

Measuring the Resiliency of Cellular Base Station Deployments

David Griffith, Richard Rouil, Antonio Izquierdo, and Nada Golmie
National Institute of Standards and Technology (NIST)
Gaithersburg, Maryland 20899–8920
Email: david.griffith@nist.gov

Abstract—The National Public Safety Telecommunications Council (NPSTC) has defined resiliency as the ability of a network to withstand the loss of assets and to recover quickly from such losses. How to measure the resiliency of a base station deployment is an important consideration for network planners and operators. In this paper, we propose a resiliency measurement method in conjunction with a performance metric such as coverage or supported throughput, where we define the resiliency as the maximum number of sites that can fail before the metric falls below a minimum acceptable threshold. Because the number of combinations of failures increases exponentially with respect to the number of sites in a given deployment, we introduce an algorithm that generates estimates of the lowest, highest, and average values of the metric for a given failure count while examining a subset of the possible failure combinations. We use an example deployment to demonstrate how the resiliency metric can be used to identify sites that have a disproportionate impact on performance; the network planner can harden these sites or, for a future deployment, adjust the site placement to reduce the effect of the high-impact sites.

I. INTRODUCTION

Modern public safety agencies depend on digital communications systems for situational awareness and to coordinate operations with other agencies. It is imperative that these systems be resistant to failure and be able to recover quickly after failures occur. The National Public Safety Telecommunications Council (NPSTC) has defined the combination of these two capabilities as a network’s *resiliency* [1, Section 5.2]. NPSTC has identified resiliency as an important design metric that will be used in the construction of the National Public Safety Broadband Network (NPSBN).

The various NPSBN stakeholders, (e.g., network planners, public safety agencies, and equipment vendors) have an interest in being able to quantitatively measure the resiliency of proposed and fielded networks. In this paper, we propose a simple resiliency measure with respect to a minimum acceptable value of a given performance metric (e.g., area coverage). The resiliency metric can help to quantify the ability of a deployment to recover rapidly from failures. Also, our resiliency analysis can show which sites are low-impact (i.e., their loss does not significantly reduce the given performance metric’s value) or high-impact (i.e., their loss causes a large reduction in performance). High-impact sites can be reinforced or augmented with backup systems to reduce the probability that they will be lost. A network operator can prioritize high-impact sites during repair and recovery operations.

Resiliency is a well-understood concept in computer network design. Early work in this area [2] defined resiliency in terms of the disconnection probability in network graphs. More recent work [3] has focused on detecting failures in wired networks by comparing multiple resiliency metrics to minimum thresholds and activating responses to maintain services when targets are missed. Our work extends this concept to the wireless case. While there has been work on wireless resiliency, it has tended to focus on the impact of mobile users while assuming a hexagonal grid deployment [4], or has focused on the robustness of network components, e.g., by developing analytical failure models [5]. These models assume that network element failure events are independent and identically distributed and that failure events follow Poisson arrival processes. While good for day-to-day operations, these models do not capture large-scale failure events. To the best of our knowledge, our study is the first to define resiliency in terms of a set of metrics and a set of corresponding thresholds for minimum acceptable performance over the range of possible base station failure combinations.

In Section II, we develop our method for obtaining the resiliency metric. In Section III, we describe our algorithm, which streamlines the analysis by reducing the number of combinations of site failures to simulate. In Section IV, we apply our approach to a hypothetical example base station deployment and use our metric to assess the resiliency of the deployment, identify low-impact and high-impact sites, and examine the effect of reducing the number of sites in the example deployment. We present our conclusions in Section V.

II. THE RESILIENCY METRIC

Our analysis uses InfoVista’s Mentum Planet cellular network planning tool, which allows a user to place base stations in a given geographical area and to specify their characteristics, such as antenna gain pattern, transmit power, and tower height¹. The tool also incorporates the effects of clutter, such as buildings, hills, and vegetation. With the base stations in place, the user can generate a prediction overlay, which gives

¹Certain commercial equipment, instruments, or materials are identified in this paper in order to specify the experimental procedure adequately. Such identification is not intended to imply recommendation or endorsement by the National Institute of Standards and Technology, nor is it intended to imply that the materials or equipment identified are necessarily the best available for the purpose.

the signal strength due to each base station at each location in the region of interest. The user can specify the physical layer characteristics of any end-user devices (e.g., smart phones), along with the traffic that they generate, and use these as input to the tool's Monte Carlo simulator. The simulator produces four coverage metrics: uplink (UL) coverage, downlink (DL) coverage, combined UL/DL (area) coverage, and population coverage. Other returned performance metrics include load, throughput, and the percentage of users served with a given reliability (i.e., user coverage)².

For any performance metric μ , we can define the resiliency with respect to μ as the number of sites that can fail while still maintaining at least a minimum performance level, μ_{\min} . Because we have an ensemble of *failure scenarios* (FS's) that result from a site deployment, we compute the site count resiliency with respect to the maximum, average, and minimum curves. A deployment is n -site resilient with respect to a given curve if n is the largest positive integer such that $\mu \geq \mu_{\min}$ for $f \leq n$, where f is the number of failed sites. For the example shown in Fig. 1, the hypothetical network is 0-site resilient with respect to the maximum value of μ , and 1-site resilient and 2-site resilient with respect to the mean and minimum values, respectively³. By combining the resiliencies associated with the worst- and minimum curves, we get a performance envelope similar to those generated for wired networks by Smith et al. [3, Fig. 3].

Given a site deployment consisting of N base stations, we obtain a set of P performance metrics $\{\mu_1, \mu_2, \dots, \mu_P\}$ when all N sites are active. If the given deployment consists of a small number of sites, we could proceed by constructing a complete set of FS's, where we define a FS to be the set of sites that have failed. For a given FS, we modify the deployment in Mentum Planet by deactivating the desired sites, generating a new prediction map, and finally running new Monte Carlo simulations to produce new values for each of our P metrics. This results in a set of values for each metric that we can plot versus the number of failed sites, as shown in Fig. 1. This figure depicts results associated with a hypothetical deployment where $N = 3$, and is intended for illustrative purposes only.

III. THE FAILURE SCENARIO PRUNING ALGORITHM

The exhaustive approach described in Section II does not scale, since an N -site deployment produces 2^N FS's (e.g., full analysis of a 30-site deployment would require us to examine more than 1 billion failure scenarios). For some deployments, the reduction in performance that results from a few site losses is severe enough that a full analysis is not necessary⁴. Since evaluating a single FS with Mentum Planet takes from minutes

²These metrics should be considered jointly when evaluating the overall resiliency of a deployment.

³A deployment can be m -site resilient with respect to one performance metric and n -site resilient with respect to a different metric, with $m \neq n$.

⁴However, there are $\binom{N}{n} = N!/(n!(N-n)!)$ FS's associated with n failures, so the exhaustive analysis can still be expensive; if (for example) $N = 30$, analyzing up to 8 failures requires examining 8.7 million FS's.

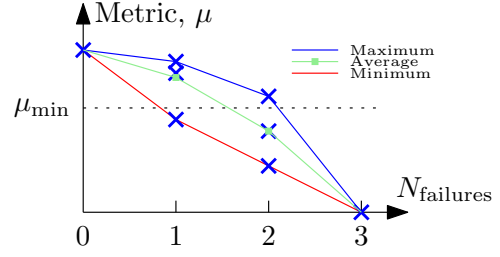


Fig. 1. Hypothetical plot of metric μ with respect to number of failed sites in a three-site deployment, with all failure scenario outcomes shown as 'x's.

to tens of minutes for small deployments (on the order of 10 sites) to tens of hours for large deployments involving dozens of sites and hundreds of users, an exhaustive evaluation of all FS's is not feasible.

We solved this problem by developing a heuristic algorithm that drastically reduces the number of failure scenarios that we have to simulate while still obtaining accurate estimates for the minimum, maximum, and average performance metrics versus the number of site failures.⁵

A. Algorithm Description

The algorithm first simulates the zero-failure case (i.e., FS \emptyset , which is the empty set because there are no failed sites) to compute the metric, $\mu(\emptyset)$, which it assigns to $\mu_{\min}(0)$, $\mu_{\max}(0)$, and $\mu_{\text{avg}}(0)$; these are respectively the 0th elements of the vectors of minimum, maximum, and mean value metrics that are the algorithm's outputs.

Next, the algorithm iteratively progresses through successive values of the site failure count, n . For $n = 1$, the algorithm examines all of the single-failure FS's by simulating each one and generating a corresponding value of the metric μ ⁶. Using this set of values, the algorithm identifies four FS's of interest: the FS for which the metric μ is a minimum, $\mu_{\min}(1)$, the FS for which μ is a maximum, $\mu_{\max}(1)$, and the two FS's whose associated values of μ are closest to $\mu_{\text{avg}}(1)$, which is the average value of μ taken over all the single-failure FS's. We retain these four FS's, which we respectively denote as $F_{\min}(1)$, $F_{\max}(1)$, and $F_{\text{avg}_\ell}(1)$ and $F_{\text{avg}_u}(1)$, where $\mu(F_{\text{avg}_\ell}(1)) \leq \mu(F_{\text{avg}_u}(1))$ ⁷.

For $n \geq 2$, the algorithm repeats the following process up to a desired maximum value of n . For each of the four retained FS's from the preceding iteration ($F_{\min}(n-1)$, $F_{\max}(n-1)$, $F_{\text{avg}_\ell}(n-1)$, and $F_{\text{avg}_u}(n-1)$), the algorithm constructs a set of FS's, each with n failures. Each FS in the set consists of the $n-1$ sites in the retained FS plus an additional site that is not in the retained FS (i.e., the added site was active when there were $n-1$ failures). One can view these new n -failure FS's as child FS's of the parent $(n-1)$ -failure FS.

⁵We tested the algorithm with the following metrics: DL/UL coverage, area coverage, population coverage, and the fraction of users served at 50 % and 95 % reliability, but show results for a subset due to space constraints.

⁶For clarity's sake, we use one metric in the description. The algorithm generates multiple metrics in parallel and simulates each FS once.

⁷In some cases, two or more of these are the same FS.

To get $\mu_{\min}(n)$, the algorithm creates $\mathcal{F}_{\min}(n)$, the set of n -failure FS's that branch from $F_{\min}(n-1)$:

$$\mathcal{F}_{\min}(n) = \{F_{\min}(n-1) \cup \{S\} \mid S \in \mathcal{S} \setminus F_{\min}(n-1)\},$$

where \mathcal{S} is the set of all the sites in the deployment and $\mathcal{S} \setminus F_{\min}(n-1)$ is the set of all sites S that are not in $F_{\min}(n-1)$. The algorithm simulates the site deployment generated by each FS in this set, and obtains a corresponding value for μ . Using the set of values of μ , it identifies the smallest one, $\mu_{\min}(n)$, and uses it to get the associated FS, $F_{\min}(n) \in \mathcal{F}_{\min}(n)$. The algorithm will use $F_{\min}(n)$ as the parent for the next iteration when it creates the next set of FS's, $\mathcal{F}_{\min}(n+1)$. The algorithm generates $\mu_{\max}(n)$, the estimated maximum value of μ with n failures, in a similar fashion. It creates a set of failure scenarios from $F_{\max}(n-1)$, simulates them, and uses the resulting set of values of μ to obtain $\mu_{\max}(n)$ and the associated FS $F_{\max}(n)$.

The algorithm estimates the mean of μ by generating the sets of n -failure FS's that branch from the retained $(n-1)$ -failure FS's $F_{\text{avg}_\ell}(n-1)$ and $F_{\text{avg}_u}(n-1)$ and taking the union of the two sets to produce the set of n -failure FS's, $\mathcal{F}_{\text{avg}}(n)$:

$$\begin{aligned} \mathcal{F}_{\text{avg}}(n) &= \{F_{\text{avg}_\ell}(n-1) \cup \{S\} \mid S \in \mathcal{S} \setminus F_{\text{avg}_\ell}(n-1)\} \\ &\quad \cup \{F_{\text{avg}_u}(n-1) \cup \{S\} \mid S \in \mathcal{S} \setminus F_{\text{avg}_u}(n-1)\} \\ &= \mathcal{F}_{\text{avg}_\ell}(n) \cup \mathcal{F}_{\text{avg}_u}(n). \end{aligned}$$

The algorithm simulates all of the networks in $\mathcal{F}_{\text{avg}}(n)$, generates values of μ for each, and uses them to compute the average, $\mu_{\text{avg}}(n) = \sum_{F \in \mathcal{F}_{\text{avg}}(n)} \mu(F) / |\mathcal{F}_{\text{avg}}(n)|$. The algorithm selects the two FS's whose associated values of μ are closest to $\mu_{\text{avg}}(n)$. The FS whose value of μ is the smaller of the two becomes $F_{\text{avg}_\ell}(n)$, and the other becomes $F_{\text{avg}_u}(n)$.

In order to increase its efficiency, the algorithm keeps track of the FS's that it simulates during each iteration, since it is possible that the sets of FS's may overlap (e.g., an FS in $\mathcal{F}_{\text{avg}}(n)$ may also be in $\mathcal{F}_{\min}(n)$). The algorithm does not simulate an FS that has already been simulated; the algorithm copies the FS's associated metric value μ and uses it to compute whatever statistic it is generating.

B. Algorithm Performance

We assessed the accuracy of the pruning algorithm by performing resiliency analyses on randomly chosen deployments from our previous simulation work [6]. In that study, we used Menthum Planet to estimate the deployment requirements for the NPSBN by partitioning the United States into a grid of 20 km \times 20 km square regions that we sorted into groups based on terrain type and population density. We used stratified random sampling to estimate the total number of sites that would be required for the NPSBN, given various assumptions about the network parameters.

In Fig. 2, we show an example plot of the percentage of users served with 95 % reliability versus the number of site failures. The figure shows metrics for all of the failure scenarios as \times 's and plots the true values of the maximum, minimum, and average metric values (using solid lines) and

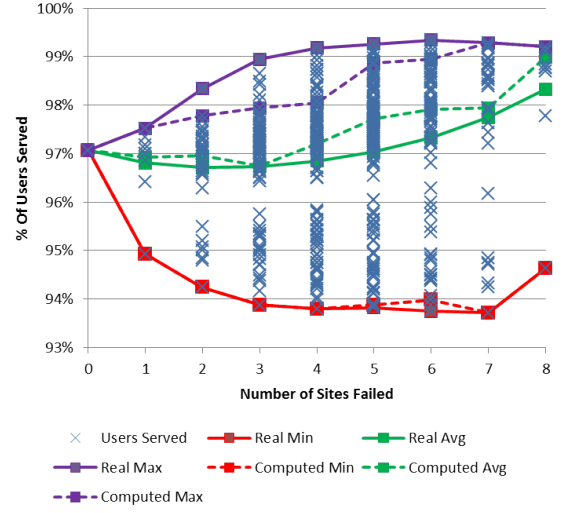


Fig. 2. Pruning algorithm and exhaustive simulation in a deployment consisting of eight sites.

the pruning algorithm's estimates of these quantities (using dashed lines). We observe the greatest error in the estimate of the maximum value of the metric, while the actual value and the result of the pruning algorithm agree much more closely for the minimum value of the metric.

We examined site deployments of various sizes (i.e., values of N). Fig. 3 shows the maximum observed error between the pruning algorithm and the actual values of the maximum, minimum, and average metric values over the set of deployments. As also indicated by Fig. 2, the pruning algorithm is most accurate when estimating the minimum metric value, and does less well for the average and maximum values. Also, in no case did the minimum and average metric estimation errors exceed 1.5 %, while the error in the maximum metric's estimate was at most about 3.5 %. These results indicate that the pruning algorithm provides reasonably accurate estimates of the maximum, minimum, and average values of μ , at a lower computational cost than an exhaustive evaluation, as we will show.

The pruning algorithm's relatively low error rate for the minimum metric appears to be due to the minimum performance's typically being driven by a small set of high-impact sites; the pruning algorithm works well in this case since the FS associated with the lowest value of the performance metric when there are n failures tends to be the minimum branch from the FS associated with the lowest-value performance when there are $n-1$ failures. In contrast, for the average and maximum metrics, there are many combinations of sites that result in similar performance with a given number of failures, and the pruning algorithm is more likely to follow a suboptimal path in these cases.

The number of FS's that the pruning algorithm examines varies; greater overlap between the sets $\mathcal{F}_{\min}(n)$, $\mathcal{F}_{\max}(n)$, and $\mathcal{F}_{\text{avg}}(n)$ means that fewer FS's need to be simulated. If these sets overlap completely for each n , the total number of

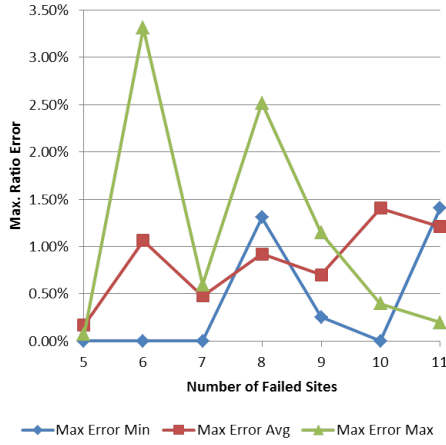


Fig. 3. Maximum error due to pruning algorithm in maximum, minimum, and average metric estimates.

examined FS's is $1 + N + (N-1) + \dots + 2 + 1 = (N^2 + N + 2)/2$. For the no overlap case, we have to examine 1 FS when $n = 0$ (the empty set) and N FS's when $n = 1$; we assume that for $2 \leq n \leq N-1$, the sets $\mathcal{F}_{\min}(n)$, $\mathcal{F}_{\max}(n)$, $\mathcal{F}_{\text{avg}_e}(n)$, and $\mathcal{F}_{\text{avg}_u}(n)$ do not intersect, and there are $N - n$ FS's in each one⁸. When $n = N$, there is only one FS to examine: \mathcal{S} , where all sites are inactive. The resulting upper bound is $1 + N + 4(N-1) + \dots + 4(N - (N-1)) + 1 = 2N^2 - N + 2$. Thus the pruning algorithm's complexity is $\mathcal{O}(N^2)$, which is a significant improvement over the 2^N FS's that the exhaustive approach requires.

IV. EXAMPLE ANALYSIS

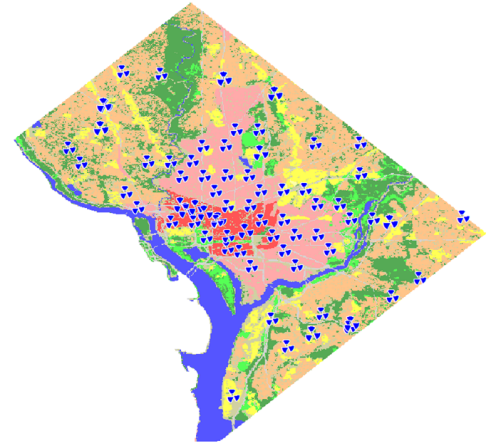
A. Description

In this section, we present an example, hypothetical, site deployment and apply the resiliency analysis using the pruning algorithm. The deployment in Washington, DC, consists of 70 sites as shown in the clutter map in Fig. 4a, and we use the user coverage metric. In Fig. 4b we show the region with better than 95 % user coverage in green, and we show the remaining part of the District in red.

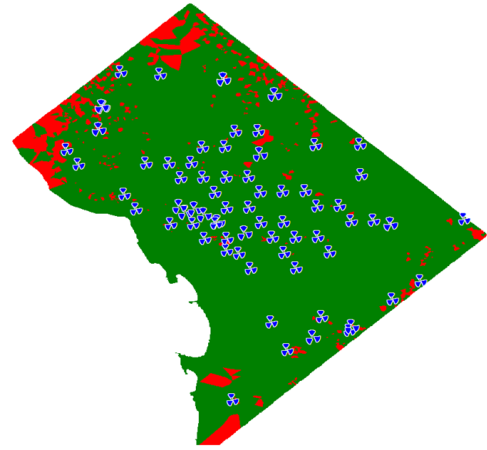
B. Impact of site failures on user coverage

We plot the estimates for the minimum, maximum, and average in Fig. 5 for up to 18 failures. The minimum curve drops below 95 % when there is a single failure (i.e., 0-site resiliency), which indicates the existence of a high-impact site. For more than four failures, the minimum coverage is below 90 %, and at 18 failures, it drops precipitously to around 50 %. Also, Fig. 5 shows that, on average, the network can have up to two simultaneous failures while still maintaining at least 95 % user coverage. In contrast, the maximum curve is above the 95 % threshold for up to 16 site failures. However, the loss of the particular set of sites associated with this result may be a very low-probability event.

⁸This cannot occur in general, but it gives us a loose upper bound.



(a) Site locations



(b) Coverage map

Fig. 4. Example site placement scenario: Washington, DC

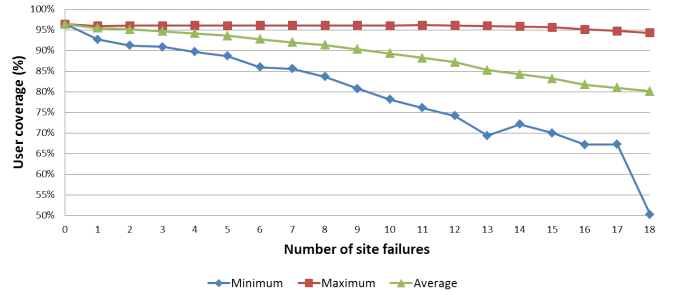


Fig. 5. Resiliency results for example 70-site deployment.

C. High and Low Impact Sites

Next, we discuss how the analysis helps identify high-impact sites that are candidates for additional hardening or redundancy, and how it also identifies low-impact sites whose loss does not dramatically reduce performance. The results in Fig. 5 show that we have a single site whose loss reduces the user coverage from 95.7 % to 92.7 %. In Fig. 6, we show the location of this site. We also show, in green and red respectively, the regions of above- and below-threshold

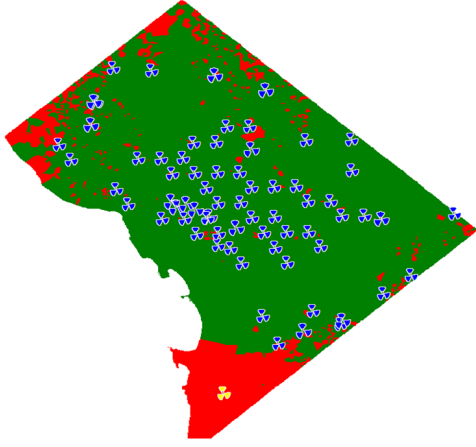


Fig. 6. User coverage and locations of high-impact sites.

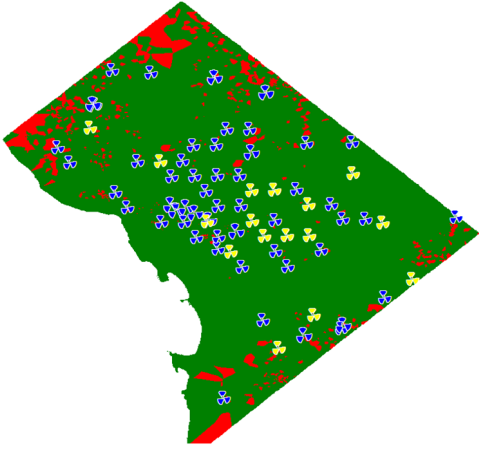


Fig. 7. User coverage and locations of low-impact sites.

coverage with the site lost. Comparing this figure to Fig. 4b, we observe that the negative impact is confined to the area around Congress Heights and Joint Base Anacostia-Bolling in the southeastern portion of the city. Also, the decrease in coverage is due to the relative isolation of this site; the presence of an additional site would be likely to reduce the impact of the site's loss.

Fig. 7 shows the locations of the 16 low-impact sites as yellow icons and shows the areas of above-threshold and below-threshold coverage. The network maintained 95.1 % user coverage when all the low-impact sites were lost, as shown in Fig. 5. Comparing Fig. 7 with Fig. 4b, we see little qualitative difference in user coverage.

Given that the loss of the low-impact sites has no appreciable impact on the user coverage, one may ask whether those sites can be removed from the deployment entirely. While selectively reducing the site count may reduce deployment and maintenance costs, this tactic has a price: reduced resiliency. To show this, we removed the 16 low-impact sites shown in Fig. 7 from the 70-site deployment, and generated a new set of estimated metrics using the pruning algorithm. Fig. 8

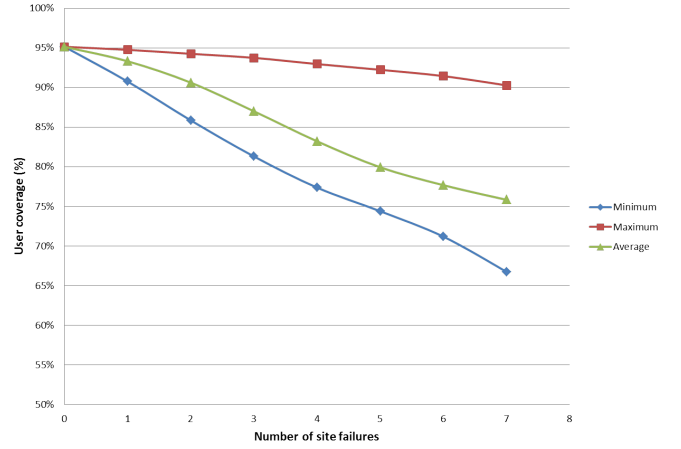


Fig. 8. Resiliency results for example 54-site deployment.

shows the minimum, maximum, and average values of the user coverage metric for the reduced 54-site deployment. The figure shows that the elimination of the low-impact sites results in a deployment that is 0-site resilient even with respect to the maximum user coverage metric.

Culling low-impact sites also reduces the ability of the network to handle large-scale incidents that require the deployment of large numbers of public safety resources. To demonstrate this, we simulated the gas leak incident use case defined by NPSTC in [7], which models the “report of a toxic gas leak in a large public assembly building near the National Mall in Washington, DC.” We considered 118 possible locations for the incident, which we show in Fig. 9. At each location, the incident occupies a $1.6 \text{ km} \times 1.6 \text{ km}$ square. The incident command personnel are concentrated in a small area; the remainder of the 327 responders and 127 vehicles that compose the response force are deployed uniformly in the incident area. We simulated background traffic associated with routine public safety operations, and superimposed it on the incident traffic. We used the mix of applications and associated offered loads in Exhibit 9 on p. 26 of [8], but reduced by half, to generate the background traffic.

Fig. 10 shows the negative effect of the reduced site count: in the 54-site deployment, 95 % of the users were served in only 4 % of the incidents, while, in the 70-site deployment, 95 % of the users were served in 48 % of the incidents. Fig. 10 also shows that the performance gap is greater when the subscriber coverage threshold is higher. Thus, the surge capacity of the network suffers when low-impact sites are pared away, even when none of the remaining sites experience outages.

V. CONCLUSIONS

The resiliency analysis technique in this paper gives network operators a useful tool to assess site deployments. The operator can identify high-impact sites whose loss has a disproportionate effect on performance. The network operator can then expend its limited resources to harden these sites or perform

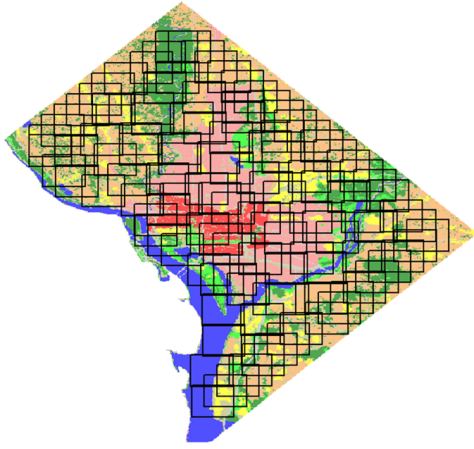


Fig. 9. Incident locations for gas leak study (118 locations).

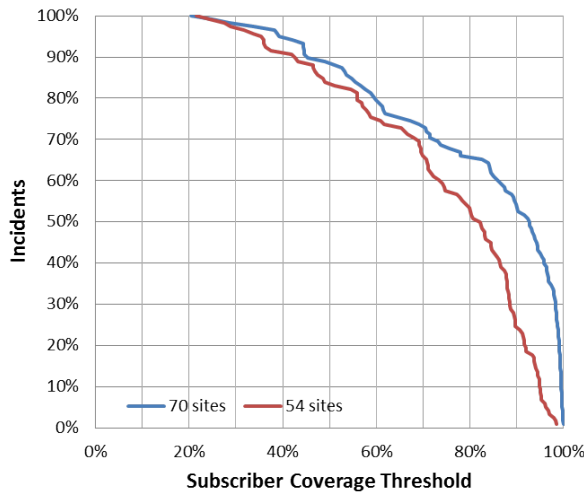
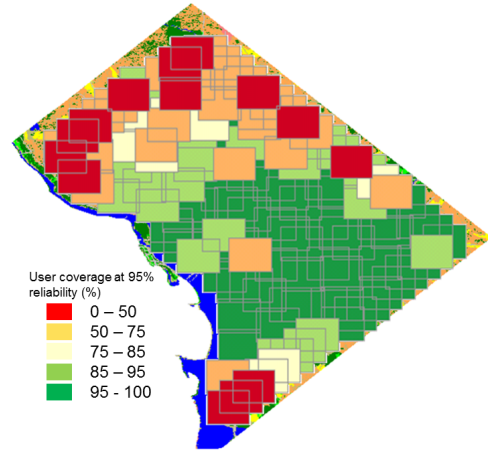


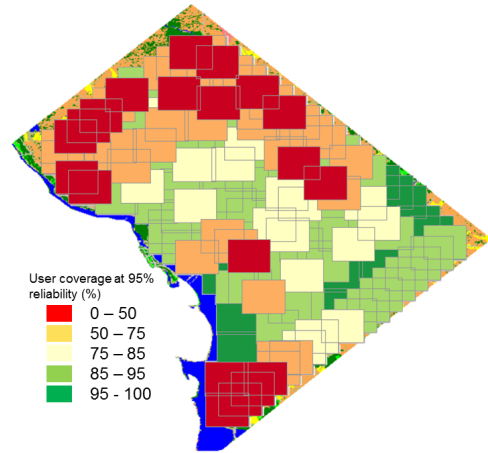
Fig. 10. Percentage of incidents covered at given threshold for 70-site and 54-site deployments.

additional monitoring and maintenance, using some of the best practices identified by NPSTC [1, Section 5.3]. Operators can estimate the probability of below-threshold performance with respect to multiple performance metrics, which also can inform decisions regarding which sites to harden. When examining multiple candidate deployments, an operator can identify those deployments that “bake in” resiliency (e.g., by having few high-impact sites) and that in turn free resources that can be expended on building additional capacity.

We also showed that while low-impact sites need less hardening or redundancy than high-impact sites, removing low-impact sites increases a deployment’s vulnerability to failures. An excessive reduction in the number of sites can leave a network vulnerable to even a single failure, if a high-impact site is lost. In addition, the overall network capacity will be decreased by removing sites, which reduces the network’s ability to handle the additional traffic associated with large-scale incidents.



(a) 70-site deployment.



(b) 54-site deployment.

Fig. 11. User coverage results for individual incidents using gas leak scenario and day-to-day traffic.

REFERENCES

- [1] *Defining Public Safety Grade Systems & Facilities*, National Public Safety Telecommunications Council, Littleton, CO, May 22, 2014. [Online]. Available: http://www.npstc.org/download.jsp?tableId=37&column=217&id=3066&file=Public_Safety_Grade_Report_140522.pdf
- [2] W. Najjar, J.L. Gaudiot, “Network resilience: a measure of network fault tolerance,” *IEEE Trans. Comput.*, vol. 39, no. 2, pp. 174-181, Feb. 1990.
- [3] P. Smith *et al.*, “Network resilience: a systematic approach,” *IEEE Commun. Mag.*, vol. 49, no. 7, pp. 88-97, July 2011.
- [4] D. Tipper *et al.*, “Providing fault tolerance in wireless access networks,” *IEEE Commun. Mag.*, vol. 40, no. 1, pp. 58-64, Jan. 2002.
- [5] S. Dharmaraja, V. Jindal, and U. Varshney, “Reliability and survivability analysis for UMTS networks: An analytical approach,” *IEEE Trans. Network & Service Management*, vol. 5, no. 3, pp. 132-142, Sept. 2008.
- [6] R. Rouil *et al.*, “Nationwide Safety: Nationwide Modeling for Broadband Network Services,” *IEEE Veh. Technol. Mag.*, vol. 8, no. 2, pp. 83-91, June 2013.
- [7] *Public Safety Communications Assessment 2012-2022: Technology, Operations, & Spectrum Roadmap*, National Public Safety Telecommunications Council, Littleton, CO, June 5, 2012. [Online]. Available: http://www.npstc.org/download.jsp?tableId=37&column=217&id=2446&file=AFST_NPSTC_Report_06232012.pdf
- [8] J. Peha *et al.*, *The Public Safety Nationwide Interoperable Broadband Network: A New Model for Capacity, Performance and Cost*, Federal Communications Commission, Washington, DC, June 2010. [Online]. Available: <http://transition.fcc.gov/pshs/docs/releases/DOC-298799A1.pdf>