The Risk of Successive Disasters: A Blow-by-Blow Network Vulnerability Analysis

Jorik Oostenbrink
Delft University of Technology
J.Oostenbrink@tudelft.nl

Fernando Kuipers
Delft University of Technology
F.A.Kuipers@tudelft.nl

Abstract—It is often assumed that a network will not be struck by multiple disasters in a relatively short period of time; that is, a subsequent disaster will not strike within the recovery phase of a previous disaster. However, recent events have shown that combinations of disasters are not implausible. This realization calls for a new perspective on how we assess the vulnerability of our networks and shows a need for a framework to assess the vulnerability of networks to successive independent disasters.

We propose a network and disaster model capable of modeling a sequence of disasters in time, while taking into account recovery operations. Based on that model, we develop both an exact and a Monte Carlo method to compute the vulnerability of a network to successive disasters. By applying our approach to real empirical disaster data, we show that the probability of a second disaster striking the network during recovery can be significant even for short repair times. Our framework is a first step towards determining the vulnerability of networks to such successive disasters.

I. INTRODUCTION

Disasters can inflict significant damage on networks. The 2011 earthquake near the coast of Japan, for example, caused extensive damage to telecommunications buildings and equipment. The total cost of emergency restoration and reconstruction of the local NTT East network was around 80 billion yen (1 billion dollars at the time) [1]. Large network outages such as these can have a massive impact on our economy and further exacerbate the impact of disasters on society. Hence, efforts into developing new methods to improve the resilience of communication networks to disasters have increased significantly in the last decade.

The rate at which disasters strike an area is typically very low. Therefore, it is commonly assumed that a network will only be affected by single (possibly composite\(^1\)) isolated disasters. The probability that two or more independent disasters will occur shortly after one another is seen as negligible and safe to ignore. Recent events have shown that this assumption might not be as rock solid as first thought.

The 2017 Atlantic hurricane season was extremely active and, due to global warming, the intensity of hurricanes is projected to keep increasing [2]. The continental United States was hit by 3 hurricanes (Harvey, Irma, and Nate), of which two where categorized as major hurricanes (Harvey and Irma) [3]. Hurricane Irma hit the East Coast only 16 days after Harvey [4], [5]. Out of the top 5 costliest US mainland tropical cyclones on record, 3 occurred in 2017 [6].

In total, there were 16 billion-dollar weather and climate disaster events in the United States in 2017 [7]. The total cost of these events exceeded 300 billion dollars. For the past five years (2013-2017), the United States has had an average of 11.6 major disasters per year with a cost of more than 1 billion dollars.

Also in 2017, Mexico was hit by two major earthquakes in two weeks (where the second quake is not considered an aftershock of the first [8]), leading to a combined economic loss of nearly 6 billion dollars [9], [10].

Recovering a network after a disaster can take several weeks to months, as a large amount of hardware will need to be replaced or repaired in a potentially very inaccessible area [1]. In the context of this paper, a network is said to be affected by multiple successive disasters if a disaster strikes the network during its recovery from a previous disaster. Depending on the moment in the recovery phase when the next disaster occurs, the total impact and final recovery time will differ significantly.

To increase the resilience of our networks to disasters, it is essential to be able to compute the vulnerability of networks to these disasters. While previous work has been instrumental in computing the vulnerability of a network to a single disaster, it has not addressed multiple successive disasters. In this paper, we propose a framework to assess the vulnerability of a network to successive disasters. Our main contributions are as follows:

- We compose a network and disaster model capable of modeling a sequence of disasters in time (Sec. II).
- We develop a method to compute the vulnerability of a network to successive disasters by modeling the network state as a discrete-time Markov chain (Sec. IV). Our methodology allows for arbitrary precision by only computing the effect of at most \( k \) successive disasters, with corresponding error bounds. Our results for the Markov chain are subsequently used to derive a faster Monte Carlo method in Sec. V.
- We apply our methods to empirical disaster data in Sec. VI. These experiments show that the probability of a second disaster striking the network during recovery can be significant, even for short repair times.
To the best of our knowledge, we are the first to propose models and methods for assessing the impact of successive disasters on networks, while taking into account recovery operations.

II. NETWORK AND DISASTER MODEL

We model the network as a directed multigraph \( G = (V, E, \psi) \) with nodes \( v \in V \) connected by links \( e \in E \), where \( \psi : E \rightarrow V \times V \) and \( e \in E \) connects \( v_1 \) to \( v_2 \) if and only if \( \psi(e) = (v_1, v_2) \). Thus, we permit the same pair of nodes to be connected by multiple links. We define a failure set \( s \), where network component \( c \in V \cup E \) is functioning if and only if \( c \notin s \). In the remainder of the paper, we refer to the failure set of a network as the state of that network.

Given such a network, we are interested in three factors: (1) the number of successive disasters we can expect the network to be struck by, (2) the impact of being struck by one or more disasters, and (3) the total time it takes to fully recover from these disasters. To assess these attributes, we need to model the occurrence of disasters over time.

The occurrence of disasters is inherently unpredictable. A common stochastic model for disaster occurrences \([11]–[13]\), which we will also employ, is the Poisson process. We model all disaster processes as mutually independent Poisson processes and assume we are given a multiset of disaster processes \( d = (a_d, \lambda_d) \in D^\ast \), where \( a_d \subseteq V \cup E \) are the components affected by \( d \) and \( \lambda_d \) is the rate of \( d \).

If disaster process \( d \) triggers at time \( t \), when the network state is \( s \), the new network state at time \( t \) will be \( s \cup a_d \). That is, all components in \( a_d \) fail. We assume at most one disaster can strike the network at any given time \( t \).

The combination of multiple Poisson processes is again Poissonian, with as rate the sum of its component rates. Thus, we can merge all disaster processes that affect the same components without affecting the outcome of our analysis. Hence, we transform the set \( D^\ast \) to

\[
D = \{(a_d, \lambda_d) | a_d \neq \emptyset \land \lambda_d = \sum_{(a_d, \lambda_{d'}) \in D^\ast} \lambda_{d'} > 0\}
\]

Let \( (T_n)_{n=1}^\infty \) be the ordered sequence such that \( T_1 \) is the occurrence time of the first disaster, and for all \( n > 1 \), \( T_n \) is the time between disasters \( n-1 \) and \( n \). Let \( (D_n)_{n=1}^\infty \) be the ordered sequence of disasters. In other words, the first disaster \( D_1 \in D \) occurs at time \( T_1 \in \mathbb{R} \), the second \( D_2 \in D \) at \( T_1 + T_2 \in \mathbb{R} \), etc. Then, for all \( n \in \mathbb{N} \):

\[
T_n \sim \text{Exp}(\lambda_D) \text{ (where } \lambda_D := \sum_{(a_d, \lambda_d) \in D} \lambda_d) \]

the \( T_n \) are exponentially distributed with rate \( \lambda_D \), and \( D_n \) and \( T_n \) are independent for all \( n \in \mathbb{N} \):

\[
P(D_n = d \land T_n = t) = P(D_n = d)P(T_n = t)
\]

A. Example Network and Disasters Instance

To illustrate our network and disaster model, we give an example in Fig. 1. We consider a small triangle network of 3 nodes and 3 links. Its representational set of disasters contains four (types of) disasters. As each of these disasters affects a different set of components, \( D^\ast = D \). The total disaster rate is \( \lambda_D = 1.6 \) disasters per year.

A network topology and set of disasters are not sufficient to properly compute the vulnerability of the network to successive disasters, as the impact of these disasters significantly depends on how quickly, and in what order, the network can be repaired. Thus, we also need to include some repair properties.

Our framework can include any repair function, but in the example the following repair rules hold: nodes can be repaired in half a month, while links take a full month to repair, and repairs are performed according to a predetermined priority and cannot be performed concurrently.

III. PROBLEM STATEMENT

We consider a deterministic repair model. We assume that, given a certain starting state, the recovery of the network is fixed (until a new disaster occurs). For example, if disaster 4 of the example instance occurs, all nodes will be damaged. Afterwards, the nodes will be repaired one by one. Thus, unless another disaster occurs during repair, the state of the network will be

- \( \{n1, n2, n3\} \) at time 0
- \( \{n2, n3\} \) at time \( \frac{1}{3} \)
- \( \{n3\} \) at time \( \frac{2}{3} \)
- \( \emptyset \) at time 1

Generalizing the above example, we define repair functions \( r_{s_0} : \mathbb{R}^+ \rightarrow V \cup E \) for each \( s_0 \in V \cup E \). \( r(t)_{s_0} \in V \cup E \) is the state of the network at time \( t + C \), given that the state of the network was \( s_0 \) after being struck by a disaster at some time \( C \). We assume the network does not degrade further in the recovery phase:

\[
r(b)_{s_0} \subseteq r(a)_{s_0} 0 \leq a \leq b, s_0 \in V \cup E
\]

Different repair strategies can be compared by changing the repair functions. Additionally, by increasing the amount of components being repaired simultaneously, the benefits of acquiring more personnel can be assessed and compared to the additional cost in salary.

In the following, we elaborate on our research objectives with respect to three properties.

A. Number of Successive Disasters \( N \)

Network operators should decide on how many successive disasters they prepare for. To do so, knowing the probability of at least \( n \) successive disasters is essential. In addition, the expected number of successive disasters is also of interest. Hence, our goal is to compute \( P(N \geq n) \), as well as \( E[N] \).

B. Impact

While knowing the expected number of successive disasters is useful, it is also important to consider their impact. Suppose we have a measure \( M : V \times X \rightarrow [0, 1] \) that assigns a value \( M(s) \) between 0 (worst case) and 1 (best case) to each state \( s \) of the network. We require that \( M(a) \leq M(b) \) if \( b \subseteq a \).
We analyze the minimum value of M during the disaster-
and-recovery process. In the one-disaster case, this would
simply be the value of M directly after the disaster. Successive
disasters, although rare, can have a significantly higher impact
on the network than single disasters. Therefore, given a critical
value m, we want to compute the probability that the network
reaches a state at least as bad as m during the disaster-
and-recovery process, \( P(M_{\text{min}} \leq m) \), where \( M_{\text{min}} \) is the
minimum value of M between \( T_1 \) and full recovery.

C. Total Time to Full Recovery

Let \( T_{\text{total}} \) be the total repair time, from the start of the
first disaster to the time when all damage from all previous
disasters has been repaired. We aim to compute the expected
time to full recovery, \( E[T_{\text{total}}] \).

IV. ANALYSIS

In this section, we describe methods for computing the
properties introduced in the previous section by modeling
the state of the network as a Discrete-Time Markov Chain
(DTMC).

A. Markov Chain

Let \( A_n \) be the state of network \( G \) directly after the \( n \)th dis-
aster strikes the network. Now, because the disaster processes
are independent and memoryless, and the repair function is
deterministic,

\[
P(A_n = a_n | A_1 = a_1, A_2 = a_2, \ldots, A_{n-1} = a_{n-1}) =
\]

that is, \((A_n)_{n=1}^\infty\) satisfy the Markov property and form a
(discrete-time) Markov chain.

The transition probabilities of this Markov chain depend on
which disaster strikes next, as well as at which stage of the
repair process this disaster strikes. By property (3), these two
factors are independent. Thus, the transition probabilities can
be calculated by summing over all possible disasters \( d \in D \):

\[
P(A_n = a_n | A_{n-1} = a_{n-1}) =
\]

Here, \( \frac{\lambda_d}{\lambda_D} \) is the probability that the network will be struck
by disaster \( d = (a_d, \lambda_d) \). \( [M_{a_{n-1}, d, a_n}, S_{a_{n-1}, d, a_n}] \) is the
period of time during which the occurrence of disaster \( d \)
will result in network state \( a_n \) and \( \exp(-\lambda_D M_{a_{n-1}, d, a_n}) - \exp(-\lambda_D S_{a_{n-1}, d, a_n}) \) the probability that the next disaster will
occur in this period of time².

We are specifically interested in the chain of network states
until full recovery. Thus, we construct an additional Markov chain \((S_n)_{n=1}^\infty\) by adding an absorbing state \( \emptyset \) to \((A_n)_{n=1}^\infty\)
such that \( S_n = \emptyset \) if and only if the network has been fully
repaired.

Let \( R_s := \min\{t \geq 0 | r(t) = \emptyset\} \) be the time it takes
to fully repair the network (assuming no subsequent disasters occur),
starting from network state \( s \in V \cup E \). The proba-
bility that, starting in state \( s \), the network is fully recovered before the next disaster strikes is \( \exp(-\lambda_D R_s) \). Therefore, the
transition probabilities to the absorbing state \( \emptyset \) are

\[
P(S_n = \emptyset | S_{n-1} = s_{n-1}) =
\]

and the transition probabilities to all other states are

\[
P(S_n = s_{n-1} | S_{n-1} = s_{n-1}) =
\]

\[
S_1 = A_1 = a_{D_1}, \text{ so the initial distribution of the Markov chain } (S_n)_{n=1}^\infty \text{ is }
\]

\[
P(S_1 = s_1) = \left\{ \begin{array}{ll}
\frac{\lambda_d}{\lambda_D} & \exists d \in D \text{ s.t. } a_d = s_1 \\
0 & \text{otherwise}
\end{array} \right.
\]

B. Number of Successive Disasters \( N \)

We can now compute the probability \( P(N \geq n) = 1 -
P(S_n = \emptyset) \) of at least \( n \) successive disasters without full
recovery. This probability decreases exponentially with \( n \).

Lemma 1:

\[
P(N \geq n) \leq (1 - \exp(-\lambda_D R))^n
\]

²\( M_{a_{n-1}, d, a_n} \) is the first time at which \( r_{a_{n-1}} \cup a_d = a_n \) (or \( \infty \) if no such time exists), and \( S_{a_{n-1}, d, a_n} \) is the first time after \( M_{a_{n-1}, d, a_n} \) at
which \( r_{a_{n-1}} \cup a_d \neq a_n \) (or \( \infty \)).
where \( R := \max_{s \in V \cup E} R_s \).

**Proof:** See Appendix.

**Remark 1:** If \( R_s = R \forall s \in V \cup E - \emptyset \), then
\[
P(N \geq n) = (1 - \exp(-\lambda_D R))^n - 1
\]

Typically, \( R = \max_{s \in V \cup E} R_s \) will be the amount of time it takes to repair all network components \( (R_{V \cup E}) \).

Unfortunately, computing \( E[N] \) directly is intractable in most cases, as the number of possible states can be as high as \( 2^{|V|+|E|} \). However, we can approximate (from below) the expected number of successive disasters by only constructing the Markov model for \( k \) successive disasters and computing the distribution of \( S_1 \) to \( S_k \). The choice of \( k \) depends on the required accuracy.

**Theorem 1 (Stopping conditions 1):** Let \( \hat{E}[N] = \sum_{n=1}^{\infty} P(N \geq n), \) then
\[
0 \leq E[N] - \hat{E}[N] \leq \frac{(1 - \exp(-\lambda_D R))^k}{\exp(-\lambda_D R)}
\]

In addition, if \( P(N \geq k) \leq \epsilon \frac{\exp(-\lambda_D R)}{1 - \exp(-\lambda_D R)}, \) then
\[
E[N] - \hat{E}[N] \leq \epsilon
\]

**Proof:** We start by proving (11).
\[
E[N] - \hat{E}[N] = \sum_{n=k+1}^{\infty} P(N \geq n)
\leq \sum_{n=k+1}^{\infty} (1 - \exp(-\lambda_D R))^{n-1} \quad \text{(Lemma 1)}
\leq \frac{(1 - \exp(-\lambda_D R))^k}{\exp(-\lambda_D R)}
\]

If \( P(N \geq k) \leq \epsilon \frac{\exp(-\lambda_D R)}{1 - \exp(-\lambda_D R)}, \) then (for \( n \geq k\):
\[
P(N \geq n) \leq \epsilon \exp(-\lambda_D R)(1 - \exp(-\lambda_D R))^{n-k-1}
\]

This can be proved analogously to Lemma 1. But this means that the absolute error
\[
E[N] - \hat{E}[N] \leq \sum_{n=k+1}^{\infty} \epsilon \exp(-\lambda_D R)(1 - \exp(-\lambda_D R))^{n-k-1}
\leq \sum_{n=0}^{\infty} \epsilon \exp(-\lambda_D R)(1 - \exp(-\lambda_D R))^n
\leq \epsilon
\]

Thus, to guarantee an upper bound on the absolute error, we can either choose the number of steps \( k \) beforehand, or test if \( P(N \geq k) \) is below the threshold after every iteration, where the latter requires fewer iterations than the former.

**C. Impact**

As \( M \) is minimal directly after a disaster, \( M_{min} = \min_n M(S_n) \). The cumulative distribution function \( \hat{P}(M_{min} \leq m) \) is the hitting probability of \( M^{\leq m} := \{s \in V \times E | M(s) \leq m\} \). We can take a similar approach as before and approximate these probabilities as
\[
\hat{P}(M_{min} \leq m) := P(M_{min}^k \leq m)
\]

where \( M_{min}^k = \min_{n \leq k} M(S_n) \).

Suppose we have computed the first \( k \) states and corresponding transition probabilities of the Markov chain \( (S_n)_{n=1}^{\infty} \). To compute \( P(M_{min}^k \leq m) \) we construct a new Markov chain \( (S_n^\leq m)_{n=1}^{\infty} \) by replacing all \( s \in M^{\leq m} \) with a single absorbing state \( A^{\leq m} \). Now,
\[
P(M_{min}^k \leq m) = P(S_n^\leq m = A^{\leq m})
\]

**Theorem 2 (Stopping conditions 2):** Let
\[
\hat{P}(M_{min} \leq m) = P(M_{min}^k \leq m) = P(S_n^\leq m = A^{\leq m})
\]

Then
\[
0 \leq P(M_{min} \leq m) - \hat{P}(M_{min} \leq m) \leq 1 - \hat{P}(M_{min} \leq m) - P(S_n^\leq m = \emptyset) \leq P(N \geq k)
\leq (1 - \exp(-\lambda_D R))^{k-1}
\]

**Proof:** If \( m \geq 1 \), then
\[
P(M_{min} \leq m) = \hat{P}(M_{min} \leq m) = 1, \quad \text{so we assume that} \quad m < 1.
\]

In this case
\[
P(M_{min} \leq m) = \hat{P}(M_{min} \leq m) = P(M_{min} \leq m) - P(M_{min}^k \leq m)
= P(M_{min} \leq m \land M_{min}^k > m)
\leq 1 - P(M_{min}^k \leq m) - P(S_n^\leq m = \emptyset)
\leq P(N \geq k)
\]

**D. Total Time to Full Recovery**

The total time to full recovery, or the total repair time, \( T_{total} \), is equivalent to the sum of the time spent on repair in all states of \( (S_n)_{n=1}^{\infty} \):
\[
T_{total} = \sum_{n=1}^{\infty} R_n
\]

where \( R_n \) is the time spent on repairs between the \( n \)th and \((n + 1)\)th disaster. Thus, \( R_n \) is 0 if \( S_n = \emptyset \) and \( R_n \) is the minimum between the total repair time of failures \( S_n \) and the time till the next disaster otherwise:
\[
R_n = \begin{cases} \min(R_{S_n}, T_{n+1}) & \text{if } S_n \neq \emptyset \\ 0 & \text{if } S_n = \emptyset \end{cases}
\]
The expected value of $R_n$ is

$$E[R_n] = \sum_{s \neq \emptyset} P(S_n = s) \int_0^\infty \lambda_D \exp(-\lambda_D t) t dt + \exp(-\lambda_D R_s) R_s$$

$$= \sum_{s \neq \emptyset} P(S_n = s) \left( \frac{1}{\lambda_D} (1 - \exp(-\lambda_D R_s)) \right)$$

$$= \frac{1}{\lambda_D} \sum_{s \neq \emptyset} P(S_n = s) (1 - \exp(-\lambda_D R_s)) \quad (18)$$

As before, we propose approximating $E[T_{\text{total}}]$ by truncating (16). That is, we approximate $E[T_{\text{total}}]$ by summing the expected values of $R_1$ to $R_k$, which only requires the distributions of $S_1$ to $S_k$.

**Theorem 3 (Stopping conditions 3):** Let $\hat{E}[T_{\text{total}}] := \sum_{n=1}^k E[R_n]$, then

$$0 \leq E[T_{\text{total}}] - \hat{E}[T_{\text{total}}] \leq \frac{(1 - \exp(-\lambda_D R))^k}{\lambda_D \exp(-\lambda_D R)} \quad (19)$$

In addition, if $P(N \geq k) \leq \epsilon \lambda_D \exp(-\lambda_D R)$, then

$$E[T_{\text{total}}] - \hat{E}[T_{\text{total}}] \leq \epsilon \quad (20)$$

**Proof:** By the monotone convergence theorem,

$$E[T_{\text{total}}] = E[\sum_{n=1}^\infty R_n] = \sum_{n=1}^\infty E[R_n]$$

In addition, by (18), $E[R_n] \leq \frac{1}{\lambda_D} P(N \geq n)$. Now, the proof follows analogously to that of Theorem 1.

**V. MONTE CARLO**

The Markov chain in Sec. IV has a large number of states. Most of these states have a very small probability of ever being reached. However, we can not simply ignore these states, as the aggregate of their probabilities is relatively high. This is a perfect use case for Monte Carlo simulations.

We propose an efficient Monte Carlo method, based on the results from Sec. IV, for estimating $P(N \geq n)$, $E[N]$, $E[R_{\min}]$, and $E[T_{\text{total}}]$. The method is given in detail in Fig. 2. The main idea is to simulate many sequences of successive disasters simultaneously, and cut off these sequences when the error bounds on the values of interest are small enough. As all sequences are cut off after the same number $n$ of successive disasters, we only allow transitions to subsequent disaster states and keep track of the probability of reaching the absorbing state separately. This allows us to closer estimate the values of interest.

In essence, we approximate the lower bounds described in Sec. IV. By Theorems 1 to 3, these lower bounds, combined with $P(N \leq n)$, give us the upper bounds as well. The method can be tuned with respect to two values: Stopping condition $\beta$ gives the maximum difference between the approximated bounds, while the number of simulations $\eta$ can be adjusted to affect the accuracy of the approximation of the bounds itself. When the probability of subsequent disasters is too high, lowering $\beta$ can keep computation times manageable by reducing the number of successive disasters taken into account.

**VI. EXPERIMENTS**

To demonstrate our methods, we apply them to a version of the Sinet topology (Fig. 3) from the Topology Zoo [14], where all nodes without geographical information have been removed. This backbone network of 47 nodes connected by
49 bidirectional links is located in Japan, and hence is vulnerable to a variety of different disasters such as earthquakes, landslides, and typhoons. All experiments are performed on an Intel Xeon Processor E5-2620 v3.

A. Dataset

We create a set of disasters $D^*$ by combining datasets from two sources: (1) the Japan Seismic Hazard Information Station (J-SHIS) [15] and (2) the International Best Track Archive for Climate Stewardship (IBTrACS) [16].

1) Earthquake Data (J-SHIS): The National Research Institute for Earth Science and Disaster Resilience (NIED) provides a large amount of data on Japanese earthquakes through the Japan Seismic Hazard Information Station (J-SHIS). We use the 2016 version of this dataset. J-SHIS provides maps of the effect of a significant number of modeled earthquakes: the Scenario Earthquake Shaking Maps. These maps give, among other data, the JMA seismic intensities for each affected Divided Quarter Grid Square [17] cell in Japan.

We create a disaster process $d \in D^*$ for each earthquake scenario. The affected components $a_d$ of each scenario are the set of network components that intersect (or lie within) one or more grid cells with a seismic intensity larger than or equal to 5.5. The disaster rates $\lambda_d$ are the inverse of the mean recurrence intervals of each fault, divided by the total number of scenarios of the fault.

2) Tropical Cyclone Data (IBTrACS): IBTrACS is a collection of tropical cyclone data from numerous agencies maintained by the National Centers for Environmental Information (NCEI) of the (U.S.) National Oceanic and Atmospheric Administration (NOAA). In our experiments, we use IBTrACS beta version 4 and limit ourselves to cyclones from 1980 to 2017. We filter out any storms that never reached wind speeds of 74 mph, leaving us with a set of 1649 historical storms. As disaster area, we would prefer to use the regions that reached 74 mph winds. Unfortunately, this information is only available for some storms (in the form of the radius maximum extent per quadrant). Therefore, we apply the concept of the hurricane strike circle instead.

A strike circle is a circle with diameter 231.5 km, centered 23.15 km to the right of the hurricane center (based on its direction of motion). It is meant to depict the typical extent of hurricane force winds [18].

For each typhoon-level storm, we find the first registered center point $p_a$ where the storm had a maximum sustained wind speed of at least 74 mph, as well as the last center point $p_b$ with at least 74 mph maximum sustained wind speed. Then, we select the range of center points from $p_a$ up to and including the first registered center point after $p_b$. Connecting these points forms a track. $a_d$ is the selection of all components within or intersecting a strike circle of any point (including points on the line segment between registered center points) on this track. The resulting set of disasters includes many storms that do not affect any components of Sinet (e.g. hurricanes striking the U.S.). However, this is not an issue, as empty $a_d$ are filtered out when generating $D$.

The final set $D^*$ is the union of the earthquake scenarios and historical tropical cyclones. This set of 2304 potential disasters can be reduced to a set $D$ of 160 unique scenarios affecting Sinet. The total rate $\lambda_D$ of these scenarios is 1.648 per year.

B. The Effect of Component Repair Time

We first examine the effect of repair time. In a one-disaster scenario, the relation between component repair time and total repair time is simple: Ignoring start-up time, if repairing components takes twice as long, the total time to full recovery will also take twice as long. However, if we take the possibility of multiple disasters into account, we encounter another effect of repair time: When the time to repair the network increases, so does the probability that the network will be struck by a subsequent disaster during recovery. These subsequent disasters further increase the expected total recovery time on top of the increase in component repair time itself. Our experiments show this effect can be significant.

We consider a situation where components are repaired one-by-one, using a greedy strategy that tries to maximize the number of connected node-pairs. We vary the time it takes...
time is 20 days, it can take more than 5 years to fully repair Sinet. Interestingly, even with a component repair time of less than 5

as could be expected, for more reasonable repair times rapidly increases with the (component) repair time. Although, we approximate all

could be expected results for higher repair times, we approximate all results. We use \( \eta = 10,000 \) simulations for each Monte Carlo approximation and set \( \beta = 0.05 \).

The expected number of successive disasters and the expected time to full network recovery are plotted in Fig. 4. \( E[N] \) rapidly increases with the (component) repair time. Although, as could be expected, for more reasonable repair times \( E[N] \) remains below 2. Due to the influence of subsequent disasters, \( E[T_{\text{total}}] \) grows exponentially in the component repair time.

Fig. 5 shows the probability of a subsequent disaster during recovery of the first disaster, \( P(N > 1) \). This value can be computed exactly by computing one step of the DTMC. Interestingly, even with a component repair time of less than 5

days, the probability of facing more than 1 disaster is relatively high. Probabilities of around 0.2, or even 0.1, are significant enough to stop ignoring the possibility of subsequent disasters.

Next, we consider the connection between repair time and network performance. To do so, we analyze the minimum value of the Average Two-Terminal Reliability (ATTR) survivability measure in the period after the first disaster strikes and before all damage has been repaired.

The ATTR of a network is the number of connected node pairs, divided by the total number of node pairs. We choose this metric because it is of vital importance that as many areas remain connected as possible after a disaster.

In Fig. 6, we have plotted \( E[\text{ATTR}_{\text{min}}] \) against the component repair time. While the repair time does affect the expected minimum ATTR, this effect is much smaller than that on the expected time to full recovery.

A similar outcome can be observed when computing the probability that at most half of all node pairs remain connected (Fig. 6). However, while \( P(\text{ATTR}_{\text{min}} \leq 0.5) \) increases relatively slowly with the repair time, \( P(\text{ATTR}_{\text{min}} \leq 0.1) \) increases much faster.

Fig. 7 shows the computation time of the Monte Carlo approximations against the component repair time.

The repair time has a significant effect on both the total recovery time and ATTR during the recovery process. Thus, reducing it, by repairing more components at once or by decreasing the time it takes to repair individual components, should be a high priority.

C. Concurrent Repair

To evaluate our methods, we consider a use-case in which multiple components can be repaired simultaneously. In addi-
TABLE I
COMPARISON OF THE EXACT RESULTS FROM SEC. IV AND THE RESULTS OF THE MONTE CARLO METHOD FROM SEC. V. THE EXACT COLUMN SHOWS THE LOWER AND UPPER BOUNDS OF THE VALUE. THE RUNTIME OF THE EXACT COMPUTATION ONLY INCLUDES THE TIME TO COMPUTE \( S_1 \) TO \( S_k \). THE MONTE CARLO APPROXIMATION IS OBTAINED BY PERFORMING 50,000 MONTE CARLO SIMULATIONS WITH STOPPING CONDITION \( P(N \geq n) \leq 0.0001 \exp(-\lambda_D R_f^*) \).

<table>
<thead>
<tr>
<th>Property</th>
<th>Exact</th>
<th>Monte Carlo</th>
</tr>
</thead>
<tbody>
<tr>
<td>( E[N] )</td>
<td>1.0850 - 1.0851</td>
<td>1.0851</td>
</tr>
<tr>
<td>( P(N &gt; 1) )</td>
<td>0.0763</td>
<td>0.0763</td>
</tr>
<tr>
<td>( P(\text{ATTR}_{\text{min}} \leq 0.5) )</td>
<td>0.3834 - 0.3834</td>
<td>0.3825</td>
</tr>
<tr>
<td>( P(\text{ATTR}_{\text{min}} \leq 0.1) )</td>
<td>0.0021 - 0.0022</td>
<td>0.0021</td>
</tr>
<tr>
<td>( E[T_{\text{total}}] ) (days)</td>
<td>19.4576 - 19.4674</td>
<td>19.4816</td>
</tr>
</tbody>
</table>

1-Threaded Computation Time (s) | 1.556.7398 | 120.1504 |

...}

...
done well in advance, this computation time can be considered to be very fast.

We have applied our model to empirical disaster data. Our experiments show that when considering successive disasters, the expected time to complete recovery grows exponentially in the time it takes to repair a network component. Additionally, the probability of a second disaster striking the network during recovery can be significant, even for short repair times. Our framework is a first step towards determining the vulnerability of a network to these successive disasters.

REFERENCES


APPENDIX A

PROOF OF LEMA 1

By induction:

Trivially, \( P(N \geq 1) = 1 \leq (1 - \exp(-\lambda_D R))^{0} \).

Now, suppose

\[
\forall k < n \, P(N \geq k) \leq (1 - \exp(-\lambda_D R))^{k-1},
\]

then

\[
P(N \geq n) = P(N \geq n - 1)P(N \geq n | N \geq n - 1) \leq (1 - \exp(-\lambda_D R))^{n-2} P(N \geq n | N \geq n - 1)
\]

By direct application of (7):

\[
P(N \geq n | N \geq n - 1) = 1 - \frac{1}{P(S_n - \emptyset | S_{n-1} - \emptyset)} \sum_{s \neq \emptyset} P(S_n - s | S_{n-1} - \emptyset) = 1 - \frac{1}{P(S_n - \emptyset | S_{n-1} - \emptyset)} \sum_{s \neq \emptyset} P(S_n - s | S_{n-1} - \emptyset) \exp(-\lambda_D R_s)
\]

\[
\leq 1 - \frac{1}{P(S_n - \emptyset | S_{n-1} - \emptyset)} \sum_{s \neq \emptyset} P(S_n - s | S_{n-1} = s) \exp(-\lambda_D R)
\]

So,

\[
P(N \geq n) \leq (1 - \exp(-\lambda_D R))^{n-1}
\]