International Journal of Mathematical, Engineering and Management Sciences
Vol. 8, No. 1, 1-14, 2023
https://doi.org/10.33889/IJMEMS.2023.8.1.001

Load Redistribution-based Reliability Enhancement for Storage Area Networks

Guixiang Lv
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
E-mail: glv@umassd.edu

Liudong Xing
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
Corresponding author: lxing@umassd.edu

Honggang Wang
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
E-mail: hwang1@umassd.edu

Hong Liu
Department of Electrical and Computer Engineering,
University of Massachusetts, Dartmouth, MA, USA.
E-mail: hliu@umassd.edu

(Received on August 16, 2022; Accepted on October 30, 2022)

Abstract
Storage area networks (SANs) are one of the prevalent reliable data storage solutions. However, cascading failures triggered by data overloading have become a major threat to SANs, preventing the desired quality of service from being delivered to users. Based on our preliminary works on studying the impacts of data loading on the reliability performance of SANs, this paper advances the state of the art by implementing node degree-based load redistribution strategies to enhance the SAN reliability, thus mitigating or even preventing the occurrence of cascading failures during the mission time. Load-based and reliability-based node selection rules are considered, which choose nodes with the highest load level and the lowest reliability for load redistribution, respectively. The relationship between data loading and reliability of an individual SAN component is modeled using the accelerated failure-time model with the power law. The SAN reliability is assessed using a combinatorial decision diagram-based approach. The application and effectiveness of the proposed load redistribution strategies are demonstrated and compared through a case study of an SAN with the mesh topology.

Keywords- Storage area network, Cascading failure, Load redistribution, Node degree, Accelerated failure-time model.

1. Introduction

Telecommuting is being widely adopted, inducing significant growth in the remote data access and storage (Bright and Raschid, 2000; Hutanu et al., 2010). As a prevalent network-based storage solution, a storage area network (SAN) implements any-to-any interconnections between servers and storage devices in the network (Honma et al., 2004; Xing et al., 2017). The SAN solution has advantages of low latency, high throughput and concurrent shared access. Due to those benefits, SANs have been used by many enterprises like IBM, NetApp, and Tintri (Garber, 2012). To deliver robust and desired services, reliability modeling and enhancement of SANs play a pivotal role.

Examples of SAN reliability models include analytical reliability models for optical-based and IP-based SAN extensions, suggested by Qiu et al. (2005). Jiang et al. (2007) put forward a fuzzy, Markov model-based reliability analysis approach for an iSCSI protocol-based SAN. Uwaechia and Akinsanmi (2013) evaluated the reliability of an SAN using the redundancy array of independent disks. Xing et al. (2017) modeled the reliability of a mesh SAN using dynamic fault trees, network graph model, and binary decision diagrams (BDDs). Lv and Xing (2021) investigated the effects of data loading on the reliability performance of SAN systems. In addition, Lin et al. (2007) suggested a reliable multicast mechanism for an Ethernet-based SAN to greatly reduce the end-to-end data transmission time.

A major threat to the reliable operation of SANs is cascading failures, where a single incident (e.g., overloading, a device’s malfunction, operator errors, physical attacks, cyberattacks, natural disasters) triggers a chain reaction, causing extensive damage to the entire system (Chen et al., 2019; Li et al., 2022; Mishra et al., 2020; Nguyen et al., 2021; Zhou et al., 2021). A rich body of research efforts have been devoted to modeling and mitigating cascading failures in various technological systems like power systems and high-performance computing systems. Major models of studying the mechanism of cascading failures include, for example, the self-organized critical model, complex network theory-based models, and simulation-based models (Bialek et al., 2016; Shi et al., 2010). The analysis approaches considering effects of cascading failures can be roughly categorized into topological approaches, combinatorial approaches, state-space methods, and simulations (Xing, 2021). However, to the best of our knowledge, no works from literature considered the mitigation of cascading failures for SAN systems.

In this work, we advance the state of the art by proposing node degree-based load redistribution strategies to enhance the SAN reliability, further mitigating and even preventing the occurrence of cascading failures during the mission time. Overload is a major case of cascading failures occurring in SANs, which often contain multiple switches used for relaying data transmissions between storage arrays and servers. In the case of the overloading causing the failure of one switch, its load is redistributed to other available switches. The redistribution may further lead to overloading on those remaining switches in a domino way, causing the outage of the entire SAN system. Based on our preliminary work in Lv and Xing (2021), we propose load redistribution-based mitigation strategies, where before the SAN reliability drops to a pre-specified threshold value, we redistribute the load of selected nodes to other nodes to enhance the overall system reliability, further mitigating the high risk of cascading failures. Two node selection rules are considered, load-based and reliability-based, which choose nodes with the highest load level and the lowest reliability for load redistribution, respectively. We demonstrate and compare the effectiveness of the proposed strategies using a detailed case study of a mesh SAN system.

The structure for the rest of the paper is: Section 2 depicts the accelerated failure-time model (AFTM) for modeling the load-failure rate relationship of a device in SANs. Section 3 presents the node selection rules and two node degree-based load-redistribution strategies, leading to four mitigation schemes. Section 4 gives an example of a mesh SAN system and its reliability analysis using BDDs. Section 5 examines effects of reliability threshold used for triggering load redistribution. Section 6 evaluates and compares performance of the four mitigation schemes. Section 7 presents an application of the proposed mitigation scheme for accomplishing a mission of a certain specified time. Section 8 concludes the work and indicates directions for future research.

2. The AFTM Model

To model the relationship between the loading and reliability/failure rate behavior of a device in the SAN, the AFTM model using the power law (Kay and Kinnersley, 2002; Khanal et al., 2014; Levitin and Amari, 2009; Lin et al., 1998) is applied.

Under the AFTM, the reliability of a component under loading L at mission time *t* is

$$R(t;L) = R_0(t\phi(L)), \tag{1}$$

where R_0 represents the baseline reliability function of an arbitrary distribution, and $\phi(L)$ represents a multiplicative factor used to reflect different levels of stresses under different loads.

In the case of one type of load, $\phi(L)$ can be specified using the power law or the exponential law as shown in (2), where α is an effect parameter.

$$\phi(L) = \begin{cases} L^{\alpha}, & Power\ law \\ e^{L\alpha}, & Exponential\ law \end{cases} \tag{2}$$

The power law is adopted in this work. When the baseline time-to-failure distribution is exponential, we have the reliability function $R(t;L_0)$, failure function $F(t;L_0)$, probability density function $f(t;L_0)$, and failure rate $\lambda(L_0)$ under load L_0 as

$$R(t;L_0) = R_0\big(t\phi(L_0)\big) = e^{-\lambda(t\phi(L_0))} = e^{-\lambda t {L_0}^{\alpha}} \tag{3}$$

$$F(t;L_0) = 1 - R(t;L_0) = 1 - e^{-\lambda t {L_0}^{\alpha}} \tag{4}$$

$$f(t;L_0) = \frac{dF(t;L_0)}{dt} = {L_0}^{\alpha}\lambda e^{-\lambda t {L_0}^{\alpha}} \tag{5}$$

$$\lambda(L_0) = \frac{f(t;L_0)}{R(t;L_0)} = {L_0}^{\alpha}\lambda \tag{6}$$

Similarly, we have the failure rate at another load L as $\lambda(L) = L^{\alpha}\lambda$. The relationship between $\lambda(L)$ and $\lambda(L_0)$ can then be expressed as

$$\lambda(L) = \lambda(L_0)(\frac{L}{L_0})^{\alpha} \tag{7}$$

3. Proposed Reliability Enhancement Strategies

This section presents the load redistribution-based strategies, where before the SAN reliability drops to a pre-specified threshold value, we redistribute the load of selected nodes to other nodes with the aim to enhance the overall SAN system reliability, further mitigating the high risk of cascading failures. Two node selection rules and two node degree-based redistribution rules are considered, which are described in Section 3.1 and Section 3.2, respectively.

3.1 Node Selection Rules

Consider two nodes selection rules: load-based and reliability-based. In the event of the redistribution being triggered, under the load-based selection rule, we select the nodes with the top *u* (e.g., 3 used in the case study) highest load level for load redistribution; under the reliability-based selection rule, we select the nodes with the top *w* (e.g., 3 used in the case study) lowest reliability for load redistribution.

3.2 Redistribution Rules

Based on the load local redistribution rule (Wang et al., 2008), we consider two node degree-based redistribution rules in this work: proportional rule and inverse-proportional rule. Briefly speaking, a node with a higher node degree tends to get more load during the redistribution in the proportional rule, but less load in the inverse-proportional rule.

Proportional Rule: Consider the redistribution of node *i*'s load. Let N_i represent the set of neighboring nodes for node *i*, including node *i* itself. According to Wang et al. (2008), Eq. (8) defines the weight of any node *j* belonging to set N_i, where d_j represents the degree of node *j*, β represents a tunable parameter used to control the strength of the initial load, which is a function of degree of the corresponding node. Eq. (9) computes the actual load reassigned from load *i* to load *j*. Eq. (10) gives the updated load after redistribution of node *i*'s load. Eq. (11) gives the updated load of node *i* after the redistribution.

$$\Pi_j = \begin{cases} \frac{d_j^{\beta}}{\sum_{m \in N_i} d_m^{\beta}}, \ for\ any\ \ j \in N_i \\ \qquad\quad 0, \ for\ any\ \ j \notin N_i \end{cases} \tag{8}$$

$$\Delta L_{ji} = L_i \Pi_j \tag{9}$$

$$\tilde{L}_j = L_j + \Delta L_{ji} \tag{10}$$

$$\tilde{L}_i = \Delta L_{ii} \tag{11}$$

In the case of redistributing loads of multiple nodes represented by set Φ at the same time, Eqs. (8) and (9) are still applicable. Eq. (12) gives the updated load of any node *j* not in set Φ after the redistribution of loads of nodes in set Φ. Eq. (13) gives the updated load of any node *i* in set Φ after the redistribution.

$$\tilde{L}_j = L_j + \sum_{i \in \Phi} \Delta L_{ji} \tag{12}$$

$$\tilde{L}_i = \sum_{y \in \Phi} \Delta L_{yi} \tag{13}$$

Inverse-Proportional Rule: Consider the redistribution of node *i*'s load. Eq. (14) defines the weight of any node *j* belonging to set N_i, where Π_j is evaluated using Eq. (8). Similar to Eq. (9), Eq. (15) computes the actual load reassigned from load *i* to load *j* under the inverse-proportional rule. The updated load after redistribution of node *i*'s load is given in Eq. (10). The updated load of node *i* after the redistribution is given in Eq. (11).

$$\Gamma_j = \begin{cases} \frac{1-\Pi_j}{\sum_{m \in N_i}(1-\Pi_j)}, \ for\ any\ \ j \in N_i \\ \qquad\quad 0, \ for\ any\ \ j \notin N_i \end{cases} \tag{14}$$

$$\Delta L_{ji} = L_i \Gamma_j \tag{15}$$

3.3 Summary

Based on the two node selection rules and the two redistribution rules presented above, we study four schemes for reliability enhancement, summarized in Table 1.

Table 1. Four schemes.

	Reliability-based selection	**Load-based selection**
Proportional redistribution	Scheme 1	Scheme 2
Inverse-proportional redistribution	Scheme 3	Scheme 4

4. An Illustrative Mesh SAN

To illustrate and compare the performance of the four reliability enhancement schemes (Table 1), we consider an example of a mesh SAN. As shown in Figure 1, the example SAN consists of two servers (Sr_1 and Sr_2) providing data services, two storage arrays (Sa_1 and Sa_2), and five switches (Sw_1, Sw_2, Sw_3, Sw_4, Sw_5) forming a network facilitating any-to-any communication between servers and storage arrays.

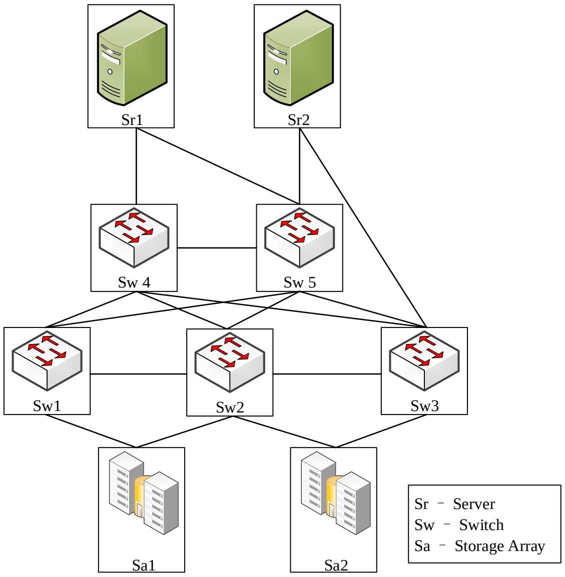

Figure 1. An example of mesh SANs.

In this work, it is assumed that the servers and storage arrays do not participate in the load redistribution activity. Thus, the degree of a switch node is defined as the number of connections/links that it has to other nodes excluding the servers and the storage arrays (i.e., other switches) in the SAN. For the example SAN, the node degrees for the five switches (Sw_1, Sw_2, Sw_3, Sw_4, Sw_5) are d_{sw1}=3, d_{sw2}=4, d_{sw3}=3, d_{sw4}=4, and $d_{sw5=}$4, respectively.

Table 2 lists baseline failure rate parameters and initial load applied for the five switches in the example mesh SAN. According to the parameters used in Lv and Xing (2021), both servers (*Sr1*, *Sr2*) and storage arrays (*Sa1*, *Sa2*) are assumed to have the failure rate of 4.756469781e-11 per hour.

Table 2. Baseline failure rates and initial load for switches of the example mesh SAN.

Switch	Failure rate λ (per hour)	Initial load applied L_0
Sw_1	3.0e-6	15
Sw_2	5.0e-6	50
Sw_3	3.0e-5	5
Sw_4	3.0e-6	1
Sw_5	3.5e-6	8

4.1 Fault Tree Modeling

The example SAN functions as long as at least one server can communicate with at least one storage array. Thus, the SAN failure can be modeled using the fault tree as shown in Figure 2.

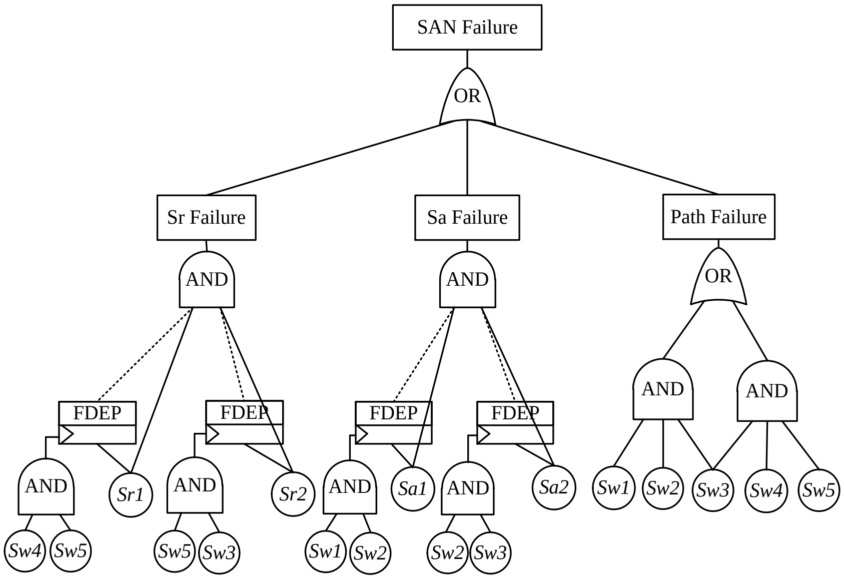

Figure 2. Fault tree of the example mesh SAN.

The server subsystem fails when both servers malfunction (modeled by the left subtree in Figure 2). Similarly, the storage array subsystem fails when both storage arrays are failed (modeled by the middle subtree). The connections or paths between servers and storage arrays fail when either switches Sw_1, Sw_2, Sw_3 all malfunction or switches Sw_3, Sw_4, Sw_5 all malfunction (modeled by the right subtree). The FDEP gates in Figure 2 are used to model the functional dependence behavior between servers or storage arrays and related switches. For example, the storage array $Sa1$ is accessible only through Sw_1 and Sw_2, meaning that $Sa1$ has function dependence on Sw_1 and Sw_2, which is modeled by an FDEP gate in the middle subtree.

For the system reliability analysis, the FDEP gates can be replaced by logic OR gates (Xing et al., 2014). Figure 3 shows the equivalent fault tree model after replacing each FDEP gate in Figure 2 with an OR gate.

4.2 Reliability Analysis Using BDDs

BDDs were first adapted to system reliability analysis in 1993 and has become a state-of-the-art method for efficient reliability analysis of diverse systems (Xing and Amari, 2015; Xing and Dugan, 2002). It is adopted in this work to analyze the reliability of the example SAN system. Using the ordering of component variables $Sr_1< Sr_2< Sa_1<Sa_2<Sw_1<Sw_2<Sw_3<Sw_4<Sw_5$ and the open-source python library PyEDA (Drake, 2022), we generate the BDD for the fault tree of Figure 3 as shown in Figure 4. The BDD has two sink nodes ‘0’ and ‘1’, representing the SAN system is operational and failed, respectively. Each non-sink node has two outgoing edges, 0-edge and 1-edge, respectively modeling the corresponding SAN component is operational and failed. Let q_i represent component i’s failure probability (i.e., the probability associated with the corresponding node’s 1-edge). The unreliability of the example SAN, denoted by U_{sys} can be obtained by summing probabilities of all paths from the root node to sink node ‘1’ of the BDD, as shown in Eq. (16).

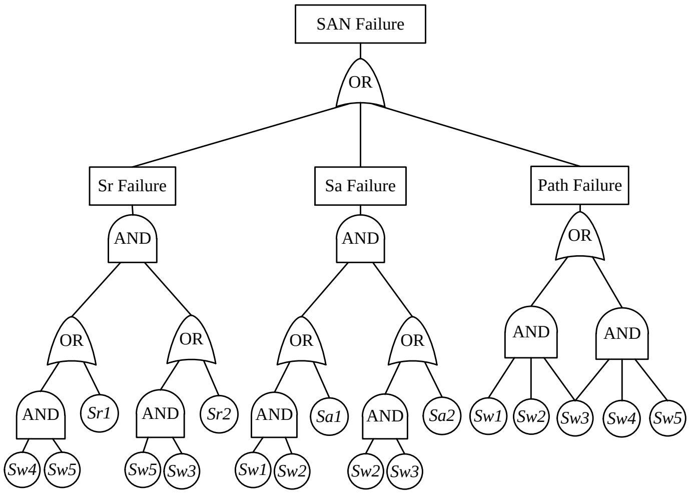

Figure 3. Equivalent fault tree of the example mesh SAN.

$$
\begin{aligned}
U_{sys} &= q_{Sr1}\times q_{Sr2}+q_{Sr1}\times(1-q_{Sr2})\times q_{Sa1}\times q_{Sa2}\\
&+q_{Sr1}\times(1-q_{Sr2})\times q_{Sa1}\times(1-q_{Sa2})\times q_{Sw2}\times q_{Sw3}\\
&+q_{Sr1}\times(1-q_{Sr2})\times q_{Sa1}\times(1-q_{Sa2})\times(1-q_{Sw2})\times q_{Sw3}\times q_{Sw5}\\
&+q_{Sr1}\times(1-q_{Sr2})\times(1-q_{Sa1})\times q_{Sa2}\times q_{Sw1}\times q_{Sw2}\\
&+q_{Sr1}\times(1-q_{Sr2})\times(1-q_{Sa1})\times q_{Sa2}\times(1-q_{Sw1})\times q_{Sw3}\times q_{Sw5}\\
&+q_{Sr1}\times(1-q_{Sr2})\times(1-q_{Sa1})\times q_{Sa2}\times q_{Sw1}\times(1-q_{Sw2})\times q_{Sw3}\times q_{Sw5}\\
&+q_{Sr1}\times(1-q_{Sr2})\times(1-q_{Sa1})\times(1-q_{Sa2})\times q_{Sw1}\times q_{Sw2}\times q_{Sw3}\\
&+q_{Sr1}\times(1-q_{Sr2})\times(1-q_{Sa1})\times(1-q_{Sa2})\times(1-q_{Sw1})\times q_{Sw3}\times q_{Sw5}\\
&+(1-q_{Sr1})\times q_{Sr2}\times q_{Sa1}\times q_{Sa2}+(1-q_{Sr1})\times q_{Sr2}\times q_{Sa1}\times(1-q_{Sa2})\times q_{Sw2}\times q_{Sw3}\\
&+(1-q_{Sr1})\times q_{Sr2}\times q_{Sa1}\times(1-q_{Sa2})\times q_{Sw2}\times(1-q_{Sw3})\times q_{Sw4}\times q_{Sw5}\\
&+(1-q_{Sr1})\times q_{Sr2}\times(1-q_{Sa1})\times q_{Sa2}\times q_{Sw1}\times q_{Sw2}\\
&+(1-q_{Sr1})\times q_{Sr2}\times(1-q_{Sa1})\times q_{Sa2}\times q_{Sw1}\times(1-q_{Sw2})\times q_{Sw4}\times q_{Sw5}\\
&+(1-q_{Sr1})\times q_{Sr2}\times(1-q_{Sa1})\times(1-q_{Sa2})\times q_{Sw1}\times q_{Sw2}\times q_{Sw3}\\
&+(1-q_{Sr1})\times q_{Sr2}\times(1-q_{Sa1})\times(1-q_{Sa2})\times q_{Sw1}\times q_{Sw2}\times(1-q_{Sw3})\times q_{Sw4}\times q_{Sw5}\\
&+(1-q_{Sr1})\times q_{Sr2}\times(1-q_{Sa1})\times(1-q_{Sa2})\times q_{Sw1}\times(1-q_{Sw2})\times q_{Sw4}\times q_{Sw5}\\
&+(1-q_{Sr1})\times q_{Sr2}\times(1-q_{Sa1})\times(1-q_{Sa2})\times(1-q_{Sw1})\times q_{Sw4}\times q_{Sw5}\\
&+(1-q_{Sr1})\times(1-q_{Sr2})\times q_{Sa1}\times q_{Sa2}+(1-q_{Sr1})\times(1-q_{Sr2})\times q_{Sa1}\times(1-q_{Sa2})\times q_{Sw2}\times q_{Sw3}\\
&+(1-q_{Sr1})\times(1-q_{Sr2})\times q_{Sa1}\times(1-q_{Sa2})\times(1-q_{Sw2})\times q_{Sw3}\times q_{Sw4}\times q_{Sw5}\\
&+(1-q_{Sr1})\times(1-q_{Sr2})\times(1-q_{Sa1})\times(1-q_{Sa2})\times q_{Sw1}\times(1-q_{Sw2})\times q_{Sw3}\times q_{Sw4}\times q_{Sw5}\\
&+(1-q_{Sr1})\times(1-q_{Sr2})\times(1-q_{Sa1})\times(1-q_{Sa2})\times q_{Sw2}\times q_{Sw3}\\
&+(1-q_{Sr1})\times(1-q_{Sr2})\times(1-q_{Sa1})\times q_{Sa2}\times q_{Sw1}\times q_{Sw2}\\
&+(1-q_{Sr1})\times(1-q_{Sr2})\times(1-q_{Sa1})\times q_{Sa2}\times q_{Sw1}\times(1-q_{Sw2})\times q_{Sw3}\times q_{Sw4}\times q_{Sw5}\\
&+(1-q_{Sr1})\times(1-q_{Sr2})\times(1-q_{Sa1})\times q_{Sa2}\times(1-q_{Sw1})\times q_{Sw2}\times q_{Sw3}\times q_{Sw4}\times q_{Sw5}\\
&+(1-q_{Sr1})\times(1-q_{Sr2})\times(1-q_{Sa1})\times(1-q_{Sa2})\times(1-q_{Sw1})\times q_{Sw3}\times q_{Sw4}\times q_{Sw5}
\end{aligned}
\tag{16}
$$

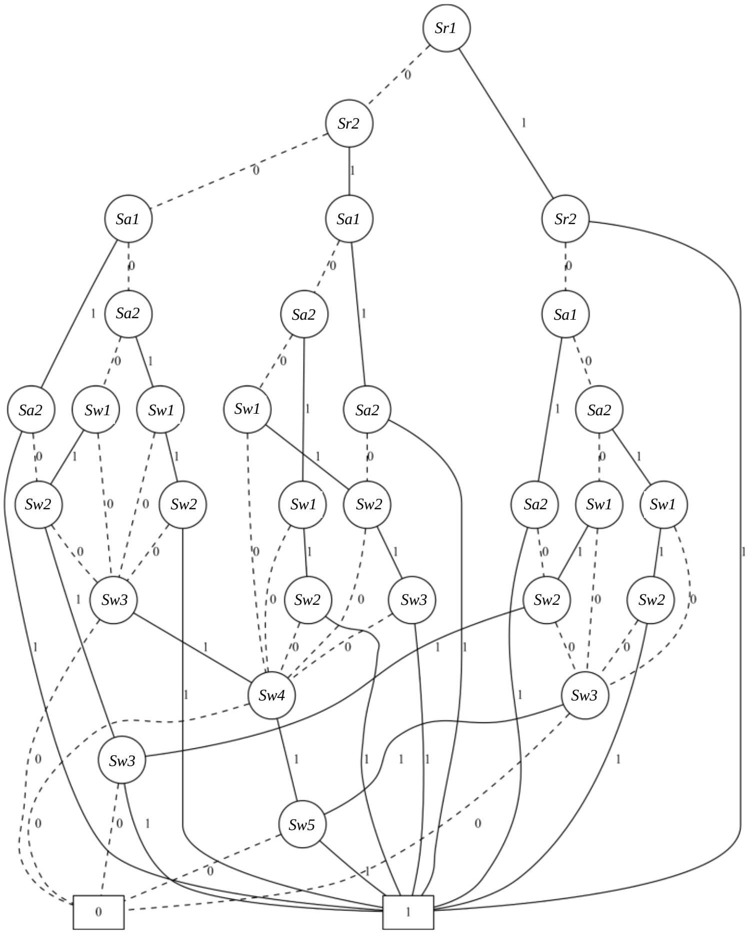

Figure 4. BDD model of the example mesh SAN.

5. Influence of Reliability Threshold on Redistribution Performance

In this section, we investigate the effects of the SAN reliability threshold used for triggering load redistribution using Scheme 1 based on the proportional rule and the reliability-based node selection rule. Specifically, we compute the reliability of five switches and the entire SAN system before and after the redistribution using three different SAN reliability threshold values R*= 0.9, 0.8, 0.7. Other parameters used in this study are α=1, β=1, u=3, and parameters in Table 2.

The results are summarized in Table 3 and illustrated graphically in Figure 5. The sharp drop and increase in Figure 5 indicate the effects of the load redistribution on the reliability of each switch and the overall SAN system. Specifically, the loads of Sw_1, Sw_2 and Sw_3 are redistributed because these three switches have the lowest reliability values. After the load redistribution, the reliability of Sw_1 and Sw_2 improves because the loads of Sw_1 and Sw_2 redistributed to other switches are more than those received from other switches during the redistribution and thus their loads decrease after the redistribution. On the other hand, the reliability of Sw_3 decreases because the load of Sw_3 redistributed to other switches is less than those received from Sw_1 and Sw_2 during the redistribution. It is intuitive that the reliability of Sw_4 and Sw_5 decreases because they receive more load from Sw_1, Sw_2 and Sw_3 during the redistribution process in addition to their original load. Table 4 summarizes the mission time when the redistribution is triggered, the SAN reliability values before and after the redistribution, their difference, as well as the improvement ratio (calculated as the difference divided by the entire SAN reliability before the load redistribution). It can be observed that the reliability improvement ratio increases as the system reliability threshold value used for triggering the redistribution decreases, implying that the redistribution scheme becomes more effective as the system reliability requirement is less restrictive.

Table 3. Load and reliabilities of all the switches and the entire SAN under Scheme 1.

Before redistribution						
	***Sw*₁**	***Sw*₂**	***Sw*₃**	***Sw*₄**	***Sw*₅**	**SAN**
Load	15	50	5	1	8	
*R**=0.9	0.914960	0.610333	0.743602	0.994093	0.946201	0.900011
*R**=0.8	0.870145	0.461742	0.628984	0.990770	0.917092	0.800033
*R**=0.7	0.829576	0.354162	0.536440	0.987621	0.890247	0.700055
After redistribution						
Load	11.33	16.44	9.33	17.44	24.44	
*R**=0.9	0.935055	0.850111	0.575222	0.901804	0.844532	0.929909
*R**=0.8	0.900240	0.775576	0.420849	0.850643	0.767626	0.850375
*R**=0.7	0.868343	0.710784	0.312685	0.804698	0.701015	0.762612

Table 4. Summary of reliability improvements for different thresholds.

Threshold	**Triggering time**	**Before redistribution**	**After redistribution**	**Difference**	**Improvement ratio**
*R**=0.9	1976h	0.900011	0.929909	0.029898	0.033220
*R**=0.8	3092h	0.800033	0.850375	0.050342	0.062925
*R**=0.7	4153h	0.700055	0.762612	0.062557	0.089360

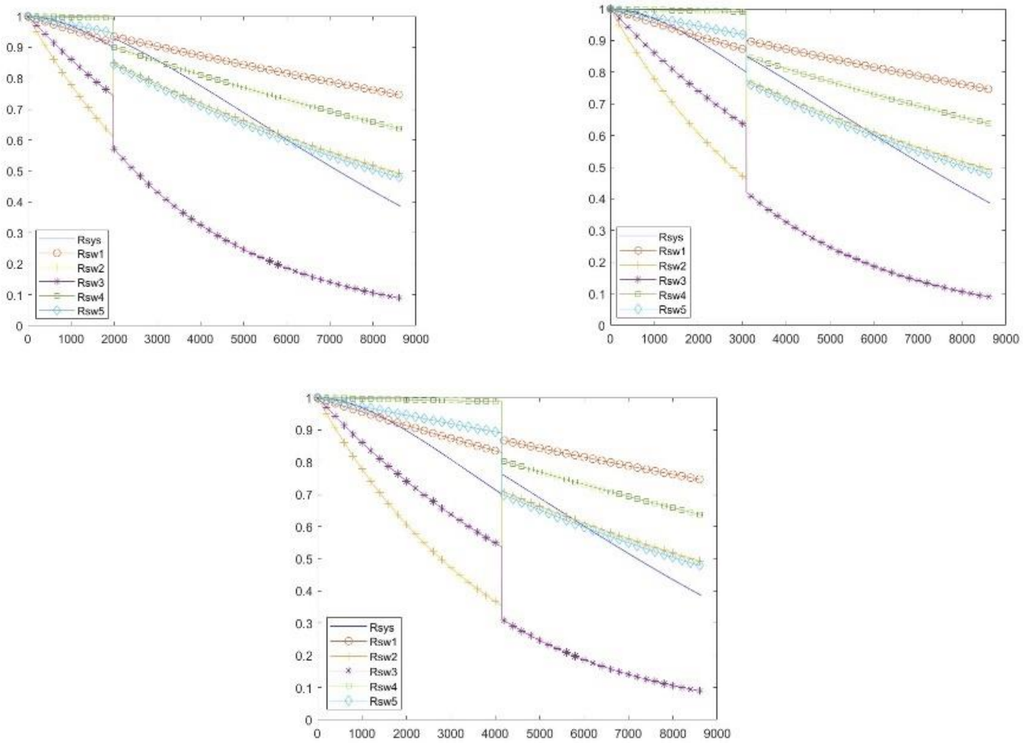

Figure 5. Switch and SAN reliabilities under *R**=0.9, 0.8, and 0.7 (from left to right).

6. Comparison of Four Schemes

In this section, we evaluate and compare the performance of the four mitigation schemes. The system reliability threshold R^*=0.8, w=3, u=3 are used. We also investigate the effects of the AFTM model parameter α and the strength control parameter β on the performance of the schemes. Three different values of parameter α (0.7, 1, 1.1) and three different values of parameter β (0.8, 1, 1.2) are considered, leading to nine cases. Table 5 summarizes the reliability improvement ratio under each case and each scheme.

Table 5. Summary of reliability improvements for nine cases.

α	β	**Scheme 1**	**Scheme 2**	**Scheme 3**	**Scheme 4**
0.7	0.8	0.052449	0.020564	0.051196	0.027277
	1.0	0.053299	0.019682	0.051271	0.027700
	1.2	0.054338	0.018943	0.051353	0.028119
1.0	0.8	0.061449	0.014656	0.058564	0.024611
	1.0	0.062925	0.013379	0.058629	0.025241
	1.2	0.064679	0.012330	0.058708	0.025865
1.1	0.8	0.061904	0.009740	0.058424	0.021050
	1.0	0.063612	0.008293	0.058487	0.021765
	1.2	0.065631	0.007107	0.058565	0.022472

Comparing the values under Scheme 1 and Scheme 2, it can be concluded that the reliability-based node selection rule performs significantly better than the load-based selection rule under the proportional redistribution strategy. Comparing the values under Scheme 3 and Scheme 4, it can be concluded that the reliability-based node selection rule still performs significantly better than the load-based selection rule under the inverse-proportional redistribution strategy.

Comparing the values under Scheme 1 and Scheme 3, it can be concluded that the proportional redistribution strategy performs slightly better than the inverse-proportional redistribution strategy when the reliability-based node selection rule is adopted. However, comparing the values under Scheme 2 and Scheme 4, it can be concluded that the inverse-proportional redistribution strategy performs better than the proportional redistribution strategy when the load-based node selection rule is adopted. Thus, the relative performance of the two redistribution strategies (proportional and inverse-proportional) depends on the node selection rule adopted.

It can also be observed from Table 5 that as β increases, the reliability improvement ratio increases under Schemes 1, 3 and 4 but decreases under Scheme 2. Unlike the influence of β, with increasing α, the reliability improvement ratio does not have a consistent varying trend. Specifically, as α increases, the reliability improvement ratio increases under Scheme 1, decreases under Schemes 2 and 4, increases first then decreases under Scheme 3.

7. Application

This section presents an application of the proposed mitigation scheme for accomplishing a mission of a certain specified time. For illustration purpose, the mitigation Scheme 1 and parameters w=3, u=3, α=1, β=1 are used.

For example, to maintain the system reliability over the desired threshold value R^*= 0.8 for a specified mission time t=4805h, Figure 6 illustrates that three load redistributions, respectively taking place at 3092h, 3712h, and 4595h are necessary. The system reliability increases sharply after each load redistribution while having a decreasing trend in general as the mission time proceeds, leading to the chainsaw shape of the SAN reliability curve. The three switches involved in the load redistributions are (Sw_1, Sw_2, Sw_3) at

t_1=3092h, (Sw_2, Sw_3, Sw_5) at t_2=3712h, and (Sw_2, Sw_3, Sw_4) at t_3=4595h. Because different subsets of switches are involved in different redistributions, the reliabilities of different switches have different trends. For example, the reliability of Sw_2 always increases after each redistribution since its load decreases during each load redistribution process. However, the reliability of Sw_4 decreases in the first two redistributions as Sw_4 shares/receives more load during the redistribution process, but increases significantly in the third redistribution as the load of Sw_4 is redistributed to other switches. Table 6 summarizes the load and reliability of each switch as well as the reliability of the entire SAN before and after each redistribution.

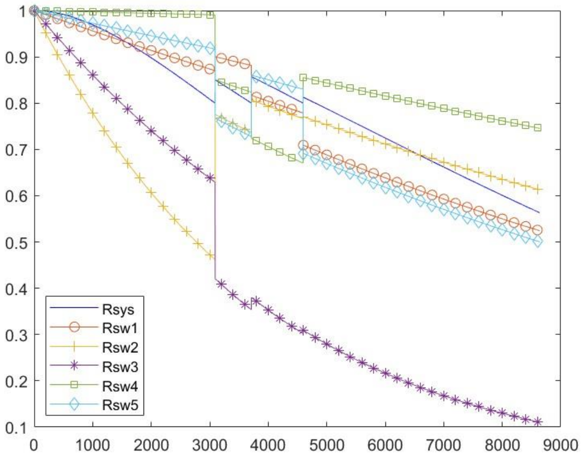

Figure 6. Illustrative application of three redistributions to accomplish a mission.

Table 6. Switch and SAN reliability before and after each redistribution.

Before redistribution						
	t_1=3092h		t_2=3712h		t_3=4595h	
	Load	Reliability	Load	Reliability	Load	Reliability
Sw_1	15	0.870145	11.33	0.881461	18.15	0.778709
Sw_2	50	0.461742	16.44	0.737030	11.58	0.766527
Sw_3	5	0.628984	9.33	0.353780	8.68	0.302256
Sw_4	1	0.990770	17.45	0.823486	29.02	0.670353
Sw_5	8	0.917092	24.45	0.727969	11.57	0.830174
R_{sys}		0.800033		0.800001		0.800053
After redistribution						
	Load	Reliability	Load	Reliability	Load	Reliability
Sw_1	11.33	0.900240	18.15	0.817060	24.91	0.709380
Sw_2	16.44	0.775576	11.58	0.806719	11.34	0.770749
Sw_3	9.33	0.420849	8.68	0.380409	8.50	0.309820
Sw_4	17.45	0.850643	29.02	0.723918	11.34	0.855358
Sw_5	24.45	0.767626	11.57	0.860410	22.91	0.691845
R_{sys}		0.850375		0.857211		0.813062

8. Conclusion and Future Work

In this paper, based on two node degree-based redistribution rules (proportional and inverse-proportional) and two node selection rules (load-based and reliability-based), we investigate four reliability enhancement schemes for SANs. Triggered by the SAN reliability dropping to a pre-specified threshold, loads of selected nodes are redistributed leading to improvement in the overall SAN reliability and thus mitigating the risk of cascading failures due to the overload. Detailed case studies are provided to demonstrate and compare the performance of the four schemes. Effects of key model parameters including the SAN reliability threshold, the AFTM model parameter α and the strength control parameters β are also investigated through the case studies. Major findings include:

(i) The proposed redistribution schemes become more effective (cause more improvements in the SAN reliability) as the SAN reliability threshold decreases (i.e., the system reliability requirement is less restrictive).

(ii) When α increases, the reliability improvements ratio does not have a consistent varying trend. Specifically, as α increases, the reliability improvement ratio increases under Scheme 1, decreases under Schemes 2 and 4, increases first then decreases under Scheme 3.

(iii) When β increases, the reliability improvement ratio increases under Schemes 1, 3 and 4 but decreases under Scheme 2.

An application of the proposed scheme to accomplish a mission with a specified duration via multiple load redistributions is also showcased.

In the future, we plan to extend the proposed work to explore and develop resilience metrics for SANs and study effective mitigation schemes to build the resilience of SANs against cascading failures (Xing, 2020). Different from the reliability metric addressed in this work that is concerned with the system's ability to perform the intended function for a certain time under a specified environment, the resilience metric is concerned with the system's ability to withstand and recover from failures or hazardous conditions.

Conflict of Interest

The authors confirm that there is no conflict of interest to declare for this publication.

Acknowledgments

This research did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors. The authors thank the editor and anonymous reviewers for their comments that help improve the quality of this work.

References

Bialek, J., Ciapessoni, E., Cirio, D., Sanchez, E., Dent, C., Dobson, I., Henneaux, P., Hines, P., Jardim, J., Miller, S., Panteli, M., Papic, M., Pitto, A., Tortos, J., Wu, D. (2016). Benchmarking and validation of cascading failure analysis tools. *IEEE Transactions on Power Systems*, *31*(6), 4887-4900.

Bright, L., & Raschid, L. (2000). Efficient remote data access in a mobile computing environment. In *Proceedings 2000 International Workshop on Parallel Processing* (pp. 57-64). IEEE. Toronto, ON, Canada.

Chen, X., Qiu, J., Reedman, L., & Dong, Z.Y. (2019). A statistical risk assessment framework for distribution network resilience. *IEEE Transactions on Power Systems*, *34*(6), 4773-4783.

Drake, C. (2022). Python EDA Documentation. Available: https://pyeda.readthedocs.io/en/latest/. Accessed in September 2022.

Garber, L. (2012). Converged infrastructure: Addressing the efficiency challenge. *Computer*, *45*(8), 17-20.

Honma, S., Morishima, H., Tsukiyama, T., Matsushima, H., Oeda, T., & Tomono, Y. (2004). Computer system using a storage area network and method of handling data in the computer system. Available: https://www.google.com/patents/US20040073677. Accessed in September 2022.

Hutanu, A., Allen, G., & Kosar, T. (2010). High-performance remote data access for remote visualization. In *2010 11th IEEE/ACM International Conference on Grid Computing* (pp. 121-128). IEEE. Brussels, Belgium.

Jiang, M., Zhou, J., & Hu, M. (2007). Fuzzy reliability analysis of an iSCSI-based fault tolerant storage system organization. In *Fourth International Conference on Fuzzy Systems and Knowledge Discovery (FSKD 2007)* (Vol. 4, pp. 598-602). IEEE. Haikou, China.

Kay, R., & Kinnersley, N. (2002). On the use of the accelerated failure time model as an alternative to the proportional hazards model in the treatment of time to event data: a case study in influenza. *Drug Information Journal*, *36*(3), 571-579.

Khanal, S.P., Sreenivas, V., & Acharya, K.S. (2014) Accelerated failure time models: an application in the survival of acute liver failure patients in India. *International Journal of Science and Research*, *3*(6), 161-166.

Levitin, G., & Amari, S.V. (2009). Optimal load distribution in series–parallel systems. *Reliability Engineering & System Safety*, *94*(2), 254-260.

Li, J., Wang, Y., Zhong, J., Sun, Y., Guo, Z., Chen, Z., & Fu, C. (2022). Network resilience assessment and reinforcement strategy against cascading failure. *Chaos, Solitons & Fractals*, *160*, 112271. https://doi.org/10.1016/j.chaos.2022.112271.

Lin, D.Y., Wei, L.J., & Ying, Z. (1998). Accelerated failure time models for counting processes. *Biometrika*, *85*(3), 605-618.

Lin, S., Lu, M., & Chiueh, T.C. (2007). Transparent reliable multicast for ethernet-based storage area networks. In *Sixth IEEE International Symposium on Network Computing and Applications (NCA 2007)* (pp. 87-94). IEEE. Cambridge, MA, USA.

Lv, G., & Xing, L. (2021). Influence of load on reliability of storage area networks. *International Journal of Mathematical, Engineering and Management Sciences*, *6*(6), 1533-1552.

Mishra, S., Anderson, K., Miller, B., Boyer, K., & Warren, A. (2020). Microgrid resilience: A holistic approach for assessing threats, identifying vulnerabilities, and designing corresponding mitigation strategies. *Applied Energy*, *264*, 114726. https://doi.org/10.1016/j.apenergy.2020.114726.

Nguyen, T., Liu, B., Nguyen, N., Dumba, B., & Chou, J. (2021). Smart grid vulnerability and defense analysis under cascading failure attacks. *IEEE Transactions on Power Delivery*, *36*(4), 2264-2273.

Qiu, X., Telikepalli, R., Drwiega, T., & Yan, J. (2005). Reliability and availability assessment of storage area network extension solutions. *IEEE Communications Magazine*, *43*(3), 80-85.

Shi, L., Shi, Z., Yao, L., Ni, Y., & Bazarga, M. (2010). A review of mechanism of large cascading failure blackouts of modern power system. *Power System Technology*, *34*(3), 48-54.

Uwaechia, A.N., & Akinsanmi, O. (2013). Reliability assessment on the performance model of Ahmadu Bello University data network repositories for storage area network design. *International Journal of Innovative Research in Science, Engineering and Technology*, *2*(7), 3311-3315.

Wang, J., Rong, L., Zhang, L., & Zhang, Z. (2008). Attack vulnerability of scale-free networks due to cascading failures. *Physica A: Statistical Mechanics and its Applications*, *387*(26), 6671-6678.

Xing, L. (2020). Reliability in internet of things: Current status and future perspectives. *IEEE Internet of Things Journal*, *7*(8), 6704-6721.

Xing, L. (2021). Cascading failures in internet of things: Review and perspectives on reliability and resilience. *IEEE Internet of Things Journal*, *8*(1), 44-64.

Xing, L., & Amari, S.V. (2015). *Binary decision diagrams and extensions for system reliability analysis*. Wiley-Scrivener, MA, Canada.

Xing, L., & Dugan, J.B. (2002). Analysis of generalized phased mission system reliability, performance and sensitivity. *IEEE Transactions on Reliability*, *51*(2), 199-211.

Xing, L., Morrissette, B.A., & Dugan, J.B. (2014). Combinatorial reliability analysis of imperfect coverage systems subject to functional dependence. *IEEE Transaction on Reliability*, *63*(1), 367-382.

Xing, L., Tannous, M., Vokkarane, V.M., Wang, H., & Guo J. (2017). Reliability modeling of mesh storage area networks for Internet of things. *IEEE Internet of Things Journal*, *4*(6), 2047-2057.

Zhou, J., Coit, D.W., Felder, F., & Wang, D. (2021). Resiliency-based restoration optimization for dependent network systems against cascading failures. *Reliability Engineering & System Safety*, *207*, 107383. https://doi.org/10.1016/j.ress.2020.107383.

Publisher's Note- Ram Arti Publishers remains neutral regarding jurisdictional claims in published maps and institutional affiliations.