

Sensitivity Analysis Techniques Applied in Video Streaming Service on Eucalyptus Cloud Environments

Rosangela Melo^{1,2*}, Maria Clara Bezerra¹, Jamilson Dantas¹, Rubens Matos¹,
Ivanildo José de Melo Filho^{1,2}, Aline Santana Oliveira¹, Fábio Feliciano², Paulo Romero Martins Maciel¹

¹ Informatics Center, Federal University of Pernambuco, Recife, BRAZIL

² Federal Institute of Education, Science, and Technology of Pernambuco (IFPE), Belo Jardim Campus, BRAZIL

*Corresponding Author: rosangela.melo@belojardim.ifpe.edu.br

Citation: Melo, R., Bezerra, M. C., Dantas, J., Matos, R., de Melo Filho, I. J., Oliveira, A. S., Feliciano, F. and Maciel, P. R. M. (2018). Sensitivity Analysis Techniques Applied in Video Streaming Service on Eucalyptus Cloud Environments. *Journal of Information Systems Engineering & Management*, 3(1), 02. <https://doi.org/10.20897/jisem.201802>

Published: January 23, 2018

ABSTRACT

Nowdays, several streaming servers are available to provide a variety of multimedia applications such as Video on Demand in cloud computing environments. These environments have the business potential because of the pay-per-use model, as well as the advantages of easy scalability and, up-to-date of the packages and programs. This paper uses hierarchical modeling and different sensitivity analysis techniques to determine the parameters that cause the greatest impact on the availability of a Video on Demand. The results show that distinct approaches provide similar results regarding the sensitivity ranking, with specific exceptions. A combined evaluation indicates that system availability may be improved effectively by focusing on a reduced set of factors that produce large variation on the measure of interest.

Keywords: availability, cloud computing, sensitivity analysis, video on demand

INTRODUCTION

Cloud Computing may be defined as a model for enabling on-demand network access to a shared pool of configurable computing resources that can be rapidly provisioned and released with minimal management effort or service provider interaction (NIST, 2013). This resource set is typically utilized as a service model where in the client only pays for what is consumed. Video on Demand (VoD) streaming services are run on cloud infrastructure platforms in order to offer cost savings, easy scalability, and high availability to users. To evaluate the availability of the service, hierarchical analytical models are employed to represent the architecture (Dantas et al., 2012).

In this paper, we propose an availability model of a VoD service based on Eucalyptus cloud environment, for evaluating the sensitivity of VoD service components. For guiding the implementation of system improvements, Design of Experiments (DoE) and percentage difference were used for identifying availability bottlenecks of the VoD service.

The remainder of the paper is organized as follows: Section II introduces related works on system availability and sensitivity analysis, Section III introduces basic concepts of cloud computing technologies, video streaming, dependability models and analysis and sensitivity analysis. Section IV presents the architecture of the system analyzed in this paper. Section V presents the availability model designed for architecture. Section VI conducts a case study about architecture analysis and presents sensitivity analysis. Finally, Section VII shows the conclusions of the study and suggests possible future work.

RELATED WORKS

Recent research have employed hierarchical modeling to represent cloud computing architectures, it is possible to compare the different solutions and estimations of dependability measures (Dantas et al., 2012; Chuob et al., 2011). Moreover, some works have also employed sensitivity analysis to identify the critical system components, and thereby propose infrastructure improvements (Matos et al., 2012). Khazaei *et al.* (2012) integrated an availability model in overall analytical sub-models of cloud system. Each sub-model captures a specific aspect of cloud centers. The key performance metrics such as task blocking probability and total delay incurred on user tasks are obtained.

Choub *et al.* (2011) proposed a private cloud, with a modeling of the cloud computing based upon the Eucalyptus architecture. To understand the behavior of Eucalyptus, it was considered Ubuntu Enterprise Cloud (UEC) as reference architecture for our Cloud Test bed environment in Lab. With UEC architecture, it was addressed the availability of each component of the cloud base on Markov chain through the level analysis of hierarchical available model (HAM). Malik et al. (2013) provided a formal analysis modeling and verification of open source state-of-art VM-based cloud management plataforms to model and analyze the structural and behavioral properties of the systems have used high-Level Petri Nets. In (Araujo et al., 2014), an availability model of a digital cloud library through a OpenNebula cloud manager is proposed, using an hierarchical approach to model and evaluate the digital library environment. Measurements were performed to obtain the availability parameters of the library service deployed in a private cloud. In this paper, the authors proposed availability models applied to a cloud environment for VoD streaming service. Design of Experiments (DoE) and percentage difference were applied in order to find the bottlenecks of system availability.

BACKGROUND

This section presents the concepts which provide a background for this paper, including: cloud computing technologies, video streaming, dependability modeling and analysis and sensitivity analysis techniques.

Cloud Computing and the Eucalyptus Platform

A cloud computing system is comprised of a bundle of resources, such as hardware, software, development platforms and services, readily usable and accessible through the Internet (Armbrust et al., 2010). Services can be provisioned on different levels by cloud computing providers, including Infrastructure as a Service (IaaS), Platform as a Service (PaaS), and Software as a Service (SaaS). In a cloud computing environment, virtual resources can also be dynamically allocated and resized to deal with a varying workload, enabling optimal use of physical resources. Cloud computing services are typically accessed through a pay-per-use model (Pereira, 2010) and the provider offers is determined and guaranteed by means of service level agreements.

Eucalyptus is a software architecture based on Linux that leverage the implementation of private and hybrid IaaS clouds. This means that users can utilize their own collections of resources (hardware, storage and network) through a selfservice interface according to their needs. The Eucalyptus software framework is modular (Eucalyptus Systems, 2010), and consists of five high level components, each with its own Web Service: Cloud Controller (CLC), Cluster Controller (CC), Node Controller (NC), Storage Controller (SC), e Walrus (Eucalyptus Systems, 2010).

Video Streaming

Video streaming is a technology used in the transmission of digital multimedia contenton the Internet (Delgado et al., 2006). Streaming enables data to be sent and flow without the need to wait for content to load completely. This requires smaller network bandwidth and less storage space. As the multimedia data arrives, it is stored in a fast buffer before execution starts. Encoding, protocols and buffering mechanism are factors that might affect transmission of a video streaming (Diaz-Sanchez et al., 2011). The video streaming service works with the protocol Real Time Streaming Protocol (RTSP) which allows the control in the data transfer with real time properties. The RTSP makes the transfer possible on demand of real time data with audio and video (Valeriana and Marcelo, 2008).

Models for Dependability Analysis

In early 1980s Laprie coined the term dependability for encompassing concepts such reliability, availability, safety, confidentiality, maintainability, security and integrity etc. (Laprie, 1992; Laprie, 1995), whereas reliability of system at t is the probability that the system performs its functions without failing up to time instant t . Availability can be expressed as the ratio of the expected system uptime to the expected system up and downtime: $A = E[Uptime]/(E[Uptime] + E[Downtime])$. It may also be represented by: $A = MTTF/(MTTF + MTTR)$ where $MTTF$ and $MTTR$ is the mean time to failure and to recovery, respectively.

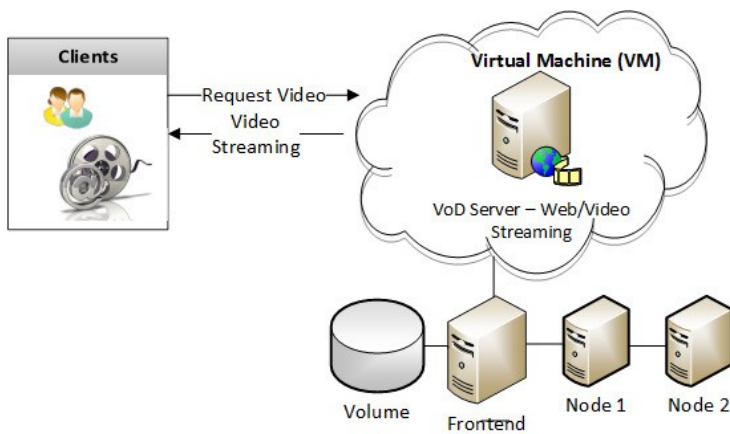


Figure 1. VoD Architecture

There are several types of models that can be used for analytical evaluation of dependability. Reliability Block Diagrams (RBD), Fault Trees, Stochastic Petri Nets (SPN) and Continuous Time Markov Chains (CTMC) have been used to model fault-tolerant systems and evaluate various dependability measures.

Sensitivity Analysis

The Design of Experiments, known as DoE, is a method used to perform sensitivity analysis. Through this method it is possible to assess the importance of each parameter of the system and furthermore it can be used to simultaneously determine the individual and interactive effects of many factors that may affect the output measures. In the DoE method, parameters are called factors and the value that is assigned to each factor is called a level. There are numerous types of DoE and among the most commonly used are: full factorial design, fractional factorial design and simple design (Jain, 1991). Sometimes the number of experiments required for a full factorial design is too large. This may happen if either the number of factors or their levels is large. It may not be possible to use a full factorial design due to the cost or the time required. In such cases, one can only use a fraction of the full factorial design (Mathews, 2004). Fractional factorial designs are commonly used to reduce the number of runs required to build an experiment (Mathews, 2004). In this study we adopted the fractional factorial design, because there are more than five variables that we need to analyze. For 2^k designs with five or more variables consider using the fractional factorial designs to reduce the number of runs required by the experiment.

SYSTEM ARCHITECTURE

The system architecture is based on Eucalyptus cloud computing platform. A broad view of the components of the VoD service architecture is seen in [Figure 1](#) (Melo et al., 2017).

The VoD architecture is divided into two sides: client and the server. The physical structure is composed of three machines. One machine is used for frontend and two machines for the nodes. The client connects to the video streaming server through the Internet. A storage volume is allocated in the frontend for storing the collection of videos. A Virtual Machine (VM), running the Apache and VLC applications, is instantiated in the nodes. VLC provides the video streaming features, whereas Apache is responsible for hosting the service on a dedicated Web page. The user issues a request for displaying a video hosted on a specific web page. VLC, in turn, grabs the requested video from the remote storage volume and transmits the stream to the user.

AVAILABILITY MODELS

This section discusses the availability models employed to represent the redundant architecture evaluated by this research, from which the availability values were calculated. Such values were obtained by the hierarchical combinatorial method, which combines the system state representation of Markov chains with RBDs (Sahner and Trivedi, 1987; Kim et al., 2009), and is the method commonly employed to evaluate complex IT systems. The systems were modeled and evaluated with the SHARPE (Silva et al., 2012) and Mercury tools (MoDCS, 2017), which were specifically designed for the analysis of such models.

Model for Architecture

RBD and CTMC models were used to represent the subsystems of the architectures presented in [Figure 1](#). These models are then combined, constituting a hierarchical model. The architecture can be divided into three

In state DUW the system is unavailable due to service application failure, although the first node is up and the second node is in warm standby mode. From DUW the following states can be reached; DUD (failure of the warm standby node), DDW (first node failure), or U UW (service repair). With service and first node operational, and the second node in warm standby, U UW indicates system availability. From this position in the model, the possibilities are warm standby failure (UUD), service failure (DUW) or first node failure, which would cause the service to become unavailable (DDW). The state DDW indicates system unavailability, with service and first node down, and second node in warm standby. From DDW, it is possible to reach three other states; failure of all the components of the system (DDD), initialization of the second node (DDU), or repair of the first node (DUW).

System failure is an event that occurs when the provided service deviates from the intended service (Silva et al., 2012; MoDCS, 2017). The failure rates of the two nodes are represented by λ_N , whilst μ_N represents the rate of node repair. A node in warm standby has the failure rate of λ_{WN} , and the repair rate to return it to standby is μ_N . A warm standby node is transformed to available mode at the rate of μ_{WN} . The failure rate of the service application is λ_{V_0D} , while the repair rate is μ_{V_0D} . The λ_{V_0D} was obtained from the inverse of the time to failure of the service module. To calculate this result, we used the CTMC model of **Figure 3** and the Mercury tool (Silva et al., 2012; MoDCS, 2017). The repair rate of the service is considered as the instantiation of a new virtual machine, including all the applications necessary to its operation (Apache and VLC).

Through the CTMC of **Figure 3** it is possible to obtain the closed-form equation for calculating the availability of the redundant model (A_{V_0D}) as shown below in Equation 1. The equation for calculating the availability of the whole infrastructure of the service (A_{V_0D}) can also be obtained from its corresponding RBD model (Dantas et al., 2012).

$$A_{V_0D} = \frac{\alpha(2\beta^2\alpha_1 + \beta\alpha_2\beta_1 + \alpha_3)\beta_1^2 + \alpha_4\beta_1^3}{\alpha_5(\beta^2\alpha_1(\lambda_N + 2\mu_N) + \beta\varphi\beta_1 + \varphi_1\beta_1^2 + \beta_7\beta^3)} \quad (1)$$

where:

$$\begin{aligned} \beta &= \lambda_{WN}, \\ \beta_1 &= \mu_{WN}, \\ \alpha &= \mu_N\mu_{V_0D}, \\ \alpha_1 &= (\lambda_N + \beta + \mu_N)^2(\lambda_N + \beta + \mu_N), \\ \alpha_2 &= 4\lambda_N^2 + 17\lambda_N\beta + 13\beta + 5\lambda\mu_N + 12\beta\mu_N + 2\mu_N^2, \\ \alpha_3 &= (8\beta(2\beta + \mu_N) + \lambda_N(11\beta + \mu_N)), \\ \alpha_3 &= (8\beta(2\beta + \mu_N) + \lambda_N(11\beta + \mu_N)), \\ \alpha_4 &= (7\beta + 2\mu_N)(\mu_{WN}^3), \\ \alpha_5 &= \lambda_N + \lambda_{V_0D} + \mu_{V_0D}, \\ \alpha_6 &= 2\lambda_N(\lambda_N + \beta)(\lambda_N + 3\beta), \\ \beta_2 &= (8\lambda_N^2 + 24\lambda_N\lambda_{WN} + 13\beta^2)\mu_N, \\ \beta_3 &= (7\lambda_N + 12\beta)\mu_N^2 + 2\mu_N^3, \\ \beta_4 &= 2\lambda_N\beta(2\lambda_N + 3\beta), \\ \beta_5 &= (\lambda_N + \beta)(\lambda_N + 16\beta)\mu_N, \\ \beta_6 &= (\lambda_N + 8\beta)\mu_N^2, \\ \beta_7 &= (\lambda_N(\beta + 2\mu_N) + \mu_N(7\beta + 2\mu_N)), \\ \varphi &= \alpha_6 + \beta_2 + \beta_3 \text{ and} \\ \varphi_1 &= \beta_4 + \beta_5 + \beta_6. \end{aligned}$$

A closed-form equation for computing the availability of the complete service ($A_{Service}$) can also be obtained, as demonstrated by Equation 2. A_f (availability of the frontend) and A_v (availability of the volume) can be computed from the RBD of **Figure 2**, whilst A_{V_0D} is calculated from Equation 1. In this equation, A_f , A_v , and A_{V_0D} correspond to the availability of the frontend, volume, and service, respectively.

$$A_{Service} = A_f \times A_v \times A_{V_0D} \quad (2)$$

CASE STUDIES

Three case studies were designed to analyze the system availability (Melo et al., 2017). The case studies can be summarized as follows:

- i) Case Study I: availability analysis of architecture.
- ii) Case Study II: sensitivity analysis of all the system components (see **Figure 2**) to establish a ranking of the most important parameters in the video streaming service.
- iii) Case Study III: analysis the behavior of the system using other technique of sensitivity analysis, by means of a percentage difference.

Table 1. Parameters for RBD of the system

Component	MTTF	MTTR
Frontend	180.72 h	0.96999h
Volume	100000 h	1 h
Service	217.77 h	0.92633

Table 2. Redundant system CTMC parameters

Parameter	Description	Value h^{-1}
λ_N	Mean Time to Node Failure	1/481.83
λ_{WN}	Mean Time to Standby Node Failure	1/578.196
μ_N	Mean Time to Node Repair	1/0.91
μ_{WN}	Mean Time to Standby Node Repair	1/0.0333
λ_{V_0D}	Mean Time to Service Failure	1/217.779
μ_{V_0D}	Mean Time to Service Instantiate	1/0.0275

Table 3. Input parameters for the DoE

Metrics	Parameters	Minimum Value (h)	Maximum Value (h)
MTTF	λ_f	180.72	4380
	λ_{V_0D}	217.779	4380
	λ_{V_0l}	10000	110000
	λ_N	481.83	4380
	λ_{WN}	578.196	4380
MTTR	μ_{V_0l}	1	2
	μ_f	0.96999	2
	μ_{V_0D}	0.0275	0.083333
	μ_{WN}	0.0333	0.083333
	μ_N	0.091	2

Case Study I

Figure 1 depicts the architecture, which has the dedicated frontend machine and two machines for the nodes. The **Table 1** shows the values of mean time to failure (MTTF) and mean time to repair (MTTR) used the model of the Frontend. Those values were obtained from (Kim et al., 2009; Dantas et al., 2012) and were used to compute the dependability metrics for the frontend, and subsequently for the whole system. This subsystem has a MTTF of 180.72 h and a MTTR of 0.96999 h.

The values used of MTTF and MTTR for Volume subsystem are described in the **Table 1**. Those values were obtained from Melo et al. (2017) and Kim et al. (2009). **Table 1** presents the parameters for the blocks of the RBD model for the structure shown in the **Figure 2**. The values of MTTF and MTTR of the Frontend and volume modules are based on Dantas et al. (2012) and Melo et al. (2017). The availability of the Service module is computed from the CTMC depicted in **Figure 3**. **Table 2** presents all values of the parameters used for computing the availability of the Service module. The values are based on the analyses shown in Dantas et al. (2012) and Melo et al. (2017).

For the presented configuration of parameters, we find a value of 0.994401 for the availability of the video streaming system, without redundancy. This availability corresponds to about 49,05 hours of downtime in a year, and therefore highlights the importance of searching for effective solutions to improve this system.

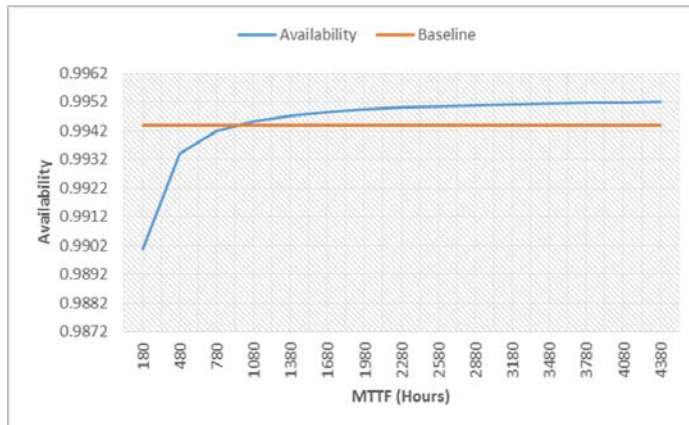
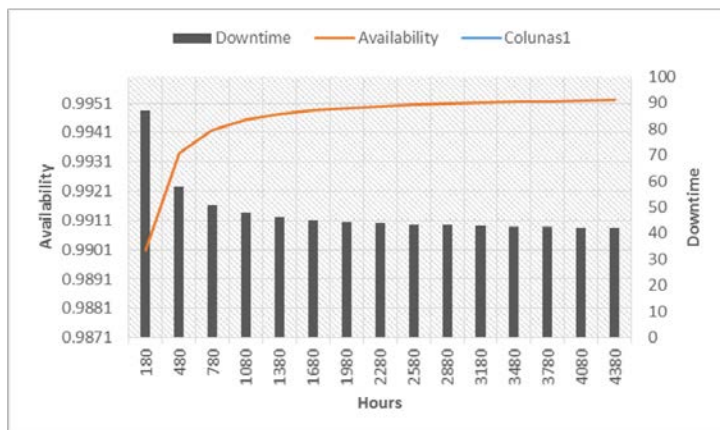
Case Study II

For 2^k designs with five or more variables consider using the fractional factorial designs to reduce the number of runs required by the experiment (Mathews, 2005). This way, we performed the analysis of a fractional factorial design of experiments to provide point of view on the sensitivity of Video Stream Service availability with respect to each parameter. This analysis was performed on the 10 parameters shown in the ranking based on partial derivatives. Two levels were considered for each parameter: the minimum and maximum values used in the graphical analysis. This 2^k factorial experiment was evaluated according to the individual effects for the system availability, and these values are shown in **Table 4**. The measures of interest were calculated with the values given in **Table 1** and **Table 3**, and the sensitivity ranking of availability for all parameters of the streaming service were given in **Table 4**. The tools Minitab was used for calculated sensitivity analysis (Minitab, 2000).

The results are ordered according to absolute values. Negative values indicate that there is an inverse relationship between parameters and system availability. For example, the sensitivity with respect to failure rates is negative due to the fact that as failure rate increases (that is, the MTTF decreases) availability decreases. The index in **Table 4** indicates that the parameters λ_f , λ_{WN} and μ_f have the largest effect values. Based on the results it can

Table 4. Sensitivity ranking from 2^k experiment analysis

Parameter	Effect
λ_f	0.009044
λ_{WN}	0.003580
μ_f	-0.002920
μ_{WN}	-0.001342
μ_{V_0L}	-0.000905
λ_{V_0L}	0.000896
μ_{V_0D}	0.000895
λ_{V_0D}	-0.000447
λ_N	-0.000447
μ_N	-0.000001

**Figure 4.** Effect of MTTF on system availability**Figure 5.** Availability versus Downtime

be concluded that the Fontend is the critical point of the video streaming system in terms of availability, and should therefore receive priority when improvements to the system are considered. **Table 4** also demonstrates that the rate λ_N , which is the rate of repair the node, is shown to have the second least impact on system availability, with only μ_N having less. The ranking obtained from design of experiments sensitivity analysis provides a direct view of the order of importance of all parameters.

Figure 4 is a graphic representation of the system availability where the MTTF parameters for Frontend were altered one at a time, whilst fixing all other parameters to the values given in **Table 1** and **Table 2**. The plot confirms that increasing the failure times of the Frontend modules results in increased availability. The need to implement improvements in the Frontend modules is confirmed by the ranking given in **Table 4**. An alternative method for reaching this conclusion would be to implement failover mechanisms in one or both of the components and evaluate on impact on system availability.

Figure 5 illustrates the system availability as a function of the MTTF for the Frontend module, and clearly demonstrates the impact that MTTF increases have on system availability. Increasing the MTTF of the Frontend results in a reduction in downtime of 44.80 hours in the year.

Table 5. Sensitivity ranking from percentage difference

Parameter	SI (A)
μ_f	0.00563709501
λ_f	-0.00514476562
μ_{V_0D}	0.00037202503
μ_{V_0L}	0.00018996201
λ_{V_0D}	-0.00011998807
λ_N	-0.00009585142
μ_{WN}	0.00003967628
λ_N	0.00002953291
λ_{V_0L}	-0.00000090908
μ_{WN}	-0.00000001800

Case Study III

In this case study analysis the behavior of the system using another technique of sensitivity analysis. In this case we used the percentage difference with this method we can also identify components that affect the availability of the systems. The percentage difference is calculated using a Equation

3. This equation shows the expression for this approach, where $\max\{Y(\theta_i)\}$ and $\min\{Y(\theta_i)\}$ are the maximum and minimum output values, respectively, computed when varying the parameter θ over the range of its n possible values of interest. If $Y(\theta_i)$ is known to vary monotonically, so only the extreme values of θ (i.e., θ_i and θ_n) may be used to compute $\max\{Y(\theta_i)\}$ and $\min\{Y(\theta_i)\}$ subsequently $S_{\theta_i}\{Y\}$ citejain.

$$S_{\theta_i}\{Y\} = \frac{\max\{Y(\theta_i)\} - \min\{Y(\theta_i)\}}{\max\{Y(\theta_i)\}} \quad (3)$$

Table 5 presents the ranking of the sensitivity analysis in descending order. In order to generate the results of this ranking we used the input data from **Table 3** and applied in Equation (3). **Table 5** gives the percentage difference of all system components calculated from the input parameter values given in **Table 3**.

The **Table 5** indicates that the repair and rate Frontend (μ_f) and (λ_f) module are the most important component with respect to availability. A refined analysis combining the two rankings - 2^k DoE and percentage difference indices may provide a reduced list of parameters which deserve the highest priority to improve the system availability. We perform such a combined analysis by checking the parameters which appear among the first three positions of the rankings. The parameters which match such a criterion are: λ_f , μ_f , μ_{V_0D} and μ_{V_0L} .

CONCLUSION

This paper presented sensitivity analysis of video streaming service availability based on hierarchical analytical models. Design of Experiments (DoE) and percentage difference were used to assess the impact of each input parameter. The results show that the system availability may be improved effectively by focusing on a reduced set of factors which produce large variation on steady-state availability. Most parameters ranked in the highest positions of sensitivity rankings is related to the frontend component. This component should receive the highest priority to achieve effective improvements on system availability. We also performed a combined analysis from the two techniques which is useful to reduce the list of parameters which an analyst should focus on, and solve possible conflicting results from distinct approaches. For future work, the authors propose to implement further sensitivity techniques, and also to extend the scope of the models considered in different topologies.

REFERENCES

- Araujo, J., Maciel, P., Torquato, M., Callou, G. and Andrade, E. (2014). Availability evaluation of digital library cloud services, in *The Fourth International Workshop on Dependability of Clouds, Data Centers and Virtual Machine Technology*. <https://doi.org/10.1109/DSN.2014.65>
- Armbrust, M., Fox, A., Griffith, R., Joseph, A. D., Katz, R., Konwinski, A., Lee, G., Patterson, D., Rabkin, A., Stoica, I. et al. (2010) A view of cloud computing. *Communications of the ACM*, 53(4), 50-58. <https://doi.org/10.1145/1721654.1721672>
- Chuob, S., Pokharel, M. and Park, J. S. (2011). Modeling and analysis of cloud computing availability based on eucalyptus platform for e-government data center, in *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on. IEEE* (pp. 289-296). <https://doi.org/10.1109/IMIS.2011.135>

- Chuob, S., Pokharel, M. and Park, J. S. (2011). Modeling and analysis of cloud computing availability based on eucalyptus platform for e-government data center, in *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2011 Fifth International Conference on. IEEE*, pp. 289-296.
- Dantas, J., Matos, R., Araujo, J. and Maciel, P. (2012). An availability model for eucalyptus platform: An analysis of warm-standby replication mechanism, in *Systems, Man, and Cybernetics (SMC), 2012 IEEE International Conference on. IEEE* (pp. 1664-1669). <https://doi.org/10.1109/ICSMC.2012.6377976>
- Delgado, G. D., Frias, V. C. and Igartua, M. A. (2006). Video-streaming Transmission with QoS over Cross-Layered Ad hoc Networks, in *Software in Telecommunications and Computer Networks, 2006. SoftCOM 2006. International Conference on. IEEE*, pp. 102-106. <https://doi.org/10.1109/SOFTCOM.2006.329729>
- Diaz-Sanchez, D., Almenarez, F., Marin, A., Proserpio, D. and Arias Cabarcos, P. (2011). Media cloud: an open cloud computing middleware for content management. *Consumer Electronics, IEEE Transactions on*, 57(2), 970-978. <https://doi.org/10.1109/TCE.2011.5955247>
- Eucalyptus Systems (2010). *Eucalyptus cloud computing platform - administrator guide*. Technical report, Eucalyptus Systems, Inc., Version 2.0.
- Jain, R. (1991). *The art of computer systems performance analysis: Techniques for experimental measurement, simulation and modeling*. Wiley India Pvt. Ltd.
- Khazaei, H., Mistic, J., Mistic, V. B. and Mohammadi, N. B. (2012). Availability analysis of cloud computing centers, in *Global Communications Conference (GLOBECOM), 2012 IEEE. IEEE*, pp. 1957-1962. <https://doi.org/10.1109/GLOCOM.2012.6503402>
- Kim, D. S., Machida, F. and Trivedi, K. S. (2009). Availability modeling and analysis of a virtualized system, in *Dependable Computing, 2009. PRDC'09. 15th IEEE Pacific Rim International Symposium on. IEEE*, pp. 365-371.
- Laprie, J. (1992). *Dependability: Basic Concepts and Terminology*. SpringerVerlag. <https://doi.org/10.1007/978-3-7091-9170-5>
- Laprie, J.-C. (1995). Dependable computing and fault tolerance: Concepts terminology, in *Fault-Tolerant Computing, Twenty-Fifth International Symposium on. IEEE*, pp. 2-11. <https://doi.org/10.1109/FTCSH.1995.532603>
- Malik, S. U. R., Khan, S. U. and Srinivasan, S. K. (2013). Modeling and analysis of state-of-the-art vm-based cloud management platforms. *IEEE Transactions on Cloud Computing*, 1(1). <https://doi.org/10.1109/TCC.2013.3>
- Mathews, P. G. (2004). *Design of Experiment with MINITAB*. William A. Tony, EUA.
- Mathews, P. G. (2005). *Design of Experiments with MINITAB*. ASQ Quality Press.
- Matos, R., Maciel, P. R. M., Machida, F., Kim, D. S. and Trivedi, K. S. (2012). Sensitivity analysis of server virtualized system availability. *Reliability, IEEE Transactions on*, 61(4), 994-1006. <https://doi.org/10.1109/TR.2012.2220711>
- Melo, R. M., Bezerra, M.C., Dantas, J., Matos, R., Melo Filho, I.J., Oliveira, A., Feliciano, F., Maciel, P. R.M. (2017). Sensitivity Analysis Techniques applied in Cloud Computing Environments, in *Information Systems and Technologies (CISTI), 2017 12th Iberian Conference on*, pp. 1-6. <https://doi.org/10.23919/CISTI.2017.7975950>
- Minitab (2000). *Minitab statistical software*, Minitab Release, vol. 13.
- MoDCS (2017). Mercury tool developed by modcs group. Available at: <https://sites.google.com/site/mercurytooldownload/>
- NIST (2013). *Cloud computing*. Available at <http://www.nist.gov>
- Pereira, R., Azambuja, M., Breitman, K. and Endler, M. (2010, July). An architecture for distributed high performance video processing in the cloud. In *Cloud Computing (CLOUD), 2010 IEEE 3rd International Conference on (pp. 482-489). IEEE*. <https://doi.org/10.1109/TR.1987.5222336>
- Sahner, R. A. and Trivedi, K. S. (1987). Reliability modeling using sharpe. *Reliability, IEEE Transactions on*, 36(2), 186-193. <https://doi.org/10.1109/TR.1987.5222336>
- Silva, B., Callou, G., Tavares, E. A. G., Maciel, P., de Figueiredo, J. C., Souza, E., Araujo, C. J. M., Magnani, F. and Neves, F. A. S. (2012). Astro: An integrated environment for dependability and sustainability evaluation. *Sustainable Computing: Informatics and Systems*, 2, 1-31. <https://doi.org/10.1016/j.suscom.2012.10.004>
- Valeriana, A. M. A. and Marcelo, R. (2008). *Monitoramento do Protocolo RTSP (Real Time Streaming Protocol) utilizando NTop (Network Top)*. Centro Brasileiro de Pesquisas Fisicas.