Including Failure Correlation in Availability Modelling of a Software-Defined Backbone Network

Gianfranco Nencioni, Bjarne E. Helvik and Poul E. Heegaard
Department of Information Security and Communication Technology,
NTNU – Norwegian University of Science and Technology, Trondheim, Norway
{gianfranco.nencioni, bjarne.e.helvik, poul.heegaard}@ntnu.no

Software-Defined Networking (SDN) promises to improve the programmability and flexibility of networks, but also brings new challenges that need to be explored. The main objective of this paper is to include failure correlation in a quantitative assessment of the properties of SDN backbone networks to determine whether they can provide similar availability as the traditional IP backbone networks. To achieve this goal, this paper has formalised a two-level availability model that captures the global network connectivity without neglecting the essential details and which includes a failure correlation assessment. The paper proposes a modular and systematic approach for characterising the principal minimal-cut sets in both SDN and traditional networks, and Stochastic Activity Network (SAN) models for characterising the single network elements. To demonstrate the feasibility of the model, an extensive sensitivity analysis has been carried out on a national backbone network.

***Index Terms*—SDN, availability modelling, SAN, failure correlation, dependability.**

I. Introduction

Software-Defined Networking (SDN) is emerging as a new networking paradigm. The potential of this technology is large, but it is a significant challenge to attain the required dependability. The objective of this paper is to provide a comprehensive, holistic model that may be used in dealing with architectural and design issues of SDN backbone networks.

SDN is based on an idea of programmable network devices in which it is assumed that the forwarding plane is decoupled from the control plane [1], [2]. Although programmable networks have been studied for decades, SDN is experiencing a growing success for a number of reasons. Among these are: the expected ability to easily change network protocols and to add new services and applications, reduced CAPEX due to use of commodity computing facilities, less costly network devices, and reduced OPEX due to better monitoring, and ease of maintenance and operation. It is also expected that the SDN will foster future network innovation and that the networks more easily may serve as an abstraction for applications. For a further discussion of the potential of SDN, see for instance [3].

SDN represents a significant shift in networking technology. A simplified sketch of the SDN architecture presented in the IRFT RFC 7426 [1] is shown in Figure 1. The control data planes are separated from each other. In addition, the control plane is logically centralised in a software-based controller, while the data plane is composed of simple network devices that forward packets.

When SDNs are introduced into the backbone network, these networks must have a dependability that is at least as good as in the current networks. Taken into account that it is the core element of the modern society's infrastructure, it should have the potential to become far better [4]. This is a strong requirement, as the current technology was designed to be inherently survivable due to its distributed nature [5] and has been constantly improved for several decades.

Relative to the importance, strict requirement and the challenges in achieving it, the dependability (e.g., measured as the availability and reliability) of SDN has received little attention. It has been suggested that the centralised and automated management may improve the dependability (e.g., [6]). However, several questions have been raised by network operators and researchers concerning the dependability issues introduced by SDN. These are due to the logical centralisation, increased complexity, interdependence between the forwarding plane and the control plane, and other factors [7], [8]. In addition to the challenges posed by centralisation and the potential increased complexity, SDN also introduces a structural challenge. In addition to traffic forwarding paths, there must be paths between the forwarding elements and control sites, which results in an increased number of minimal-cut sets [9].

Most network dependability models assume independent failing of network elements and and it is often assumed that there are only link failures. In assessing the dependability of backbone network, this kind of assumptions are highly unrealistic and will result in too optimistic predictions. The objective of this paper is to establish a modelling approach that includes both the structural (static) as well as the dynamic (temporal) aspects of failing and recovery of network elements, where also interdependencies due to geographical and physical proximity, common operation and maintenance, misconfiguration, compatibility issues, homogeneous equipment and traffic migration are taken into account. Furthermore, the model should include all types of faults and be applicable to assess both a conventional network and in an SDN network to enable a comparison and an identification of the sensitivity to parameter changes to identify the potential dependability bottlenecks of an SDN backbone network.

To achieve the above objective, an original modelling approach is introduced. The two-level modelling of networks, which hereto has been based in independence between network elements, is extended by regarding the dominant minimal-cut sets as subjects for detailed models an level two, rather

than individual elements. This enable us to include well founded models of the correlations and achieve more realistic predictions for network availability.

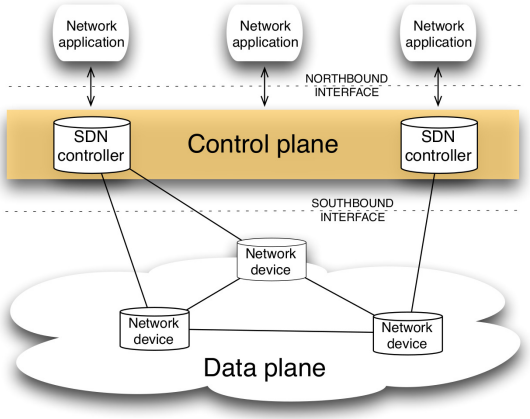


Fig. 1: SDN architecture (excluding the management plane).

The remainder of this paper is organised as follows: in Section II, we briefly review the related literature. Then, the two-level hierarchical model to evaluate the network availability is presented in the sections III, IV, V, and VI. The modelling approach is general, but a case is used for illustration. The core idea when interdependencies are accounted for, is to use the lowest cardinality cut sets from a structural model as a basis for dynamic models capturing the dependencies. Finally, in Section VII, we discuss the results of the sensitivity analysis based on the selected set of parameters that will potentially affect the dependability of SDN. Section VIII summarises and gives some concluding remarks.

II. Related works

Dependability is an important issue to make SDN a success [10]. However, so far the focus has been on design, see for instance [11], [12] and references in the latter, and less on modelling the dependability in SDN. Some pioneering works have been done in order to assure that SDN might meet the carrier grade dependability requirements [13], [14], which are followed by later work dealing with more complicated failure scenarios, for instance [15]. These works are focused on the forwarding part of the network.

In [8], the potential dependability challenges with SDN are discussed which are partially illustrated by a small case study with a structural analysis of SDN enabled network. In [16], probabilistic model checking is used to investigate the probabilities of different kinds of failures, i.e. shutdown caused by the different SDN architectural elements (packet forwarding, application, northbound/southbound interface, and controller). In [17], dependability modelling of SDN is assessed by developing an availability/reliability model of a hierarchical SDN controller.

The above approaches are not considering the network topology to evaluate the dependability of SDN. In [18], this aspect has been investigated by proposing a tool to assess the reliability of SDN by network failure injection.

Both [19] and [20] propose availability models of SDN. In [19], an hierarchical availability model is proposed using Reliability Graph and Stochastic Reward Net. In [20], a stochastic model is proposed by using Stochastic Reward Net. In both papers, the approaches are used in a very small SDN case study (the topology is different in the two papers but they both have three switches and one controller) to compute the availability of a particular application (storage in [19] and virtual machine in [20]).

In none of above papers, the overall availability of SDN is evaluated in a application-agnostic way by considering both the network topology and the sources of failure in the network elements and compared to the traditional network.

In [9], a two-level availability model is presented, which is an extension of the model used as an example in [21]. The model allows to study how the SDN paradigm modifies the overall availability of the network relative to the traditional distributed IP network and analyse which factors dominate in this new scenario.

This paper further extends the above two-level availability model for evaluating the overall availability of a SDN backbone. The model in [9] assumed independent failing of network elements, thus it models the dynamic aspects of the individual network elements. Instead, the model proposed in this paper allows to consider the *failure correlation* among the network elements in both traditional network and SDN by modelling together sets of network elements. For this reason, the model proposed in this paper permits to take into account several failure correlation sources, such as geographical and physical proximity, common operation and maintenance, misconfiguration, compatibility issues, homogeneous equipment, and traffic migration.

III. Two-level availability model

In this section a two-level model is introduced to evaluate the dependability of SDN in a global backbone. In particular, the dependability is measured in terms of steady-state availability, henceforth referred to as availability.

A. Hierarchical availability modelling approach

The two-level hierarchical availability modelling approach consists of:

- *Structural* model of the network topology;
- *Dynamic* model of network elements and dominant minimal-cut sets.

The approach seeks to avoid the potential uncontrolled growth in model size by compromising the need for modelling details and at the same time modelling a (very) large scale network. The detailed modelling is necessary to capture the dependencies that exist between network elements. The model also describes multiple failure modes in the network elements and in the controllers. The structural model assumes independence between the components considered, where a component can be either a single network element with one failure mode, or a set of elements that are interdependent and/or experience several failure modes with an advanced recovery strategy. For the dynamic models we can use a

Markov model or Stochastic Petrinet (e.g., Stochastic Reward Network [22]). For the structural model we can use reliability block diagram, fault trees, or structure functions based on minimal-cut or -path sets.

In the following sections, we will demonstrate the use of this approach in a particular case study. In Section IV, we present the structural models of IP legacy systems and SDN backbone networks. In Section V, the failure correlation assessment is carried out. It is based on the previous structural analysis and consists in the identification of principal minimal-cut sets and the related failure correlation sources. In Section VI, for the dynamic level we propose a modular and systematic approach for characterising the principal minimal-cut sets in both SDN and traditional networks. The failure correlation assessment and the approach for modelling the principal minimal-cut sets are the main innovations with respect to the model presented in [9] Within the approach, for obtaining the availability model of the principal minimal-cut sets we propose and combine Stochastic Activity Network (SAN) models of the individual network elements (links, IP routers, SDN switches, and SDN controllers). The SAN model of the individual network elements are based on the Markov models proposed earlier in [9]. Finally, we explain how to integrate the structural and dynamic models in a hierarchical model.

B. Model case study

In this paper, we analyse the availability of a nation-wide backbone network that consists of 10 nodes across 4 cities, and two dual-homed SDN controllers. See Figure 2 for an illustration of the topology. The nodes are located in the four major cities in Norway, Bergen (BRG), Trondheim (TRD), Stavanger (STV), and Oslo (OSL). Each town has duplicated nodes, except Oslo which has four nodes (OSL1 and OSL2). The duplicated nodes are labelled, X_1 and X_2, where X=OSL1, OSL2, BRG, STV, and TRD. In addition to the forwarding nodes, there are two dual-homed SDN controllers (SC_1 and SC_2), which are connected to TRD and OSL1. We have considered two dual-homed SDN controllers based on the deployment consideration in [23], where the impact of the SDN controller deployment (number, connectivity to the transport network, and location) on the network availability is evaluated by using the two-level model proposed in [9]. In [23], the results highlight that from a network operator prospective the best solution for providing an availability similar to the one provided by a traditional IP network is by deploying two dual-homed SDN controllers and that the location of the SDN controllers has a marginal impact.

The objective of the study is to compare the availability of SDN with a traditional IP network with the same topology of network elements (SDN forwarding switches and IP routers). We assume that nodes, links, and controllers in the system may fail. The peering traffic in a city is routed through an access and metro network with a connection to both (all four) nodes in the city. The system is working (up), when all the access and metro networks are connected. Note that for SDN, at least one controller must be reachable from all nodes along a working path.

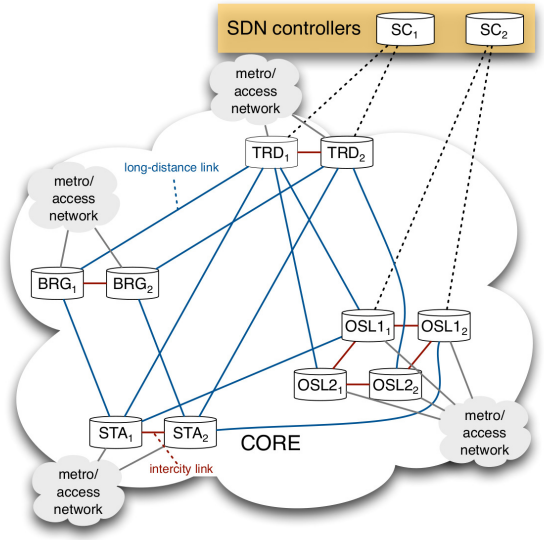


Fig. 2: Nation-wide backbone network

In the following sections, this topology has been used as a case study for presenting the modelling approach. If the structural model is a general and well-known procedure, the failure correlation assessment and the dynamic model have been developed to evaluate the case study but the same procedure can be generalized and applied to more general scenarios. In the related section, we will explain how to generalize the procedures.

IV. Structural model

As already introduced, one of the consequences of moving the control logic from distributed to centralised is the increase in "connectivity" required to consider the network available. For this reason we focus on the dependability issues for the control plane by investigating the reactive SDN mode. More formally, given a traffic that needs to be routed from an origin node o to a destination node d, the following connections must be considered in SDN:

- *flow triggering*: on arrival of a new flow, a path for the trigger message that should be sent from o to the SDN controller;
- *network state update and route directives*: a path from the SDN controller to each node in the path from o to d;
- *forwarding*: forwarding path from o to d.

The first two connections are related to the *control plane* in SDN, they concern about the connectivity among the controller and the nodes in the data network. The last connection is associated to the *data plane* and concerns about the connectivity of the forwarding nodes.

For traditional (legacy) IP networks, the structure of the data plane and control plane is the same, and identical to the structure of the data plane in SDN.

The critical parts of the connection between the traffic origin and destination (and between the controller and any network node in SDN) can be determined using structural analysis

based on either *minimal-cut sets* or *minimal-path sets* [24]. In this paper we use minimal-cut sets:

Definition 1: Minimal-cut set - A system is failed, if and only if, all the subsystems in a minimal-cut set are failed, even if all the other subsystems that are not in the set are working.

The minimal-cut sets form the basis for a *structure function*.

Definition 2: Structure function - Each max-term of the structure function expressed in a minimal product-of-sums form corresponds to a minimal-cut set.

The structural analysis for all the possible connections in the SDN case study, shows that the total number of minimal-cut sets S is $\|S\| = 2916$. The cardinality $c_j = \|s_j\|$ of each of the minimal-cut sets, $j = 1, \cdots, 2916$ is given in Table I. Each column contains the number of sets that is $C_k = \|\{s_j \in S | c_j = k\}\|$, $k = 1, \cdots, 13$. The table compares the minimal-cut sets of SDN with a conventional IP network where the control plane is embedded in the nodes,

The number of minimal-cut sets with cardinality one is equal to zero because traffic sources are at least dual-homed and there are two dual-homed control sites.

The number of minimal-cut sets C_2 increases from 3 to 4 due to the control nodes. Note also that the number of minimal-cut sets C_3 almost doubles. This indicates that in this case study, a significant increase in vulnerability is observed for the SDN case that is not explained solely by the introduction of a control node, but the fact that a controller must be reachable from every node across the backbone in order for the network to work.

V. Failure correlation assessment

In the following section we present the second step in the procedure to model the availability of an SDN backbone network. In the previous section, we determined the minimal-cut sets for the traditional IP network and SDN, now we identify the principal minimal-cut sets and evaluate which are the failure correlation sources among the elements composing the principal minimal-cut sets. For assessing the case study, we have selected the minimal-cut sets with the lower cardinality as principal because we assume a contribution of similar magnitude from each element composing the minimal-cut sets. More generally, the principal minimal-cut sets can be identified in other ways but the failure correlation source should be always considered in this process in order to not neglect minimal-cut sets that have an important contribution to the network unavailability.

A. Identification of principal minimal-cut sets

In our evaluation we have selected and assessed the minimal-cut sets with cardinality equal to 2 and 3. Table II gathers the four types of minimal-cut sets depending on the combination of the kind of network elements (node or link) and highlights in which kind of network (TN: traditional network, F-SDN: forwarding part of SDN, C-SDN: control part of SDN) the minimal-cut sets belong.

The first type of selected minimal-cut sets, $\{n, n\}$, is the only one with cardinality equal to 2 and is composed by two nodes belonging to the same city or the two SDN controllers.

The second type, $\{n, n, n\}$, is composed by three nodes in three different cities or by the SDN controller and the two nodes which the other SDN controller is attached to. They are due to the topology of the backbone network or the dual-homing of the SDN controllers.

The third type, $\{n, n, l\}$, is composed by two nodes in different cities and a link that connects the other node of one of these cities to one node in another city. Furthermore, this type can also be composed by one SDN controller, one node in the city which the other SDN controller is connected to, and the link from the other node of this city and the SDN controller.

The forth type, $\{n, l, l\}$, is composed by one node and two links attached to the other node to the same city or by one SDN controller and the two links connected to the other SDN controller.

The first type of minimal-cut sets is motivated by the redundancy, i.e. the number of nodes in the city or the number of SDN controllers.

In the other types, the minimal-cut sets are related the topology of the backbone network in both traditional and SDN, instead in the control part of SDN they are due to the dual-homing of the SDN controllers.

B. Evaluation of failure correlation sources

In this work we have considered the following failure correlation sources:

- **Geographical Proximity (GEO)**: a small distance among the network elements (i.e. same city or area) triggers a common sensitivity to bad weather and natural disasters;
- **Physical Proximity (PHY)**: some network elements are adjacent (e.g. a router and a link) causing a strong failure correlation (e.g. blackout);
- **Common O&M (COM)**: there are cases (e.g. routers in the same city/PoP) where the O&M is actually the same in multiple network elements, thus an O&M failure lets all the network elements to fail;
- **Misconfiguration (MIS)**: there are elements (e.g. SDN switches in the same city/PoP or SDN controller) that share the same configuration or have a correlated logic;
- **Compatibility Issue (CIS)**: a simultaneous (SW or HW) failure on multiple network elements due to incompatibility issues among them;
- **Homogeneous Equipment (HEQ)**: if a (SW or HW) failure happens in a network element, another element with the same (SW or HW) equipment may likely fail as well;
- **Traffic Migration (TMI)**: when a network element (e.g. an edge router or SDN controller) fails, it could happen that the replacement network element is not able to take over.

VI. Dynamic model of principal minimal-cut sets

For modelling the availability of the minimal-cuts sets, we used a modular and systematic approach similar to the one proposed in [25]. The approach is composed of two steps, (i)

TABLE I: Distribution of cardinality of the minimal-cut sets for the IP network and SDN

	C_1	C_2	C_3	C_4	C_5	C_6	C_7	C_8	C_9	C_{10}	C_{11}	C_{12}	C_{13}	sum
IP network	0	3	8	91	304	360	356	189	70	13				1394
SDN	0	4	15	107	340	520	780	584	302	170	59	31	4	2916

TABLE II: Principal minimal-cut sets (2 and 3 cardinality) for the different networks

cardinality	type	TN & F-SDN	C-SDN
2	{n,n}	$\{n_{BRG_1}, n_{BRG_2}\}$ $\{n_{STV_1}, n_{STV_2}\}$ $\{n_{TRD_1}, n_{TRD_2}\}$	$\{n_{SC_1}, n_{SC_2}\}$
3	{n,n,n}	$\{n_{BRG_1}, n_{STV_2}, n_{TRD_2}\}$ $\{n_{BRG_2}, n_{STV_1}, n_{TRD_1}\}$	$\{n_{OSL1_1}, n_{OSL1_2}, n_{SC_1}\}$
	{n,n,l}	$\{n_{BRG_1}, n_{STV_2}, l_{TRD_2-BRG_2}\}$ $\{n_{BRG_1}, n_{TRD_2}, l_{STV_2-BRG_2}\}$ $\{n_{BRG_2}, n_{STV_1}, l_{TRD_1-BRG_1}\}$ $\{n_{BRG_2}, n_{TRD_1}, l_{STV_1-BRG_1}\}$	$\{n_{OSL1_1}, n_{SC_1}, l_{OSL1_2-SC_2}\}$ $\{n_{OSL1_2}, n_{SC_1}, l_{OSL1_1-SC_2}\}$ $\{n_{SC_2}, n_{TRD_1}, l_{TRD_2-SC_1}\}$ $\{n_{SC_2}, n_{TRD_2}, l_{TRD_1-SC_1}\}$
	{n,l,l}	$\{n_{BRG_1}, l_{STV_2-BRG_2}, l_{TRD_2-BRG_2}\}$ $\{n_{BRG_2}, l_{STV_1-BRG_1}, l_{TRD_1-BRG_1}\}$	$\{n_{SC_1}, l_{OSL1_1-SC_2}, l_{OSL1_2-SC_2}\}$ $\{n_{SC_2}, l_{TRD_1-SC_1}, l_{TRD_2-SC_1}\}$

derive block models at a high level of abstraction, (ii) build detailed models of the minimal-cut sets by Stochastic Activity Networks (SANs). Two kinds of blocks have been considered: the *component* blocks and the *dependency* blocks. The component blocks consist of the network element models, while dependency blocks describe the failure correlation among the network elements.

This procedure is general and not only related to the presented case study. The model blocks are independent to the network topology. The composition of the block is related to the principal minimal-cut sets of the case study but the approach is generally valid.

A. Component blocks

In the following we introduce the models that constitute the component blocks. We present the SAN models of the network elements: links (which are the same in both SDN and traditional network), traditional IP routers, SDN switches, and SDN controllers.

a) Link (L): The model of a link is assumed to be dominated by physical link failures. Therefore, a simple two-state Markov model could be used. Figure 3 shows the SAN representation. The links are either up or down due to hardware

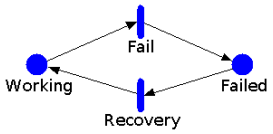


Fig. 3: SAN of a link

failure. We use the same model for both traditional network and SDN. Given failure rate λ_L and repair rate μ_L, the availability of a link is $A_L = \frac{\mu_L}{\lambda_L+\mu_L}$. This model is assumed for each of the link components in the structural model. We don't know the geographical location of the nodes and therefore the distance between them either, which implies that the length of the links connecting the nodes in the network cannot be determined. Hence, in our case studies we have to assume that the link failure rate is not dependent of the link length. Note that in general the failure rate is expected to be proportional to the length of the link.

b) Traditional IP router (R): The SAN model of a traditional router is depicted in Figure 4. In the model we focus on the router functionalities and the related failure sources, each component of the router has not been considered because it would be dependent on a particular router architecture. In any case, we assume 1+1 redundancy of the controller hardware, which is a common best practice in any architecture. Multiple failures are not included in the model since they are assumed to be less frequent and will probably not have significant impact on the expected accuracy of the approach.

The SAN model of the traditional router is composed of eight places:

- *Working* represents the state when the system is fully working and it is initialised with one token;
- *failed_MAN* is equal to 1 when there is a failure of the O&M, 0 otherwise;
- *spare_CHW* represents the state when one of the two redundant control hardware is failed but the other is correctly working;
- *sys_down* is a coverage state and is equal to 1 if there is an unsuccessful activation of the stand-by hardware after a failure (manual recovery).
- *failed_CHW* represents the state when both controllers have a hardware failure;
- *failed_SW* is equal to 1 when there is a software failure, 0 otherwise;
- *failed_FHW* represents the state when there is a permanent hardware failure in the forwarding plane
- *failed_FHWt* represents the presence of a transient hardware failure in the forwarding plane;

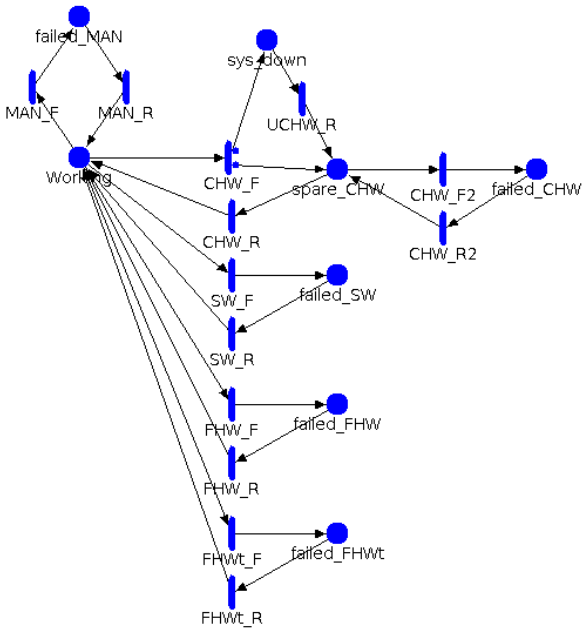


Fig. 4: SAN of a traditional IP router

The router is failed when the token is not in *Working* or *spare_CHW*.

The places are connected by mean of the following timed activities with exponential time distribution:

- *MAN_F* and *MAN_R* represent the failure and the recovery events of the O&M with a rate of λ_{dO} and μ_{dO}, respectively;
- *CHW_F* represents the failure event of the control hardware with a rate of 2 λ_{dC} and there are two cases, with probability C_{dC} a token is put into *spare_CHW*, otherwise (with probability $1 - C_{dC}$) the system is not able to manage the control hardware failure and the system goes down;
- *CHW_F2* represents the failure event of the spare control with a rate of λ_{dC};
- *CHW_R* and *CHW_R2* both represent the recovery of the control hardware with a rate of μ_{dC};
- *UCHW_R* represents the recovery after an unsuccessful activation of the stand-by hardware with a rate of μ_{dUC};
- *SW_F* and *SW_R* represent the failure and the recovery events of the software with a rate of λ_{dS} and μ_{dS}, respectively;
- *FHW_F* and *FHW_R* represent the permanent failure and the recovery events of the forwarding hardware with a rate of λ_{dF} and μ_{dF}, respectively;
- *FHWt_F* and *FHWt_R* represent the transient failure and the recovery events of the forwarding hardware with a rate of λ_{dFt} and μ_{dFt}, respectively;

All the model parameters are defined in Table III. Note that for sake of simplicity we have assumed homogeneous equipment, and link failures independent of the link length. The table includes the numerical values used in the case studies and that are inspired by and taken from several studies [26], [27], [28].

TABLE III: Model parameters for the IP network with numerical values used in the case studies

intensity	[time]	description
$1/\lambda_L = 4$	[months]	expected time to next link failure
$1/\mu_L = 15$	[minutes]	expected time to link repair
$1/\lambda_{dF} = 6$	[months]	expected time to next permanent forwarding hardware failure
$1/\mu_{dF} = 12$	[hours]	expected time to repair permanent forwarding hardware
$1/\lambda_{dFt} = 1$	[week]	expected time to next transient forwarding hardware failure
$1/\mu_{dFt} = 3$	[minutes]	expected time to repair transient forwarding hardware
$1/\lambda_{dC} = 6$	[months]	expected time to next control hardware failure
$1/\mu_{dC} = 12$	[hours]	expected time to repair control hardware
$1/\lambda_{dS} = 1$	[week]	expected time to next software failure
$1/\mu_{dS} = 3$	[minutes]	expected time to software repair
$1/\lambda_{dO} = 1$	[month]	expected time to next O&M failure
$1/\mu_{dO} = 3$	[hours]	expected time to O&M repair
$1/\mu_{dUC} = 8$	[hours]	expected time to recover from uncovered control hardware failure
$C_{dC} = 0.97$		coverage factor

c) SDN switch (S): Figure 5 shows the model of the switch in SDN, which is significantly simpler than the router in a traditional network. The states related to the control hardware failures are not contained in this model, since all the control logic is located in the controller. O&M associated with the SDN switch has been also omitted because we assume that the complexity of the O&M operations done on a single switch is likely to be small relative to a router and globally in the controller. The software is still present but its failure rate will be very low since the functionality is much simpler. Table IV

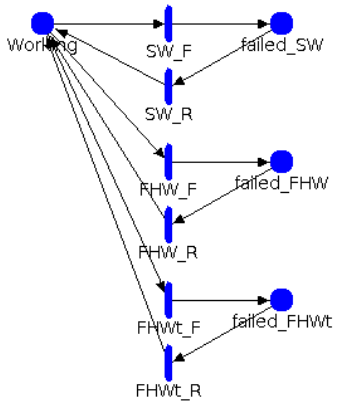


Fig. 5: SAN of a SDN switch

describes the parameters for modelling the SDN switch.

All the parameters of the SDN model are expressed relative to the parameters for the traditional network (Table III). In an SDN switch, the failure/repair intensities of (permanent/transient) hardware failures are the same because failures with the same cause have the same intensities in both models. However, we assume that the software on an SDN switch will be much less complicated than on a traditional IP router because the control logic has been moved to the controllers,

TABLE IV: Model parameters for the SDN switch

intensity	description
$\lambda_F = \lambda_{dF}$	intensity of permanent hardware failures
$\mu_F = \mu_{dF}$	repair intensity of permanent hardware failures
$\lambda_{Ft} = \lambda_{dFt}$	intensity of transient hardware failures
$\mu_{Ft} = \mu_{dFt}$	restoration intensity after transient hardware failures
$\lambda_{sS} = 0$	intensity of software failure

and we have set the failure rate to zero, for the sake of simplicity.

d) SDN controller (C): The SDN controller is modelled with the SAN depicted in Figure 6. We assume that the SDN controller is a cluster of M processors and that the system is working, i.e., has sufficient capacity, if K out of the M processors are working, which means that both software and hardware are working. In the following the term "active processors" is used to identify the working processors. The other main assumptions of the model are:

- single repairman for a hardware failure;
- load dependency of software failure when the system is working, $\lambda_S(N_a) = \lambda_S/N_a$, where N_a is the number of active processors and the meaning of λ_S is explained in more detail in Section VII;
- only processors failed due to hardware failures will be down until the system recovers when the entire system fails;
- load independence of software failure when the system has failed, $\lambda_S(N_a) = \lambda_S$, since the remaining processors are working at the full capacity.

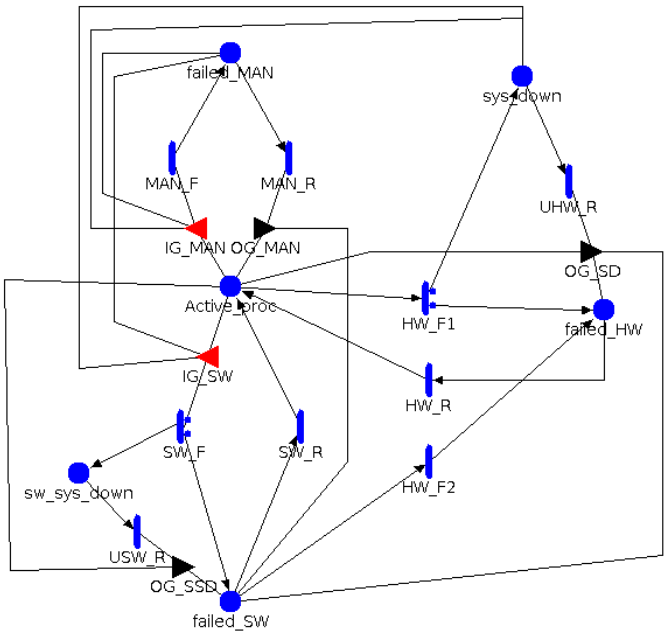


Fig. 6: SAN of SDN controller

The SAN model of the SDN controller is composed of six places:

- *Active_proc* represents the number of active processors and it is initialised to the total number of processors;
- *failed_MAN* is equal to 1 when there is a failure of the O&M, 0 otherwise;
- *failed_SW* represents the number of processors where the software has failed;
- *failed_HW* represents the number of processors where the hardware has failed;
- *sys_down* is a coverage state and is equal to 1 if the hardware failure in one processor forces all the system to be down;
- *sw_sys_down* is a coverage state and is equal to 1 if the software failure in one processor causes the crash of all the processors.

The places are connected by mean of the following timed activities with exponential time distribution:

- *MAN_F* and *MAN_R* represent the failure and the recovery of the O&M with a rate of λ_O and μ_O, respectively;
- *SW_F* represents the failure of the software with a rate of λ_S, if the number of active processors is at least K, or N_a λ_S, otherwise; there are two cases, with probability C_S a token is put into *failed_SW* (if there are enough working processors, the system is still working), otherwise (with probability $1 - C_S$) the system is not able to manage the software failure and the system goes down;
- *SW_R* represents the recovery of the software with a rate of μ_S;
- *USW_R* represents the recovery of the software crash with a rate of μ_{US};
- *HW_F1* represents the failure of the hardware of the active processors with a rate of N_a λ_H and there are two cases, with probability C_C a token is put into *failed_HW* (the hardware is failed but if there are enough working processors, the system is working), otherwise (with probability $1-C_C$) the system is not able to manage the hardware failure and the system goes down (note that if there is already a token in *failed_MAN* or
- *sys_down*, the token is forced to be put in *failed_HW*);
- *HW_F2* represents the failure of the hardware of the processors with a failed software with a rate of N_s λ_H, where N_s is the number of token in *failed_SW*;
- *HW_R* represents the recovery of the hardware with a rate of μ_H;
- *UHW_R* represent the recovery after an unsuccessful activation of the stand-by hardware with a rate of μ_{UH};

Furthermore, the following input and output gates are included:

- *IG_MAN* enables the O&M failure activity only if there are no tokens in *failed_MAN*, *sys_down*, and *sw_sys_down*;
- *IG_SW* enables the software failure activity only if there are no tokens in *failed_MAN*, *sys_down*, *sw_sys_down*, and there are active processors and implies the decrease of the number of active processors;
- *OG_MAN* and *OG_SSD* resets the number of software failures and sets the number of active processors to

TABLE V: Model parameters for the SDN controller

intensity	description
$\lambda_H = \alpha_H\ \lambda_{dC}\ N/K$	intensity of hardware failures
$\mu_H = \mu_{dC}$	hardware repair intensity
$1/\mu_{UH} = 0.5h$	restoration time after uncovered hardware failure
$\lambda_S = \alpha_S\ \lambda_{dS}\ N$	intensity of software failures
$\mu_S = \mu_{dS}$	restoration intensity after software failure
$1/\mu_{US} = 0.5h$	restoration time after uncovered software failure
$\lambda_O = \alpha_O\ \lambda_{dO}\ N$	intensity of O&M failures
$\mu_O = \mu_{dO}$	rectification intensity after O&M failures
$C_H = C_{dC}$	hardware failure coverage factor
$C_S = 0.9$	software failure coverage factor

the total number of processors minus the number of processors with failed hardware;

- *OG_SD* increases the number of failed hardware, resets the number of software failure, and sets the number of active processors to the total number of processors minus the number of processors with failed hardware.

In the proposed model the system is down where the number of tokens in *Active_proc* is lower than K or there is a token in *failed_MAN*, in *sys_down*, or in *sw_sys_down*.

As from the SDN switches, we have not found good data from SDN controllers. Therefore, the parameters for the components in the model of SDN are assumed to be relative to the parameters of the traditional network in Table III. The parameters the SDN controller model are listed in Table V, where the proportionality factors α_H, α_S, and α_O has been varied to study the sensitivity of these parameters, and hence also the uncertainty of your model parameters assumptions, on previous work [9]. In this paper based on the previous outcome we set $\alpha_H = 1$, $\alpha_S = 1$, $\alpha_O = 0.2$, and $\alpha_C = 1$.

Moreover, all failure rates are N-times larger than in the traditional network, where N is the number of network nodes (10 in the national network case study). This is because we assume that the SDN needs roughly the same processing capacity and amount of hardware than in the traditional network. Therefore, the failure intensity is assumed to be proportional to N, and of the same order of magnitude as the total failure intensity of the traditional distributed IP router system. For the hardware failures the total failure intensity is divided by the number of needed processors $K = \lfloor 0.8 \cdot M \rceil$, where $M = N$ is the total number of processors.

B. Dependency blocks

For modelling the principal minimal-cut sets, the component blocks need to be composed and the dependency blocks need to be introduced for considering the failure correlation among the network elements. We define three categories of dependency blocks to reflect the different sources of failure correlation as was presented in Section V:

- *add* - a new failure state is added to the model; the majority of the dependency blocks falls into this category; in the example of Figure 7, the GEO failure state is added between two SDN switches (S1 and S2); note that *IG_GF* enables the GEO failure activity only if both S1 and S2 are working;

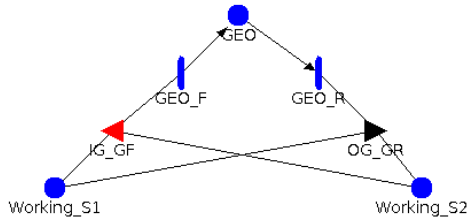


Fig. 7: Adding the GEO failure state between S1 and S2

- *modify* - the model is modified to take into account the interdependency of two (or more) states in different component blocks; Figure 8 shows a modification that reflect the case where a SW failure of an SDN switch (S1 or S2), it likely that TMI could cause a SW failure also in the other SDN switch (S2 or S1); given *SW_F_S1* (or *SW_F_S2*), one case has been added, in which case, if S1 and S2 are both working and with probability $1 - C_{TMI}$, the software failure on S1 (or S2) leads also to a failure in S2 (or S1), otherwise a token will be put into only *failed_SW_S1* (or *failed_SW_S2*);

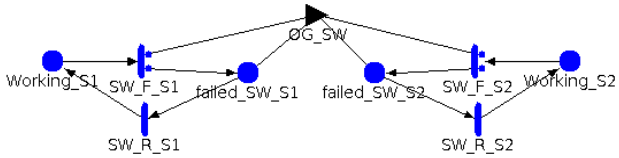


Fig. 8: Modification for considering SW failures caused by TMI between S1 and S2

- *merge* - instead of considering two (or more) separate failure states with the same cause, all these states are merged in on common state; e.g. the *failed_MAN* state of two traditional IP routers can be eliminated and a new failure state can be added as previously presented in Figure 7.

Table VI shows the parameters related to the failure correlation. In [29], it is reported that 10% of failures are multiple, simultaneous failures. Hence, we have considered an intensity of the correlated failures that is ten times lower than the "original" one. In particular, the "original" intensity of the GEO, PHY, MIS, and CIS are related to the permanent forwarding hardware or link (depending on the correlated elements), link, O&M, and SDN controller software, respectively. Since the COM failure is a merge failure correlation, we have considered a failure intensity equal to the intensity of distributed O&M failure. For the GEO and CIS recovery, we have considered a rate three times lower than the "original" rate since they need more time for restoring from the failure source (e.g. blackout) or to discover the origin of the failure. Instead, for the PHY, MIS and COM recovery, the rate for restoring the single element as been considered. Moreover, for sensitivity analysis the multiplicative factors α_X ($X \in \{GEO, PHY, MIS, COM, CIS\}$) and β_Y ($Y \in \{TMI, HEQ\}$) are introduced.

TABLE VI: Model parameters for failure correlation sources

intensity	description
$\lambda_{GEO} = \frac{\alpha_{GEO}\ \lambda_F}{10}$	intensity of geographical-spread failure
$\mu_{GEO} = \mu_F/3$	repair rate after a geographical-spread failure
$\lambda_{PHY} = \alpha_{PHY}\ \lambda_L/10$	intensity of physical-spread failure
$\mu_{PHY} = \mu_L$	repair rate after a physical-spread failure
$\lambda_{COM} = \alpha_{COM}\ \lambda_{dO}$	failure intensity caused by a shared O&M
$\mu_{COM} = \mu_{dO}$	recovery rate from a shared-O&M failure
$\lambda_{MIS} = \alpha_{MIS}\ \lambda_O/10$	misconfiguration failure intensity
$\mu_{MIS} = \mu_O$	intensity to recover from a misconfiguration failure
$\lambda_{CIS} = \alpha_{CIS}\ \lambda_S/10$	failure intensity caused by a compatibility issue among different elements
$\mu_{CIS} = \mu_S/3$	recovery rate from a incompatibility failure
$C_{TMI} = 0.95 + \beta_{TMI}$	coverage factor for considering failures induced by traffic migration
$C_{HEQ} = 0.99 + \beta_{HEQ}$	coverage factor for taking into account failures due to homogeneous equipment

C. Composition of block models

Considering the principal minimal-cut sets and the failure correlation sources highlighted in Section V, we can map the failure correlation sources to the elements composing the 12 kinds of minimal-cut sets (4 for the traditional network, 8 for the SDN) as summarised in Table VII.

TABLE VII: Type of minimal-cut sets for the different networks vs failure correlation source

type	network	GEO	PHY	COM	MIS	CIS	TMI	HEQ
{n,n}	TN	✓		✓			✓	
	F-SDN	✓			✓		✓	
	C-SDN				✓		✓	
{n,n,n}	TN							✓
	F-SDN							✓
	C-SDN	✓				✓		
{n,n,l}	TN	✓						✓
	F-SDN							✓
	C-SDN	✓				✓		
{n,l,l}	TN	✓	✓					
	F-SDN	✓	✓					
	C-SDN	✓	✓					

Basing on the table, for each of the minimal-cut set the block diagram of the related model is presented. In the diagrams, the component blocks are depicted as square boxes, instead the dependency as rounded boxes. The dependency blocks have different line styles depending on the category: solid line for "add", dashed line for "modify", and bold line for "merge".

Note that further details on the the actual implementation in Möbius Tool [30] of the SAN models are reported in [31].

1) $\{n, n\}$ **in traditional network**, the two routers are in the same city (*GEO*), share the O&M (*COM*), and if one fails all the traffic is managed by the other one (*TMI*);

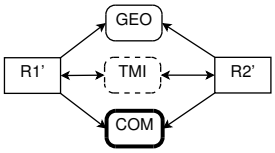


Fig. 9: Block diagram of $\{n, n\}$ in TN

2) $\{n, n\}$ **in SDN (forwarding part)**, the two SDN switches are in the same city (*GEO*), if one fails all the traffic is managed by the other one (*TMI*), and share a common configuration (*MIS*);

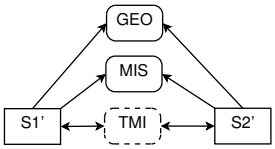


Fig. 10: Block diagram of $\{n, n\}$ in F-SDN

3) $\{n, n\}$ **in SDN (control part)**, the two SDN controllers share a common configuration (*MIS*) and if one fails the other one takes over the control (*TMI*);

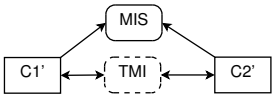


Fig. 11: Block diagram of $\{n, n\}$ in C-SDN

4) $\{n, n, n\}$ **in traditional network**, the three routers have both HW and SW homogeneous equipment (*HEQ*);

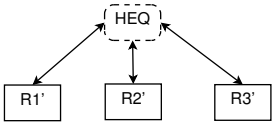


Fig. 12: Block diagram of $\{n, n, n\}$ in TN

5) $\{n, n, n\}$ **in SDN (forwarding part)**, the three SDN switches have mainly HW homogeneous equipment (*HEQ*);

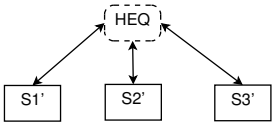


Fig. 13: Block diagram of $\{n, n, n\}$ in F-SDN

6) $\{n, n, n\}$ **in SDN (control part)**, the two SDN switches are in the same city (*GEO*), instead the controller and the switches can have compatibility issues (CIS);

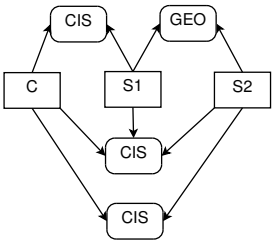


Fig. 14: Block diagram of $\{n, n, n\}$ in C-SDN

7) $\{n, n, l\}$ **in traditional network**, one router and the link are in the same city (*GEO*) and the two routers have homogeneous equipment (*HEQ*);

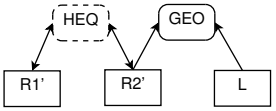


Fig. 15: Block diagram of $\{n, n, l\}$ in TN

8) $\{n, n, l\}$ **in SDN (forwarding part)**, one SDN switch and the link are in the same city (*GEO*) and the two SDN switches have homogeneous equipment (*HEQ*);

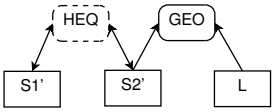


Fig. 16: Block diagram of $\{n, n, l\}$ in F-SDN

9) $\{n, n, l\}$ **in SDN (control part)**, the SDN switch and the link are in the same city (*GEO*), instead the controller and the switch can have compatibility issues (CIS);

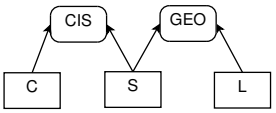


Fig. 17: Block diagram of $\{n, n, l\}$ in C-SDN

10) $\{n, l, l\}$ **in traditional network**, the two links are connected to the same router (*PHY*) and the router and the two links are in the same city (*GEO*);

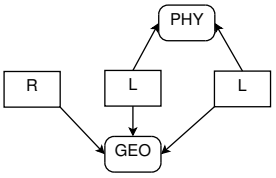


Fig. 18: Block diagram of $\{n, l, l\}$ in TN

11) $\{n, l, l\}$ **in SDN (forwarding part)**, the two links are connected to the same SDN switch (*PHY*) and the SDN switch and the two links are in the same city (*GEO*);

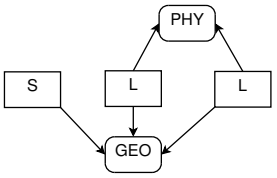


Fig. 19: Block diagram of $\{n, l, l\}$ in F-SDN

12) $\{n, l, l\}$ **in SDN (control part)**, the two links are connected to the same SDN switch (*GEO*, *PHY*), instead the SDN controller is independent.

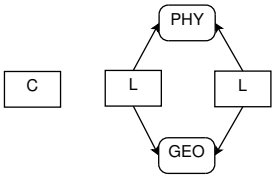


Fig. 20: Block diagram of $\{n, l, l\}$ in C-SDN

Note that we did not consider the homogeneous equipment in the case 1, 2, 3, 6, 7, and 8 because we assumed that its correlation contribution is negligible (it is dominated by TMI). Note that in the case 6 the failure correlation sources are different than that in the case 2 because in the case 2 the SDN switches are the only ones connected to the metro/access network instead in the case 6 there are two more ($OSL2_1$ and $OSL2_2$).

D. Merging the two levels to evaluate network availability

The next step is to obtain the overall network availability by merging the structure model and the principal minimal-cut defined in Sections IV and V with the availability of the principal minimal-cut sets computed by using the SAN models in Section VI.

The *inclusion-exclusion principle*, which is a technique to obtain the elements in the union of finite sets, is applied. Using the inclusion-exclusion principle on the structure function, we can write the network unavailability as the probability of the union of all the minimal-cut sets:

$$U_N = P\left(\bigcup_{i=1}^{||S||} s_i\right) = \sum_{k=1}^{||S||} (-1)^{k-1} \sum_{\substack{\emptyset \neq I \subseteq S \\ |I|=k}} P\left(\bigcap_{i \in I} s_i\right), \quad (1)$$

where $s_i \in S$ are the minimal-cut sets (see Section IV), and $P(s_i)$ is the probability of set s_i. Since we model the principal minimal-cut sets by using the modular and systematic approach proposed in this section, in the inclusion-exclusion procedure we consider the minimal-cut sets as a single independent entity.

VII. Numerical evaluation

The target of this section is to evaluate how the failure correlation affects the availability of both traditional network and SDN. In particular, we investigate the impact of the different failure correlation sources on the overall availability

in both traditional network and SDN. For this purpose we use the α_X factors where $X = GEO, PHY, COM, MIS, CIS$ and β_Y addends where $Y = TMI, HEQ$ (see Table VI), which affect the intensity of the related failure correlation sources, and are defined as follows:

- $\alpha_{GEO} = 10\ \lambda_{GEO}/\lambda_F$;
- $\alpha_{PHY} = 10\ \lambda_{PHY}/\lambda_L$;
- $\alpha_{COM} = \lambda_{COM}/\lambda_{dO}$;
- $\alpha_{MIS} = 10\ \lambda_{MIS}/\lambda_O = \dfrac{10\ \lambda_{MIS}}{10 \cdot 0.2 \cdot \lambda_{dO}} = 5\ \dfrac{\lambda_{MIS}}{\lambda_{dO}}$;
- $\alpha_{CIS} = 10\ \lambda_{CIS}/\lambda_S$;
- $\beta_{TMI} = C_{TMI} - 0.95$;
- $\beta_{HEQ} = C_{HEQ} - 0.99$.

The evaluation has been carried out by realising a simulation on Möbius Tool [30] and considering a wide range of scaling factor, spanning: $\alpha_X \in \{10^i\}$ with $i = -2, \cdots, 2$, $\beta_{TMI} \in \{-0.05, -0.02, 0, 0.02, 0.05\}$, and $\beta_{HEQ} \in \{-0.01, 0, 0.01\}$. Note that further details on the the settings of the simulation on Möbius are included in [31].

Figures 21 show the impact of the different sources of failure correlation on the unavailability of both traditional network and SDN. The figures illustrates the sensitivity to the individual failure correlation: we can notice that the failure correlation has a significant contribution to the network unavailability when $\alpha_X \geq 0$, but as long as the failure correlation is less than the reference value, i.e. $\alpha_X < 0$, the effect is moderate.

The greatest impact on the network unavailability of SDN is caused by GEO and TMI correlation sources with a gap of 0.07 and 0.05, respectively (see Figures 21(a) and 21(e)). This behaviour is likely due the fact that GEO and TMI affect the minimal-cut sets with the lowest cardinality (see Table VII). Moreover, GEO has an impact in most of the principal minimal-cut sets and TMI is the most relevant joint failure correlation source.

At the same time, GEO and TMI have a low impact on the unavailability of the traditional network, which is instead affected by PHY and COM failure correlation sources with a gap of 0.02 and 0.04, respectively (see Figures 21(b) and 21(c)). In both traditional network and SDN, PHY has the same impact because it is affecting only the links, which are the common element between the legacy and SDN networks. The impact of COM on traditional network unavailability is higher than the impact of MIS on SDN because, given the definition of λ_{COM} and λ_{MIS}, if $\alpha_{COM} = \alpha_{MIS}$ then $\lambda_{COM} = 5\ \lambda_{MIS}$. Therefore an α_{MIS} is needed to be five times higher than α_{COM} to have the comparable effect on the network unavailability.

Regarding the CIS failure correlation source (see Figure 21(d)), the impact on SDN unavailability is limited because it affects only the control part of the principal minimal-cut sets with high cardinality.

Finally, the unavailability is much less sensitive to HEQ, compared to the TMI as source of correlation. This is not due affected by variation of the coverage factor: even with $C_{HEQ} = 0.95$ the unavailability is not significantly varying for neither legacy network nor SDN. The lack of sensitivity is probably caused by the fact that HEQ is affecting only principal minimal-cut sets with higher cardinality.

TABLE VIII: Unavailability of the component blocks

component block	unavailability
Link (L)	1.01×10^{-4}
Traditional IP router (R)	5.967×10^{-3}
SDN switch (S)	7.78×10^{-4}
SDN controller (C)	1.439×10^{-2}

TABLE IX: Network unavailability vs correlation scenario

scenario	network unavailability	
	TN	SDN
No correlation	1.1×10^{-4}	2.1×10^{-4}
Reference correlation	3.92×10^{-3}	3.156×10^{-3}
High correlation	6.908×10^{-2}	0.3005

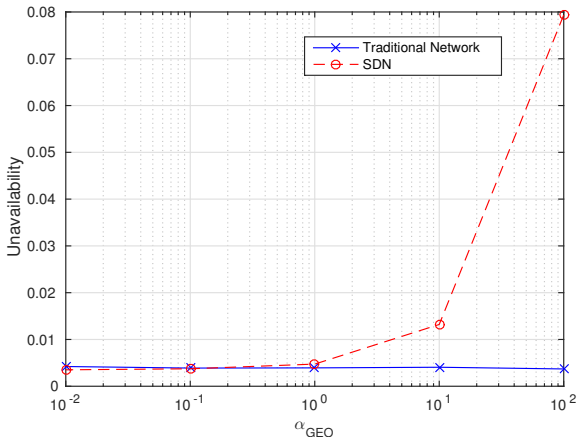
For an in-depth analysis of the impact of the failure correlation on the unavailability of traditional network and SDN, we have evaluated the unavailability of the principal minimal-cut sets in three scenarios:

- *No correlation,* the network availability is computed by considering the unavailability of the single network elements (as in [9]) and not the principal minimal-cut sets. Table VIII shows the unavailability values of the individual component blocks (i.e. network elements) computed by using the SAN models presented in Section VI-A.
- *Reference correlation,* the network availability is computed by using the reference values of correlation, i.e. $\alpha_X = 1$ with $X \in \{GEO, PHY, COM, MIS, CIS\}$ and $\beta_Y = 0$ with $Y \in \{TMI, HEQ\}$.
- *High correlation,* the network availability is computed by using high values of correlation, i.e. $\alpha_X \in \{10^2\}$, $\beta_{TMI} = -0.05$, and $\beta_{HEQ} = -0.01$.

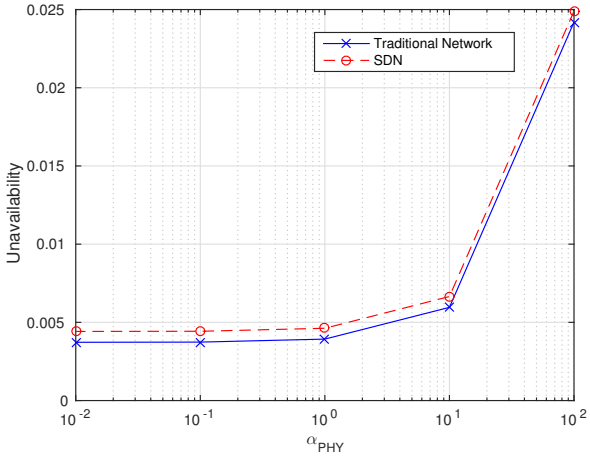
The network unavailability in the three scenarios is shown in Table IX. The table highlights that the unavailability of traditional network and of SDN are in the same order of magnitude when there is no correlation and at the reference correlation, instead the unavailability of traditional network is one order of magnitude lower than the unavailability of SDN when there is high correlation. This may indicate that the inherent distribution in the traditional network has a higher robustness towards extreme failure correlations with respect to SDN.

The results of the evaluation of the unavailability of the principal minimal-cut sets in three scenarios are depicted in Figure 22. The figure highlights the following observations:

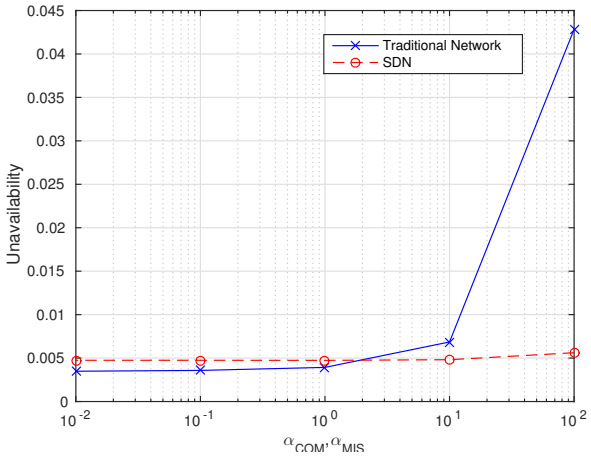
- The effect of failure correlation on the double-node minimal-cut sets, i.e. {n,n}, is similar in order of magnitude in the traditional network and the SDN. However, there is an higher increase in the forwarding part of SDN (F-SDN), which may lead to the dual forwarding node failure becoming as significant as the dual control node failure. This is an important insight that should be taken into account in the design of the network.
- In the case of the triple-node failure, i.e. {n, n, n}, the



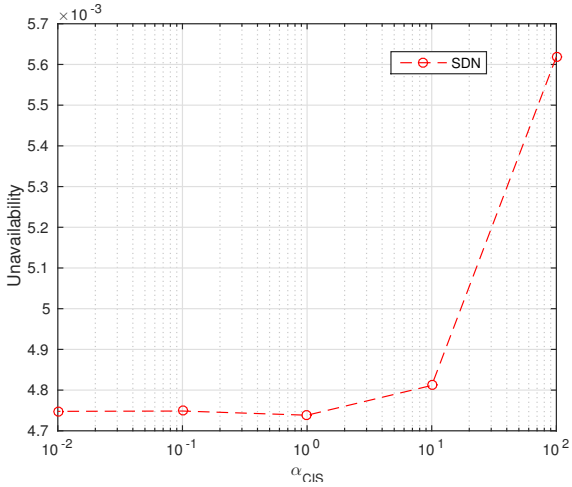
(a) Unavailability with varying GEO failure intensity



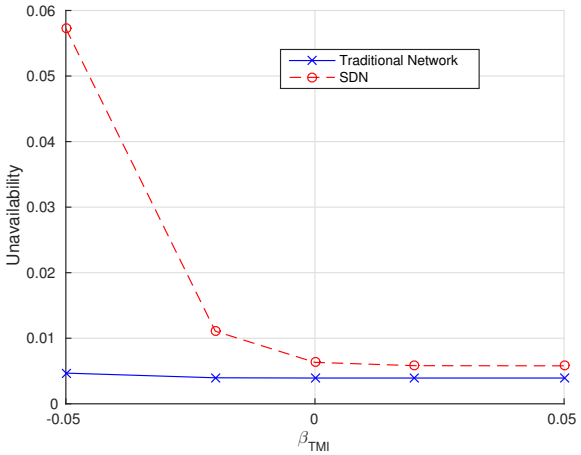
(b) Unavailability with varying PHY failure intensity



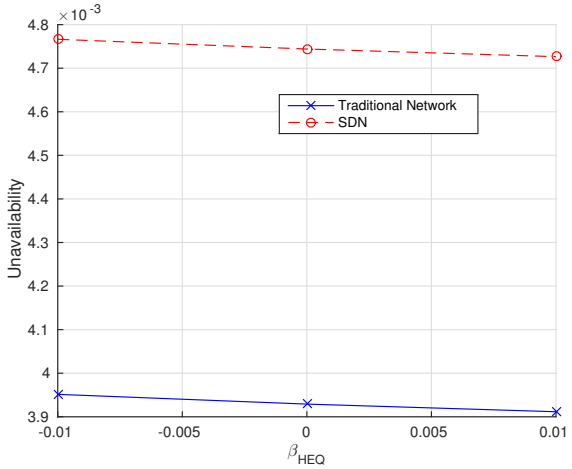
(c) Unavailability of traditional network with varying COM failure intensity and of SDN with varying MIS failure intensity



(d) Unavailability of SDN with varying CIS failure intensity



(e) Unavailability with varying TMI failure intensity



(f) Unavailability with varying HEQ failure intensity

Fig. 21: Unavailability of traditional network and of SDN with varying failure intensity of correlation sources

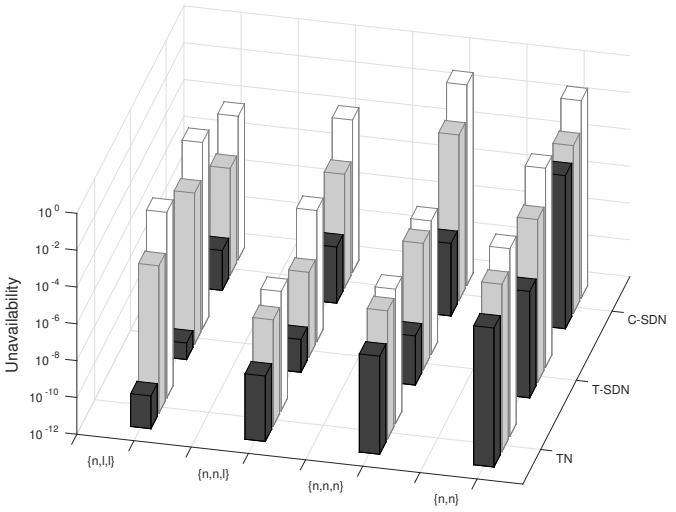


Fig. 22: Unavailability of the principal minimal-cut sets: no correlation (black bars), reference correlation (grey bars), and high correlation (white bars)

impact of correlated failures are significantly stronger for SDN than for the traditional network. In fact for the SDN control (C-SDN), it may become the dominant failure mode in case of high correlation.

- A similar effect can be noticed for the double-node one-link failure, i.e. {n, n, l}, although the contribution to system failure is less than the previous failure mode.
- The single-node double-link failure, i.e. {n, l, l} is an interesting case. It is almost neglectable in the scenario with no failure correlation, but it becomes comparable to the other failure cases when the failure correlation increases. In particular, it may become the dominant failure mode in the traditional network when the failure correlation is high.

VIII. Summary and Concluding Remarks

There is a need to identify potential dependability bottlenecks of the SDN backbone architecture, as well as to investigate the dependability of SDN relative to legacy networks. To address these issues, a holistic and reasonable comprehensive dependability model of the entire network has been presented in this paper. It uses a two-level modelling technique and takes inter-relations and -dependencies between the various network elements into account, considering the dynamic behaviour of the elements are modelled which takes into account the relevant types of failures like permanent and transient hardware failures, software design failures, operation and maintenance failures and imperfect coverage.

The approach avoids excessive complexity, state explosions and large computational demands. The models are implemented as SANs and are applied to a nation-wide backbone network. A comparison between a traditional and an SDN implementation of the network is performed for similar parameters. Issues that should receive attention in the transformation into an SDN backbone network are identified.

The main findings from our case study are the following:

- Correlated failures may have a significant effect on the overall network unavailability.
- Analysing the impact of the failure correlation sources on network unavailability (see Figure 21), SDN is more sensitive to different failure correlation sources than traditional network, e.g., SDNs are more sensitive to geographical proximity and homogeneous equipment, instead traditional networks are more sensitive to common O&M. The effect of the different failure correlation sources is summarized in Table X.
- Aggregating the failure correlation sources, SDN is more prone to be unavailable than traditional network at high correlation (see Table IX). The relative difference seems to be least both in case of moderate correlation and no correlation.
- When the failure correlation increases, the failure modes with three failed network elements get an increased impact on the network unavailability in both traditional network and SDN, although the failure modes getting an increasing dominance are different (see Figure 22).

There are few empirical studies on correlated network failures, apart from those due to catastrophic environmental events, such as hurricanes and earthquakes. With the increased vulnerability of SDN to such failures, a better basis for a solid network dimensioning is need. Being aware of this vulnerability, ensuring the robustness of SDN should be a design objective.

Acknowledgment

This work is funded by collaboration project between Telenor and NTNU, and by the NTNU QUAM research lab (https://www.ntnu.edu/iik/quam).

References

[1] E. Haleplidis, K. Pentikousis, S. Denazis, J. H. Salim, D. Meyer, and O. Koufopavlou, "Software-defined networking (SDN): Layers and architecture terminology," Internet Research Task Force (IRTF), Request for Comments RFC 7426, January 2015.

[2] D. Kreutz, F. M. V. Ramos, P. J. E. Veríssimo, C. E. Rothenberg, S. Azodolmolky, and S. Uhlig, "Software-defined networking: A comprehensive survey," *Proceedings of the IEEE*, vol. 103, no. 1, pp. 14–76, 2015.

[3] B. Nunes, M. Mendonca, X.-N. Nguyen, K. Obraczka, and T. Turletti, "A survey of software-defined networking: Past, present, and future of programmable networks," *Communications Surveys Tutorials, IEEE*, vol. 16, no. 3, pp. 1617–1634, Third 2014.

[4] NGMN Alliance. 5G White paper.

[5] P. Baran, "On distributed communications networks," *IEEE Transactions on Communications*, vol. 12, no. 1, pp. 1 – 9, march 1964.

[6] ONF, "Software-defined networking: The new norm for networks," Open Networking Foundation, ONF White Paper, April 13 2012.

[7] C. Wilson. Verizon: Reliability a key SDN concern. [Online]. Available: http://www.lightreading.com/carrier-sdn/sdn-technology/verizon-reliability-a-key-sdn-concern/d/d-id/715582

[8] P. E. Heegaard, V. B. Mendiratta, and B. E. Helvik., "Achieving dependability in software-defined networking - a perspective," in *7th International Workshop on Reliable Networks Design and Modeling (RNDM)*, Munich, Germany, October 2015.

[9] G. Nencioni, B. E. Helvik, A. J. Gonzalez, P. E. Heegaard, and A. Kamisinski, "Availability modelling of software-defined backbone networks," in *2016 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks Workshop (DSN-W)*, June 2016, pp. 105–112.

TABLE X: Effect of the different failure correlation sources on network unavailability

failure correlation source	Traditional Network (TN)	Software-defined Network (SDN)
Geographical Proximity (GEO)	Low sensitivity	High sensitivity
Physical Proximity (PHY)	Basically the same, moderate sensitivity	
Common O&M (COM)	Medium sensitivity	N/A
Misconfiguration (MIS)	N/A	Almost insensitive
Compatibility Issue (CIS)	N/A	Low sensitivity in absolute numbers
Traffic Migration (TMI)	Insensitive in the investigated range	Moderate sensitivity
Homogeneous Equipment (HEQ)	Insensitive in the investigated range	

[10] S. Fernandes and M. Santos, "SDN dependability: Assessment, techniques, and tools," in *SDN Research Group, IETF 93*, Prague, Czech Republic, July 19-24 2015.

[11] D. Kreutz, F. Ramos, and P. Verissimo, "Towards secure and dependable software-defined networks," in *Proceedings of the second ACM SIGCOMM workshop on Hot topics in software defined networking*. ACM, 2013, pp. 55–60.

[12] S. Song, H. Park, B. Y. Choi, T. Choi, and H. Zhu, "Control path management framework for enhancing software-defined network (SDN) reliability," *IEEE Transactions on Network and Service Management*, vol. 14, no. 2, pp. 302–316, June 2017.

[13] D. Staessens, S. Sharma, D. Colle, M. Pickavet, and P. Demeester, "Software defined networking: Meeting carrier grade requirements," in *Local Metropolitan Area Networks (LANMAN), 2011 18th IEEE Workshop on*, 2011, pp. 1–6.

[14] S. Sharma, D. Staessens, D. Colle, M. Pickavet, and P. Demeester, "Enabling fast failure recovery in openflow networks," in *Design of Reliable Communication Networks (DRCN), 2011 8th International Workshop on the*, 2011, pp. 164–171.

[15] S. A. Astaneh and S. S. Heydari, "Optimization of SDN flow operations in multi-failure restoration scenarios," *IEEE Transactions on Network and Service Management*, vol. 13, no. 3, pp. 421–432, Sept 2016.

[16] J. Wu, Y. Huang, J. Kong, Q. Tang, and X. Huang, "A study on the dependability of software defined networks," in *2015 International Conference on Materials Engineering and Information Technology Applications (MEITA 2015)*, September 2015.

[17] F. Longo, S. Distefano, D. Bruneo, and M. Scarpa, "Dependability modeling of software defined networking," *Computer Networks*, vol. 83, pp. 280 – 296, 2015.

[18] M. A. Chang, B. Tschaen, T. Benson, and L. Vanbever, "Chaos monkey: Increasing sdn reliability through systematic network destruction," *SIGCOMM Comput. Commun. Rev.*, vol. 45, no. 4, pp. 371–372, Aug. 2015.

[19] T. A. Nguyen, T. Eom, S. An, J. S. Park, J. B. Hong, and D. S. Kim, "Availability modeling and analysis for software defined networks," in *Dependable Computing (PRDC), 2015 IEEE 21st Pacific Rim International Symposium on*, Nov 2015, pp. 159–168.

[20] K. Han, T. A. Nguyen, D. Min, and E. M. Choi, "An evaluation of availability, reliability and power consumption for a SDN infrastructure using stochastic reward net," J. J. J. H. Park, Y. Pan, G. Yi, and V. Loia, Eds.

[21] P. E. Heegaard, B. E. Helvik, G. Nencioni, and J. Wäfler, "Managed dependability in interacting systems," in *Principles of Performance and Reliability Modeling and Evaluation*, L. Fiondella and A. Puliafito, Eds. Springer, 2016.

[22] G. Ciardo and K. S. Trivedi, "A decomposition approach for stochastic reward net models," *Perf. Eval*, vol. 18, pp. 37–59, 1993.

[23] G. Nencioni, B. E. Helvik, A. J. Gonzalez, P. E. Heegaard, and A. Kamisiński, "Impact of SDN Controllers Deployment on Network Availability," Department of Telematics, NTNU, Tech. Rep., March 2016. [Online]. Available: http://people.item.ntnu.no/~gianfran/SDNctrlDep.pdf

[24] R. E. Barlow and F. Proschan, *Statistical Theory of Reliability and Life Testing: Probability Models*. To BEGIN WITH, 1975.

[25] K. Kanoun, M. Borrel, T. Morteveille, and A. Peytavin, "Availability of CAUTRA, a Subset of the French Air Traffic Control System." *IEEE Trans. Computers*, vol. 48, no. 5, pp. 528–535, 1999.

[26] A. J. Gonzalez and B. E. Helvik, "Characterization of router and link failure processes in UNINETT's IP backbone network," *International Journal of Space-Based and Situated Computing*, 2012.

[27] P. Kuusela and I. Norros, "On/off process modeling of ip network failures," in *Dependable Systems and Networks (DSN), 2010 IEEE/IFIP International Conference on*, June 2010, pp. 585–594.

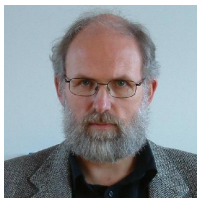
[28] S. Verbrugge, D. Colle, P. Demeester, R. Huelsermann, and M. Jaeger, "General availability model for multilayer transport networks," in *Proceedings.5th International Workshop on Design of Reliable Communication Networks, 2005. (DRCN 2005)*. IEEE, October 16-19 2005, pp. 85 – 92.

[29] A. J. Gonzalez, B. E. Helvik, J. K. Hellan, and P. Kuusela, "Analysis of dependencies between failures in the uninett ip backbone network," in *2010 IEEE 16th Pacific Rim International Symposium on Dependable Computing*, Dec 2010, pp. 149–156.

[30] "Möbius: Model-based environment for validation of system reliability, availability, security, and performance," https://www.mobius.illinois.edu/, accessed: 2017-03-02.

[31] G. Nencioni, B. E. Helvik, and P. E. Heegaard, "Implementing the Availability Model of a Software-Defined Backbone Network in Möbius," IIK, NTNU, Tech. Rep., March 2017. [Online]. Available: http://people.item.ntnu.no/~gianfran/SANmodelSDN.pdf

Gianfranco Nencioni received his Ph.D. degree in information engineering from the University of Pisa (Italy) in 2012. In the Fall of 2011, he was a visiting Ph.D. student with the Computer Laboratory, University of Cambridge (UK). From 2012 to 2015, he was a Postdoctoral Fellow at the University of Pisa, and now is a Postdoctoral Fellow at NTNU (Norway). His past research activity has regarded energy-aware routing and design in both wired and wireless networks. His current research activity regards dependability on SDN and NFV.



Bjarne E. Helvik (1952) received his Siv.ing. degree (MSc in technology) from the Norwegian Institute of Technology (NTH), Trondheim, Norway in 1975. He was awarded the degree Dr. Techn. from NTH in 1982. He has since 1997 been Professor at the Norwegian University of Science and Technology (NTNU), the Department of Telematics. Since August 2009, he has been Vice Dean with responsibility for research at the Faculty of Information Technology and Electrical Engineering at NTNU. He has previously held various positions at ELAB and SINTEF Telecom and Informatics. In the period 1988-1997 he was appointed as Adjunct Professor at the Department of Computer Engineering and Telematics at NTH.

His field of interests includes QoS, dependability modelling, measurements, analysis and simulation, fault-tolerant computing systems and survivable networks, as well as related system architectural issues. His current research focus is on ensuring dependability in services provided by multi-domain, virtualised ICT systems.

Poul E. Heegaard received his Ph.D. in telematics from NTNU in 1998. He has been a full professor at NTNU since 2010. His main research interests are performance and dependability modeling and simulations of communication networks, currently focusing on resource optimization and management in distributed autonomous systems in a multi-domain context. He was head of the Department of Telematics (20092013), and is now head of the NTNU Quantitative Modelling of Dependability and Performance (QUAM) research lab.