Framework for Probabilistic Vulnerability Analysis of Interdependent Infrastructure Systems

Chloe Johansen Applegate, M.ASCE[1]; and Iris Tien, Ph.D., A.M.ASCE[2]

Abstract: Critical infrastructure systems are deteriorating and experiencing increased cascading failures. In this paper, we propose a new probabilistic framework for modeling interdependent infrastructure networks. As part of this framework, we present our modeling approach and the accompanying sets of algorithms that enable the computationally efficient probabilistic modeling of large infrastructure systems considering interdependencies between networks. The proposed method creates a representative Bayesian network (BN) of the system, which provides exact inferences over the network compared to simulation-based approximations. With the traditional computational limitations of BNs to tens of parent component nodes per child system node, our generalized framework enables computationally efficient BN modeling of systems of, for example, hundreds of component nodes, including component-level performance and interdependent connections across networks. Our approach requires only simple inputs based on the basic component characteristics of location, type, connectivity, and initial failure probabilities. We modeled three types of interdependencies across the system—service provision, geographic, and access for repair. We combined a minimum link set (MLS) formulation with the idea of supercomponents in order to reduce network complexity without any approximating assumptions. We present algorithms to efficiently identify MLSs and supercomponents, as well as to identify and remove any cyclic dependencies that arise across the network. Once the BN was constructed, we were able to perform exact inference analyses over a range of component state and hazard event scenarios in order to identify vulnerabilities across the network. The main novelty of the paper is to enable the probabilistic assessment of large, complex, interdependent infrastructure systems. We were able to consider performance from the component level and model hundreds of component nodes in a computationally efficient manner without approximating assumptions. We accounted for interdependencies between systems with exact inference results. The model can be used to investigate the potential for cascading failures and to prioritize critical components for repair, replacement, or reinforcement. We applied the proposed methodology and algorithms to the water distribution network in Atlanta, Georgia, and its dependencies with the power system. We validated the model using the results from a recent interdependent outage event. **DOI: 10.1061/(ASCE)CP.1943-5487.0000801.**

Introduction

Critical infrastructure systems, such as water, power, transportation, communication, and fuel networks, are deteriorating and experiencing increased cascading failures. This is due to a combination of aging infrastructure components, vulnerability to natural disasters, and susceptibility to organized attacks. For these reasons, it is necessary to assess and improve the resilience of infrastructure systems—"the ability to prepare for and adapt to changing conditions and withstand and recover rapidly from disruptions" (White House 2013). Cascading failures occur when a single infrastructure component outage causes outages in other infrastructure systems and components.

A quantitative method to assess infrastructure vulnerabilities is necessary in order to improve resilience for communities and to determine how to invest resources in order to reinforce and recover infrastructure systems (Johansen et al. 2016). In the process, it is also necessary to account for other networks on which certain infrastructures depend. For example, each of the other fifteen critical infrastructure sectors defined by the White House (2013) connect to the energy sector in some way. Power is necessary to operate water pumps and treatment plants, maintain critical manufacturing operations, support healthcare facilities, and control and operate transportation systems. Because of these connections, the risk of cascading failures across systems increases. An example occurred during the blackout in the Northeast in 2003, when a tree contacting power lines caused outages for approximately 50 million people and led to disruptions in communication systems, public transit, and water distribution systems (Lu et al. 2006).

In this paper, we present a new methodology that models multiple infrastructure systems and the interdependencies that exist between them using a probabilistic Bayesian network (BN) framework. BNs account for uncertainty while enabling exact inference over the network in contrast with approximate or bounded results from simulation-based approaches. While BN modeling has traditionally been computationally limited to systems with tens of component nodes per child system node, our generalized framework enables tractable modeling of systems of, for example, hundreds of component nodes, including component-specific information and interdependent connections across networks. In this paper, we describe the full approach and the set of algorithms that enable the computationally efficient modeling of interdependent infrastructure systems from the component level to the system-of-systems level using the proposed framework. The main novelty of the contribution is to enable probabilistic assessment of large, complex infrastructure systems. We were able to model hundreds of component nodes in a computationally efficient manner without approximating

[1]Graduate Student, School of Civil and Environmental Engineering, Georgia Institute of Technology, 790 Atlantic Dr. NW, Atlanta, GA 30332 (corresponding author). ORCID: https://orcid.org/0000-0002-5558-0802. Email: cjohansen6@gatech.edu; chloeajohansen@gmail.com

[2]Assistant Professor, School of Civil and Environmental Engineering, Georgia Institute of Technology, 790 Atlantic Dr. NW, Atlanta, GA 30332. Email: itien@ce.gatech.edu

Note. This manuscript was submitted on March 7, 2018; approved on June 28, 2018; published online on October 31, 2018. Discussion period open until March 31, 2019; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Computing in Civil Engineering***, © ASCE, ISSN 0887-3801.

assumptions, accounting for interdependencies between systems with exact inference results.

The remainder of this paper gives a detailed description of our methodology, including how the BN was constructed with required inputs, component connection models, and interdependency definitions within the framework. We describe our approaches and the accompanying algorithms we developed in order to address challenges associated with the BN modeling of complex infrastructure systems. These include algorithms for minimum link set identification in infrastructure systems, supercomponent identification, and defining nodes modeling different types of interdependencies. These also include a method to identify and remove cyclic component configurations within the network. The overall methodology creates a computationally tractable probabilistic BN model of large infrastructure systems considering interdependencies between networks. We applied the methodology to interdependent water and power distribution systems in Atlanta, Georgia. We provide inference results from a historic cascading failure event for validation. We compared the performance of the proposed methodology to that of prior approaches. The main contribution of this work is to provide a new framework for the probabilistic vulnerability analysis of interdependent infrastructure systems, including modeling methodology and algorithms for analysis. It provides a generalized method by which complex infrastructure systems can be efficiently modeled, including their interdependencies, using BNs. We were able to analyze systems from the component level and obtain exact inference results in order to conduct detailed probabilistic assessments of these systems in order to increase reliability and resilience.

Background and Related Work

Probabilistic Vulnerability Analysis of Infrastructure Systems

Analyzing the vulnerability of infrastructure systems can be done using empirical, agent-based, system dynamics-based, economic theory-based, and network-based approaches (Ouyang 2014). The approach used in this paper is a network-based approach, with nodes and links that represent infrastructure components and the connections between them, respectively. Network-based approaches can be used to effectively prevent catastrophic effects, improve absorptive capacities, and analyze the propagation of decisions made on the network (Ouyang 2014). A network-based approach was chosen because it allowed the use of a probabilistic, dynamic analysis of interdependent infrastructure systems. Network-based approaches also enable the evaluation of the ability of the interdependent infrastructure networks to remain resilient in the face of a hazard. Vulnerability studies have been performed using network-based approaches in several ways—under random failures, under natural hazards, and under intentional attacks (Ouyang 2016). The methodology developed in this paper can be used for each of these analyses based on the conditional probabilities that are input for any type of hazard or disruption.

Bayesian Networks

Bayesian networks model systems in order to account for the probabilistic dependencies between components and facilitate the updating of system assessments with new information. A BN is a directed (i.e., the edges are directional) and acyclic (i.e., no closed path exists in the network) probabilistic graph composed of nodes and links. Based on the dependency relationships between components, nodes are defined as parent or children nodes. Children depend on the states of their parents. Each node represents a random variable and, for discrete networks, is defined by a conditional probability table (CPT). The CPT consists of the conditional probabilities of the states of the child node given the states of the parents. Parent nodes are defined by their marginal probabilities.

In computing applications in civil engineering, BNs have been used in several ways. One is to identify damage location on civil structures using electromechanical (E/M) impedance (Naidu et al. 2006). In Naidu et al. (2006), BNs were used to reduce the amount of input data needed for traditional damage identification methods, which require large amounts of training data. Cheng and Hoang (2016) probabilistically estimated slope stability using BNs in order to calculate posterior probabilities of slope collapse without requiring prior knowledge of data distributions. As they were in these studies, BNs were useful in the approach described in this paper because a large amount of input data was not necessary in order to learn information about the network and to calculate probabilities of failure of infrastructure components based on different scenarios.

BNs have been used to model single infrastructure networks such as inland waterway ports (Hosseini and Barker 2016), railway lines (Castillo and Grande 2016), highways (Grande et al. 2017), and power (Tien and Der Kiureghian 2017) and water networks (Leu and Bui 2016). These studies did not consider interdependencies between different networks. In Leu and Bui (2016), the BN nodes were defined based on general properties of the water network (e.g., pipe diameter and depth) and other factors that could affect the water network (e.g., pipe corrosion and construction activities). Hosseini and Barker (2016) built a BN model where resilience metrics, such as backup utility systems and quick evacuation, were nodes in the network. A BN was used to analyze the risk of domino effects, similar to cascading failures, in chemical plant infrastructure in Khakzad (2015). Nodes represented parts of fuel storage plants, such as tanks, the states of which could be safe, on fire, or burned out. The cascading failures modeled were in time slices by applying a dynamic BN model. Similarly, a dynamic BN was used to evaluate cascading effects in a power grid in Codetta-Raiteri et al. (2012). In this study, electrical lines were considered as series or parallel modules that connected nodes in the power grid.

BNs have also been used to model the security of interdependent critical infrastructure (Schaberreiter et al. 2013). This approach used service outputs and high-level system measurements as nodes in the network. Aung and Watanabe (2010) similarly modeled interdependent infrastructure systems using BNs at a very high level, in which each node in the network represented an entire critical infrastructure sector. The BN was used to determine the cascading effects of infrastructure sector outages. The critical infrastructure BN model in Di Giorgio and Liberati (2011) also included nodes representing services supplied and single nodes representing infrastructure systems such as, for example, the electrical transmission system as a whole, along with nodes representing adverse events.

Limitations of Prior Work

Compared to previous studies using BNs to model infrastructure systems, our focus was on large, complex infrastructure networks, accounting for the states of individual components of each system. In studies where BNs have been used to model single infrastructure networks, nodes in the BN were used to represent the properties of a network and other global factors that affected the network. By contrast, the approach in this paper used nodes to represent the states of the individual components in the network and links to represent the connectivity between them. In infrastructure networks, overall system states are governed by individual component states. Our approach enabled us to consider the states of specific components whose performance impacted overall infrastructure system performance. The resulting model can be used to analyze diverse scenarios, including component-level events, with levels of service outcomes measuring the resilience of the network under different conditions. Previous single network approaches have not considered interdependencies between the networks modeled and other networks on which they depend.

Previous studies modeling the security of interdependent infrastructure systems using BNs differ from the framework proposed in this paper in that rather than modeling entire infrastructure systems or the services provided as single nodes, our approach modeled infrastructures starting from the constituent components of a system. We considered from the level of the individual infrastructure components the topology and connectivity characteristics of infrastructure networks. In practice, this is the level at which the complex relationships between systems, including the interdependencies between them, arise. For example, the probability of being able to provide a service at a distribution component is dependent on the number and reliabilities of redundant paths, which are themselves composed of other components, to that distribution point. For instance, for a water distribution system, the probability of being able to provide service at a distribution component is also dependent on the reliability of the power components supplying electricity for the water treatment plants and pump stations.

In contrast with previous studies, each node in our BN model represented an individual component of an infrastructure network. This enabled us to capture component-to-component relationships and to incorporate component-level information, such as updates about component states from monitoring or inspection information, into system assessments. In addition, decisions for infrastructure systems occur at the component level—for example, decisions about which component should have an additional backup or between which components a redundant path should be built. Our proposed framework supports these component-level inferences. The resulting analyses allow infrastructure owners to identify specific nodes (representing individual components) in the network considered critical for replacement, repair, or additional buildouts in order to increase overall system performance.

The methodology proposed used a BN-based approach to capture probabilistic relationships between components and incorporate both prior information about the network and update assessments when new information is learned about the network (Johansen and Tien 2017). Prior information was incorporated during construction of the BN. Updating information was incorporated during inference of the BN. For example, if it was learned that a certain hazard occurred on the system or that a certain component failed, the new information was propagated to all nodes in the network in order to calculate updated probabilities across all component and system states.

The authors advance upon previous work (Johansen and Tien 2017), which defined three interdependency types—service provision, geographic, and access for repair—and propose a modeling methodology for these three types. These interdependency types were one element of the framework proposed in this paper. However, many other elements were required to build the full interdependent infrastructure system model in a computationally tractable manner. The previous methodology was applied to three components. This paper advances the approach to model an entire network of interdependent infrastructure systems. The example application was composed of hundreds of components. This study focuses on computational methods and describes the proposed full approach and accompanying algorithms to automatically model the interdependent systems.

Methodology

Fig. 1 shows the overall methodology. In order to create the interdependent infrastructure system model, we began with inputs of

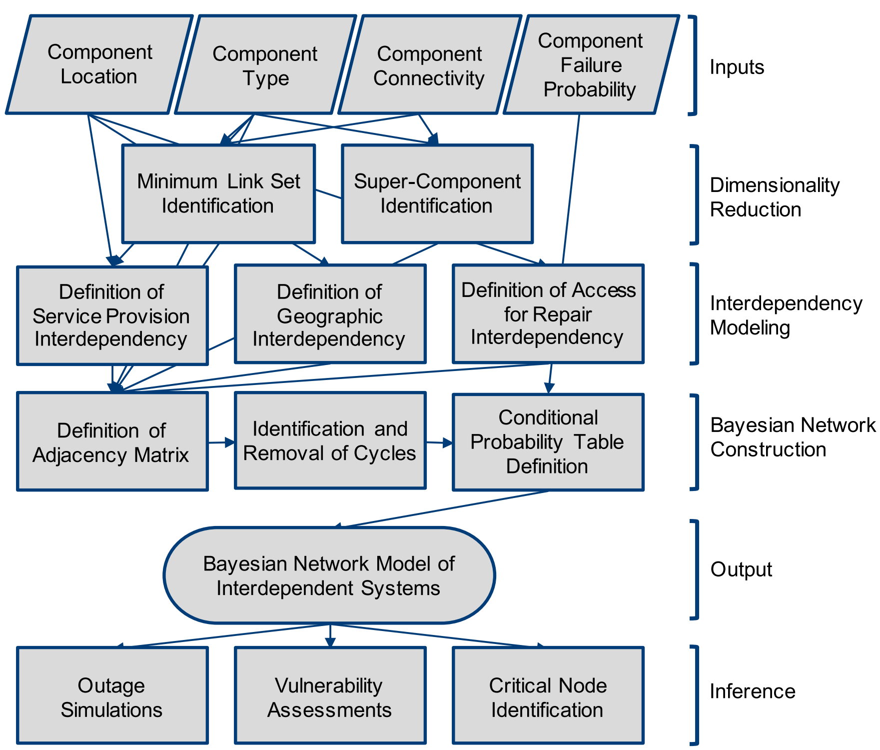

Fig. 1. Flowchart of overall methodology.

component locations, connectivity, types, and failure probabilities. We considered a binary system in this case, in which components could be in one of two possible states, i.e., failure or survival. Therefore, failure probabilities were defined. The method easily extended to systems of multiple states, such as those modeling flow capacity (Tong and Tien 2017). In those cases, probability distributions over all possible component states are defined.

We then reduced the dimensionality of the BN by using a minimum link set (MLS) formulation combined with supercomponent identification. Algorithms to do this in the context of infrastructure systems are presented. Next, we specified the interdependencies and constructed the BN, which included defining the adjacency matrix, identifying and removing cycles in the graph, and defining conditional probability distributions. We were then able to use the resulting model to perform exact inference to probabilistically assess the vulnerability of interdependent infrastructure networks. We describe each of these steps of the methodology in detail in the following sections.

Inputs

The construction of the BN was largely based on geospatial information about the interdependent infrastructure networks. We assumed that any given infrastructure system is composed of individual components whose performance contributes to the performance of the overall network. The required inputs for the methodology were component locations, connectivity, types, and failure probabilities.

The first inputs necessary were the locations of all components in the networks. For example, for a water distribution network, locations include coordinates of supply components (water treatment plants, pump stations, tanks, and reservoirs), pipe junctions, and terminal distribution nodes. The locations provide information on the components that provide infrastructure resources or services to specific parts of the community. In addition, locations account for the relation between components and hazards that were considered for risk assessments, because most hazards can be described geospatially. For example, the distance from an earthquake fault line can be computed based on the component locations. Any specification of locations is acceptable as long as the frame of reference is the same across the networks and hazards.

The component connectivity inputs were expressed as pairs of infrastructure components that were connected in an individual network. For example, for a water distribution system, the connectivity was provided as a list of the two system components that each pipe connected. The connectivity inputs were translated into a connectivity matrix composed of ones and zeros in which a value of one signified that the components were connected and a zero signified no connection.

Another input needed was the type of each component. There were three component types considered in the methodology, each corresponding with a different physical function within an infrastructure system. The first was supply components. These are components that generate or output the resource that flows through the network. For example, for a water distribution system, the resource is water, and the supply nodes are water treatment plants, pump stations, tanks, and reservoirs. The second type was distribution components. These are endpoints in the network that distribute the infrastructure resource to customers or end users. These can also be smaller distribution points such as small pumps or power lines for water and power systems, respectively, which feed individual houses or facilities. The final type was transshipment components. These are intersections between several links in the system that are not endpoints in the network. These enable the infrastructure resource to be distributed more easily across the network along multiple paths compared to having single flow paths. The component type inputs were necessary to define the role of each component within overall system functioning.

The final input was the failure probabilities for each component. These were determined from empirical calculations, asset health scores, or estimated probabilities from infrastructure owners or other domain-specific experts. An example of empirical calculations is to use fragility curves to calculate failure probabilities given a specific hazard (González et al. 2016). In other cases, infrastructure owners calculate failure probabilities for each component, such as those corresponding with varying asset health scores.

If precise failure probability values cannot be determined, relative values can be used, particularly if a goal of the analysis is to provide a comparative ranking and prioritization of system components. These failure probabilities can also be updated when new information is learned about the system. For example, if the results of an inspection update the estimated probability distribution of the state of a component, that information updates the prior failure probability of the component, and through its connections, the distributions of the states of surrounding nodes as well.

Dimensionality Reduction

With these inputs, we then reduced the dimensionality of the BN. Previously, BNs have been used to model smaller systems of five to ten components (e.g., Bobbio et al. 2001; Kim 2011). Algorithms have been developed to use BNs to model and assess the reliability of much larger systems (Tien and Der Kiureghian 2015, 2016) and increase computational efficiency in conducting inferences for critical infrastructure systems (Tien and Der Kiureghian 2017). We used both a minimum link set (MLS) formulation and supercomponent identification to decrease the dimensionality of the network and make it computationally tractable for modeling systems of hundreds of nodes.

In assessing the performance of a system, a minimum link set is a minimum set of components that are required to be functioning in order for the system to function. For a physical infrastructure system, we defined a MLS as the components that must be working for a resource, such as water, power, or gas, to be conveyed from a source node to any other node in the network. If a single component in the MLS fails, the MLS fails. For this application, MLSs linked supply components to transshipment or distribution components. They mapped paths through the network for providing infrastructure services to end points in communities. Algorithms to identify MLSs for entire networks have not been found in the literature. This can be done manually for small networks; however, this is time consuming for networks of larger than 10 components.

Complementary to MLSs are minimum cut sets (MCSs). For infrastructure networks, a MCS is the minimum set of components that must fail for a resource to fail to be conveyed from a source node to any other node in the network. Several algorithms have been developed to identify MCSs in networks. One such algorithm is EG-CUT, developed by Shin and Koh (1998). This algorithm builds a MCS generation tree and backtracks from a leaf when it fails to generate a MCS. However, this method does not enumerate MLSs.

A robust, efficient algorithm to define the MLSs of the system enabled us to capture the functionality of the network while reducing the dimensional complexity of the BN. It models the influence of every combination of individual component states on overall system performance through the MLSs. To define the MLSs, we propose a recursive algorithm based on a depth-first search method (Jiang et al. 2016). For large infrastructure networks, we also created a cutoff for the maximum size of a MLS, based on the logic that a resource will not deviate far from the shortest path, in order to increase computational efficiency. For example, water will not weave through a grid in a network in order to travel between two points on a single line.

The recursive MLS identification algorithm, presented as Algorithm 1, is run for each pair of supply components and target components, which include all transshipment and distribution components in the network. Inputs to the algorithm are the start component S, target component T, connectivity matrix **Con**, the shortest distance D_S from any supply component to the target component of interest, and a matrix of the physical length **L** of all links in the network. Unbolded italics denote scalar values; small bolded letters denote vectors; capital bolded variables denote matrices. A comparative distance D_C is calculated to create a distance cutoff for the maximum physical distance of the MLS using a multiplier M. In the application example in this paper, the cutoff distance is twice the shortest distance between the supply and target nodes, i.e., $M = 2$.

As the algorithm proceeds, it visits an increasing number of components. Several variables are created during the recursion of the algorithm. These include a visited vector $(\mathbf{Vis})_{1\times n}$ of ones and zeros the length of the number of components n that represents the components that have been visited (one representing an unvisited component and zero representing a visited node during the course of the algorithm), a current path vector $\mathbf{P_C}$ that represents the path calculated within the recursive algorithm, and a current length variable L_C that represents the length of the current path.

Algorithm 1. MLS identification algorithm.

Input: S, T, **Con**, D_S, **L**, M, **Vis**, $\mathbf{P_C}$, L_C
$D_c = M \cdot D_S$
$\mathbf{Vis}(S) = 0$
$\mathbf{P_C} = [\mathbf{P_C}, S]$
Add length of link to length of current path L_C
ch = unvisited connections in **Con**
If $S = T$
    $\mathbf{MLS} = \mathbf{P_C}$
Else for each element (i) in **ch**:
    $S = \mathbf{ch}(i)$
    If $L_C > D_C$, break
    newVis = **Vis**
    $\mathbf{newVis}(S) = 0$
    $newPaths = \text{MLSalg}(S, T, \mathbf{Con}, \mathbf{newVis}, \mathbf{P_C}, L_C)$

where S = supply node; T = target node; **Con** = connectivity matrix for the network; D_S = shortest distance from any supply node to the target node; **L** = matrix of link lengths; M = multiplier for maximum physical distance of MLS; $\mathbf{Vis} = 1 \times n$ vector of visited nodes, initiated as all 1s representing that all nodes are unvisited; $\mathbf{P_C}$ = vector of components in the current path of the MLS, initiated as an empty vector representing that no nodes are yet included in the path; L_C = length of current path, initiated as 0 indicating that the current path length is zero; and **ch** = vector of children nodes of S found in **Con** not yet visited.

In the algorithm, first, a comparative distance D_c is calculated as a multiplier M times the shortest distance D_S. This step is performed because it is not logical that a resource will take an excessively long path if a much shorter path is available. The multiplier used in the application in this paper is two, meaning that the longest distance considered is two times longer than the shortest distance from any supply node to the target node. Next, the start component S is marked as visited in the visited vector **Vis**. The start component is then added to the current path $\mathbf{P_C}$. Unless the current path only contains one node, the length of the link added to the current path is added to the current length L_C. The children variable **ch** is defined as the children of the start component S—found in **Con**—that have not been visited as found in **Vis**. A MLS is discovered if the source component is equal to the target component ($S = T$) and is defined as the current path $\mathbf{P_C}$. If an MLS is not found, the algorithm then cycles through each element i of the child vector **ch** and sets the child as the source component S. If the current path length L_C is greater than the cutoff distance D_C, the algorithm moves on to the next supply node. Otherwise, new variables are defined to move on to the next recursion. The new visited vector is set as **newVis** and the new start component is marked as visited. Finally, the algorithm calls itself to repeat with the next S component until a MLS is reached or the algorithm has visited all elements of the **ch** vector.

A second method for reducing the dimensionality of the network is to use supercomponents (Der Kiureghian and Song 2008; Tien 2017), which combine multiple components in order to model them using a representative single node. One way to define a supercomponent is as a subset of components in the system that are connected in series or parallel (Tien and Der Kiuregian 2017). Supercomponents reduce the effective number of nodes in the BN while still representing the state of each node in the system (Bensi et al. 2013). The components that comprise the supercomponent are represented as parents of the supercomponent in the BN, reducing the computational complexity of the model without making any approximating assumptions.

For our algorithm, we defined a supercomponent when its state was known given the failure of any one of its constituent components. In this approach, components in a series configuration are grouped into a supercomponent. Those components become parents of a supercomponent node in the BN. Algorithm 2, which defines the supercomponents, uses only the connectivity matrix **Con** and a vector of the supply components **s** as inputs. The algorithm is as follows:

Algorithm 2. Supercomponent identification algorithm.

Input: $\mathbf{Con}, \mathbf{s}$
For each transshipment and distribution component in the network (i)
    $\mathbf{c} = \mathbf{Con}(i, :)$
    If $\mathbf{s} \notin \mathbf{c}$ and length of $\mathbf{c} = 2$
        For each element of $\mathbf{c}(j)$
            If the length of $\mathbf{Con}(j, :) = 2$, supercomponent identified

The algorithm loops over each nonsupply component i in the network and defines the connection vector **c** as all components indicated in the connectivity matrix with values of one. Supply components are not included, because their functionality differs from that of transshipment and distribution components. The algorithm then selects connections in **c** that have exactly two connections, representing a component in series. Finally, the algorithm loops through all elements of the connection vector **c** and goes through the same selection process to either add components to the current supercomponent or to identify a new supercomponent.

Defining Interdependencies

In the modeling methodology, we included three comprehensive interdependency types (service provision, geographic, and access for repair) that affect the resilience of infrastructure systems. These three interdependency types were used because they encompass the possible connections between infrastructures and are well defined compared to previous definitions of interdependencies (Johansen and Tien 2017). For example, Rinaldi et al. (2001) defined four types of interdependencies—physical, cyber, logical, and geographic. In our framework, a service provision interdependency refers to the function of a component in one system relying on the function of a component from another system. This covers both physical and cyber interdependencies as defined by Rinaldi et al. (2001). An example of this is a water pump station requiring electricity from a power substation to function. If there is an outage at the power substation, the water pump station will fail, particularly if no backup power is present.

A geographic interdependency refers to the relationship between two or more components in the same geographic area that are likely to experience similar effects given a local hazard. This interdependency type is consistent with the geographic interdependency defined by Rinaldi et al. (2001). For example, components in proximity to one another are likely to fail concurrently if a hazard occurs in their vicinity. It is common for gas and water lines to be routed along the same roads so that only one trench is necessary; this represents a geographic interdependency in which the two lines are more likely to fail together given a common hazard.

An access for repair interdependency is defined for certain infrastructures that must be functioning in order to gain cyber or physical access to a failed component to repair it. For example, if a water network component loses function, communication systems are necessary in order to report the failure or gather information about the event through monitoring systems. Transportation systems must also be working so that repair crews can access the failed component. This interdependency type was developed specifically to address infrastructure resilience, taking into account systems necessary for postdisaster recovery and restoration (Johansen and Tien 2017). Rinaldi et al. (2001) defined a logical interdependency when the states of two infrastructures each depend "on the state of the other via a mechanism that is not a physical, cyber, or geographic connection." We instead used the three explicit interdependency types of service provision, geographic, and access for repair that can be quantitatively rather than subjectively modeled.

Previous work in identifying and accounting for interdependencies in infrastructure networks includes Chou and Tseng (2010) and Halfawy (2008). In Chou and Tseng (2010), failure records of different infrastructure types were used to predict interdependencies through sequence-based failure events. Halfawy (2008) focused on how to integrate management of multiple municipalities in order to optimize asset management decisions over multiple infrastructure types that may have different owners. Both of these approaches could easily be integrated into our proposed framework. New interdependencies learned or predicted could be added to the model through the defined interdependency relationships in order to assess potential cascading failures. The results from the models presented in this paper can be used across infrastructure owners to address priorities in investment to mutually benefit multiple infrastructure stakeholders. Another approach to analyzing infrastructure interdependencies is the inoperability input-output model. This model analyzes how disruptions to one infrastructure system propagate to other infrastructure systems through the exchange of input and output resources that are transferred between systems (Satumtira and Dueñas-Osorio 2010). These models are typically applied to account for economic interdependencies between infrastructure systems (Akhtar and Santos 2012; Santos et al. 2014). If desired, nodes representing economic variables could be added to our proposed framework, both at the component and system levels. Here, we focused on the physical performance of the infrastructure systems.

The next step in our methodology was to identify the interdependencies that exist in the infrastructure networks being modeled. We modeled each of the three comprehensive interdependency types (service provision, geographic, and access for repair) affecting the resilience of infrastructure systems.

To model a service provision interdependency, a direct link is added from the supplying (parent) component providing the service to the dependent (child) component. For example, to model the dependency of a water pump station on power, a link is added from the power substation to the water pump station. An assumption made in building the BN was that that the closest supplying component provides the resource to the dependent component (Dueñas-Osorio et al. 2007). If other information on how resources are supplied across infrastructure networks is available, that information is easily incorporated into the BN through direct links between those components. Child components can have multiple parents. For example, water treatment plants often have feeds from multiple power substations. In that case, each of the substations is represented as a parent node. Components typically requiring a service provision interdependency include natural gas and water supply components that depend on power. Other components in each of these systems do not require power in order to function. For example, water distribution components, particularly in older systems, can usually operate without power. As systems become increasing automated, however, service provision interdependencies will increase.

An example BN for a service provision interdependency is shown in Fig. 2, in which the service provision interdependencies are shown as dashed arrows. The example BN comprises a power system of components $C_{1p}, \ldots, C_{mp}$, where m is the number of power components, and a water system of components $C_{1w}, \ldots, C_{nw}$, where n is the number of water components. The MLSs are numbered $\text{MLS}_1, \ldots, \text{MLS}_t$, where t represents the number of MLSs for the water network. Using the previous example, each component, $C_{\_p}$, represents a power substation and $C_{\_w}$ represents a water pump station.

We model geographic interdependencies by grouping components into zones. For each zone, we add a common hazard node, which is a parent of all of the components in the zone. Hazard nodes can be created for any set of components in order to capture multiple hazard impacts. These include cyber nodes that represent cyber threats on infrastructure systems, and natural disaster nodes that represent earthquake or hurricane threats. Zones can be determined based on proximity to certain hazards, collocation of components, or service areas around supply components. An example

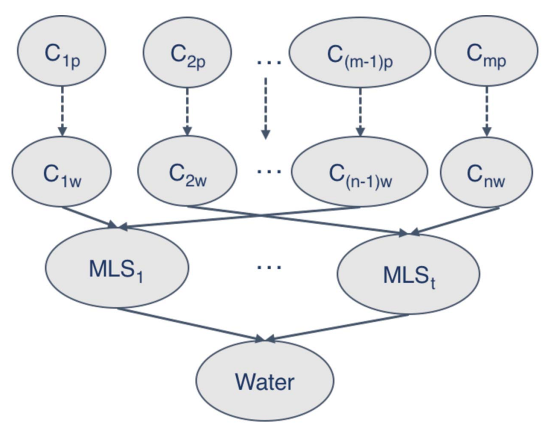

Fig. 2. Example BN for service provision interdependency.

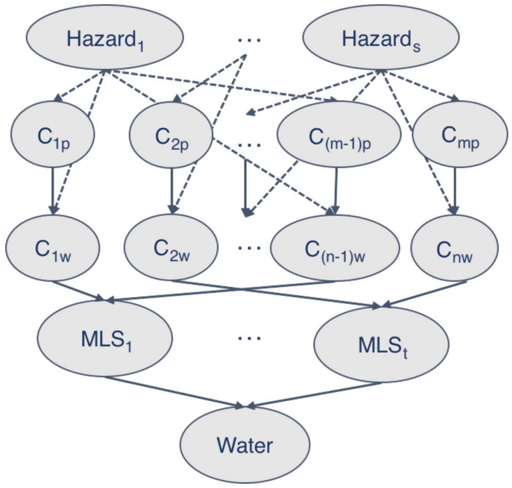

Fig. 3. Example BN for geographic interdependency.

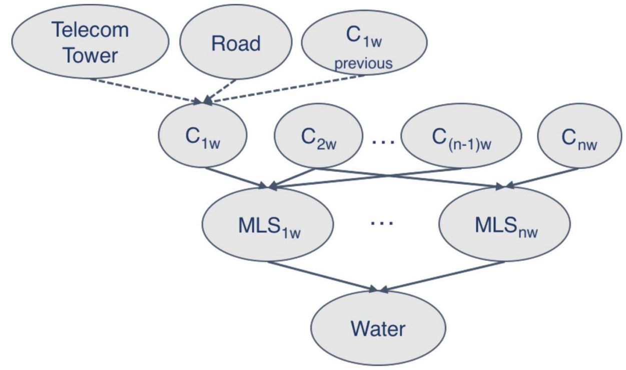

Fig. 4. Example BN for access for repair interdependency.

of using proximity to a hazard is partitioning components based on their distance from an earthquake fault line. An example of partitioning components using service areas is using a k-nearest neighbor search to group components by their closest supply nodes.

Fig. 3 shows an example BN for a geographic interdependency, in which the dashed lines represent the geographic interdependencies. Hazard zones are represented by Hazard_1 to Hazard_s, where s denotes the number of hazard zones. Components with the same hazard parents are in the same hazard zone. In the example, components C_{1p}, C_{1w}, $C_{(m-1)p}$, and $C_{(n-1)w}$ are all in Hazard zone 1.

System nodes can also be created based on the geographic partitions that represent the resources provided to particular service areas, which are used when performing inference on the system. Each system node is a child of all of the components within that system, such as those within a geographic partition.

When modeling an access for repair interdependency, the change in the operational status of infrastructure components over time is taken into account. Access nodes are created as parent nodes of the components that depend on them. Access nodes only affect the state of a child component in the case of component failure. A working component is independent, for example, of the state of its connected communication or transportation networks. Therefore, a node representing the state of the component in its previous time step is created, allowing the determination of the need to account for the state of an access node. In defining the access nodes, for cyber access, these nodes account for required communication with the dependent component and the robustness of the communication channels to disruptions. For physical access, the access nodes represent remoteness and redundancy in transportation paths to reach the component. The probability of repair of a component given access can be defined based on the criticality of the component or the availability of resources for repair.

An example BN is shown in Fig. 4 with C_{1w} as the potentially failed node and the access nodes defined as a telecommunications tower and road providing cyber and physical access, respectively. The state of the water node in the previous time step is represented by $C_{1w\text{previous}}$.

Adjacency Matrix

The structure of a BN is defined by an adjacency matrix. An adjacency matrix is similar to a connectivity matrix, which is composed of ones and zeros. In this case, a value of one indicates a connection between a parent node in a row and a child node in a column, and zero indicates no connection. In addition, rather than merely capturing the original network topology with components as in a connectivity matrix, the adjacency matrix includes the defined MLSs, supercomponents, and interdependency relationships. The adjacency matrix also accounts for the directionality of the dependency and is not symmetric. The adjacency matrix was constructed based on the parent–child relationships defined in previous steps. For MLSs in particular, each component that comprises an MLS is a parent of an MLS node, and each MLS node is a parent of the component for which it is an MLS, providing the paths for a resource to reach a component. This functional relationship introduces potential cycles into the graph. The method to address these cycles is presented in the following section.

Cycles

In the modeling and assessment of complex infrastructure systems, it is possible for cycles to arise in the creation of the BN graph. For example, suppose component C_1 in the water network, denoted C_{1w}, is a part of a MLS for component C_{nw}. Suppose at the same time that, based on the topology of the network, C_{nw} is a part of a MLS for component C_{1w}. With these two components each a part of the other's MLSs, a cycle is introduced. BNs, however, must be acyclic graphs. Typically in such a case, the system would no longer be able to be modeled as a BN. We developed a method to identify and remove cycles from the BN (discussed subsequently) and account for the dependency that cycles introduce (also discussed subsequently).

An example of a cycle that arises from MLS formulation is shown in Fig. 5, in which component C_1 is part of the MLS_2 for C_2 and C_2 is included in MLS_1 for C_1. Here we present a method to remove the cycles in the graph while retaining the dependency relationships between the nodes. Specifically, we

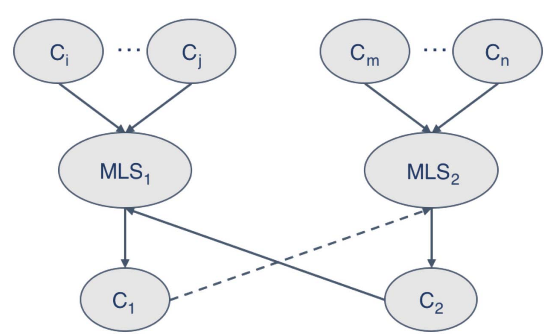

Fig. 5. Example BN with cyclic dependency.

J. Comput. Civ. Eng., 2019, 33(1): 04018058

defined the components by their joint probability distribution and removed one of the links from a component to a MLS, removing the cycle. In the example, we removed the link from C_1 to MLS_2 (shown by the dotted line). The joint probabilities were calculated by considering the possible configurations of each component state and using total probability to calculate the remaining values. This calculation is further described in Algorithm 3, which is presented subsequently, to define the MLS conditional probability tables.

In this step, we must identify all cycles in the graph. We do this by traversing the graph represented by the adjacency matrix and identifying when the path traversed reaches a previously visited node. This represents a cycle in the graph. Once the cycle is identified, a link from a component to a MLS node is removed in the adjacency matrix.

When traversing the graph, we begin with component C_i. A visited vector is constructed as $[\mathbf{C_i}]$. The graph traverse algorithm then moves to C_i's connections—in this case, MLS_1. The visited vector is now $[C_i, \mathrm{MLS}_1]$. This continues, traversing to each connection until we reach $[C_i, \mathrm{MLS}_1, C_1, \mathrm{MLS}_2, C_2, \mathrm{MLS}_1]$. In this step, a node in the visited vector is repeated, indicating a cycle. Thus, we remove a link in the adjacency matrix, in this case the link from C_1 to MLS_2, by setting that entry to zero. We note the removed link to define the conditional probability distribution in the next step.

Defining Conditional Probability Tables

Each node in the BN must be defined by a conditional probability distribution of its state given the states of its parents, typically represented for discrete or discretized variables in a conditional probability table. The calculation of the CPT varies for each node type. Details for the CPT calculation for nodes defining geographic interdependencies, access for repair interdependencies, supply components including service provision interdependencies, transshipment and distribution components, and MLSs both noncyclic and cyclic are provided in this section

Geographic Interdependency Nodes

When defining the geographic interdependency in the network, hazard nodes and zone nodes are created that represent components in the same physical area that are likely to experience correlated outcomes in the case of a hazard. Hazard nodes do not have parent nodes, so their CPTs are defined as the marginal probabilities of hazard occurrence, as shown in Eq. (1), where H_i, $i = 1, \ldots, n_H$ represents a hazard node and n_H is the number of hazard zones. The probability of occurrence of hazard H_i is p_{H_i}

$$\begin{aligned} &\mathrm{P}(H_i \text{ occurs}) = p_{H_i} \\ &\mathrm{P}(H_i \text{ does not occur}) = 1 - p_{H_i} \end{aligned} \tag{1}$$

To assess the overall performance of the infrastructure systems, we evaluate levels of service provided in each service area in the community. We use zone nodes to represent these service areas; the parents of zone nodes include all of the nodes within the zone. Eq. (2) defines the CPTs for zone nodes, where Z_j, $j = 1, \ldots, n_Z$ represents a zone node and n_Z is the number of zones. The components within each zone are represented by $\mathbf{C}^{\boldsymbol{k}}_{\mathbf{Z_j}}$, $k = 1, \ldots, n_{Z_j}$, where n_{Z_j} is the number of components in zone Z_j. In Eq. (2), the percent level of service is denoted as N. A zone's level of service is defined as the percentage of components that are in the working state within the zone's partition

$$\begin{aligned} &\mathrm{P}(Z_j \text{ is at } N\% \text{ Service}) \\ &\quad = \begin{cases} 1 \text{ if } N\% \text{ of } \mathbf{C}^{\boldsymbol{k}}_{\mathbf{Z_j}}, & k = 1, \ldots, n_{Z_j} \text{ survive} \\ 0 & \text{otherwise} \end{cases} \end{aligned} \tag{2}$$

Access for Repair Interdependency Nodes

Access nodes are created when defining access for repair interdependencies. We define these for all supply, transshipment, and distribution components in the networks that depend on them for repair. Eq. (3) defines a component CPT when it has an access node as a parent, where C_m denotes the component, as acc_m denotes the access node, and $C_{m\,prev}$ denotes the component's state in the previous time step. The probability of failure of component C_m is p_{f_m}. The probability of repairing component C_m is defined as p_{repair_m}

$$\begin{aligned} &\mathrm{P}(C_m \text{ survives}) = \begin{cases} 1 - p_{f_m} & \text{if } acc_m \text{ survives and } C_{m\,prev} \text{ survives} \\ 1 - p_{f_m} & \text{if } acc_m \text{ fails and } C_{m\,prev} \text{ survives} \\ p_{repair_m} & \text{if } acc_m \text{ survives and } C_{m\,prev} \text{ fails} \\ 0 & \text{if } acc_m \text{ fails and } C_{m\,prev} \text{ fails} \end{cases} \\ &\mathrm{P}(C_m \text{ fails}) = \begin{cases} p_{f_m} & \text{if } acc_m \text{ survives and } C_{m\,prev} \text{ survives} \\ p_{f_m} & \text{if } acc_m \text{ fails and } C_{m\,prev} \text{ survives} \\ 1 - p_{repair_m} & \text{if } acc_m \text{ survives and } C_{m\,prev} \text{ fails} \\ 1 & \text{if } acc_m \text{ fails and } C_{m\,prev} \text{ fails} \end{cases} \end{aligned} \tag{3}$$

Supply Components

In defining CPTs for supply components, the parent nodes of supply components typically include hazard nodes and service provision interdependency nodes. Eq. (4) represents the CPT formulation for supply nodes. The supply node is represented by S_q, $q = 1, \ldots, n_S$, where n_S is the number of supply nodes. The conditional probabilities of component failure given that a hazard occurs or does not occur are represented by $p_{f_{q|haz}}$ and $p_{f_{q|no\,haz}}$, respectively. The hazard node of which the supply component is a child is represented by H_i, and the service provision interdependency parent is represented by R_s

$$\begin{aligned} &\mathrm{P}(S_q \text{ survives}) = \begin{cases} 1 - p_{f_{q|haz}} & \text{if } H_i \text{ occurs and } R_s \text{ survives} \\ 0 & \text{if } H_i \text{ occurs and } R_s \text{ fails} \\ 1 - p_{f_{q|no\,haz}} & \text{if } H_i \text{ does not occur and } R_s \text{ survives} \\ 0 & \text{if } H_i \text{ does not occur and } R_s \text{ fails} \end{cases} \\ &\mathrm{P}(S_q \text{ fails}) = \begin{cases} p_{f_{q|haz}} & \text{if } H_i \text{ occurs and } R_s \text{ survives} \\ 1 & \text{if } H_i \text{ occurs and } R_s \text{ fails} \\ p_{f_{q|no\,haz}} & \text{if } H_i \text{ does not occur and } R_s \text{ survives} \\ 1 & \text{if } H_i \text{ does not occur and } R_s \text{ fails} \end{cases} \end{aligned} \tag{4}$$

Transshipment and Distribution Components

Transshipment and distribution components have MLSs and hazard nodes as parents. Eq. (5) shows the CPT formulation for transshipment and distribution components. The components are represented by the variable C_t, $t = 1, \ldots, n_d$, where n_d is the number of nonsupply components. The MLSs that are parents of the component are represented by MLS_{d_v}, $v = 1, \ldots, n_M$, where n_M is the number of MLS parent nodes of component C_t. In Eq. (5), the component is also a child of a single hazard node H_i. The conditional probabilities of component failure given that the hazard occurs or does not occur are represented by $p_{f_{t|haz}}$ and $p_{f_{t|no\,haz}}$, respectively

$$P(C_t \text{ survives}) = \begin{cases} 1 - p_{f_{t|haz}} & \text{if } H_i \text{ occurs and any MLS}_{d_v} \text{ survives} \\ 1 - p_{f_{t|no\,haz}} & \text{if } H_i \text{ does not occur and any MLS}_{d_v} \text{ survives} \\ 0 & \text{if } H_i \text{ occurs and no MLS}_{d_v} \text{ survive} \\ 0 & \text{if } H_i \text{ does not occur and no MLS}_{d_v} \text{ survive} \end{cases}$$

$$P(C_t \text{ fails}) = \begin{cases} p_{f_{t|haz}} & \text{if } H_i \text{ occurs and any MLS}_{d_v} \text{ survives} \\ p_{f_{t|no\,haz}} & \text{if } H_i \text{ does not occur and any MLS}_{d_v} \text{ survives} \\ 1 & \text{if } H_i \text{ occurs and no MLS}_{d_v} \text{ survive} \\ 1 & \text{if } H_i \text{ does not occur and no MLS}_{d_v} \text{ survive} \end{cases} \tag{5}$$

Minimum Link Set Nodes

The functioning of an MLS depends on the functioning of the components in the MLS. Therefore, the parents of the MLS nodes are the components that comprise the MLS. However, there are two types of MLSs—those without cyclic links and those with cyclic links that have been removed. The formulations for the CPTs for MLSs in these two cases are described subsequently.

Eq. (6) shows the CPT formulation for MLSs that did not contain cycles, and therefore do not contain links that have been removed with the cycle removal algorithm. The MLS nodes are represented by $\text{MLS}_w, w = 1, \ldots, n_{M_{NC}}$, where $n_{M_{NC}}$ is the number of noncyclic MLSs. The components comprising the MLS are denoted $\mathbf{C}_{\mathbf{w}_x}$, $x = 1, \ldots, n_{w_x}$, where n_{w_x} represents the number of components in MLS_w

$$P(\text{MLS}_w \text{ survives}) = \begin{cases} 1 & \text{if all } \mathbf{C}_{\mathbf{w}_x} \text{ survive} \\ 0 & \text{if any } \mathbf{C}_{\mathbf{w}_x} \text{ fails} \end{cases}$$

$$P(\text{MLS}_w \text{ fails}) = \begin{cases} 0 & \text{if all } \mathbf{C}_{\mathbf{w}_x} \text{ survive} \\ 1 & \text{if any } \mathbf{C}_{\mathbf{w}_x} \text{ fails} \end{cases} \tag{6}$$

For MLSs containing cycles and, therefore, links that have been removed during the cycle identification process, the CPTs defined for the MLS nodes must account for the removed links. This is done using values from the joint probability distribution of the nodes for which links have been removed. Let the cyclic MLSs be denoted MLS_a, $a = 1, \ldots, n_{M_C}$, where n_{M_C} is the number of cyclic MLSs. The components that comprise an MLS are denoted $\mathbf{C}_{\boldsymbol{a_b}}$, $b = 1, \ldots, n_{a_b}$, where n_{a_b} represents the number of components in MLS_a. The removed links for each MLS are represented in $\mathbf{L}_{\mathbf{rem}}$, a $y \times 2$ matrix in which y is the number of removed links for a specific MLS, and each row represents the parent to child link that was removed. Each removed link is defined as $\mathbf{L}_{\mathbf{rem}_z}$, $z = 1, \ldots, y$. The conditional probabilities of failure of components C_{a_b} given that a hazard occurs or does not occur are represented by $p_{f_{b|haz}}$ and $p_{f_{b|no\,haz}}$, respectively. The probability of a hazard occurring is p_{H_i}, and the joint probability value calculated for use in the CPT is $p_{MLS\,cyc}$. The probability of failure of the link that is removed is calculated as the product of marginal failure probabilities of the parents of $\mathbf{L}_{\mathbf{rem}_z}$. The algorithm for formulating the CPT for MLSs with cyclic links is presented in Algorithm 3 as follows:

Algorithm 3. Algorithm for cyclic MLS CPT formulation.

For each row z in $\mathbf{L}_{\mathbf{rem}}$:

Parents of $\mathbf{L}_{\mathbf{rem}_z}$ are components corresponding to 1 values in adjacency matrix in row $\mathbf{L}_{\mathbf{rem}_z}$

$$P(\mathbf{L}_{\mathbf{rem}_z} \text{ fails}) = \prod_{b=1}^{n_{a_b}} [p_{f_{b|haz}} \cdot p_{H_i} + p_{f_{b|no\,haz}} \cdot (1 - p_{H_i})]$$

$$\begin{aligned} p_{MLS\,cyc} = {} & \sum_{z=1}^{y} P(\mathbf{L}_{\mathbf{rem}_z} \text{ fails}) - \sum_{1 \le c < d \le y} P(\mathbf{L}_{\mathbf{rem}_c} \text{ fails}) \cap P(\mathbf{L}_{\mathbf{rem}_d} \text{ fails}) \\ & + \sum_{1 \le c < d < f \le y} P(\mathbf{L}_{\mathbf{rem}_c} \text{ fails}) \cap P(\mathbf{L}_{\mathbf{rem}_d} \text{ fails}) \cap P(\mathbf{L}_{\mathbf{rem}_f} \text{ fails}) - \cdots \\ & + (-1)^{n-1} [P(\mathbf{L}_{\mathbf{rem}_1} \text{ fails}) \cap \ldots \cap P(\mathbf{L}_{\mathbf{rem}_y} \text{ fails})] \end{aligned}$$

$$P(MLS_a \text{ survives}) = \begin{cases} 1 - p_{MLS\,cyc} & \text{if all } C_{a_b} \text{ survive} \\ 0 & \text{if any } C_{a_b} \text{ fails} \end{cases}$$

$$P(MLS_a \text{ survives}) = \begin{cases} p_{MLS\,cyc} & \text{if all } C_{a_b} \text{ survive} \\ 1 & \text{if any } C_{a_b} \text{ fails} \end{cases}$$

Once the CPTs for all nodes are defined, the BN model can be built.

J. Comput. Civ. Eng., 2019, 33(1): 04018058

Application

To demonstrate the proposed framework and our approach, we applied it to the interdependent water and power distribution networks in Atlanta, Georgia. We modeled the system and performed inference on the network using the model. We validated the methodology by comparing the results from inference using the constructed model to a real-world scenario in which a power outage led to cascading failures in the water system.

System Overview

For the water system, we analyzed pipes greater than or equal to 18 inches in diameter. This included 112 components, seven of which were supply stations and 105 of which were transshipment or distribution nodes. There were 244 links, or pipes, in the network. For the power system, we modeled the power substations that were located at each supply node. Supply nodes had between one and three electrical feeds, varying with each supply component.

Fig. 6 shows the system with supply nodes shown as empty circles and distribution and transshipment nodes shown as solid points. The supply nodes are also the locations of the power components.

Bayesian Network Model

We then used the proposed framework to create the BN model of the interdependent infrastructure systems.

Inputs

The input file was 4 MB and included identification numbers and locations of 112 junctions in the water network. The start and end junctions and size characteristics of 350 pipes were included. The junctions were condensed to represent the start and end junctions of each pipe rather than accounting for all on-pipe junctions.

Component Locations. The component locations were given as state-plane coordinates in the example.

Component Connectivity. The component connectivity for the application was obtained from a list of each link in the network used in the hydraulic model of the system.

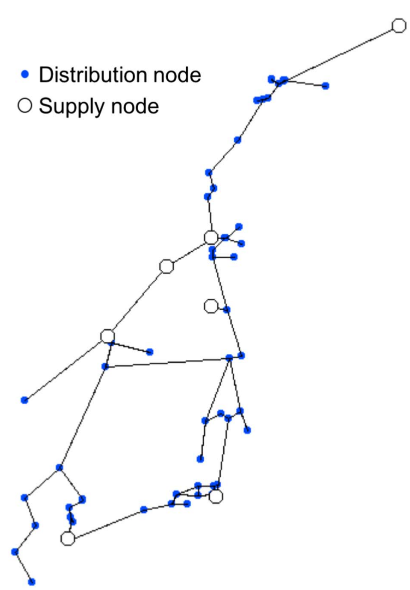

Fig. 6. Atlanta water and power distribution systems.

Component Type. The component types for the application were defined depending on their function, i.e., supply, transshipment, or distribution. The constituent elements of supply nodes, i.e., for pump stations and treatment plants, were aggregated into a single node for each supply. If such element-level information is available, it can easily be incorporated into the model as parents of the supply node. Supercomponent identification can be utilized to reduce dimensionality as needed.

Component Failure Probability. For the application, component failure probabilities were assumed to be consistent across each component in order to better assess relative component vulnerabilities. The failure probabilities given that a hazard occurred or did not occur were assumed to be 1×10^{-2} or 1×10^{-4}, respectively. The hazard in the example was generalized and could, for example, represent a storm. The equal prior failure probabilities across components resulted in ranking and component prioritization rather than specific failure probability values. If more information is learned about the components, the failure probabilities can easily be updated as inputs to the model.

Dimensionality Reduction

Running Algorithm 1 for the full system identified the MLSs from a supply node to each of the transshipment and distribution nodes in the network. This took approximately 2.19 s on a computer with 4 GB RAM and a 1.3 GHz Intel Core i5 processor using MATLAB 2017b for the entire network. This was a novel algorithm to identify MLSs, because none were found in previous literature. There were 246 MLSs in the full system. The maximum number of MLSs for a component was 5 and the maximum length of an MLS was 17 components. An example set of MLSs for node $C7$ is

$$\begin{bmatrix} C108, & C58, & C59, & C7 \\ C108, & C60, & C59, & C7 \end{bmatrix}$$

where the first component in each row is a supply node and the middle nodes are on the path to the final node. Supercomponents were not needed for this example.

Defining Interdependencies

The interdependencies modeled in the application were service provision and geographic. Service provision interdependencies were based on information provided by the owners of the water network. There were power substations located at each of the water supply stations. Each supply station had between one and three power substations. To model the service provision interdependencies, direct links were added from each power substation to the water supply node that it supplied. Backup generators could also be incorporated to account for continued power in the case of an outage of a main substation. There were a total of 15 power substations in the network that provided power to seven water supply nodes.

We partitioned the water and power networks into hazard zones that were used to represent geographic interdependencies. These hazard zones also represented service areas surrounding each of the water supply nodes. The seven zone partitions for the network are shown in Fig. 7.

Four of the service areas were split into two groups for ease of computation during inference. Therefore, in total, there were 11 partitions with hazard nodes as parents for the nodes in each of them. Nodes were also created as children of the distribution nodes in each zone, representing levels of service in each service area.

Adjacency Matrix

We build the adjacency matrix from identified MLSs and interdependency relationships between nodes. Each MLS was a parent of its dependent component node, and the components that

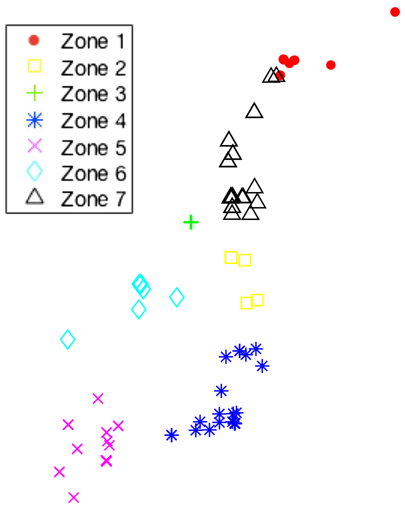

Fig. 7. Service areas partitioned by zone for application network.

comprised each MLS were parents of the MLS node. Links created by service provision interdependencies included defined parent and dependent children nodes. Geographic interdependency links included those from a hazard node to each node in the hazard zone, as well as links from each distribution component in a zone to the service level nodes used for posterior inference. There were 1,868 parent–child relationships defined in the adjacency matrix. The Bayes Net Toolbox (version 1.0.4) (Murphy 2001) was used to construct and perform inference in MATLAB. This toolbox required the components to be sorted topologically from parent to child nodes. The sorting and construction of the adjacency matrix took approximately 1.26 s.

Cycles

Defining all MLSs and parent–child relationships for the example system created 52 cycles in the network. We used the graph traverse algorithm to first identify these cycles and then remove one of the links from a component to a MLS for each cycle. This removed the cycles from the network in a negligible amount of time.

Defining Conditional Probability Tables

Overall, the definition of conditional probabilities took about 62.6 s. This built CPTs for each of the 392 nodes in the BN. The minimum CPT size was 2×1 (for hazard nodes). For nodes with n parents, the size of the CPT was $2 \times 2 \times n + 1$.

Geographic Interdependency Nodes. The probability that a hazard occurs was assumed to be 0.01. This was for generalized hazards and can be changed to the probability of occurrence of any specific hazard of concern. As an example, the CPT for service zone 1 was

$$\mathrm{P}(Z_1 \text{ is at } N\% \text{ Service}) = \begin{cases} 1 & \text{if } N\% \text{ of } C^k_{Z_j}, k = 1, \ldots, 12 \text{ survive} \\ 0 & \text{otherwise} \end{cases}$$

where $C_{Z_1} = C8, C9, C10, C11, C12, C13, C14, C15, C16, C17, C18, C19$. All transshipment and distribution components in zone 1 are included in C_{Z_1}. The CPT in this example was $2 \times 2 \times 13$.

Supply Components. The CPT for an example supply component, $C108$, is given in Table 1. In Table 1, H_9 indicates the hazard in its zone partition; R_1 and R_2 are the two power substations that supply $C108$; and S and F denote survival and failure, respectively, of the substations.

Table 1. CPT for example supply component, $C108$

	H_9 occurs				H_9 does not occur			
	R_1S		R_1F		R_1S		R_1F	
$C108$	R_2S	R_2F	R_2S	R_2F	R_2S	R_2F	R_2S	R_2F
Survives	0.99	0.99	0.99	0	0.9999	0.9999	0.9999	0
Fails	0.01	0.01	0.01	1	0.0001	0.0001	0.0001	1

Note: F = failure; and S = survival.

Table 2. CPT for example distribution component, $C99$

	H_3 occurs		H_3 does not occur	
$C99$	MLS_{C99_1}S	MLS_{C99_1}F	MLS_{C99_1}S	MLS_{C99_1}F
Survives	0.99	0	0.9999	0
Fails	0.01	1	0.0001	1

Table 3. CPT for example noncyclic MLS, MLS_{C70_1}

	$C_{\mathrm{MLS}_{C70_{1_1}}}$S		$C_{\mathrm{MLS}_{C70_{1_1}}}$F	
MLS_{C70_1}	$C_{\mathrm{MLS}_{C70_{1_2}}}$S	$C_{\mathrm{MLS}_{C70_{1_2}}}$F	$C_{\mathrm{MLS}_{C70_{1_2}}}$S	$C_{\mathrm{MLS}_{C70_{1_2}}}$F
Survives	1	0	0	0
Fails	0	1	1	1

Table 4. CPT for example cyclic MLS, MLS_{C66_3}

MLS_{C66_3}	$C_{\mathrm{MLS}_{C66_{3_1}}}$S	$C_{\mathrm{MLS}_{C66_{3_1}}}$F
Survives	P($C3$ survives)	0
Fails	$1 -$ P($C3$ survives)	1

Transshipment and Distribution Components. The CPT for an example transshipment or distribution component, $C99$, is presented in Table 2. In Table 2, H_3 indicates the hazard in its zone partition. In this example, component $C99$ has only one MLS parent, and S and F denote survival and failure, respectively, of the MLS.

Minimum Link Set Nodes. An example of the CPT for a noncyclic MLS, MLS_{C70_1} is given in Table 3, where $C_{\mathrm{MLS}_{C70_{1_1}}}$ is $C111$ and $C_{\mathrm{MLS}_{C70_{1_2}}}$ is $C91$; these are the two nodes that comprise MLS_{C70_1}.

An example of the CPT for a cyclic MLS, MLS_{C66_3}, is given in Table 4. In this example, the link that was removed was from $C3$ to MLS_{C66_3}. Therefore, the remaining parent of MLS_{C66_3} was $C_{\mathrm{MLS}_{C66_{3_1}}}$, which represents $C110$. The probability that $C3$ survives was calculated from Algorithm 3.

Output

Fig. 8 shows the overall BN model. The hazard nodes are denoted $\mathrm{Hazard}_1, \ldots, \mathrm{Hazard}_9$. These are parents of the power and water components and represent the geographic interdependencies. The power supply components are denoted $\mathrm{Power\ Supply}_1, \ldots, \mathrm{Power\ Supply}_{15}$. These are parents of water supply components, representing service provision connections. Water supply components are denoted $\mathrm{Water\ Supply}_1, \ldots, \mathrm{Water\ Supply}_7$, and water distribution components are denoted $\mathrm{Water\ Distribution}_1, \ldots, \mathrm{Water\ Distribution}_{105}$. Water distribution components are parents of zone partitions $\mathrm{Zone}_1, \ldots, \mathrm{Zone}_9$, which represent levels of service throughout the network. Both water supply and distribution components are parents of

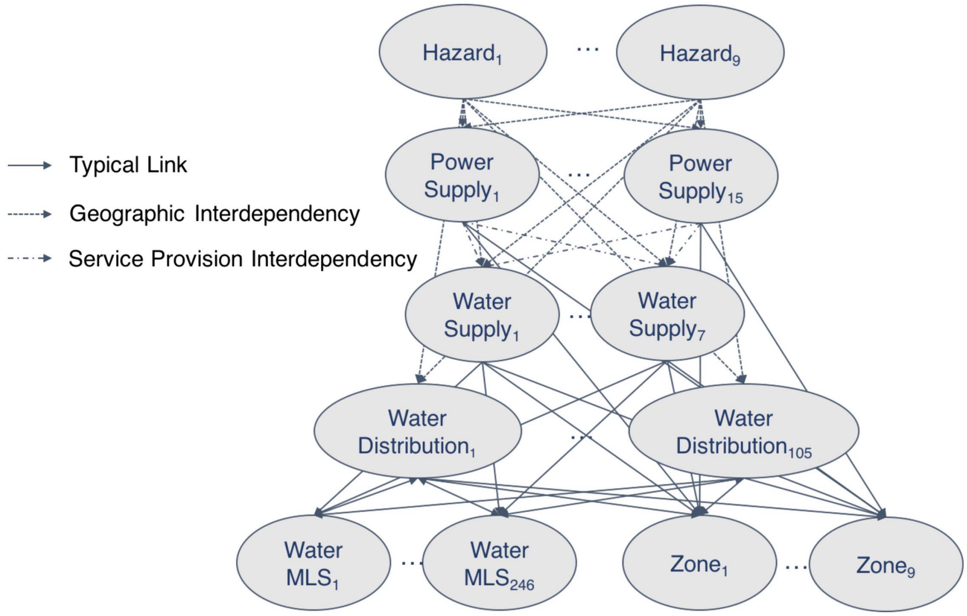

Fig. 8. Overall BN model of Atlanta water and power distribution networks.

MLSs denoted Water MLS$_1$, . . . , Water MLS$_{246}$. MLSs are parents of the distribution components that they supply. The subscripts represent the number of nodes of each type in the network; the BN comprised 382 total nodes.

Validation

We validated the model using a real-world scenario of cascading failures due to the interdependent nature of infrastructure networks that occurred in both 2014 and 2017. In these instances, a water pump station lost power from both of its dual feeds and caused outages throughout Atlanta's downtown area. The water system lost pressure in both cases and a boil water advisory became necessary. To test the scenario with the model, we simulated an outage to the power components supplying the affected pump station. The resulting network showed outages throughout the downtown area, as shown in Fig. 9. This was consistent with the outcomes of the event in which the downtown area lost water pressure. We used the loss of water pressure as an indicator for failure at the distribution level in the example. The BN model included all the complexities of the functionality and interdependencies of the networks, and showed the effects of the outage directly.

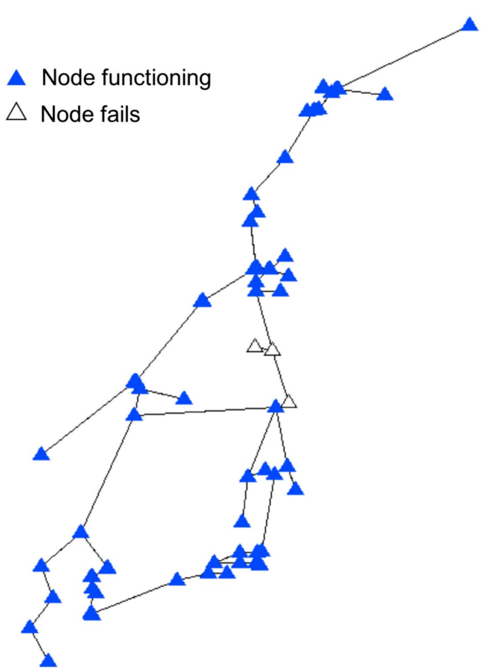

Fig. 9. Atlanta outage scenario for validation.

Example Inferences

With the BN model built, varying inferences could be conducted over the networks. The validation scenario above is an example of assessing the impacts of a service provision interdependency, in which the power supply of a water pump station failed and caused cascading outages in the water system. Examples of other probabilistic vulnerability analyses include assessing the impacts of hazards occurring in specific zones—geographic interdependencies—or evaluating the effects of failures within the water system itself.

Fig. 10 shows inference results from a hazard occurring in hazard zones 1 and 2. The gradient on the right represents failure probabilities. Hazard zones 1 and 2 are in the upper right corner of the system, so components in that area experienced increased probabilities of failure. Because the supply nodes were distributed throughout the rest of the network, no additional outages were experienced due to this event scenario.

Another example of inference is to assess the effects of an observed outage or the failure of a specific component in the network. Inference over the BN updates the failure probabilities of all nodes throughout the network. Fig. 11 shows the results from learning that a large supply component in the bottom right area of the network has failed. The figure shows the effects of such an outage on the ability to provide service in that part of the network.

These inferences were performed to highlight the abilities of the proposed framework. The results shown are a small subset of information that can be gained from the interdependent infrastructure model. The model allows a user to input information across a wide

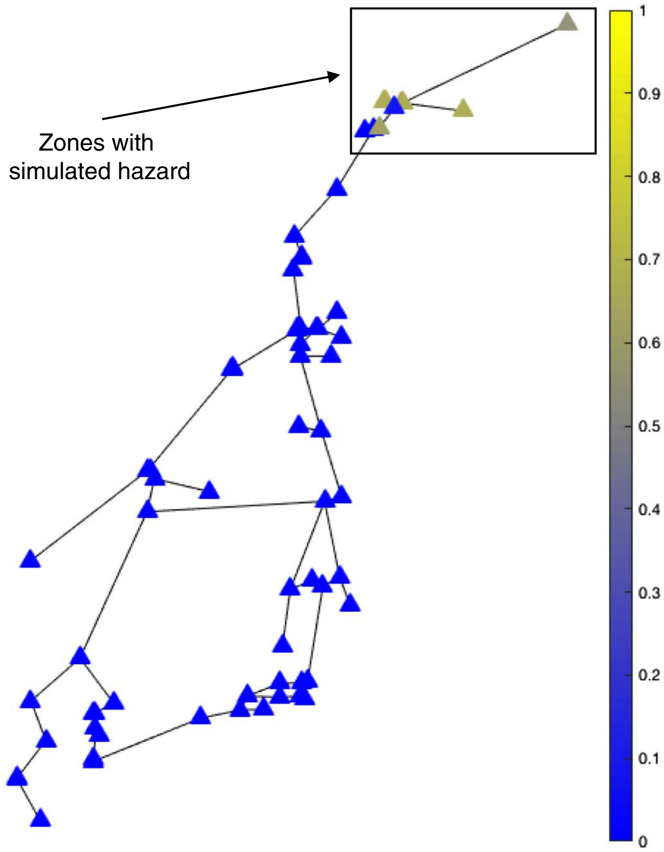

Fig. 10. Inference results from hazard occurrence in Zones 1 and 2.

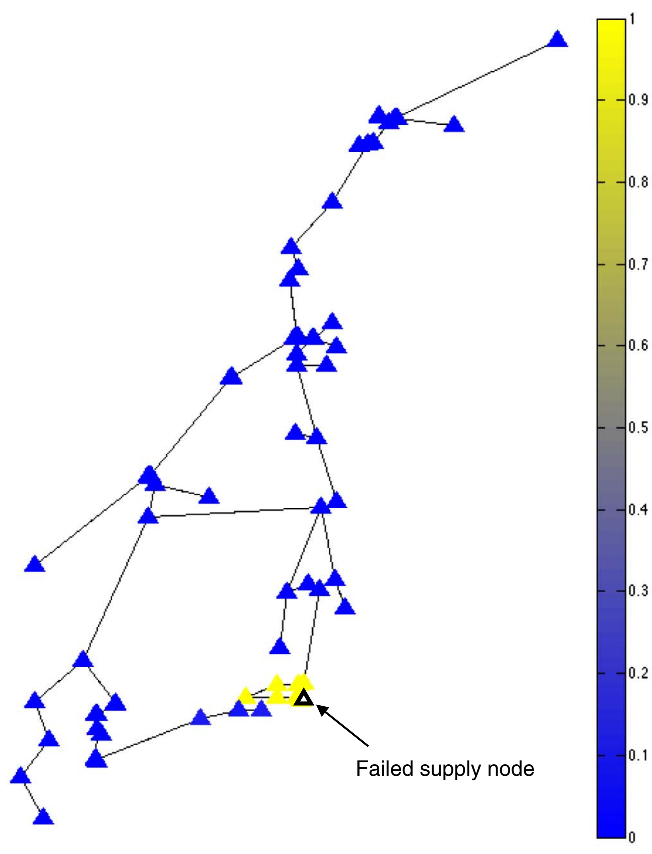

Fig. 11. Inference results from supply node failure.

range of possible scenarios—for example, outages that are experienced or expected, hazard occurrences, or updated information on a component such as failure, retrofit, or replacement. The user can then visualize and observe the updated probabilities of failure in components throughout the network. The aforementioned inferences were performed in approximately 4 s each. The output was achieved in a computationally efficient manner and was based on a full representation of the network, including the performance of its constituent individual components and the interdependencies that exist across systems.

Performance Compared to Prior Approaches

To further assess the performance of the proposed methodology, we compared it to that of prior approaches in several steps of the framework. The MLS enumeration took 2.19 s using the proposed method. While there were no previous algorithms found in the literature to identify these MLSs, an algorithm was developed to enumerate the complement to MLSs, MCSs. Mishra et al. (2015) proposed an algorithm to identify MCSs that used the connectivity matrix of a graph to check the connection between nodes in a network as nodes are progressively removed. The largest system that this algorithm was tested on in the study contained 21 nodes and 26 links. The enumeration of the MCSs took approximately 2,600 s. This was over 1,000 times longer for a network that was approximately five times smaller than the application used in this paper. The MLS formulation we have presented allows us to expand the number of components that are included in the network with increased computational efficiency compared to other methods of identifying minimum sets in a network.

Prior approaches to modeling interdependent infrastructure systems using BNs have focused on network characteristics at the global level rather than including system topologies from the component level to study system reliability and prioritize repair and retrofit for components. Therefore, the inference examples in this paper are not comparable to works such as Aung and Watanabe (2010) and Di Giorgio and Liberati (2011). A BN approach without MLS formulation was explored by Schaberreiter et al. (2013). However, that study applied to a system of four infrastructure component nodes and four service nodes. The approach was not scalable to infrastructure systems of the size we were interested in for this paper or the network used in the application.

Finally, we compared the performance of the proposed methodology to results from Monte Carlo simulation. Samples of probabilities of failure for each component were selected based on hazard occurrence using the same probabilities as described in the application for the proposed model. The failure or survival of each component was used to update the survival or failure of each MLS. These updated MLSs were then used to update the survival or failure of the nodes that depended on them. The outcome was the probability of survival of each component node. The Monte Carlo was performed using 10^3, 10^4, 10^5, and 10^6, and 10^7 simulations. The calculated probabilities of survival of all components are shown in Fig. 12. The solid circles represent the probabilities of survival of each component calculated using the proposed framework. These are the exact solutions. The open diamonds, circles, squares, and triangles and represent the probabilities obtained from 10^3, 10^4, 10^5, and 10^6 simulations, respectively, and $\times$s represent those obtained from 10^7 simulations. The computation times required for each method and the average percent errors over all components are presented in Table 5. As expected for Monte Carlo, the error decreased with an increase in the number of simulations. However, the average errors decreased slowly as the order of magnitude

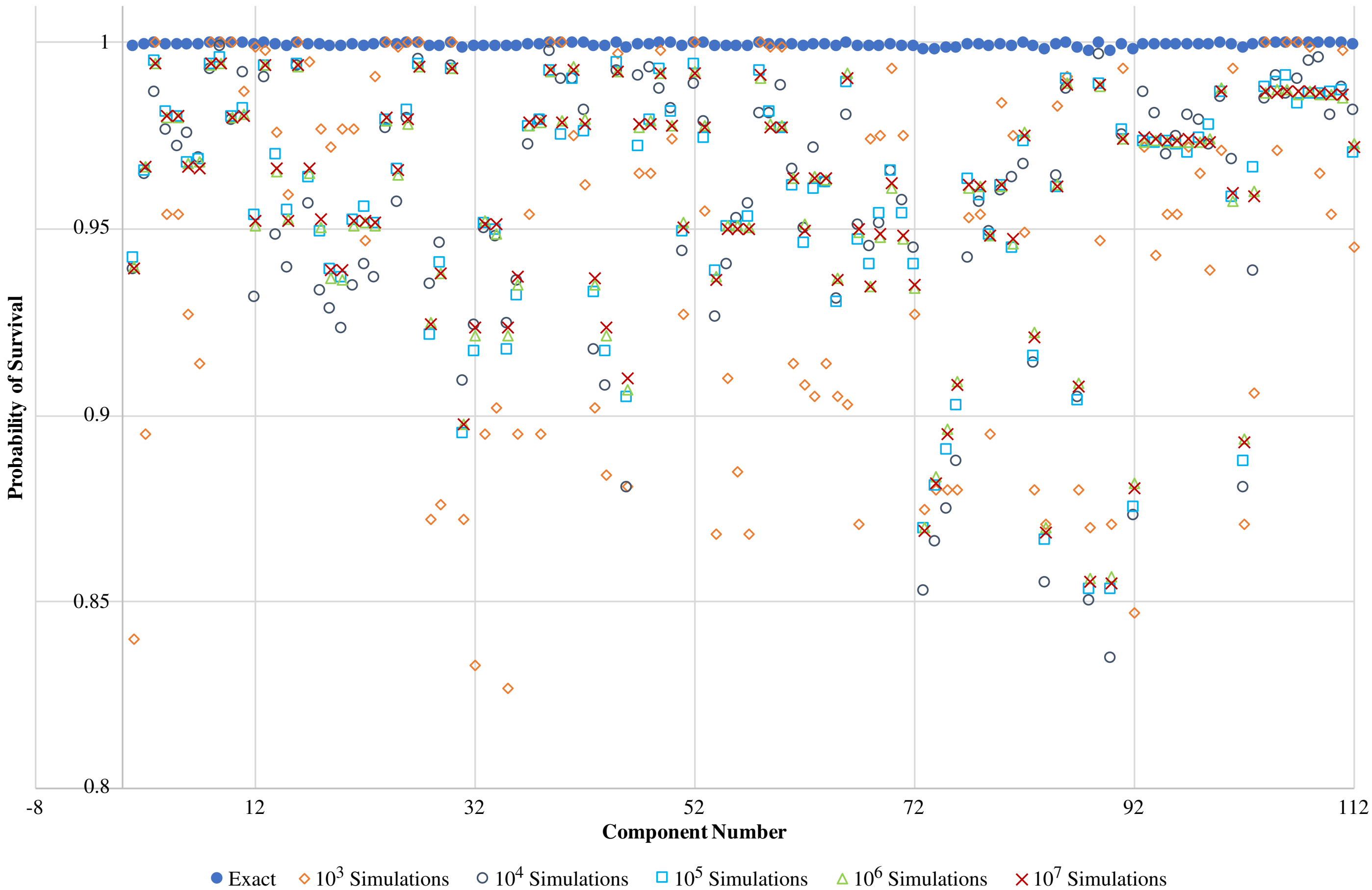

Fig. 12. Comparison of results from proposed methodology to Monte Carlo simulations.

Table 5. Comparison of performance of proposed methodology to Monte Carlo simulations

Result	Proposed	10^3 simulations	10^4 simulations	10^5 simulations	10^6 simulations	10^7 simulations
Time (s)	88.66	0.17	0.67	5.31	55.49	3,684.67
Average error (%)	0	5.49	4.36	4.18	4.14	4.13

of the number of simulations increased. The computation time for 10^7 simulations was much higher than the proposed approach and also required a large amount of memory, with a 112×10^7 matrix representing the state of each component for every simulation. Results from the simulation approach did not capture the small differences in failure probabilities that the exact solutions obtained using the proposed method provided. The errors were significant given the importance of detailed granularity in the probabilities of survival in the results, particularly if they are to be used to rank component criticality in the interdependent network.

Conclusion

We have developed and validated a new generalized framework to perform probabilistic vulnerability analyses of interdependent infrastructures. In this paper, we have presented the proposed approach, including algorithms to construct a BN model of the interdependent infrastructure systems. The method results in computationally efficient modeling and analysis of large infrastructure networks with exact inference possible over any number of system states. Inference over the network enables scenario-based analyses and prioritization from the component level for repair, replacement, or reinforcement decisions. This is useful before hazards occur in order to assess where the greatest extent of damage is possible and how to invest resources to prevent large outages. The model is useful during hazards in order to determine where to dispatch resources and repair crews to bring the most customers or the most critical customers back online as quickly as possible. Finally, the model is useful after hazards in order to prioritize components for interventions to prevent similar incidents and impacts from occurring again in the future.

The BN formulation accounts for uncertainty within the system as well as the complex interdependencies between different infrastructure systems. Uncertainties in both individual component failure probabilities and the probabilistic connections between components are included in the model. Only simple inputs of the basic component characteristics of location, type, connectivity, and initial failure probabilities are required. Dimensionality reduction algorithms, including for minimum link sets, supercomponents, and cycle identification, allow the model to include hundreds of component nodes while remaining computationally efficient and without requiring any approximating assumptions. The proposed modeling approach and framework for analysis enable us to create more reliable and resilient networks by understanding where vulnerabilities exist within systems and the areas in which the investment of resources will lead to the greatest improvements in predicted system outcomes.

Acknowledgments

This material is based upon work supported by the National Science Foundation Graduate Research Fellowship under Grant No. DGE-165004. Support from the Department of Homeland Security through the Critical Infrastructure Resilience Institute and the National Science Foundation through Grant No. CNS-1541074 is also acknowledged.

Notation

The following symbols are used in this paper:

a = instance of cyclic MLS;
acc_m = access node m;
C_{a_b} = components that comprise a cyclic MLS;
Con = connectivity matrix;
C_m = component with access node as parent;
$C_{m\,prev}$ = component with access node as parent's state in previous time step;
C_t = transshipment or distribution component;
$\mathbf{C}_{\mathbf{w}_x}$ = components comprising $\mathrm{MLS_w}$;
$\mathbf{C}^{\mathbf{k}}_{\mathbf{Z}_j}$ = components within zone Z_j;
ch = unvisited children nodes of start component in **Con**;
D_C = comparative cutoff distance;
D_S = shortest distance from any supply component to target component of interest;
H = hazard node;
i = current element;
j = current zone number;
L = matrix of physical length of links in the network;
L_C = current length;
$\mathbf{L}_{\mathbf{rem}}$ = matrix of removed links;
M = multiplier;
MLS_a = cyclic MLSs;
MLS_{d_v} = MLSs that are parents of component C_t;
MLS_w = MLS node;
N = percent level of service;
n = number of components in the network;
n_{a_b} = number of components in MLS_a;
n_d = number of transshipment and distribution components;
newVis = new visited vector in recursive algorithm;
n_H = number of hazard zones;
n_M = number of MLS parent nodes of component C_t;
n_{M_C} = number of cyclic MLSs;
$n_{M_{NC}}$ = number of noncyclic MLSs;
n_S = number of supply nodes;
n_{w_x} = number of components in $\mathrm{MLS_w}$;
n_Z = number of zone nodes;
n_{Z_j} = number of components in zone Z_j;
$\mathbf{P}_\mathrm{C}$ = current path vector;
p_{f_m} = probability of failure of component C_m;
$p_{f_{q|haz}}$ = probability of failure of component q given a hazard occurs;
$p_{f_{q|no\,haz}}$ = probability of failure of component q given a hazard does not occur;
p_{H_i} = probability of occurrence of hazard H_i;
$p_{MLS\,cyc}$ = joint probability value calculated for the cyclic MLSs;
p_{repair_m} = probability of repair of component C_m;
q = instance of supply component;
R_s = service provision interdependency parent;
S = start component;
S_q = supply component q;
s = vector of supply components;
T = target component;
t = instance of transshipment or distribution component;
Vis = visited vector;
v = instance of MLS for a specific component;
w = instance of MLS node;
x = instance of component in MLS;
y = number of removed links;
Z = zone node; and
z = instance of removed link.

References

Akhtar, R., and J. R. Santos. 2012. "Risk-based input-output analysis of hurricane impacts on interdependent regional workforce systems." *Nat. Hazard.* 65 (1): 391–405. https://doi.org/10.1007/s11069-012-0369-0.

Aung, Z. Z., and K. Watanabe. 2010. "Modeling inoperability propagation using Bayesian networks." In Vol. 342 of *Proc., Int. Conf. on Critical Infrastructure Protection*, 199–212. Berlin: Springer.

Bensi, M., A. Der Kiureghian, and D. Straub. 2013. "Efficient Bayesian network modeling of systems." *Reliab. Eng. Syst. Saf.* 112: 200–213. https://doi.org/10.1016/j.ress.2012.11.017.

Bobbio, A., L. Portinale, M. Minichino, and E. Ciancamerla. 2001. "Improving the analysis of dependable systems by mapping fault trees into Bayesian networks." *Reliab. Eng. Syst. Saf.* 71 (3): 249–260. https://doi.org/10.1016/S0951-8320(00)00077-6.

Castillo, E., and Z. Grande. 2016. "Bayesian networks-based probabilistic safety analysis for railway lines." *Comput.-Aided Civ. Infrastruct. Eng.* 31 (9): 681–700. https://doi.org/10.1111/mice.12195.

Cheng, M.-Y., and N.-D. Hoang. 2016. "Slope collapse prediction using Bayesian framework with K-nearest neighbor density estimation: Case study in Taiwan." *J. Comput. Civ. Eng.* 30 (1): 04014116. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000456.

Chou, C.-C., and S.-M. Tseng. 2010. "Collection and analysis of critical infrastructure interdependency relationships." *J. Comput. Civ. Eng.* 24 (6): 539–547. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000059.

Codetta-Raiteri, D., A. Bobbio, S. Montani, and L. Portinale. 2012. "A dynamic Bayesian network based framework to evaluate cascading effects in a power grid." *Eng. Appl. Artif. Intell.* 25 (4): 683–697. https://doi.org/10.1016/j.engappai.2010.06.005.

Der Kiureghian, A., and J. Song. 2008. "Multi-scale reliability analysis and updating of complex systems by use of linear programming." *Reliab. Eng. Syst. Saf.* 93 (2): 288–297. https://doi.org/10.1016/j.ress.2006.10.022.

Di Giorgio, A., and F. Liberati. 2011. "Interdependency modeling and analysis of critical infrastructures based on dynamic Bayesian networks." In *Proc., 19th Mediterranean Conf. on Control and Automation*, 791–797. Corfu, Greece: Aquis Corfu Holiday Palace.

Dueñas-Osorio, L., J. I. Craig, and B. Goodno. 2007. "Seismic response of critical interdependent networks." *Earthquake Eng. Struct. Dyn.* 36 (2): 285–306. https://doi.org/10.1002/eqe.626.

González, A. D., L. Dueñas-Osorio, M. Sánchez-Silva, and A. L. Medaglia. 2016. "The interdependent network design problem for optimal infrastructure system restoration." *Comput.-Aided Civ. Infrastruct. Eng.* 31 (5): 334–350. https://doi.org/10.1111/mice.12171.

Grande, Z., E. Castillo, and E. Mora. 2017. "Highway and road probabilistic safety assessment based on Bayesian network models." *Comput.-Aided Civ. Infrastruct. Eng.* 32 (5): 379–396. https://doi.org/10.1111/mice.12248.

Halfawy, M. R. 2008. "Integration of municipal infrastructure asset management processes: Challenges and solutions." *J. Comput. Civ. Eng.* 22 (3): 216–229. https://doi.org/10.1061/(ASCE)0887-3801(2008)22:3(216).

Hosseini, S., and K. Barker. 2016. "Modeling infrastructure resilience using Bayesian networks: A case study of inland waterway ports." *Comput. Ind. Eng.* 93: 252–266. https://doi.org/10.1016/j.cie.2016.01.007.

Jiang, X., R. Bai, J. Atkin, and G. Kendall. 2016. "A scheme for determining vehicle routes based on Arc-based service network design." *Inf. Syst. Oper. Res.* 55 (1): 16–37. https://doi.org/10.1080/03155986.2016.1262580.

Johansen, C., J. Horney, and I. Tien. 2016. "Metrics for evaluating and improving community resilience." *J. Infrastruct. Syst.* 23 (2): 04016032. https://doi.org/10.1061/(ASCE)IS.1943-555X.0000329.

Johansen, C., and I. Tien. 2017. "Probabilistic multi-scale modeling of interdependencies between critical infrastructure systems for resilience." *Sustainable Resilient Infrastruct.* 3 (1): 1–15. https://doi.org/10.1080/23789689.2017.1345253.

Khakzad, N. 2015. "Application of dynamic Bayesian network to risk analysis of domino effects in chemical infrastructures." *Reliab. Eng. Syst. Saf.* 138: 263–272. https://doi.org/10.1016/j.ress.2015.02.007.

Kim, M. C. 2011. "Reliability block diagram with general gates and its application to system reliability analysis." *Ann. Nucl. Energy* 38 (11): 2456–2461. https://doi.org/10.1016/j.anucene.2011.07.013.

Leu, S.-S., and Q.-N. Bui. 2016. "Leak prediction model for water distribution networks created using a Bayesian network learning approach." *Water Resour. Manage.* 30 (8): 2719–2733. https://doi.org/10.1007/s11269-016-1316-8.

Lu, W., Y. Bésanger, E. Zamaï, and D. Radu. 2006. "Blackouts: Description, analysis and classification." In *Proc., 6th WSEAS Int. Conf. on Power Systems*, 429–434. Lisbon, Portugal.

Mishra, R., M. A. Saifi, and S. K. Chaturvedi. 2015. "Enumeration of minimal cutsets for directed networks with comparative reliability study for paths or cuts." *Qual. Reliab. Eng. Int.* 32 (2): 555–565. https://doi.org/10.1002/qre.1772.

Murphy, K. P. 2001. *The Bayes net toolbox for MATLAB*. Berkeley, CA: Univ. of California.

Naidu, A. S. K., C. K. Soh, and K. V. Pagalthivarthi. 2006. "Bayesian network for E/M impedance-based damage identification." *J. Comput. Civ. Eng.* 20 (4): 227–236. https://doi.org/10.1061/(ASCE)0887-3801(2006)20:4(227).

Ouyang, M. 2014. "Review on modeling and simulation of interdependent critical infrastructure systems." *Reliab. Eng. Syst. Saf.* 121: 43–60. https://doi.org/10.1016/j.ress.2013.06.040.

Ouyang, M. 2016. "Critical location identification and vulnerability analysis of interdependent infrastructure systems under spatially localized attacks." *Reliab. Eng. Syst. Saf.* 154: 106–116. https://doi.org/10.1016/j.ress.2016.05.007.

Rinaldi, S. M., J. P. Peerenboom, and T. K. Kelly. 2001. "Identifying, understanding, and analyzing critical infrastructure interdependencies." *IEEE Syst. Control Mag.* 21 (6): 11–25. https://doi.org/10.1109/37.969131.

Santos, J. R., S. T. Pagsuyoin, L. C. Herrera, R. R. Tan, and K. D. Yu. 2014. "Analysis of drought risk management strategies using dynamic inoperability input-output modeling and event tree analysis." *Environ. Syst. Decis.* 34 (4): 492–506. https://doi.org/10.1007/s10669-014-9514-5.

Satumtira, G., and L. Dueñas-Osorio. 2010. "Synthesis of modeling and simulation methods on critical infrastructure interdependencies research." In *Sustainable and resilience critical infrastructure systems*, edited by K. Gopalakrishnan and S. Peeta, 1–51. Berlin: Springer.

Schaberreiter, T., P. Bouvry, J. Röning, and D. Khadraoui. 2013. "A Bayesian network based critical infrastructure risk model." In Vol. 175 of *Proc., EVOLVE: A Bridge between Probability, Set Oriented Numerics, and Evolutionary Computation II*, 207–218. Berlin: Springer.

Shin, Y. Y., and J. S. Koh. 1998. "An algorithm for generating minimal cutsets of undirected graphs." *Korean J. Comput. Appl. Math.* 5 (3): 681–693. https://doi.org/10.1007/BF03008891.

Tien, I. 2017. "Bayesian network methods for modeling and reliability assessment of infrastructure systems." In *Risk and reliability analysis: Theory and applications*, 417–452. New York: Springer.

Tien, I., and A. Der Kiureghian. 2015. "Compression and inference algorithms for Bayesian network modeling of infrastructure systems." In *Proc., 12th Int. Conf. on Applications of Statistics and Probability in Civil Engineering*, edited by T. Haukaas, Vancouver, Canada.

Tien, I., and A. Der Kiureghian. 2016. "Algorithms for Bayesian network modeling and reliability assessment of infrastructure systems." *Reliab. Eng. Syst. Saf.* 156: 134–147. https://doi.org/10.1016/j.ress.2016.07.022.

Tien, I., and A. Der Kiureghian. 2017. "Reliability assessment of critical infrastructure using Bayesian networks." *J. Infrastruct. Syst.* 23 (4): 04017025. https://doi.org/10.1061/(ASCE)IS.1943-555X.0000384.

Tong, Y., and I. Tien. 2017. "Algorithms for Bayesian network modeling, inference, and reliability assessment for multistate flow networks." *J. Comput. Civ. Eng.* 31 (5): 04017051. https://doi.org/10.1061/(ASCE)CP.1943-5487.0000699.

White House. 2013. *Presidential policy directive/PPD 21: Critical infrastructure security and resilience*. Washington, DC: White House.

J. Comput. Civ. Eng., 2019, 33(1): 04018058