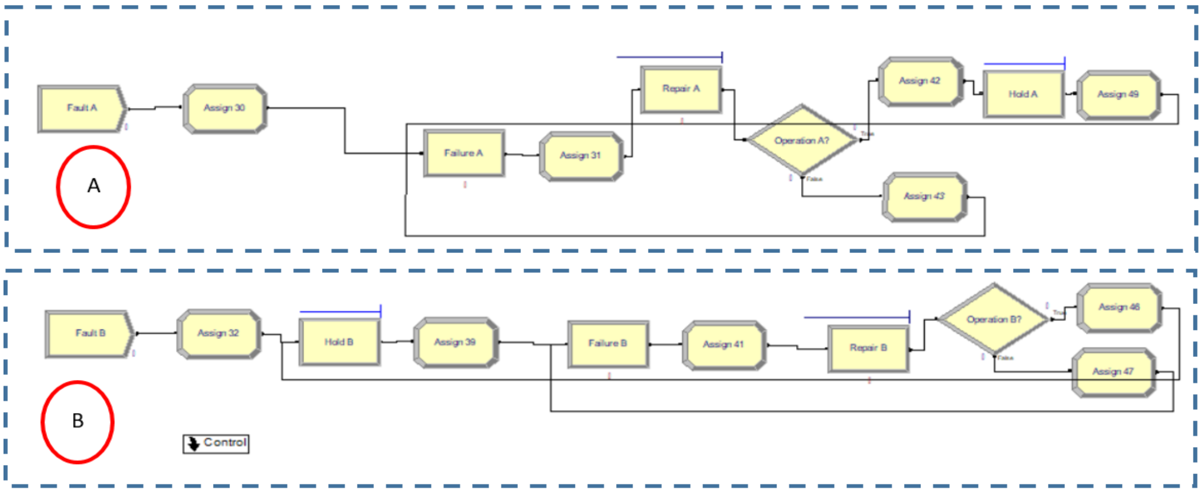

Figure 7: Discrete event simulation model.

There is a sub-model that monitors (Control) the number of blocks in operation at a given time, noticing how many elements are failing to evaluate the probabilities. It with scans in seconds (s), while events occur in hours (h). When the Failure block is repaired, it now goes into Stock, remaining in Hold (unless the other block is also in failure). The same occurs when the block that was in Stock becomes Operational. The blocks will alternate in the three states as Operation-Failure-Stock. Having a known model as the exponential is that it will be the basis for the validation of the Simulation model. Table 4 shows that the values match the ones calculated with the continuous exponential model, and validate the Simulation model.

Table 4: Validation with the Exponential model.

P0	**P1 (SR)**	**P2 (U)**
0.48156 ± 0.02592	0.30985 ± 0.02038	0.20858 ± 0.01345

Once the Simulation model is validated, we will make use of the Constant and Lognormal distributions for the repair service and the Weibull distribution for the failure distribution. The system could not be evaluated for these distributions using the usual Markovian model. All distributions were tested with the same average value. The choice is due to the fact that one has zero variance (Constant), the other is long tail, Lognormal (heavy-tailed) and the other has important characteristics to model different types of failures (Weibull). Using the same values as in the previous examples, the Lognormal distribution will have the same average as the exponential distribution, 1/0.3 = 3.33 h/ repair (MTTR) and standard deviation = 3X 3.33 = 10 h/repair. All distributions use the same MTTR and MTBF (Table 5). We can notice that the most critical values of Unavailability (P2) are worse in the Weibull distribution (for failure), then in the Lognormal distribution (for repair), and better in the Constant distribution (for repair). This is because we have a closed queue. Although the probability of Stock Rupture is lower in the Weibull distribution (next in Lognormal), it is a negative mark because this fact would alert about the need to stock spare.

Table 5: Constant and Lognormal distributions for repairs and Weibull for failures.

Distribuition	P0	P1 (SR)	P2 (U)
Constant	0.42318 ± 0.02969	0.41179 ± 0.01759	0.16504 ± 0.01386
Lognormal	0.53184 ± 0.04975	0.21531 ± 0.01827	0.25284 ± 0.06204
Weibull	0.48735 ± 0.04847	0.23038 ± 0.01447	0.28227 ± 0.03747

5 CASE STUDY

To demonstrate the calculations made in the previous topics and develop them in a real environment, a Firewall environment deployed in one of the company's units, responsible for perimeter security and client-to-site VPN as shown in Fig. 8. The MTBF, according to the manufacturer, is 7 years for the system shown.

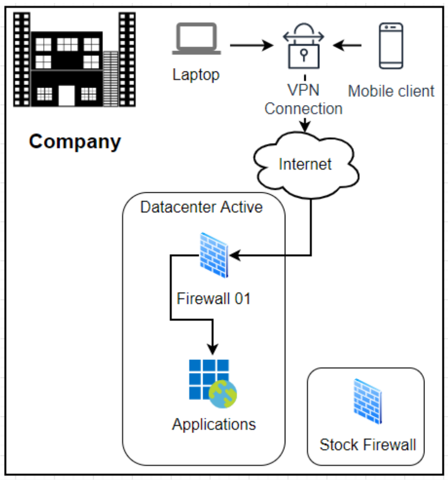

Figure 8: Case study topology.

Equipment of the same model is in stock, and in case of problems with the equipment in operation, it is possible to make the replacement. Thus, it will be necessary to remove the Firewall in Operation and put the equipment in Stock to replace it, replacing all the cabling of the same, which will take about 1 hour. In addition, it will be necessary to update the firmware version and equalize all the settings from one to the other, through backup, which should also take 1 hour, with this whole process being performed by a specialist agent. The monitoring system will open a call with critical severity, the service must be started within 1 hour after opening the ticket. Adding up all the times, the total working time should be 3 hours, if all the conditions mentioned are met within the stipulated time. In addition, it will be necessary to update the firmware version and equalize all the configurations from one to the other, through a backup, which should also take 1 hour, and this entire process is carried out by a specialized agent. The monitoring system will open a call with critical severity, the service starts within 1 hour after opening the ticket. Thus, the total working time must be 3 hours, as long as all the conditions are fulfilled within the stipulated period:

- The SLA contracted by the company studied is 99.8% per year;
- In case of failure, the monitoring system opens a “Severity 1”; the SLA for this incident is 01 hour;
- The SLA for resolution to Severity 1 is 04 hours after signing the call generated;
- If a failure is identified, the company has a support contract that guarantees that new equipment will be sent on the next business day. For failures on Friday, the Unavailability can last up to four days.

There are two maintenance scenarios: (1) the best maintenance scenario for reactivating the failed equipment, the sum of the incident signature SLA generated by the monitoring tool (up to 01 hour) and the incident resolution SLA after the subscription (up to 02 hours), without the need to send new hardware, totaling 3 hours; (2) the worst maintenance scenario for the reactivation of the failed equipment, that is, the previous three hours and the time to ship new hardware by the manufacturer (up to 04 days = 96 hours), i.e., a total of 99 hours. Other significant times will not be counted, as we do not have an average of hours to make the estimate, but, depending on the case, other times must be considered. It may happen that, when removing the equipment from stock, it is already in fault as it has not been previously tested. In this way, we can consider that the MTBF values of 7 years ($1/\lambda_1$) and 7 years + 2h ($1/\lambda_2$) for check the failure. Thus, $1/\mu$ is the sum of 01 hour of ticket signature + 01 hour of replacement in the rack + 01 hour for updating and configuration ($\mu = 1/3$ replacement/h). For maintenance and replacement of equipment, $\rho_1 = 1/99$ and $\rho_2 = 1/102$ repair/h can be considered. Thus, $1/\rho_1$ will be the time for replacing the equipment at the manufacturer and placed directly in stock and $1/\rho_2$ will be the replacement time of the equipment in the manufacturer plus 03 hours to activate the equipment and put into operation. This is an expensive equipment for the Data Center. According (Uptime Institute 2021), stated in their latest report that about a third of all reported outages cost more than US $ 250,000, and can cost much more than US $ 1 million.

5.1 Continuous Simulation

The continuous simulation, including the transient, can be seen in Fig. 9 that shows the faultless state (left side) and the single failure state (Stock Rupture). It's seen that the probability of not having failures is lower when $\mu = 0$, and the probability of having single failure is greater for $\mu = 0$. Single failure case is positive because it means earlier warning that there was stock. If the value of $\mu = 0$, the system will be with greater probability of Unavailability ($P_2 = 1 - P_1 - P_0$). The difference seems insignificant, but when it comes to high values for Availability (for four or five nines) the difference can even mean the imposition of fines.

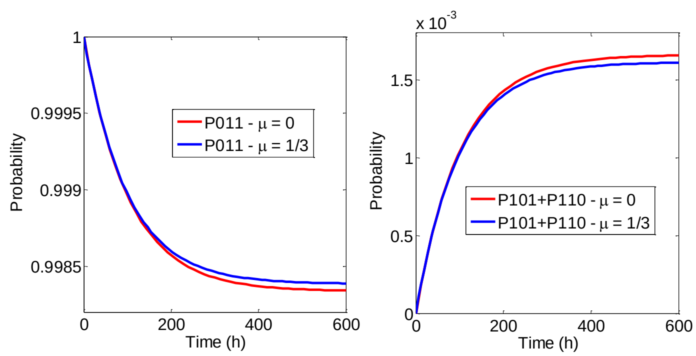

Figure 9: Probability of having 0 units failing (left) or having one unit failing (stock rupture).

Table 6 shows the stationary values of the states with the real data for the extended model.

5.1 Discrete Event Simulation

The use of Continuous Simulation may be sufficient in some situations. However, when the probability distributions are different from the exponential, it is necessary to use DES. The best approach is the one that fits the best probability distributions from measured data from the real system. Our approach intends to assess in which situations the values of Availability or Stock Rupture can be exceeded. For this we use the Constant, Exponential and Lognormal distributions for repair (R) and Weibull for failures (F). With $\mu = 0$ and with $\rho_1 = \rho_2$, the diagram in Fig. 3 converges to the diagram in Fig. 1. Thus, we use $\mu = 0$ and an average for ρ value (since they are very close) to do the DES. Table 7 shows the results of the simulations.

Table 6: Values of steady state.

	P0	P1		P2
μ	**P011**	**P101**	**P110**	**P200**
0	0.9983	0.0017	2.68e-06	2.76e-06
1/3	0.9984	4.74e-05	0.0016	2.68e-06

Table 7: Simulation values.

Distribution	P0	P1 (SR)	P2 (U)
Exponential (R)	0.99810 ± 4.91e-04	0.00190 ± 4.91e-04	0.00 (*)
Constant (R)	0.99841 ± 0.00230	0.00159 ± 0.00230	0.00 (*)
Lognormal (R)	0.99596 ± 0.02220	0.00404 ± 0.02220	0.00 (*)
Weibull (F)	0.99824 ± 0.00215	0.00173 ± 0.00252	2.89e-05 ± 3.67e-04

(*) Replications with 5.10^6 hours of duration and it was not possible to detect a double failure. Each run elapsed about four hours.

6 DISCUSSION

We made several considerations that could affect System Availability. However, in the real case analyzed, the results showed that the system has MTTR values much lower than MTBF values. This may indicate that, depending on the associated cost, the Discrete Event Simulation will have to be carried out, but it will take a long time. Thus, the use of Discrete Event Simulation can be applied, but only in specific cases. In addition, as seen, the Continuous Simulation, done initially, serves as a guide to assess the need or not of Discrete Event Simulation. On the other hand, if sufficient, the Steady State value is quickly obtained by using usual Markovian models. We also emphasize that the application of the analytical model (easier to use) guides the application of the Discrete Event Simulation model. Thus, in the considered Case Study, there was no Availability value greater than the established requirement (99.8%), even for other probability distributions. However, depending on the requirement (eg, five nines), this analysis would have to go further. We did the Simulation study knowing that the greatest importance is on the types of probability distributions (we use Fig. 1), but it might be necessary to evaluate details and thus have to simulate according to Fig. 3. We observed that the P2 probability value (Unavailability) was not obtained for the Discrete Event Simulation model after running 5 h. Availability can be obtained by adding 1- P0 - P1, which is sufficient for our analysis. However, in critical cases, we would have to run much longer than 5 h.

7 CONCLUSION

This article shows the importance of making a more in-depth analysis to obtain Availability values, especially when high cost items are involved. We must use Continuous and Discrete Simulation. Continuous Simulation, although performed with the most conventional hypotheses, serves as a guide for DES, showing the points that are the most important. DES is essential that there are measures in the System in focus so that the most appropriate probability distributions are fitted. The results of the simulation showed that the knowledge of the failure distribution may be more important than that of the repair distribution. Thus, the analyst will be able to assess, for example, the question of the cost of having more devices in stock or making improvements to Maintainability. The knowledge of the probability distributions involved is essential to achieve the desired results. However, this will depend on how much precision the result obtained must have, which is a function of the involved costs. Thus, systems that have MTTRs in units of magnitude close to MTBFs, must undergo more in-depth analysis of the various steps that can affect the final Availability result. In situations where the values of MTBFs are much higher than those of MTTRs, the models quickly converge to the Steady State. In these cases, Markovian models brings the advantages of fast calculation and the values obtained by closed equations. Availability value must be done with the support of scientific software. Equations (9-12) were obtained relatively easily using Matlab's solve function. The DES carried out lasted around 16 hours. In real cases should have a much longer duration and a greater number of replications. If possible, high-capacity computers should be used. In high cost systems, there are several different manufacturers or that the Physical-Functional configurations may be in different ways. This can be used for decision making when signing the SLA contract, or for Planning and Management of the System Operation. A future work would be the placement of a model to assist in the decision-making process in view of the involved costs.

ACKNOWLEDGMENTS

This study was funded in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior – Brasil (CAPES) - Finance Code 001.

REFERENCES

Avizienis, A., J. C. Laprie, and B. Randell. 2001. "Fundamental concepts of dependability," (pp. 7-12). University of Newcastle upon Tyne, Computing Science.

Barabadi, A. 2012. "Reliability and Spare Parts Estimation taking into Consideration the Operational Environment - A Case Study", International Journal of Performability Engineering Vol. 8, No. 5, pp. 497-506.

Barabadi, A., J. Barabady, and T. Markeset. 2011. "Maintainability analysis considering time-dependent and time-independent covariates", Reliability Engineering and System Safety, vol. 96, no.1, pp. 210- 217.

Blokus, A., and P. Dziula. 2019. "Reliability Analysis of Different Configurations of Master and Back-Up Systems Used in Maritime Navigation", Journal of Marine Science and Engineering, J. Mar. Sci. Eng. 2020, 8, 34; doi:10.3390/jmse8010034.

Bukhsh A. Z., A. Saeed, I. Stipanovica, and A. G. Doreea, "Predictive maintenance using tree-based classification techniques:A case of railway switches", Transportation Research Part C 101 (2019) 35–54, 2019.

Chumash, T. "Obtaining Five 'Nines' of Availability for Internet Services," Rutgers University. https://www.cs.rutgers.edu/~rmartin/teaching/spring06/cs553/papers/003.pdf, accessed 1th February 2017.

Hitechnectar. *What is the Importance of Predictive Maintenance in Industry 4.0*, https://www.hitechnectar.com/blogs/predictive-maintenance-industry-4-0, accessed in 9th April 2021.

ITU-T. 2012. "Quality of Service and Network Performance: Handbook", Handbooks on Standardization, ITU, Geneva.

Kanso, A., M. Toeroe, and F. Khendek. 2014. "Comparing redundancy models for high availability middleware", Computing (2014) 96:975–993.

Lenz, M., and J. Rhodin. 2011. "Reliability calculations for complex systems", thesis, Linköpings university, Sweden, http://liu.diva-portal.org/smash/get/diva2:433387/FULLTEXT01.pdf, accessed in 8th April 2021.

Reliawiki, *RBDs and Analytical System Reliability*. 2021. Chapter 3, http://reliawiki.org/index.php/RBDs_and_Analytical_System_Reliability, accessed in 4th April 2021.

Souza-Franco, V. R. L., M. V. Clavijo, A. M. Schleder, and M. R. Martins. 2019. "Monte Carlo Simulation to Consider Uncertainty in the Reliability Analysis of Dynamic Positioning Systems", 29th European Safety and Reliability Conference.

Uptime Institute. 2019. *The importance of Uptime in the Data Center*, https://datacenter.com/news_and_insight/the-importance-of-uptime-in-the-data-center, accessed in 10th April 2021.

Ursini, E. L., H. de C. L. dos Santos, and M. T. Okano. 2020. "Improving the Availability of Firewalls with a View to Increasing ICT Consumption Due Covid-19," 11th IEEE Annual IEMCON, Vancouver, BC, Canada, 2020, pp. 0647-0653.

AUTHOR BIOGRAPHIES

EDSON L. URSINI is an Associate Professor at School of Technology, Unicamp with a doctoral degree in Electrical Engineering from University of Campinas. He works with data networks, stochastic processes, queuing models, teletraffic, discrete event simulation, continuous simulation, and performance analysis. His email is ursini@ft.unicamp.br.

HENRY DE CASTRO LOBO DOS SANTOS is a Doctorate Student at School of Technology - Unicamp with a MsC in University of Campinas. He works with data networks and information security. His research area is queuing models, data traffic, discrete event simulation, continuous simulation and performance analysis. His email is h190839@dac.unicamp.br.

MARCELO T. OKANO is Permanent Professor of Professional Master in Productive Systems, holds PhD degree in Production Engineering from UNIP and PhD Degree in Business Administration from EAESP/FGV. Research areas are Innovation, ICT and Business. His email is mtokano@unicamp.br.