Modeling and Evaluating NFV-Enabled Network Services under Different Availability Modes

Besmir Tola*, Gianfranco Nencioni†, Bjarne E. Helvik*, and Yuming Jiang*
*NTNU-Norwegian University of Science and Technology, Norway
†University of Stavanger, Norway
Email:*{besmir.tola, bjarne.e.helvik, yuming.jiang}@ntnu.no, †gianfranco.nencioni@uis.no

***Abstract*—Network and Telecom operators are continuously embracing the adoption of Network Function Virtualization (NFV) as a means to provide more agile, flexible and cost-efficient services. Many telecommunication services need to possess carrier-grade quality of service; therefore, future NFV-enabled telecom services should present high levels of availability. In this paper, we present a composed availability model of NFV-enabled network services under different availability modes, namely Standard Availability, Cold Protection, and Hot Protection. We model and analyze the availability of NFV-enabled network services for each of the availability modes aiming at finding the best redundancy configuration to ensure carrier-grade quality. Through discrete-event simulation analysis we are able to identify the most suitable redundancy configuration for each of the availability modes.**

***Index Terms*—NFV, Service Function Chaining, Availability Modes, Cold Protection, Hot Protection.**

I. Introduction

Network Function Virtualization (NFV) is expected to change the way operators provide their services by entailing greater network programmability, dynamic service delivery, and service automation. Through decoupling network functions into software and hardware, NFV aims at replacing legacy network functions with virtualized instances, called Virtual Network Functions (VNFs) [1], running as software into commodity servers. By linking together many VNFs, NFV provides the ability to define specialized services as an ordered set of network functions (e.g., firewalls, intrusion protection etc.), commonly referred to as Service Function Chain (SFC).

The VNFs are network function software implementations running over an NFV infrastructure (NFVI), which provides, through a virtualisation layer commonly referred to as Virtual Machine Monitor (VMM) or hypervisor, the virtual resources needed to support the execution of VNFs. The management and orchestration of resources and services is performed by the NFV-Management and Orchestration (NFV-MANO), which represents a logically central entity in charge of service lifecycle operations. The NFV-MANO is composed of three main components: Virtual Infrastructure Manager (VIM), VNF Manager (VNFM), and NFV Orchestrator (NFVO).

The transition to NFV deployments introduces additional resilience challenges which may threaten the benefits that NFV architectures embrace [2]. In addition, NFV-enabled telecommunication services are expected to fulfill very strict carrier-grade availability requirements, i.e., five-nines or more [3]. As a result, NFV resilience challenges have drained significant attention from both academia and industry research. To this end, ETSI has provided several guidelines regarding reliability concepts and requirements [4] (and the references within).

Server virtualization represents the core enabling technology for NFV. The authors of [5] paved the way of availability modelling involving virtualized systems with multiple failure modes. Using fault-tree analysis and continuous-time Markov chains (CTMC), they perform a sensitivity analysis for the system performability, i.e., performance and reliability, and extend the analysis for different scalability considerations in [6], [7]. Zhang *et al.* [8] and Dantas *et al.* [9] use a combination of CTMC and Reliability Block Diagram (RBD) approaches to represent and evaluate the dependability of virtualized systems and cloud computing infrastructure, respectively.

An availability model of a virtualized Evolved Packet Core is presented in [10]. Using Stochastic Activity Networks (SANs), the authors assess the system availability in case of multiple and catastrophic failure events since similar events may seriously impact the system availability. In [11], the authors propose a two-level model and evaluate the availability of an SFC deployed in an NFV architecture. By merging RBDs and Stochastic Reward Nets (SRNs) they perform a sensitivity analysis to identify critical parameters. Similarly, in [12], they extend the analysis by including the VIM functionality.

In this paper, we propose an availability model which distinctively to the previous works considers multiple availability modes featuring different fault recovery mechanisms. The considered availability modes include Standard Availability (SA), Cold Protection (CP), and Hot Protection (HP), where each mode can be suitable for different service-level availability requirements. Furthermore, we investigate the impact of redundancy configuration and protection schemes on ensuring a carrier-grade level of service dependability. The availability model is implemented by using two formalisms: i) Replicate/Join, a state sharing composition model that captures the dependencies among components, and ii) the Stochastic Activity Networks (SAN), suitable for describing the failure dynamics of the individual components.

The paper is structured as follows. Section II illustrates the proposed service availability model. Section III presents the salient features of the different availability modes. The SAN models of the individual components are presented in Section IV. Numerical results of the simulation analysis for each of the availability modes are presented in Section V.

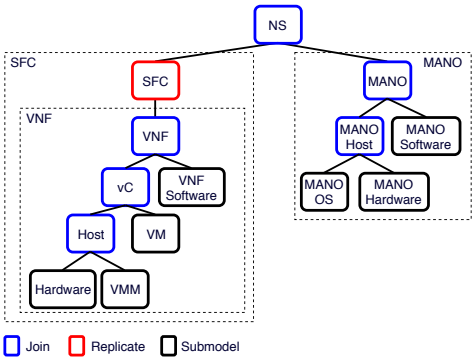

Fig. 1. Network Service SAN model using Replicate/Join formalism.

Finally, Section VI concludes the paper by highlighting the most important insights.

II. Availability Model

In this section, the composed model used to evaluate service availability is presented. The model is implemented through a Replicate/Join formalism by using the Möbius software tool [13]. The formalism enables the modeler to compose a model in the form of a tree, where each leaf node represents a system submodel and each non-leaf node can be a Join or Replicate node. A Join node is a state-sharing node used to compose two or more submodels, whereas a Replicate node is used to compose submodel replicas.

The delivery of an NFV-enabled network service results from the interaction of the SFC (as an ordered sequence of VNFs composing the service) and the MANO (which deploys, instantiates and manages the service lifecycle). While it is argued that a MANO failure shall not affect existing VNFs [14], as specified in [4] and highlighted by the authors of [15], the MANO actually plays a critical role in ensuring the VNF's resiliency. Aligned with [15], we consider the service is available when both SFC and MANO are available.

Fig. 1 depicts the composed service model. We assume a VNF is deployed through a hypervisor-based virtualization running directly on hardware, i.e., bare-metal virtualisation. In addition, we assume a Virtual Machine (VM) is dedicated to a single VNF. Therefore, the model is composed of the *Host* subsystem which symbolizes an NFVI server consisting of the computing, storage and network hardware resources. This level joins two submodels representing the *Hardware* and *VMM* components. The intermediate level represents the virtual Container (*vC*) providing the virtualized environment where a VNF is executed by joining the *VM submodel* with the *Host* level. Lastly, the *VNF* level joins the *vC* and the *VNF software* submodels.

A high-level architecture of a widely referenced solution, namely Open Baton [16], is used as a reference for the MANO model. A common deployment involves a commodity server running its own OS, e.g., Linux-based kernel OS, and the installation of the various MANO software component's packages, e.g., NFVO, VNFM etc. For simplicity, we consider the MANO software as a single component where the failure of any of its software packages causes a failure of the MANO functionality. Therefore, on the *Host* level, the MANO model is composed by joining the *MANO Hardware* with the *MANO OS*. On the higher level, the *MANO software* is joined with the *MANO Host* node. When any of the elements fails, the MANO becomes unavailable.

The SFC consists of an ordered sequence of VNFs. Therefore by replicating the same VNF non-leaf node, through the SFC replicate node, we obtain a representative model of a SFC where the number of replicas indicate the number of VNFs composing the chain. The SFC, being a replicate node, allows state-sharing among the different replicas. We assume that each replica, i.e., VNF, fails independently. Thus, by not sharing any state among the VNFs, we simulate such independence. By joining the SFC and the MANO subsystems, i.e., the top join node, the model represents a series configuration where each subsystem (MANO, VNF_1, VNF_2,..., VNF_O) needs to be working in order for the service to be available.

From a modeling perspective, there are similarities among the submodels composing the VNF model and the MANO model. Specifically, the same submodel, with related failure and repair parameters, is used to describe the failure dynamics of both the *VNF* and the *MANO software* components. The same submodel is used for the *VNF* and *MANO hardware* components, and so is the submodel used for the VNF *VMM* and *MANO OS* components.

Each component's behavior dynamics are captured through a specific SAN submodel which we introduce in more detail in Section IV.

III. Availability Modes

The VNF availability modes we investigate are Standard Availability (SA), Cold Protection (CP) and Hot Protection (HP). The former one is regarded as a baseline mode since it features the simplest recovery procedure. Whereas, the later ones, driven from typical implementations using virtualization technologies (see for example [17]), embody more sophisticated recovery strategies.

In this paper, we consider that each VNF composing the SFC is deployed as a load-sharing cluster where several VNF units, making up the cluster, are needed to satisfy a certain load demand. The VNF is considered to be operational if at least N out of the K units are working. Therefore, the cluster itself is able to provide protection for up to $K - N$ simultaneous failures. On top of load-sharing we consider an additional level of protection through our availability modes where M redundant units provide protection to the load-sharing cluster. As a result, by tuning the K, N and M parameters we investigate different redundancy configurations.

Similarly, for the MANO is implemented as a load-sharing cluster where R defines the number of MANO units and the MANO is operational if at least S out of the R units are up.

A. Standard Availability (SA)

The SA mode represents a case where a VNF does not rely on any redundancy mechanism. Failures on the different levels, which are discussed in Section II and shown in Fig. 1,

are detected through heartbeat mechanisms. Once a failure on the host level is detected, the recovery process requires the summoning of an operator to execute a manual replacing or repairing of the failed component. Whereas, in case a failure on a software level is detected, i.e., VMM, VM or VNF software, the recovery follows a two-step procedure. At first, an automatic restart/reboot of the failed component is triggered by the MANO and only if the component restart/reboot does not recover the service, a hard repair, i.e., patch fixing or software updating, is performed.

B. Cold Protection (CP)

The CP mode consists of a solution where the aim is to minimize the downtime caused by a failure on the host level.The CP mode leverages multiple hosts configured as a cluster. Specifically, for a primary host, there is a secondary host ready to takeover the VMs affected by a primary-host failure. A primary host sees the secondary one by exchanging heartbeat messages. In case of failures within the host level, i.e., hardware or VMM, the CP mode features an automatic restart of the affected VMs, activated by the MANO, by performing a similar to "live migration" procedure, on the secondary host. In case the failure is experienced within the VM/VNF software level, the MANO restarts the affected VM on the same host. Similar to the SA mode, in case a VM/VNF software restart does not successfully recover the service, a hard repair is executed. Note that the redundancy is provided only on the host level and the redundancy restoration is performed by either replacing/repairing the failed hardware component or by performing a soft repair followed by an eventual hard repair of the VMM in case the former does not restore the redundancy.

C. Hot Protection (HP)

Hypervisor-based Fault Tolerance represents a powerful technology promising continuous service availability [17]. Similarly to this solution, in the HP-mode implementation a VM, i.e., primary VM, is protected by creating and synchronizing a secondary VM, that is identical and continuously available in a different host. The secondary VM is ready to take over in the event of a failure caused in the host level, i.e, hardware and VMM, VM or VNF application level. In this mode, the failure detection uses a combination of heartbeat messages and logging traffic to monitor the status of the primary VM. In case the logging traffic and/or heartbeat miss or exceed a specific timeout interval (order of seconds), a failure is detected. Once the failure is detected, an automatic and seamless failover to the secondary VM is performed. The redundancy restoration is carried out similarly to failure recovery in SA. When the hardware fails, a manual repair is preformed. In case the VMM, VM or the VNF software fails, the same two-step procedure of SA and CP is performed.

Driven by the fact that HP provides a VM fault-tolerant solution that promises service continuity, we consider in the remaining that the MANO adopts only the HP mode.

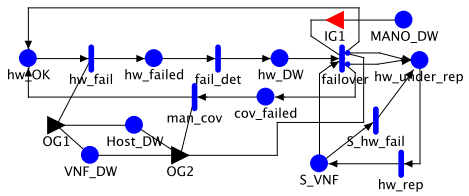

Fig. 2. Hardware SAN availability models.

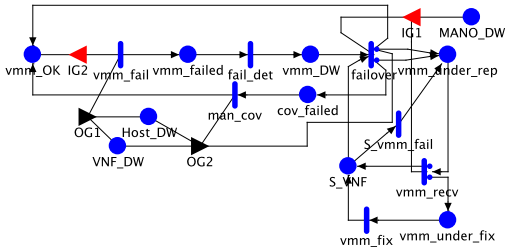

Fig. 3. VMM SAN availability models.

IV. SAN Submodels

In this section, the SAN models of the elements, composing the service, for each of the availability modes are illustrated. A SAN model is composed of *places, activities, input gates*, and *output gates* primitives. Through activity firings and following specific distributions, tokens are moved among places resulting in system state changes. Input and output gates enable and control activity firings.

The availability modes differ from each other only on the recovery mechanisms. In particular, the HP mode includes all the SAN primitives utilized in the SA and CP modes. Therefore, due to space limitations we illustrate only the HP mode since the two others may be induced from the HP mode. Note that the MANO submodels are identical to the VNF submodels as specified at the end of Section II hence, we avoid illustrating.

A. Hardware Submodel

The *hardware* SAN availability model is depicted in Fig. 2. The model comprises the following shared places, i.e., states shared among the different *hardware, VMM, VM and VNF software* submodels:

- *VNF_DW* indicate the number of failed VNF units;
- *Host_DW* represents the number of hosts that are down;
- *MANO_DW* represents the status of the MANO. In case more than $R-S$ tokens are present, the MANO is down;
- *S_VNF* is populated with M tokens and represents the secondary VNF redundant units ready to takeover the service from the failed VNFs;

In addition, the following output gates enable token marking movements for the shared places:

- *IG1* enables the failover operation activity. Only in case there are less than $R-S$ tokens in *MANO_DW*, i.e., the MANO cluster is operational, the failover is performed;
- *OG1/OG2*, when the *hw_fail/hw_rep* timed activity is completed, the output gate increases/decreases with 1 token the places *Host_DW* and *VNF_DW*;

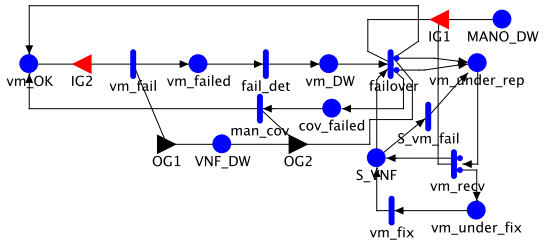

Fig. 4. VM SAN availability model.

The following places define the component operational status:

- *hw_OK* corresponds to the fully working state of the hardware components and is initialized with K tokens;
- *hw_failed* is populated with 1 token in case a hardware component fails, 0 otherwise;
- *hw_DW* represents the detection of a hardware failure;
- *hw_under_rep* represents the number of hardware components undergoing a repair process;
- *cov_failed* defines the state where the failover procedure fails and a manual coverage is required;

The places are connected by mean of the following negative exponentially distributed (n.e.d.) timed activities:

- *hw_fail* and *hw_rep* represent the hardware failure and repair events with rates λ_{hw} and μ_{hw}, respectively;
- *fail_det* represents the failure detection with rate μ_{det};
- *failover* represents the HP failover event with rate μ_{fo}. Since the failover is an automatic procedure, the MANO triggers the recovery procedure. There are two cases, with probability C_{fo} the failover is successful and 1 token is moved into *hw_under_rep* and another token is fetched from *S_VNF* and is moved into *hw_OK*. Whereas, with probability $1 - C_{fo}$ the failover procedure fails and 1 token is placed into *hw_under_rep* and the previous one fetched from spare units is put into *cov_failed*;
- *S_hw_fail* represent the hardware failure event of the redundant host with rate λ_{hw}. The redundant host provides resources to other services as well; therefore, they experience hardware failures similarly to the primary;
- *man_cov* represents the intervention of an operator performing a manual coverage with rate μ_{cov};

B. VMM Submodel

Fig. 3 illustrates the *VMM* SAN availability submodel. Compared to the *Hardware* model, the difference lies on the redundancy restoration identified by the *vmm_recv* timed activity. With probability C_{res}, a VMM restart recovers the service and with probability $1 - C_{res}$ the VMM undergoes a manual fixing. Due to space constraints, we omit further description.

C. VM and VNF Submodels

Fig. 4 illustrates the *VM* submodel. Although apparently similar to the *VMM*, the submodel slightly differs on the fact that the *VM submodel* is an element of a higher level, i.e., vC. Thus, VMs can fail only if their underlying hosts have not failed. To this end, *IG2* enables a VM failure only if the number of tokens in *VNF_DW* are less than K. Similarly, the *VNF software* submodel belongs to the higher level and the relative SAN model is identical to the *VM* model.

TABLE I
MODEL PARAMETERS USED IN THE EVALUATION.

Parameter	Time	Description [mean time to]
$1/\lambda_{hw} = 6.5$	months	next hardware failure
$1/\mu_{hw} = 1$	hour	hardware repair
$1/\mu_{fo} = 5$	secs	VM failover
$1/\mu_{det} = 5$	secs	failure detection
$C_{fo} = 0.95$		VM failover coverage factor
$1/\mu_{mig} = 1$	minute	VM migrate
$C_{mig} = 0.95$		VM migrate coverage factor
$1/\lambda_{vmm} = 4$	months	next VMM failure
$1/\mu_{vmm} = 1$	hour	VMM fix
$1/\mu_{vmm_{res}} = 30$	secs	VMM reset
$1/\lambda_{vm} = 2$	months	next VM failure
$1/\mu_{vm} = 1$	hour	VM hard fix
$1/\mu_{vm_{res}} = 30$	secs	VM reset
$1/\lambda_{sw} = 2$	weeks	next VNF software failure
$1/\mu_{sw} = 1$	hour	VNF software fix
$1/\mu_{sw_{res}} = 15$	secs	VNF software restart
$C_{res} = 0.8$		restart coverage factor
$1/\mu_{\Delta} = 30$	minutes	summon an operator
$1/\lambda_{Msw} = 1$	month	next MANO software failure
$1/\mu_{Msw} = 1$	hour	mean time to MANO software fix
$1/\mu_{MSW_{res}} = 15$	secs	MANO software restart
$1/\lambda_{OS} = 1$	month	next OS failure
$1/\mu_{OS} = 1$	hour	OS fix
$1/\mu_{OS_{res}} = 1$	minute	OS reboot
$1/\mu_{cov} = 30$	minutes	manual coverage
$O = 3$		# VNFs composing the SFC

V. NUMERICAL EVALUATION

In this section we evaluate the different VNF cluster configurations for each of the availability modes. We compute the steady-state service availability for the composed model using discrete-time simulations implemented in Möbius with 95% confidence interval for a one year time simulation. The set of numerical values regarding failure, repair intensities and coverage probabilities, retrieved from previous literature [6], [7], [11], [12], are presented in Table I.

Cluster overprovisioning is an excellent means for providing high level of protection, i.e., providing extra units to cope with multiple simultaneous failures. For this purpose, we define the VNF load-sharing cluster *overprovisioning-ratio* as $\gamma = \frac{K-N}{N}$. We assume that each VNF cluster is composed of $K = 4$ units and vary N so that γ is increased from 0 to 0.25 and 0.5. The same definition and assumption apply to the MANO cluster as well with $\gamma_M = \frac{R-S}{S}$ and $R = 4$.

Table II illustrates the service availability with varying number of redundant units M, overprovisioning-ratio γ and recovery coverage factors for the modes that make use of redundancy, i.e., CP and HP. We observe that for an increasing M there is an almost negligible availability increase irrespective of the availability mode. On the other hand, an increase of the overprovisioning-ratio is associated with up to three orders of magnitude of availability increase hence, suggesting that it is much more beneficial to scale-out a cluster than to provide the same unit(s) in the form of redundant backups. Furthermore, we notice that the HP mode is more sensitive

TABLE II
AVAILABILITY FOR DIFFERENT VNF REDUNDANCY CONFIGURATIONS AND RECOVERY COVERAGE FACTOR.

		Cold Protection		Hot Protection	
γ	M	$C_{mig} = 0.8$	$C_{mig} = 0.99$	$C_{fo} = 0.8$	$C_{fo} = 0.99$
0	1	99.25%	99.30%	99.59%	99.96%
	2	99.26%	99.31%	99.60%	99.98%
	3	99.27%	99.45%	99.61%	99.99%
0.25	1	99.9964%	99.9970%	99.9981%	99.99971%
	2	99.9967%	99.9976%	99.9992%	99.99992%
	3	99.9968%	99.9985%	99.9994%	99.99997%

For all the results $\gamma_M = 0.25$.

TABLE III
EFFECTS OF VNF CLUSTER OVERPROVISIONING ON SERVICE AVAILABILITY FOR DIFFERENT FAILURE INTENSITIES.

Failure Intensities	γ	**Standard Availability**	**Cold Protection**	**Hot Protection**
λ_{ref}	0	98.9%	99.30%	99.88%
	0.25	99.994%	99.997%	99.9997%
	0.5	99.999941%	99.99997%	99.999993%
$10 \cdot \lambda_{ref}$	0	90.35%	93.18%	97.59%
	0.25	99.47%	99.71%	99.80%
	0.5	99.91%	99.98%	99.99%

For all the results $M = 1$ and $\gamma_M = 0.25$.

to coverage factor variations compared to the CP mode. Increasing the robustness of the failover mechanism, i.e., higher coverage, may generate up to one order of magnitude higher availability. The explanation lies within the mode itself since the CP mode exploits a VM migration only for hardware and VMM failure events, whereas the HP mode fully exploits the failover procedure for all kinds of failures.

Table III shows the service availability for each mode when the provisioning ratio is varied. Two cases are considered, one with failure intensities taken from Table I, denoted with λ_{ref}, and the case where failure intensities are $10 \cdot \lambda_{ref}$. We notice that in the former case, only the HP mode achieves a carrier-grade quality (5 nines) when each VNF cluster is overprovisioned with one additional VNF unit. By providing two extra units as the means for protection, all the modes achieve more than 5 nines. On the other hand, for higher failure intensities, none of the modes reaches 5 nines availability.

With respect to the MANO provisioning ratio, Table IV illustrates the results when varying γ_M. We observe that the availability is augmented by one nine when the provisioning ratio is increased from 0 to 0.25, but remains almost unchanged when the ratio becomes higher. Therefore, while overprovisioning of the MANO cluster provides protection to the service, a high overprovisioning does not gain accordingly on the service availability.

VI. CONCLUDING REMARKS

In this paper, an availability model based on SAN composition has been proposed. The model is flexible and can be extended to incorporate even more failure types on both hardware (memory, disk, CPU) and VNF (VNF components) level. A sensitivity analysis aiming at identifying the configuration

TABLE IV
EFFECTS OF MANO CLUSTER OVERPROVISIONING ON SERVICE AVAILABILITY.

γ_M	**Standard Availability**	**Cold Protection**	**Hot Protection**
0	99.97%	99.97%	99.98%
0.25	99.99425%	99.9970%	99.999731%
0.5	99.99428%	99.9971%	99.999732%

For all the results $M = 1$ and $\gamma = 0.25$.

that achieves the so-called "fine-nines" availability has been carried out. Three different protection mechanisms have been investigated and the outcomes show that service availability is sensitive to a correct dimensioning of the VNF and MANO clusters. Increasing the VNF cluster size by one unit coincides with an increase of up to three orders of magnitude of the service availability but a high MANO overprovisioning does not bring a substantial advantage. Moreover, when a Hot Protection mode is configured, the failover robustness, i.e., higher coverage factor, can be exploited to achieve up to one order of magnitude availability boost.

ACKNOWLEDGMENT

This research was funded by the joint EU FP7 Marie Curie Actions Cleansky Project, Contract No. 607584.

REFERENCES

[1] G. N. ETSI, "ETSI GS NFV 002 v1.2.1: Network Functions Virtualisation (NFV); Architectural Framework," 2014.
[2] B. Han, V. Gopalakrishnan, G. Kathirvel, and A. Shaikh, "On the resiliency of virtual network functions," *IEEE Communications Magazine*, vol. 55, no. 7, pp. 152–157, 2017.
[3] R. Swale and D. Collins, *Carrier Grade Voice Over IP*. McGraw Hill Professional, 2013.
[4] I. N. ETSI, "ETSI GR NFV-REL 007 v1.1.2: Network Function Virtualisation (NFV); Reliability; Report on the resilience of NFV-MANO critical capabilities," 2017.
[5] D. S. Kim, F. Machida, and K. S. Trivedi, "Availability modeling and analysis of a virtualized system," in *PRDC'09*. IEEE.
[6] R. d. S. Matos, P. R. Maciel, F. Machida, D. S. Kim, and K. S. Trivedi, "Sensitivity analysis of server virtualized system availability," *IEEE Transactions on Reliability*, vol. 61, no. 4, pp. 994–1006, 2012.
[7] D. S. Kim *et al.*, "Availability modeling and analysis of a virtualized system using stochastic reward nets," in *CIT'16*. IEEE.
[8] X. Zhang, C. Lin, and X. Kong, "Model-driven dependability analysis of virtualization systems," in *ICIS'09*. IEEE, 2009, pp. 199–204.
[9] J. Dantas, R. Matos, J. Araujo, and P. Maciel, "An availability model for eucalyptus platform: An analysis of warm-standy replication mechanism," in *SMC'12*. IEEE, 2012, pp. 1664–1669.
[10] A. Gonzalez *et al.*, "Service availability in the NFV virtualized evolved packet core," in *GLOBECOM, 2015 IEEE*. IEEE.
[11] M. Di Mauro *et al.*, "Service function chaining deployed in an NFV environment: An availability modeling," in *CSCN'17*. IEEE.
[12] ——, "Availability modeling and evaluation of a network service deployed via NFV," in *TWDC'17*. Springer, pp. 31–44.
[13] Möbius: Model-based environment for validation of system reliability, availability, security and performance. [Online]. Available: www.mobius.illinois.edu
[14] I. N. ETSI, "ETSI GS NFV-REL 001 v1. 1.1: Network Functions Virtualisation (NFV); Resiliency Requirements," 2015.
[15] A. J. Gonzalez, G. Nencioni, A. Kamisiński, B. E. Helvik, and P. E. Heegaard, "Dependability of the NFV orchestrator: State of the art and research challenges," *IEEE Communications Surveys & Tutorials*, 2018.
[16] Open Baton: An open source reference implementation of the ETSI NFV MANO specification. [Online]. Available: openbaton.github.io
[17] VMware vSphere Availability. Accessed 2019 Jan. [Online]. Available: docs.vmware.com/en/VMware-vSphere/6.5/vsphere-esxi-vcenter-server-65-availability-guide.pdf