Using Centrality Measures in Dependency Risk Graphs for Efficient Risk Mitigation

George Stergiopoulos, Marianthi Theocharidou, Panayiotis Kotzanikolaou, Dimitris Gritzalis

Chapter 18

USING CENTRALITY MEASURES IN DEPENDENCY RISK GRAPHS FOR EFFICIENT RISK MITIGATION

George Stergiopoulos, Marianthi Theocharidou, Panayiotis Kotzanikolaou and Dimitris Gritzalis

Abstract One way to model cascading critical infrastructure failures is through dependency risk graphs. These graphs help assess the expected risk of critical infrastructure dependency chains. This research extends an existing dependency risk analysis methodology towards risk management. The relationship between dependency risk paths and graph centrality measures is explored in order to identify nodes that significantly impact the overall dependency risk. Experimental results using random graphs to simulate common critical infrastructure dependency characteristics are presented. Based on the experimental findings, an algorithm is proposed for efficient risk mitigation. The algorithm can be used to define priorities in selecting nodes for the application of mitigation controls.

Keywords: Dependency risk graphs, graph centrality, cascading failures, mitigation

1. Introduction

Critical infrastructure dependencies contribute to the evolution of cascading effects in the case of infrastructure failures. Previous research [5–7, 12, 13] has articulated a methodology for assessing the cumulative risk of dependency risk paths (i.e., paths of critical infrastructure nodes that are (inter)connected as a result of one or more dependencies). The methodology takes as input the risk assessment results from critical infrastructure operators and, based on the first-order dependencies between the critical infrastructure nodes, assesses the implied risk values of all the n-order dependency risk chains. Next, by sorting the estimated dependency risk chains based on the cumulative dependency risk of each chain, the most important dependency chains are identified.

Although several methods focus on the identification and assessment of the most critical chains of dependencies, they tend to underestimate the importance

of nodes that do not belong to the most critical risk paths (i.e., dependency risk paths with cumulative dependency risk levels above a risk threshold). Moreover, even when nodes belonging to critical risk paths are examined, there are certain nodes whose effects are not measured properly (e.g., nodes that participate in multiple dependency risk paths but have low-risk first-order connections). Decreasing the probability of failure of these nodes may have a greater overall benefit because they affect multiple dependency paths.

This chapter presents an enhanced methodology that uses graph centrality measures to define node priorities when applying risk mitigation controls. Experiments are conducted to determine the significance of each measure in risk mitigation. The experimental results are used to specify an algorithm for achieving an efficient risk mitigation strategy.

2. Graph Centrality Analysis

Graph centrality measures are used to estimate the relative importance or role of a node in a graph. Multiple centrality measures exist, each measuring a different characteristic:

- **Degree Centrality:** This measures the number of edges attached to each node. Given a node u, the degree centrality is defined as:

$$C_d(u) = deg(u)$$

  where $deg(u)$ is the total number of outbound and inbound edges.

- **Closeness Centrality:** This quantifies the intuitive notions of "central" or "peripheral" in a two-dimensional region; it is based on geodesic distances. Closeness centrality is defined as:

$$C_c(u) = \sum_{\forall v \in V(G)} \delta(u, v)$$

  where $\delta(u, v)$ is the average shortest path between the examined node u and any other node in the graph.

- **Betweenness Centrality:** This measures the number of paths in which a node participates. Betweenness centrality is defined as:

$$C_b(u) = \sum_{u \neq i \neq j \in V} \delta_{ij}(u)$$

  where

$$\delta_{ij}(u) = \frac{\sigma_{ij}(u)}{\sigma_{ij}}$$

  Here, $\sigma_{ij}(u)$ denotes the number of geodesic distances from i to j in which node u is present and σ_{ij} is the number of geodesic distances from i to j.

- **Bonacich (Eigenvector) Centrality:** Bonacich centrality [2] attempts to measure the influence of a node in a network. It is defined as:

$$c_i(\alpha, \beta) = \sum_j (\alpha - \beta c_i) R_{i,j}$$

  where α is a scaling factor, β reflects the extent to which centrality is weighted, R is the node adjacency matrix, I is the identity matrix and l is a matrix of ones. Note that an adjacency matrix is an $N \times N$ matrix whose elements have a value of one if an edge exists between nodes; and zero otherwise.

- **Eccentricity Centrality:** This measure is similar to closeness centrality. Essentially, it is the greatest distance from among all the shortest paths between u and any other vertex (in terms of geodesic distances).

3. Related Work

Centrality analysis has primarily been used in graph-based critical infrastructure protection approaches involving vulnerability analyses in power networks. For example, Verma et al. [16] have simulated node removal strategies that trigger cascading failures in the high-voltage European power grid. They compare: (i) random node removal; (ii) node removal based on centrality (betweenness, degree and closeness); and (iii) node removal based on node significance, a context-based measure that considers power flow through a node to its neighbors. They conclude that betweenness, closeness and node degree centrality measures underestimate power grid vulnerability. This is because removing a node with the highest significance causes much more damage than removing a node with the highest centrality or a random node.

A heuristic methodology [1] uses five centrality measures: degree centrality, betweenness centrality, eccentricity centrality, centroid centrality and radiality. The methodology ranks nodes in five lists, one list for each centrality measure. If a node is present in at least two lists, it is considered to be an important node that must be examined. The methodology has been used to evaluate the effects of targeted attacks on the Swiss power grid.

The electrical centrality measure [4] assesses the structure of a network as a function of its electrical topology instead of its physical topology. Unlike the situation with conventional measures of network structure, power networks appear to have a scale-free structure when measured electrically; specifically, power networks have a number of highly-connected "hub" buses that should be examined thoroughly. A similar approach by Wang et al. [18] concludes that, when the electrical parameters are incorporated in centrality definitions, the distributions of degree centrality and eigenvector centrality become very different from those based only on the topological structure. In the case of electrical degree centrality and electrical eigenvector centrality, a large amount of centrality can reside in a small number of nodes; this helps locate groups of important nodes that cannot be identified otherwise. Cadini et al. [3] have

extended the topological concepts of centrality measures to account for the reliability of network connections.

Zio and Piccinelli [19] have highlighted the importance of considering the actual service capacities of nodes as well as other parameters such as the probabilities of node failures and the fact that the flows between network nodes are not restricted to direct, shortest paths as typically assumed. For these reasons, Zio and Piccinelli extend the topological concept of betweenness centrality to account for random flow propagation across a network. Based on network performance characteristics and the random flow betweenness centrality measure, they have identified weaknesses in the network structure of an electrical power transmission system.

Nguyen et al. [9] have studied the optimization problem of detecting critical nodes in interdependent power networks. They introduce novel centrality measures that more accurately assess the importance of each node in interdependent networks; this is achieved by considering intra-centrality (centrality of nodes in each network) and inter-centrality (centrality due to the interconnections between two networks).

In all the approaches discussed above, centrality measures are used topologically and in combination with other parameters to provide a measure of the reliability or failure rate of a node. Many of the approaches demonstrate that a pure topological analysis of power networks is inadequate. On the other hand, the approach presented in this chapter uses centrality measures as an analysis tool for graphs that express the risk dependencies of interconnected critical infrastructures. The edges between the nodes in these graphs do not define physical or topological connections between nodes as in the case of the approaches discussed above. Parameters that indicate the significance of a node are inherently incorporated in the graphs as each risk graph considers the probability of a node failure, the probability of a cascading failure to another node and the impact of the failure. Centrality measures help identify the potentially significant nodes that have larger contributions to the overall graph risk. Thus, the application of mitigation controls to these nodes yields greater overall benefits.

4. Centrality Measures for Dependency Graphs

This research extends the dependency risk methodology of Kotzanikolaou et al. [5, 6] for analyzing multi-order cascading failures. A dependency is defined as "the one-directional reliance of an asset, system, network or collection thereof – within or across sectors – on an input, interaction or other requirement from other sources in order to function properly" [14]. The methodology of Kotzanikolaou et al. [5, 6] quantifies this concept by identifying direct relations (first-order dependencies) between pairs of critical infrastructures as assessed by critical infrastructure operators and extends them to n-order relations. Each dependency from a node CI_i to a node CI_j is assigned an impact value $I_{i,j}$ and likelihood value $L_{i,j}$ of a disruption being realized. The product of the impact and likelihood values yields the dependency risk $R_{i,j}$ to infras-

tructure CI_j due to infrastructure CI_i. Dependencies are visualized in a graph $G = (N, E)$ where N is the set of nodes (or infrastructures or components) and E is the set of edges (or dependencies). The graph is directional and the destination critical infrastructure receives a risk from the source critical infrastructure due to its dependency. The numerical value of each edge is the level of the cascade resulting risk for the receiving infrastructure due to the dependency based on a predefined risk scale $\{1, .., 9\}$.

The methodology of Kotzanikolaou et al. [6] extends the direct risk relations in order to estimate the risk of n-order dependency chains. Let $\mathbb{CI} = (CI_1, \ldots, CI_m)$ be a set of critical infrastructures. An algorithm in [6] examines each critical infrastructure as a potential root of a cascading effect. Let CI_{Y_0} denote a critical infrastructure that is the root of a dependency chain $CI_{Y_0} \rightarrow CI_{Y_1} \rightarrow \ldots \rightarrow CI_{Y_n}$ of length n. The algorithm computes the cumulative dependency risk of the n-order dependency chain as:

$$DR_{Y_0,\ldots,Y_n} = \sum_{i=1}^{n} R_{Y_0,\ldots,Y_i} \equiv \sum_{i=1}^{n} (\prod_{j=1}^{i} L_{Y_{j-1},Y_j}) \cdot I_{Y_{i-1},Y_i} \quad (1)$$

Informally, Equation (1) computes the dependency risk contributed by each affected node in the chain due to a failure realized at the source node. The computation of the risk is based on a risk matrix that combines the likelihood and the incoming impact values of each vertex in the chain. Interested readers are referred to [5] for additional details of the methodology.

4.1 Centrality Measures for Dependency Graphs

This section analyzes the effects of centrality measures in order to construct an algorithm for selecting the most appropriate nodes to apply risk mitigation controls. As mentioned above, the methodology uses risk graphs whose edges denote directed risk relations between nodes, not topological connections or service exchanges between nodes. Several centrality measures in a dependency risk graph formulation are considered in order to identify the nodes that have significant effects on the evolution of the cumulative risk in a dependency chain. Intuitively, nodes with high centrality measures would have high effects on the overall dependency risk. Thus, they are good candidates for implementing risk mitigation controls in a cost-effective mitigation strategy.

Degree Centrality. A node with high degree centrality is a node with many dependencies. Since the edges in a risk graph are directional, the degree centrality is examined for two cases: (i) inbound degree centrality (i.e., number of edges ending at a node); and (ii) outbound degree centrality (i.e., number of edges starting from a node). Nodes with high inbound degree centrality are called cascade resulting nodes while nodes with high outbound degree centrality are called cascade initiating nodes [8].

Nodes with high outbound degree centrality appear to be the most appropriate nodes to examine when prioritizing mitigation controls. Indeed, if proper

mitigation controls are applied to these nodes, then multiple cumulative dependency risk chains are simultaneously reduced. This could result in a cost-effective mitigation strategy that applies controls to high risk edges or high risk paths. Obviously, it is not certain that applying one or more security controls at a node with high outbound degree centrality would positively impact many (or all) outgoing dependencies chains involving the node. However, a mitigation strategy would definitely benefit if it were to initially examine such security controls.

Nodes with high inbound degree centrality in a risk graph are natural "sinkholes" of incoming dependency risk. These nodes are probably subject to multiple independent sources of risk, but reducing the impact of a disruption on these nodes may affect multiple paths. This research has not experimented with such nodes; however, future work will examine mitigation strategies that focus on sinkholes instead of nodes with high outbound degree centrality.

Closeness Centrality. A node with high closeness centrality has short average distances from most nodes in a graph. In the case of a dependency risk graph, nodes with high closeness tend to be part of many dependency chains; sometimes these nodes may even initiate dependency chains. Since cascading effects tend to affect relatively short chains (empirical evidence indicates that cascades rarely propagate deeply [15]), nodes with high closeness centrality would have larger effects on the overall risk of dependency chains than nodes with low closeness centrality. To formalize this idea, consider Equation (1) that computes the cumulative risk of a dependency chain: the closer a node is to the initiator of a cascading event, the greater the effect it has on the cumulative dependency risk. This is because the likelihood of its outgoing dependency affects all the partial risk values of the subsequent dependencies (edges).

A more effective way to exploit closeness centrality in mitigation decisions is to compute the closeness of every node with respect to the subset of the most important initiator nodes. Regardless of the underlying methodology, risk assessors would have *a priori* knowledge or intuition about the most important nodes in cascading failure scenarios. For example, empirical results show that energy nodes and information and communications nodes are the most common cascade initiators [15].

In addition, nodes with high outbound degree centrality are likely to participate in multiple dependency risk chains. Thus, it is possible to first identify the subset of the most important nodes for cascading failures and then compute the closeness of all other nodes relative to this subset of nodes as a secondary criterion for mitigation prioritization.

Eccentricity Centrality. Similar to closeness centrality, eccentricity centrality measures the centrality of a node in a graph that has a small maximum distance from the node to every other reachable node. Note that the small maximum distance corresponds to the greatest distance from among all the shortest-paths between the node and every other node (geodesic distances).

If the eccentricity of a critical infrastructure node is high, then all the other critical infrastructure nodes are proximal to it.

Betweenness Centrality. In a dependency risk graph, a node with high betweenness centrality lies on a high proportion of dependency risk paths. This means that, although such nodes may not be initiating nodes of cascading failures (high outbound centrality) or may not belong to a path with high cumulative dependency risk, they tend to contribute to multiple risk paths and, thus, play an important role in the overall risk calculation. Applying mitigation measures at these nodes (in the form of security controls) is likely to decrease the dependency risk of multiple chains simultaneously.

Upon comparing closeness centrality with betweenness centrality, it appears that closeness should precede betweenness as a mitigation criterion. Although nodes that are between multiple paths will eventually affect multiple chains, it is possible that a node that lies in multiple paths but tends to be at the end of a dependency chain will not (in reality) affect the cumulative dependency risk chain (recall that nodes with high-order dependencies are rarely affected).

Bonacich (Eigenvector) Centrality. A node with high Bonacich [2] (eigenvector) centrality has a high influence on other nodes. In a risk dependency graph, nodes with high eigenvector centrality where $\beta > 0$ are of interest because these nodes are connected to other nodes that also have high connectivity. This is an interesting measure for critical infrastructure risk graphs because such nodes not only can cause cascading failures to more nodes, but they can cause multiple cascading chains of high risk. In contrast, a less connected node shares fewer dependencies with other nodes and is, therefore, affected only by specific nodes in the graph. This means that applying mitigation measures to such a node may not significantly affect the overall risk. However, if mitigation controls are applied to a node with the highest eigenvector centrality (when $\beta > 0$), then the most powerful (or critical) node is modified and this, in turn, affects several other important nodes.

4.2 Centrality Measures for Risk Mitigation

This section examines how the centrality measures described above can be combined to assist in selecting the most appropriate nodes for applying mitigation controls. For example, a critical infrastructure node with high eccentricity and closeness measures might affect a large number of paths with relatively low cumulative dependency risk values. If existing risk assessment methods are applied, potentially serious cascading effects involving these nodes may go unnoticed.

Based on the analysis of the centrality measures on dependency risk graphs discussed in Section 2.4, the following generic method is proposed to assess the selection of candidate nodes for applying risk mitigation controls:

- Use the method of Kotzanikolaou et al. [5] (see Equation (1)) to assess the cumulative dependency risk of all existing dependency paths in a dependency risk graph.
- Compute all the centrality measures for each node.
- Identify alternative mitigation strategies by selecting a subset of nodes for applying risk mitigation controls based on (some) centrality measures.
- Apply the strategy to the selected subset of nodes (i.e., reduce the weights of all the outbound edges for each node in the selected set). Generate a new risk graph (reduced risk graph) by applying mitigation controls to the selected nodes.
- Evaluate the results of the strategy by comparing the new graph with the initial graph. The comparison can be based on the risk of the most critical path, the maximum risk of all paths or the number of paths that have risk values above a specified risk threshold.

The next section uses the generic method to evaluate the effects of various centrality measures on the selection of candidate nodes for risk mitigation. The experimental results are leveraged to develop the most efficient strategy for applying controls to mitigate the overall risk.

5. Experimental Results

An automatic dependency risk graph generator was developed in Java and the Neo4J graph database model was used for graph construction and analysis. Graph databases provide index-free adjacency and more effective models than relational databases, especially in situations where the relationships between elements are the driving force for data model design [11, 17]. Neo4J builds on the property graph model; nodes may have various labels and each label can serve as an informational entity. The nodes are connected via directed, typed relationships. Nodes and relationships hold arbitrary properties (key-value pairs) that make the Neo4J library ideal for building and testing dependency risk graphs and calculating centrality values. After creating a dependency risk graph, the automated dependency risk graph generator computes the cumulative dependency risk of all paths of length five and the centrality values of each node.

The first step was to study possible relationships between the most critical paths of a risk graph (calculated according to the method of Kotzanikolaou et al. [6]) and the subset of nodes with the highest centrality measures. The experiments were designed to understand how often nodes appear simultaneously in the critical paths (i.e., paths with the highest cumulative dependency risk values) and how often nodes in the paths are members of the set of nodes with the highest centrality measures. The graphs constructed in the experiments were randomized with certain restrictions [8, 15] in order to resemble critical infrastructure dependencies based on real data:

Table 1. Participation rates of nodes with high centrality measurements.

Type of Statistical Experiment	**Average**
Nodes in 1% of top paths AND 10% of highest centrality values	16.3%
Nodes in 5% of top paths AND 10% of highest centrality values	16.2%
Nodes in 10% of top paths AND 10% of highest centrality values	16.0%
Paths in 1% of top paths AND at least one node in the top 10% of nodes with the highest centrality values	49.0%

- Occasional tight coupling (i.e., occasional high dependencies between critical infrastructures). Some node relationships in a risk graph have high dependencies (randomization applies random risk values with relatively high lower and upper bounds).
- Interactive complexity (i.e., a measure of the degree to which it is not possible to foresee all the ways in which things can go wrong). Real-world critical infrastructure dependencies have high interactive complexity. To achieve this, the experiments constructed random graphs of 50 nodes with high complexity; the critical paths up to fourth-order dependencies had 230,300 to 2,118,760 possible chains.
- One to seven connections (dependencies) per critical infrastructure node.
- Critical paths of three to four hops.
- 62% of critical infrastructure nodes act as initiators.
- Initiators tend to have higher numbers of interconnections.
- 100 random repetitions.

Experiments were conducted on 5,000 random graphs with the aforementioned restrictions. The results demonstrated that the sum of nodes comprising the top 1% of critical paths also appeared in the top 10% of nodes with the highest centrality measures (average of 16%). However, the number of critical paths with at least one high centrality node was extremely high: an average of 49% of the top 1% of the most critical paths always included a high centrality node based on at least one of the measures. This percentage remained the same even for the top 10% of most critical paths, which leads to the conclusion that the top 10% of paths essentially pass through the same nodes as the top 1% of paths. These results appear to hold for all the centrality measures.

The same experiments were conducted using the top 10% of most critical paths against the top 10% of nodes the with the highest centrality values. The participation percentage appeared to remain stable (16,850 out of a total of 141,093 nodes in the top 10% critical paths). Table 1 presents the participation percentages obtained for each experiment. The results show that, with a

percentage of 49%, the top 1% of highest ranked critical paths are indeed affected by nodes with very high centrality. Analysis of larger sample sets (more than 50% of critical paths) revealed that almost all the high centrality nodes were part of some critical path.

5.1 Risk Mitigation Based on Centrality

The experimental results demonstrate that, even if nodes with high centrality are only a small fraction of the nodes in the most critical risk paths, they affect the top 1% of the most critical risk paths about half the time. Thus, it is essential to take them into consideration when deciding where to implement risk mitigation controls in a high criticality path. In practice, the controls could involve the repair prioritization of nodes (i.e., where to send a repair crew first) or increasing redundancy at a node to reduce the likelihood or consequences of a failure.

In the experiments, the implementation of mitigation controls at a node i was emulated by reducing the likelihood $L_{i,j}$ that a failure of node i would cascade to another node j with a risk dependency on node i. Specifically, the implementation of mitigation controls at a node i was emulated by reducing the $L_{i,j}$ by 20% for all nodes j that depend on node i. The reduction in cascading likelihood was selected because the focus was on cascade initiating nodes. In the case of sinkholes, the reduction in impact would be more appropriate because these are usually cascade resulting nodes. To measure the results of risk mitigation on each selected subset of nodes, the dependency risk values were computed in the same graph before and after the implementation of risk mitigation, and the corresponding risk reduction in each case was computed.

The effects on two metrics were examined: (i) risk reduction achieved in the most critical path; and (ii) risk reduction in the sum of the risks of the top 20 paths with the highest cumulative dependency risks. Mitigation controls were implemented for 6% of the nodes in the entire risk graph (three out of 50 critical infrastructure nodes in the experiments).

Effect on the Most Critical Path. Figure 1 shows that the highest risk reduction in the most critical path was achieved when implementing mitigation controls at the top three nodes (6%) with the highest aggregate values of all the centrality metrics. The average risk reduction achieved was 8.1% (over 100 experiments and 100 most critical paths). The highest risk reduction achieved in all the experiments was 31.5%.

The second highest risk reduction was achieved using a combination of the top three nodes (6%) using the eccentricity and closeness centrality measures (highest risk reduction achieved: 27.2%; average: 9.1%). The next highest risk reduction was achieved using betweenness (highest risk reduction: 26.0%; average: 8.1%) and, lastly, using the eigenvector centrality (highest risk reduction: 17.3%; average: 7.4%).

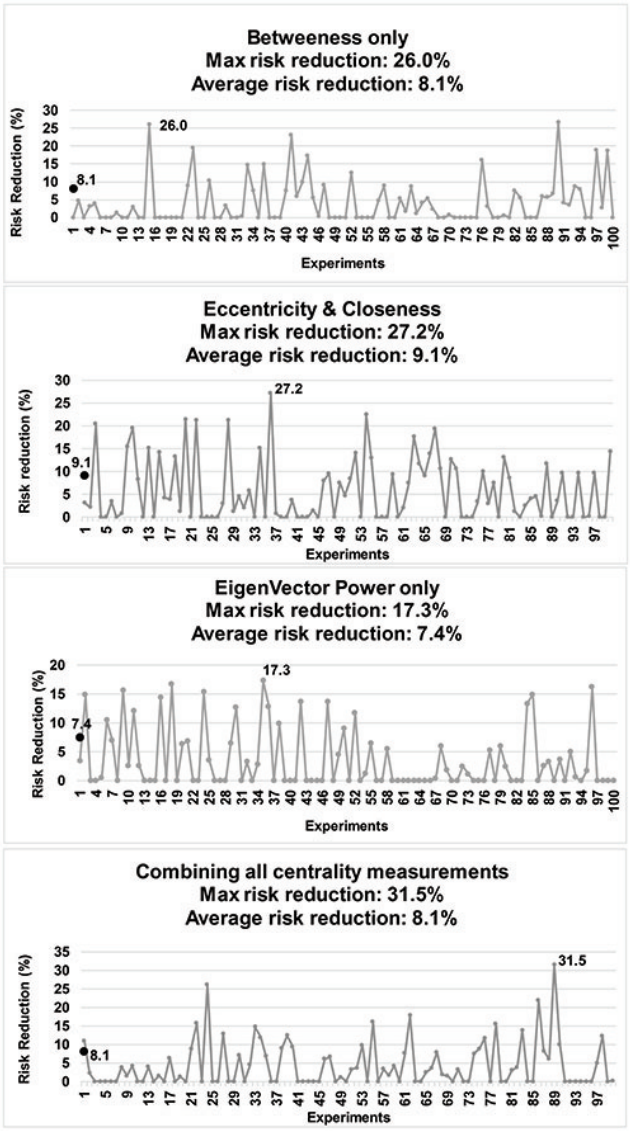

Figure 1. Risk reduction in the most critical path.

Effect on the Top 20 Risk Paths. Figure 2 shows that the highest risk reduction in the sum of risk values derived from the top 20 critical paths is, once again, achieved by implementing mitigation controls at the top three (6%) nodes with the highest centrality for different combinations of centrality metrics. However, the risk reduction achieved has the lowest average reduction of 4.4% despite the fact that the highest maximum reduction is 30.3%.

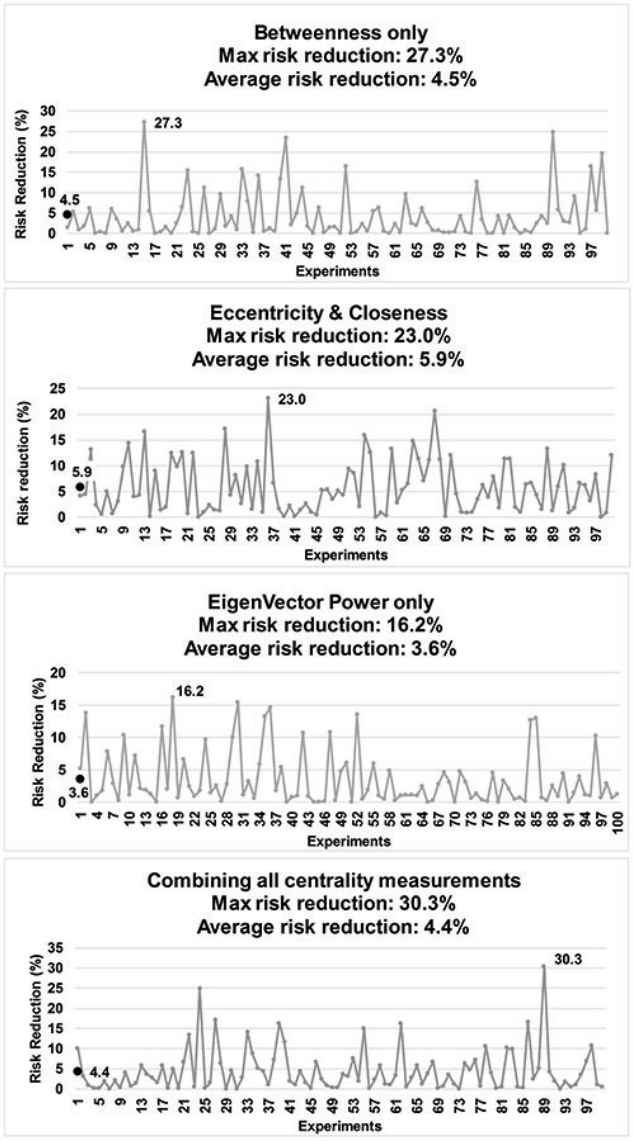

Figure 2. Average risk reduction in the 20 highest critical paths.

Despite aggregating all the centrality measurements, the second highest risk reduction was achieved using a combination of the top 6% of nodes using betweenness centrality only (highest risk reduction achieved: 27.3%; average: 4.5%), followed by the combined use of the eccentricity and closeness centralities (highest reduction: 23.0%; average: 5.9%); and finally using the eigenvector centrality (highest risk reduction: 16.2%; average: 3.6%).

5.2 Efficient Risk Mitigation Algorithm

Algorithm 1 was designed based on the experimental results presented above. In the algorithm, U_1 denotes the subset of the top $X\%$ of nodes with the highest centrality values from among all the centrality sets; U_2 denotes the subset of the top $X\%$ of nodes with the highest eccentricity and closeness centralities; U_3 denotes the subset of the top $X\%$ of nodes with the highest degree and betweenness centralities; and U_4 denotes the subset of the top $X\%$ of nodes with the highest eigenvector centrality. The parameters r_1, r_2, r_3 and r_4 correspond to the average risk reductions for U_1, U_2, U_3 and U_4, respectively, which were measured in the experiments as 8.5%, 9.0%, 4.5% and 3.6%, respectively. S is the subset of nodes belonging to the top 20 critical paths with the highest cumulative dependency risks.

6. Conclusions

The dependency risk methodology described in this chapter extends the approach of Kotzanikolaou et al. [5, 6] by incorporating graph centrality measures as additional criteria for evaluating alternative risk mitigation strategies. The goal was to identify the nodes that greatly affect the critical risk paths and, thus, are more efficient candidates for the application of risk mitigation controls. The experimental results confirm that the most critical paths in dependency risk graphs tend to involve nodes with high centrality measures. However, multiple centrality measures can be applied and these measures contribute to the overall risk mitigation in differing degrees. Experimental evaluations were performed for each centrality measure and for combinations of measures in order to determine the most appropriate combinations of measures. The results demonstrate that aggregating all the centrality measure sets to identify nodes with high overall centrality values is the best mitigation strategy, a result that matches intuition. Nevertheless, aggregation may not always be a viable choice because a dependency graph may have no nodes that exist in all the high centrality sets or there may be contextual reasons that inhibit the application of controls at these nodes.

For this reason, the methodology was extended to rank different combinations of centrality measures based on the experimental results. The results show that, if the method for calculating dependency risk chains is combined with centrality measures, an average risk mitigation of 8.1% is achieved for the most critical path by only implementing mitigation controls at three out of 50 nodes. The experimental analysis was used to design an algorithm for identifying the optimum set of nodes that achieve greater than average risk mitigation for the overall network of nodes instead of a single node. The algorithm, thus, enables "important" nodes to be targeted for mitigation even if the nodes do not belong to the most critical paths in a risk graph.

Future work will focus on enhancing the methodology by incorporating additional parameters, such as the cost of applying controls and other limitations that may arise during mitigation. These could be contextual, such as sector-

Algorithm 1 : Mitigation algorithm.

```
procedure MITIGATION(U1, U2, U3, U4, S)
    Create subset U1
    Create subset S                          ▷ S has nodes from the top 20 paths
    if S ∩ U1 not empty then
        Implement controls at the nodes in S ∩ U1
        if nodes in S ∩ U1 have less nodes than U1 then
            Implement the remaining controls at the nodes in U1
        end if
    else
        Implement controls at the nodes in U1
    end if
    if risk reduction < r1 then
        CONTINUE
    else
        FINISH
    end if
    Create subset U2
    if S ∩ U2 not empty then
        Implement controls at the nodes in S ∩ U2
        if nodes in S ⋂ U2 have less nodes than U2 then
            Implement the remaining controls at the nodes in U2
        end if
    else
        Implement controls at the nodes in U2
    end if
    if risk reduction < r2 then
        CONTINUE
    else
        FINISH
    end if
    Create subset U3;
    if S ∩ U3 not empty then
        Implement controls at the nodes in S ∩ U3
        if nodes in S ∩ U3 have less nodes than U3 then
            Implement the remaining controls at the nodes in U3
        end if
    else
        Implement controls at the nodes in U3
    end if
    if risk reduction < r3 then
        CONTINUE
    else
        FINISH
    end if
    Create subset U4;
    if S ∩ U4 not empty then
        Implement controls at the nodes in S ∩ U4
        if nodes in S ∩ U4 have less nodes than U4 then
            Implement the remaining controls at the nodes in U4
        end if
    else
        Implement controls at the nodes in U4
    end if
    if risk reduction < r4 then
        Implement controls at the nodes with the highest results for all four strategies
    end if
end procedure
```

based characteristics of nodes or constraints imposed by legislation, policy and critical infrastructure operations. Additionally, mitigation strategies for nodes with high inbound degree centrality (sinkholes) will be explored in combination

with impact reduction. Finally, since the risk graphs used in this work were based only on normal operating conditions [10], it is necessary to investigate modified risk graphs that depict other modes of operation (e.g., stressed, crisis and recovery modes).

Acknowledgement

This research was partially supported by the European Union's Seventh Framework Programme for Research, Technological Development and Demonstration under Grant no. 312450.

References

[1] E. Bilis, W. Kroger and C. Nan, Performance of electric power systems under physical malicious attacks, *IEEE Systems Journal*, vol. 7(4), pp. 854–865, 2013.

[2] P. Bonacich, Power and centrality: A family of measures, *American Journal of Sociology*, vol. 92(5), pp. 1170–1182, 1987.

[3] F. Cadini, E. Zio and C. Petrescu, Using centrality measures to rank the importance of the components of a complex network infrastructure, in *Critical Information Infrastructure Security*, R. Setola and S. Geretshuber (Eds.), Springer-Verlag, Berlin Heidelberg, Germany, pp. 155–167, 2009.

[4] P. Hines and S. Blumsack, A centrality measure for electrical networks, *Proceedings of the Forty-First Annual Hawaii International Conference on System Sciences*, 2008.

[5] P. Kotzanikolaou, M. Theoharidou and D. Gritzalis, Assessing n-order dependencies between critical infrastructures, *International Journal of Critical Infrastructures*, vol. 9(1/2), pp. 93–110, 2013.

[6] P. Kotzanikolaou, M. Theoharidou and D. Gritzalis, Cascading effects of common-cause failures in critical infrastructures, in *Critical Infrastructure Protection VII*, J. Butts and S. Shenoi (Eds.), Springer, Heidelberg, Germany, pp. 171–182, 2013.

[7] P. Kotzanikolaou, M. Theoharidou and D. Gritzalis, Interdependencies between critical infrastructures: Analyzing the risk of cascading effects, in *Critical Information Infrastructure Security*, S. Bologna, B. Hammerli, D. Gritzalis and S. Wolthusen (Eds.), Springer-Verlag, Berlin Heidelberg, Germany, pp. 104–115, 2013.

[8] E. Luiijf, A. Nieuwenhuijs, M. Klaver, M. van Eeten and E. Cruz, Empirical findings on critical infrastructure dependencies in Europe, in *Critical Information Infrastructure Security*, R. Setola and S. Geretshuber (Eds.), Springer-Verlag, Berlin Heidelberg, Germany, pp. 302–310, 2009.

[9] D. Nguyen, Y. Shen and M. Thai, Detecting critical nodes in interdependent power networks for vulnerability assessment, *IEEE Transactions on Smart Grid*, vol. 4(1), pp. 151–159, 2013.

[10] A. Nieuwenhuijs, E. Luiijf and M. Klaver, Modeling dependencies in critical infrastructures, in *Critical Infrastructure Protection II*, M. Papa and S. Shenoi (Eds.), Springer, Boston, Massachusetts, pp. 205–213, 2008.

[11] B. Shao, H. Wang and Y. Xiao, Managing and mining large graphs: Systems and implementations, *Proceedings of the ACM SIGMOD International Conference on Management of Data*, pp. 589–592, 2012.

[12] M. Theoharidou, P. Kotzanikolaou and D. Gritzalis, A multi-layer criticality assessment methodology based on interdependencies, *Computers and Security*, vol. 29(6), pp. 643–658, 2010.

[13] M. Theoharidou, P. Kotzanikolaou and D. Gritzalis, Risk assessment methodology for interdependent critical infrastructures, *International Journal of Risk Assessment and Management*, vol. 15(2/3), pp. 128–148, 2011.

[14] U.S. Department of Homeland Security, National Infrastructure Protection Plan (NIPP) 2013: Partnering for Critical Infrastructure Security and Resilience, Washington, DC, 2013.

[15] M. van Eeten, A. Nieuwenhuijs, E. Luiijf, M. Klaver and E. Cruz, The state and the threat of cascading failure across critical infrastructures: The implications of empirical evidence from media incident reports, *Public Administration*, vol. 89(2), pp. 381–400, 2011.

[16] T. Verma, W. Ellens and R. Kooij, Context-independent centrality measures underestimate the vulnerability of power grids, *International Journal of Critical Infrastructures*, vol. 11(1), pp. 62–81, 2015.

[17] C. Vicknair, M. Macias, Z. Zhao, X. Nan, Y. Chen and D. Wilkins, A comparison of a graph database and a relational database: A data provenance perspective, *Proceedings of the Forty-Eight Annual Southeast Regional Conference*, pp. 42:1–42:6, 2010.

[18] Z. Wang, A. Scaglione and R. Thomas, Electrical centrality measures for electric power grid vulnerability analysis, *Proceedings of the Forty-Ninth IEEE Conference on Decision and Control*, pp. 5792–5797, 2010.

[19] E. Zio and R. Piccinelli, Randomized flow model and centrality measure for electrical power transmission network analysis, *Reliability Engineering and System Safety*, vol. 95(4), pp. 379–385, 2010.