

Appendix of Urban-Net: A System to Understand and Analyze Critical Infrastructure Networks for Emergency Management

Anika Tabassum*, Supriya Chinthavali⁺, Sangkeun Lee⁺, Liangzhe Chen^Δ, B. Aditya Prakash*

*Department of Computer Science, Virginia Tech

⁺Oak Ridge National Laboratory

^ΔPinterest

Email: {anikat1,badityap}@cs.vt.edu, {chinthavalis,lees4}@ornl.gov, liangzhechen@pinterest.com

ACM Reference Format:

Anika Tabassum*, Supriya Chinthavali⁺, Sangkeun Lee⁺, Liangzhe Chen^Δ, B. Aditya Prakash*. . Appendix of Urban-Net: A System to Understand and Analyze Critical Infrastructure Networks for Emergency Management. In *Proceedings of* . ACM, New York, NY, USA, 4 pages.

1 ORGANIZATION DETAIL

In appendix we describe more about the detail of Urban-Net. In Sec. 2 we describe in detail about the framework two analytic modules of Urban-Net. Sec. 4 provides a brief overview of related tools and literature. Sec. 3 we describe in detail our demo plan for each component separately, and one more use-case scenario for the analytic modules.

2 SYSTEM DESIGN

To achieve the above mentioned goals and create an interactive user interface we implement several back-end tasks: data processing, interactive interface with a data server, constructing networks, building analytics tools for them and a simulator for failure analysis of CIS nodes. In this section, we give more details of our framework and internal modules.

2.1 Modules

We give more details on the interface design for each module next.

2.1.1 Topology-based Analytic module. This module aims to identify potentially affected entities across different CIS when a perturbation entity is selected from a CIS network. When converting shapefile data into a graph data to store into Neo4j database, we link nodes originating from different CIS based on a set of rules. For instance, we link a node that represents a power plant with a node that represents a substation that is geographically nearest to the power plant. When a user selects a node from the shown map, the module generates a graph query to find all entities that can be reached within k hops from the perturbation nodes. This can give users an idea about what could be the potentially affected entities and understand the influence of failure of the selected nodes.

The whole CIS network has been categorized into four layers: energy, water, transportation, and communication. With the CIS layer selection tab the users can select any node types they want

to view (and each type of node has a different color). With an interactive zoom, a user can select some random input nodes on the real-time map and see the topological effect of interdependent nodes and edges. The demo provides the interdependency analysis of CIS components in three levels. The first level contains the initial perturbation nodes selected by the user, second and third level identifies the nodes within $\leq k$ and $\leq 2k$ edge hops from the initially perturbed nodes. For visualization, we consider $k = 7$ under the assumption that these would suffice the interdependency analysis study for the existing network. Fig. 1(a) shows the initial visualization of the module considering whole CIS network. By perturbing some particular CIS nodes it shows the affected networks within $2k = 15$ hops with the help of geographic map (Fig. 2(a)) as well as tabular form (Fig 1(b)).

In short, topology-based analytic module is graph theory based approach to identify the potentially affected entities based on geographic closeness. Our main contribution in this module is to provide an efficient dynamic computational model to identify the nearest CIS using the procedure mentioned above from any set of perturbed nodes selected by a user. The dynamic model provides user to choose the types of CIS for understanding interdependencies and analyzing vulnerabilities within (2-3 sec.)

2.1.2 Simulation-based Analytic module. With topology-based analysis it is hard to estimate the temporal delay of a failure node in a cascade which is very important in order to understand the restoration period of a node. Simulation-based analytic module on an energy based heterogeneous network allows the users to analyze real-time consequences of CIS grid layers of a region by varying seed nodes for perturbation, i.e., random perturbation (randomly generated seed nodes), regional perturbation (nodes generated centered around the selected region). This module also considers temporal aspects along with physical interdependencies. When considering temporal aspects, we allow users to control four parameters for every CI node types: i) average time a CIS node can support before it loses control (β) ii) average time (hrs) a node takes to recover from failure (α) iii) average load for a node and iv) average capacity for a node. To create the heterogeneous network, we consider the current physical graph stored in the database.

Step 1: In this step, we will explain how we generate perturbed nodes to provide an environment for real-time simulation. Creating perturbed nodes based on user choice is necessary to understand interdependencies and vulnerability by simulating different consequences. We generate a list of seed nodes initially which are considered to be failed nodes. We adopt two different approaches to generate seed nodes based on user's choice. For the first, we

Permission to make digital or hard copies of part or all of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for third-party components of this work must be honored. For all other uses, contact the owner/author(s).

Copyright held by the owner/author(s).

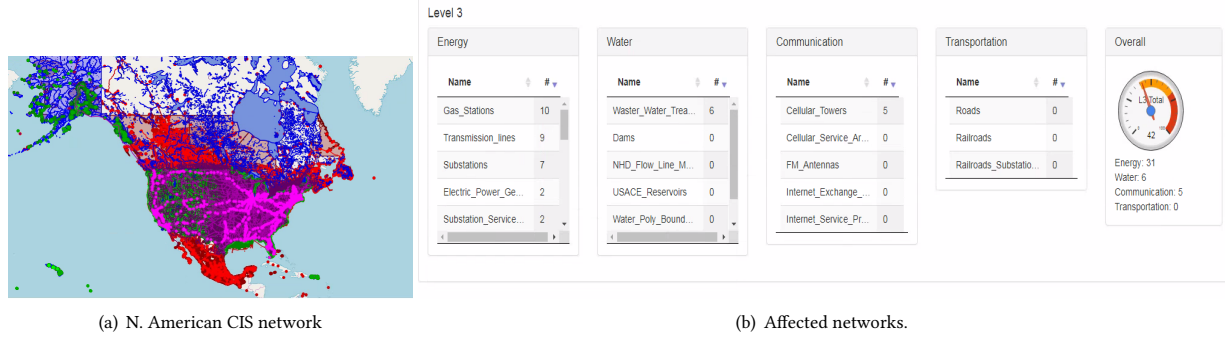


Figure 1: Visualization of Topology-based analytic module. (a) shows the whole CIS network. Red, blue, green, purple indicates energy, water, communication and transportation network respectively. Note, it is difficult to identify the physical interdependency between them from the overall visualization. (b) shows a summary of the affected CIS network within 15 hops in tabular form.

randomly sample 10 nodes of the selected state and consider them as initially failed nodes. For the second, we randomly choose one node as the center node of the disaster affected state and list all the nodes as initially failed nodes which are within a 10km geographic radius of the center node.

Step 2: We developed state-of-the-art tractable cascade models, which can be utilized using various user parameters (like load capacity etc.) and which are based on novel *path-based* failures between the various CIS components [4]. Tractable cascade models can help approximate the failure nodes fast and give the notion of simulation with more real-time consequence. We construct a heterogeneous network for energy based on two types of electric power plants (with and without using natural gas to generate power), substations, transmission or bus nodes, natural gas compressors, natural gas pipelines, road network. To build the cascade model the nodes in the network itself are connected if there is an edge and to other networks based on the idea of failure cascade proposed by [6]. Each type of node in the network has a certain power load it needs to transfer for operating conditions and the maximum power capacity it can transfer. If a node's load condition goes beyond its capacity it will be inactive for a time based on β . Once a node fails in a cascade, it can recover after a time period based on α . All these four parameter values (load, capacity, α , β) can be set by user's choice. Nodes under transmission bus, gas compressor, and power plants have the ability to distribute their load to its neighbor by probability P if they fail. The amount of load it can distribute is also based on the user's choice.

Step 3: In this step, we identify the nodes that will fail next in the cascade based on the current list of seed nodes and schedule recovery time of the new failed nodes exponentially distributed by their corresponding α . A node is considered a 'failed' and added to the list of seed nodes if it fulfills any of the following conditions. 1) For the nodes which are considered 'failed' in the list of seed nodes, it can redistribute a load of their neighbors by P . 2) For an active node, if load > capacity then it is considered a new failed node. 3) If all the neighbors of an active node fail, then the node will be considered as 'failed'. 4) If all the nodes of its connected networks according to the cascade fail, it will be added as a failed node.

Step 4: Next, we check if any nodes recover from its failed state, i.e., their scheduled recovery time starts, they will be added in the list of active nodes.

Step 5: We implement step 2 and step 3 as long the simulation continues. We consider running the cascade unless all the nodes become active or if the recovery schedule of a failed node reaches its maximum constraint. In the interface, for each timestamp t , we list the failed nodes and nodes recovered from a failed state in the current cascade and run the simulation based on the failed nodes for the next cascade. In order to understand the impact of nodes with time, we show a dynamic graph of time vs a number of failed nodes for each node type at the bottom. The nodes which fail can be shown both on the map with their detail information to the right.

The main contribution in this module is to develop a real-life tractable cascade model and integrate path-based failures among the various CIS components for the simulation. To the best of our knowledge, no one used this approach before for analyzing failures in CIS.

2.2 Implementation

Here we briefly describe the different technologies we used to implement Urban-Net. We pursue to use the open-source, easily accessible, and easy-to-integrate technologies for Urban-Net.

Interface. We design the user interaction of the interface using JavaScript, and the OpenLayers API Map library¹. We also use a Python Tornado web server² to connect the user interface with the analytic tool and the geographic database (GeoServer and Neo4j).

Analyzing data. We perform graph operations on the constructed network, and it is implemented using the python library Py2neo (a python library for the Neo4j graph database) and a separate docker.

Dynamic map and concurrent hash table. We use the OpenLayers map library to maintain a dynamic map view based on CI nodes selected by the user. OpenLayers has fast thread-safe hash table and vector structure; WMS and WFS to store features or selected layers on map and query them on the client side.

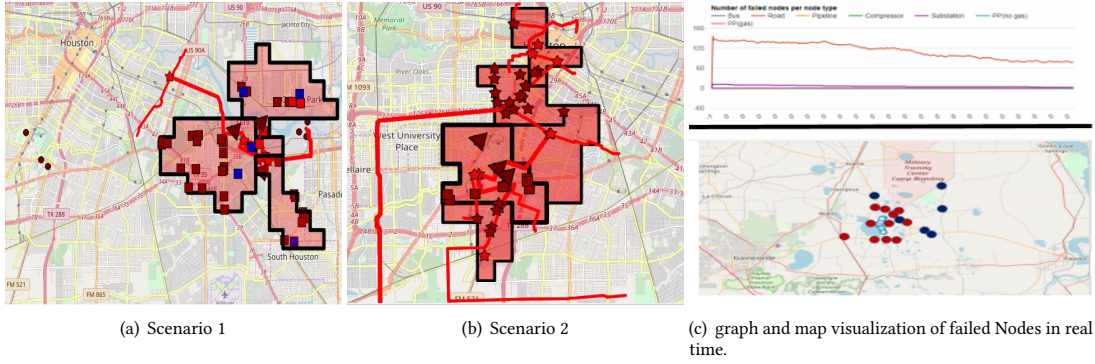


Figure 2: (a)-(b)Analyzing physical interdependency of CIS in Houston. We perturb two different sets of power plants (red triangle) in (a),(b) which (a) affect some water treatment plants (blue rectangle). (b) affect highest number of energy network. (c) Simulation-based analytic. Bottom Fig. shows the initial seed/failed nodes (red) while others are the effect of failure cascade . (Top Fig.) is a graph showing the no. of failed nodes of different CIS type where the final cascade heavily damaged the road network.

3 SCENARIOS AND DEMO. PLAN

In this section we first talk about the machine configuration for Urban-Net. Then we present the important scenarios for each module of the system.

Dataset and Machine configuration. We chose the four categories of CIS: energy, water, communication, and transportation based on the HSIP Gold data[1]. Urban-Net system has been deployed to 2.7 GHz 12-Core Intel Xeon E5 system with 64GB DDR3 Memory. The constructed graph contains 81M CI nodes and 83M interdependent CI edges, and its total size of the data is approximately 60GB. To simplify the future deployment, we made Docker containers for our backend modules including databases and geoserver.

Detail Demo. Plan. In preparation for the demo, we already store the existing CIS layers over US and their interdependent relations in the server. We also store analysis output of the simulation-based analytic module for one US state to present a faster visualization to the audience.

In the interface, users can interact with and navigate the map for each module. In the topology-based analytic module users can select perturbation nodes they want to consider using 'command' and dragging over the nodes. They can also select the CIS layers they want to consider for analysis for each of the modules. Our plan is to invite the audience, let them interact with Urban-Net, turn them into analysts and develop use-case scenarios related to understanding the CI network using our tool.

Next, for each module we will demonstrate scenarios of Urban-Net illustrating its capability, performance and our plan for the attendees.

3.1 Understanding interdependencies

We already explained the use-case scenario for Fig. 2(a) in our paper. In Fig. 2(b), we selected another three power generation plants in the middle of Houston which effected a large portion CI network consisting of 68 energy nodes in total. However, these do not affect any other type of interdependent CI nodes except electric power

plants. Thus we showcase two use cases for identifying vulnerabilities as well as a realistic effect on interdependent networks.

Plan for the attendees. The attendees will be able to load the CIS data perform interdependency analysis in different parts US for recent hurricanes or future disaster. The attendees will be able to change input nodes/node type to disturb a CIS network and analyze their physical interdependencies. They will be able to see the detail of the affected nodes (name, location, target node etc.) in the interface.

3.2 Understanding vulnerability

Here we explain in detail about the use-case scenario given in the paper. To showcase a use case for a simulation-based analytic module, we considered the state Florida, the most affected state during hurricane Irma. We selected some random nodes centered around the 10km of hurricane area using regional perturbation method used in Step 1 of Sec. 2.1 and ran the simulation for three mins. (considered maximum schedule recovery time as 10 mins.) to understand which nodes have been affected the most. We chose all types of CIS of energy network for the simulation. Bottom Fig. 2(c) shows the initial seed nodes where majority of them are gas power plant generators. The last cascade of the simulation after 3 minutes shows that the failure cascade heavily damaged the road networks, when failure of all other nodes reduced (Top Fig. 2(c)). This implies the end time of the hurricane.

Plan for the attendees. The attendees will be able to select any state in the US they want to simulate in real time. They will be allowed to change the parameter settings (α , β , load, capacity) for each network to evaluate the restoration period of the failure nodes in the network.

4 MORE RELATED WORK

Studies on understanding and analyzing CI systems have been performed extensively for the last several years [4, 8, 9]. [9], [8] perform a study on national level CI systems to understand how each system are interconnected and can identify multiple connections, types of coupling, failures, and types of interdependencies. [4]

¹<https://openlayers.org/>

²<https://www.tornadoweb.org/en/stable/index.html>

proposed failure maximization problems on CI interdependent networks which can identify the maximum failure nodes due to perturbation of initial seed nodes. [7] broadly categorized the existing study, modeling, and simulation into several categories such as empirical approaches, agent-based approaches, system dynamics based approaches, economic theory based approaches, and network-based approaches. Existing approaches of interdependency vulnerability analysis and modeling of CI systems propose different mathematical and graph-based evaluations which are mostly quantitative [3], [5]. [10] discussed several visualization tools and systems in CIs developed by different govt labs and agencies to facilitate decision making. *CIPDSS*, a simulation model to protect CIs against vulnerabilities, *CLEAR*, to assess impacts of CO_2 on the energy sector, *URBAN-CAT* to understand impacts of climate change, and *LAND-SCAN* a visualization of population distribution on the settlement. We had presented a preliminary status report of Urban-Net in a workshop [6]. In this paper we have made extensive addition of visualization and the topology-based and simulation-based analytic modules following the idea of the report.

5 DEMO VIDEO

We also provide a 1 min. 17s. video link of Urban-Net [2]. The video demonstrates in detail how users can interact with the analytic module to understand interdependencies and analyze vulnerability. Urban-Net interface name analytic module as Impact Estimator. First 40s of the video shows how users can select different CIS layer and visualize them on map. The interface also provides an option for case-studies of two major events- Wild-fire and Hurricane Sandy. In the video, it shows which part of the region and CIS nodes are most affected during these events and provide an option for analysis on these components. The last 37s shows how users can select perturbed nodes and which nodes or CIS components can be affected due to the interdependencies based on their choice.

REFERENCES

- [1] 2018. Homeland Security Infrastructure Program (HSIP). <https://gii.dhs.gov/HIFLD/hsip-guest>. (2018).
- [2] 2019. Appendix. <https://people.cs.vt.edu/anikat1/urbanet>. (2019).
- [3] Sergey V Buldyrev, Roni Parshani, Gerald Paul, H Eugene Stanley, and Shlomo Havlin. 2010. Catastrophic cascade of failures in interdependent networks. *Nature* 464, 7291 (2010), 1025.
- [4] Liangzhe Chen, Xinfeng Xu, Sangkeun Lee, Sisi Duan, Alfonso G Tarditi, Supriya Chinthavali, and B Aditya Prakash. 2017. Hotspots: Failure cascades on heterogeneous critical infrastructure networks. In *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*. ACM, 1599–1607.
- [5] Åke J Holmgren. 2006. Using graph models to analyze the vulnerability of electric power networks. *Risk analysis* 26, 4 (2006), 955–969.
- [6] Sangkeun Lee, Liangzhe Chen, Sisi Duan, Supriya Chinthavali, Mallikarjun Shankar, and B Aditya Prakash. 2016. URBAN-NET: A network-based infrastructure monitoring and analysis system for emergency management and public safety. In *IEEE BigData Workshop*. Big Data for Sustainable Development.
- [7] Min Ouyang. 2014. Review on modeling and simulation of interdependent critical infrastructure systems. *Reliability engineering & System safety* 121 (2014), 43–60.
- [8] Peter Pederson, Danile Dudenhofer, Steven Hartley, and May Permann. 2006. Critical infrastructure interdependency modeling: a survey of US and international research. *Idaho National Laboratory* 25 (2006), 27.
- [9] Steven M Rinaldi, James P Peerenboom, and Terrence K Kelly. 2001. Identifying, understanding, and analyzing critical infrastructure interdependencies. *IEEE Control Systems* 21, 6 (2001), 11–25.
- [10] Anika Tabassum, Supriya Chinthavali, Liangzhe Chen, and B Aditya Prakash. 2018. Data Mining Critical Infrastructure Systems: Models and Tools. *IEEE Intelligent Informatics Bulletin* 43 (2018), 31.