Using centrality measures in CI dependency risk graphs for efficient risk mitigation

George Stergiopoulos[1], Marianthi Theocharidou[2], Panayiotis Kotzanikolaou[3], and Dimitris Gritzalis[1]

[1]Dept. of Informatics, Athens University of Economics & Business, 76 Patission Ave., Athens GR-10434, Greece, {geostergiop, dgrit}@aueb.gr

[2]European Commission, Joint Research Center (JRC), Institute for the Protection and the Security of the Citizen (IPSC), Security Technology Assessment Unit, via E. Fermi 2749, Ispra, I-21027, Italy, marianthi.theocharidou@jrc.ec.europa.eu

[3]Dept. of Informatics, University of Piraeus, 85 Karaoli & Dimitriou, GR-18534, Piraeus, Greece, pkotzani@unipi.gr

Abstract

Cascading failures of Critical Infrastructures (CIs) can be modeled through Dependency Risk Graphs, in order to assess the expected risk of CI dependency chains. In this paper we extend our previous dependency risk analysis methodology towards risk management. We explore the relation between dependency risk paths and graph centrality measures, in order to identify nodes that significantly affect the overall dependency risk. We experiment using random graphs that simulate common CI dependency characteristics, as identified by recent empirical studies. Based on our experimental findings, we propose an algorithm that can be used to define priorities in selecting nodes for the application of mitigation controls, in order to achieve efficient risk mitigation.

Keywords: Critical Infrastructure, Dependency Risk Graphs, Graph Centrality, Cascading failure, Mitigation.

1 Introduction

Critical infrastructure dependencies allow the evolution of cascading effects in the case of failures. In our previous work [7, 6, 8, 13, 12] we have proposed a risk based methodology which can be used to assess the cumulative risk of dependency risk paths, *i.e.* paths of CI nodes that are (inter)connected due to one or more dependencies. Our methodology uses as input the risk assessment results from each CI operator and, based on the 1st-order dependencies between the CI nodes, it can be used to assess the implied risk values of all the n-order dependency risk chains. Then, by sorting the estimated dependency risk chains based on the cumulative dependency risk of each chain, the assessors are able to identify which dependency chains are the most important.

Although previous methods focus on the identification and the assessment of the most critical chains of dependencies, they tend to underestimate the importance of nodes not belonging to the most critical risk paths (*i.e.* dependency risk paths with a cumulative dependency risk level above a risk threshold). Moreover, even when examining nodes belonging to critical risk paths, there are cases of nodes whose effect is not properly measured; for example nodes which may participate in multiple dependency risk paths but with low-risk 1st-order connections. Decreasing the probability of failure in such nodes may have a greater overall benefit, since they affect multiple dependency paths.

In this paper, we enhance our methodology by using graph centrality measures, in order to define node priorities when applying risk mitigation

controls. We perform experiments in order to determine the significance of each measure in risk mitigation. Then, we propose an algorithm for achieving an efficient risk mitigation strategy.

2 Graph Centrality Analysis

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

- *Degree centrality* measures the number of edges attached to each node. Given a node u, degree centrality is defined as: $C_d(u) = deg(u)$, where $deg(u)$ is the total number of its outgoing and ingoing edges.

- *Closeness centrality* quantifies the intuitive notion of what one terms "central" or "peripheral" in a two dimensional region and is based on geodesic distances. It 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* measures the number of paths a node participates in. It 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 where node u is present and σ_{ij} the number of geodesic distances from i to j.

- *Bonacich (Eigenvector) centrality* [2] attempts to measure the influence of a node in a network 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. An *adjacency matrix* is a N×N matrix with values of 1 if there is an edge between two nodes and 0 otherwise.

- *Eccentricity centrality* is similar to closeness centrality. Essentially, it is the greatest distance in all shortest-paths between u and any other vertex (geodesic distance).

2.1 Related work on centrality analysis for CIP

Centrality analysis has been mainly used in graph-based CIP approaches for the vulnerability analysis in power networks. For example, in [15] authors simulate node removal strategies which trigger a cascading failure in the high-voltage European power grid. They compare (i) *random* removal, (ii) removal based on *centrality* (*betweenness, degree, closeness*) and (iii) based on *node significance*, a context-based measure that takes into account power flow through a node to its neighbours. They conclude that betweenness, closeness and node degree centrality measures underestimate the vulnerability of power grids, as removing a node with the highest node significance causes remarkably more damage than removing a node with the highest centrality or a random node.

The measure of *electrical centrality* [5] describes the structure of the network as a function of its electrical topology rather than its physical topology. Compared to conventional measures of network structure, power networks appear to have a scale-free network structure, when measured electrically.

Thus, unlike previous studies of power grid structure, power networks have a number of highly-connected "hub" buses, which should be examined more thoroughly. A similar approach [17] concludes that when the electrical parameters are incorporated into the centrality definition, the distribution of the degree centrality and the eigenvector centrality become very different from what are based on the topological structure alone. With the *electrical degree centrality* and the *electrical eigenvector centrality* a large amount of centrality can reside in a small number of nodes in the system and help to locate a group of important nodes that can not be identified otherwise. Researchers in [4] extend the topological concepts of centrality measures to account for the reliability of the network connections.

The work of [18] highlights the importance of considering the actual service capacity of the nodes, but also other parameters, such as the probabilities of their failure and the fact that the flow among network nodes is not restricted to only direct, shortest paths as typically assumed. For this reason, they extend the topological concept of betweenness centrality to account for random flow propagation across the network. Based on network performance characteristics and the *random flow betweenness centrality* measures, they highlight weaknesses of the network structure in an electrical power transmission system.

In the approach of [10], the authors consider interdependent power networks and they study the optimization problem of detecting critical nodes to assess their vulnerability. They introduce novel centrality measures, which assess the importance of each node more accurately on interdependent networks, as they consider both *intra-centrality* (the centrality of nodes in each

network) and *inter-centrality* (the centrality formed by the interconnections between two networks).

In all these approaches, centrality measures are used both topologically and in combination with other parameters to provide a measure of the reliability or failure rate of a node. In our approach we will use centrality metrics as an analysis tool for interconnected CIs. However, in contrast with all other approaches, we do not use graphs defining physical or logical connections between nodes. Our centrality measures will be applied on dependency *risk graphs* between interconnected critical infrastructures, which consider both the probability of a node failure and the impact of this failure.

3 Centrality measures for dependency risk graphs

Our work extends the dependency risk methodology of [7, 6] used for the analysis of multi-order cascading failures. Dependency can be 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" [1]. The methodology of [7, 6] quantifies this concept by identifying the direct relations (1st-order dependencies) between pairs of CIs, as assessed by the individual CI operators, and extends them to n-order relations. Each dependency from a node CI_i to a node CI_j has been assigned[1] an impact value, denoted as $I_{i,j}$ and the

[1]The impact and likelihood assessments are extracted by organization level risk assessments performed by each CI operator. It is reasonable to assume that a CI operator has

likelihood $L_{i,j}$ of a disruption being realized. The product of the last two values is defined as the dependency risk $R_{i,j}$ caused to the infrastructure CI_j due to a dependency to the infrastructure CI_i. Dependencies are visualized through graphs $G = (N, E)$, 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 CI is receiving a risk from the source CI due to its dependency. The numerical value in each edge refers to the level of the cascade resulting risk for the receiver due to the dependency, based on a predefined risk scale $\{1, .., 9\}$.

The methodology of [7] extends these direct risk relations in order to estimate the risk of n-order dependency chains. Let $\mathbb{CI} = (CI_1, ...CI_m)$ be the set of all the examined infrastructures. The algorithm examines each CI as a potential root of a cascading effect. Let CI_{Y_0} denote a critical infrastructure examined as the root of a dependency chain and $CI_{Y_0} \rightarrow CI_{Y_1} \rightarrow ... \rightarrow CI_{Y_n}$ denote a chain of length n. Then the algorithm computes, for each examined chain, the *cumulative Dependency Risk* of the n-order dependency chain as:

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

Informally, equation 1 computes the dependency risk contributed by each affected node in the chain, due to a failure realized in 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 [6] for a detailed analysis.

assessed the expected incoming impact and risk caused by external interconnected entities, such as other CIs.

3.1 Exploring centrality measures for dependency risk graphs

We will explore the effect of centrality measures in order to construct an algorithm for the selection of the most appropriate nodes to apply risk mitigation controls. As mentioned above, our underlying methodology uses *risk graphs* where the edges denote directed risk relations between nodes, and not a topological connection or a service exchange between the CI nodes. We will examine various centrality measures in a dependency risk graph, in order to identify which nodes have a significant effect on the evolution of the cumulative risk in the dependency chains. Intuitively, nodes with high centrality measures are expected to have a high effect in the overall dependency risk. Thus they may be good candidate nodes for the implementation of risk mitigation controls, for a cost-effective mitigation strategy. We will examine various centrality measures in order to analyze the effect of each centrality metric in a dependency risk graph.

3.1.1 Degree centrality

A node with high degree centrality is a node with many dependencies. Since the edges in risk graphs are directional, in this case, the degree centrality could be examined for two different cases:

- Inbound degree centrality, *i.e.* the number of edges ending to a node. A high inbound degree centrality would indicate nodes known as *cascading resulting nodes* [9].

- Outbound degree centrality, *i.e.* the number of edges starting from a node. This is an indication of a *cascading initiating node* [9].

Nodes with high inbound degree centrality in a risk graph are natural "sinkhole" points of incoming dependency risk. Thus such nodes may not be the most appropriate starting points for applying mitigation controls, from a cost/benefit analysis, since they are probably subject to multiple and possible independent sources of risk. On the other hand, nodes with high outbound degree centrality seem to be the most suitable nodes to examine, when prioritizing mitigation controls. Indeed, if proper mitigation controls are applied to such nodes, then multiple cumulative dependency risk chains will simultaneously be reduced. Such a strategy could be a much more cost-effective alternative, from a mitigation strategy aiming at applying controls to particular high risk edges or high risk paths. Obviously, it is not certain that applying one or more security controls to a node with high outbound degree centrality, will positively affect many (or all) outgoing dependencies chains using this node. However, a mitigation strategy would benefit if it were to initially examine such possible security controls.

3.1.2 Closeness centrality

Nodes with high closeness centrality are nodes that have "short" average distances from most nodes in a graph. In the case of a dependency risk graph, nodes with high closeness are nodes that tend to be part of many dependency chains; sometimes might even initiate them. Since cascading effects tend to affect relative short chains (empirical evidence indicates that

cascades rarely propagate deeply [14]), intuitively, nodes with high closeness centrality will have a bigger effect in the overall risk of the dependency chains, than nodes with low closeness centrality. To formalize this, consider equation 1 used to compute the cumulative risk of a dependency chain: the closer a node is to the initiator of a cascading event, the more effect will have on the cumulative dependency risk since the likelihood of its outgoing dependency will affects all the partial risk values of the following 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, the risk assessors will already have an a priori knowledge / intuition of the most important nodes for cascading failure scenarios. For example, empirical results show that energy and ICT nodes are the most common cascade initiators [14].

In addition, nodes with high outgoing degree centrality will probably be nodes that 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 others in relation with this subset of nodes, as a secondary criterion for mitigation prioritization.

3.1.3 Eccentricity centrality

Similar to closeness centrality, this is a measure of the centrality of a node in a graph which is based on having a small maximum distance from a node to every other reachable node. Here, small maximum distances are the greatest distances detected in all shortest-paths between v and any other vertex

(geodesic distance). If the eccentricity of a CI node is high, then all other CI nodes are in proximity.

3.1.4 Betweenness centrality

In a dependency risk graph, a node with high betweenness centrality will lie on a high proportion of dependency risk paths. This means that even though such nodes may not be initiating nodes of a cascading failure (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 to the overall risk. Applying mitigation measures to such nodes (in the form of security controls) will probably decrease the dependency risk of multiple chains simultaneously.

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

3.1.5 Bonacich (Eigenvector) centrality

A node with high Bonacich [3] (eigenvector) centrality is a node that has a high influence to other nodes. In a risk dependency graph, we are interested in nodes with high eigenvector centrality where $\beta > 0$, since these nodes are connected to other nodes which also have high connectivity. This is an interesting measure for CI risk graphs, as such nodes not only can cause

cascading failures to more nodes, but they can cause multiple cascade chains of high risk. On the contrary, a less connected node means that it shares fewer dependencies with other nodes. It is affected only by specific nodes in the graph. This means that applying mitigation measures to such a node may not affect significantly the overall risk. On the contrary, if one applies mitigation controls to the node with high Eigenvector centrality (when $\beta > 0$), this means that the most powerful (or critical) is modified and this, in turn, affects several other important nodes.

3.2 Assessing centrality measures for risk mitigation

We will now examine how these centrality measures can be combined to assist the selection of the most appropriate nodes for applying mitigation controls. For example, a CI node with high Eccentricity and Closeness measures might affect a large amount of paths with relatively small cumulative dependency risk ratings. Using previous methods, potentially serious cascading effects involving such nodes may go unnoticed. Based on the analysis of the centrality measures on dependency risk graphs discussed in Section 3, we define the following steps as a generic method to assess the selection of candidate nodes for applying risk mitigation controls:

1. Use the method described in [6] (see equation 1) to assess the cumulative dependency risk of all existing dependency paths in a given dependency risk graph.

2. Compute all centrality measures for every node.

3. Alternative mitigation strategies: Based on (some) centrality measures, select a subset of nodes for applying risk mitigation controls.

4. Apply the strategy to the selected subset of nodes, *i.e.* reduce the weights of all the outgoing edges for each node in the selected set. A new risk graph is generated; it depicts the reduced risk graph, after the application of mitigation controls to the selected nodes.

5. Evaluate the results of the strategy by comparing the new graph to the initial one. The comparison can be based on the risk of the most critical path, the maximum risk of all paths, or the number of paths who demonstrate risk above a specified risk threshold.

In the following section we will use the above method in order to evaluate the effect of various centrality measures in the selection of candidate nodes for risk mitigation. Using these results, we will also try to develop the most efficient strategy for applying controls for risk mitigation.

4 Experimental Results

We developed an automatic Dependency Risk Graph generator in Java, using the Neo4J graph database model for graph construction and analysis. Graph databases are defined as storage systems that provide index-free adjacency. Graph database technology is an effective tool for modeling data, in comparison with relational databases and querying languages, in cases where the relationship between elements is the driving force for the design of the data model [16, 11]. Neo4J builds upon 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. Both nodes and relationships hold arbitrary properties (key-value pairs), which makes the Neo4J library ideal for building and testing dependency risk graphs and calculating centrality measurements. After computing a Dependency Risk Graph, our generator will compute the cumulative dependency risk of all paths of length $leg5$ and the centrality measures of each node.

First we will study possible relations between the most critical paths of the risk graph (as calculated using the aforementioned method from [7]) and the subset of nodes with the highest centrality measures. This experiment will show how often nodes appear *concurrently* in the critical paths (the ones with the highest cumulative dependency risk value) and, also, how often nodes in paths appear to be members of the set of nodes with the highest centrality measures. Graphs used for this experiment were randomized with specific restrictions, proposed in recent empirical studies [14, 9], in order to resemble CI dependencies based on real data:

- "Occasional tight coupling" (Occasional high dependency between CIs). Some node relationships in the risk graph have high dependencies (randomization applies random Risk values with relatively high lower and upper bounds).

- "Interactive complexity" (a measure of the degree to which we cannot foresee all the ways things can go wrong). Considering graphs made up of 50 nodes and critical paths of 3-5 hops, possible combinations range from 230,300 to 2,118,760.

- 1 to 7 connections (dependencies) per CI node.
- Critical paths of length 3 to 4 hops.
- 62% of CI nodes in graph act as initiators.
- Initiators tend to have higher number of interconnections.
- Number of random repetitions: 1000.

We ran tests on 5000 random graphs with the aforementioned restrictions. Results showed that, the sum of nodes composing the top 1% of all critical paths, also appear in the top 10% of the nodes with the highest centrality measures, with an average of 16%. Yet, the amount of critical paths that had at least one node with high centrality measurements, was extremely high: An average of 49% of the top 1% of the most critical paths were always include a node with high centrality in at least one of the measures. This percentage seemed to remain the same even for the top 10% of most critical paths, which leads to the conclusion that, the top 10% of paths essentially passes through the same nodes as the top 1% of paths. This appears to be true for all centrality measures. Then, we ran the same experiment using the top 10% of the most critical paths against the top 10% of the nodes with the highest centrality measures. The participation percentage seems to remain stable (16,850 out of a total of 141,093 nodes in the top 10% critical paths). Detailed participation percentages for each test are presented in Table 1. With an approximate percentage of 50%, the top 1% of the highest ranked critical paths are indeed affected by nodes with very high centrality. If we

analyze even larger sample sets (more than 50% of critical paths) almost all nodes with high centrality are part of some critical path.

Type of statistical experiment	**Average**
No. of nodes in 1% of top paths AND in 10% of highest Centrality measurements	16.3%
No. of nodes in 5% of top paths AND in 10% of highest Centrality measurements	16.2%
No. of nodes in 10% of top paths AND in 10% of highest Centrality measurements	16.0%
No. of paths in 1% of top paths AND having at least 1 node also in the top 10% of nodes with highest Centrality measurements	49.0%

Table 1: Participation rates of nodes with high centrality measurements

4.1 Risk mitigation based on centrality

We have concluded from the previous experiment that, even if nodes with high centrality are only a small fraction of the nodes in the most critical risk paths, yet, half of the times they do affect the top 1% of the most critical risk paths. Thus it seems essential to take them into consideration when attempting to pinpoint where to implement risk mitigation controls within a path of high criticality. In practice, examples of controls can be the repair prioritization of nodes (which node to send a repair crew first) or increasing redundancy on a node to reduce likelihood or consequences of a failure. For the experiments, we emulate the implementation of mitigation controls to a node i, by reducing the likelihood $L_{i,j}$ that a failure will cascade to any other node j having a risk dependency for i, when node i fails. We emulated the implementation of mitigation controls at a node i, by reducing the Likelihood

$L_{i,j}$ by 20%, for all nodes j that depend on i. To measure the result of risk mitigation to each selected subset of nodes, we calculated the dependency risk values in the same graph, before and after the implementation of risk mitigation and we calculated the risk reduction succeeded in each case.

The effect on two risk metrics was examined: (i) risk reduction achieved in the most critical path and (ii) risk reduction in the total sum of the risk of the top 20 paths of the highest cumulative dependency risk. Mitigation controls were implemented each time on 6% of nodes of the entire Risk graph (three (3) out of fifty (50) CI nodes in these experiments).

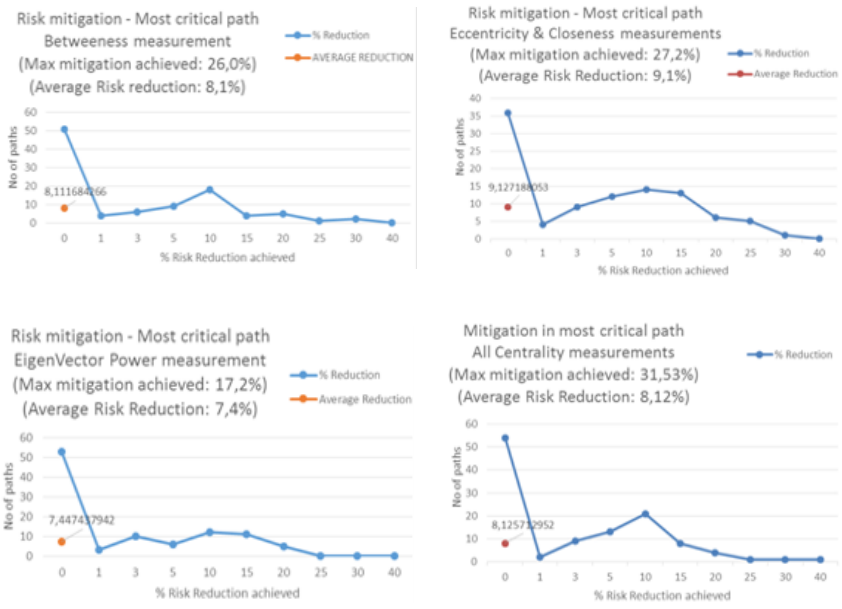

Figure 1: Risk mitigation achieved in Most Critical Path using 6% of nodes with highest measurements in four different Centrality measurements

4.1.1 Effect in the most critical path

In figure 1 we see that the highest risk reduction in the most critical path was achieved when implementing mitigation controls on the top 6% of all centrality measures together (the three nodes with the highest aggregation of all their centrality measures). The achieved risk reduction has an average of 8,1% (average risk reduction per 100 experiments/100 most critical paths). The highest risk reduction achieved in all experiments was 31,5%.

Besides aggregating all centrality measures, the second highest risk reduction was achieved using a combination of the top 6% of nodes using the Eccentricity and Closeness centralities (highest mitigation achieved: 27,2%, average: 9.0%); then using betweeness (highest mitigation achieved: 26%, average: 8,1%) and, lastly, using the EigenVector (highest mitigation achieved: 17,2%, average: 7,4%).

4.1.2 Effect in the top 20 risk paths

In figure 2, we see that the highest risk reduction in the sum of all the risk values derived from the top 20 critical paths, is, again, achieved by implementing mitigation controls on the top 6% of all centrality measurements aggregated. Yet, mitigation achieved has the lowest average of 4,4% risk reduction, albeit the highest maximum reduction, that of 30,3%.

Besides aggregating all centrality measurements, the second highest risk reduction was achieved using a combination of the top 6% of nodes using the Eccentricity and Closeness centralities (highest mitigation achieved: 23,2%, average: 5,8%); following are the mitigation achieved using only Betwee-

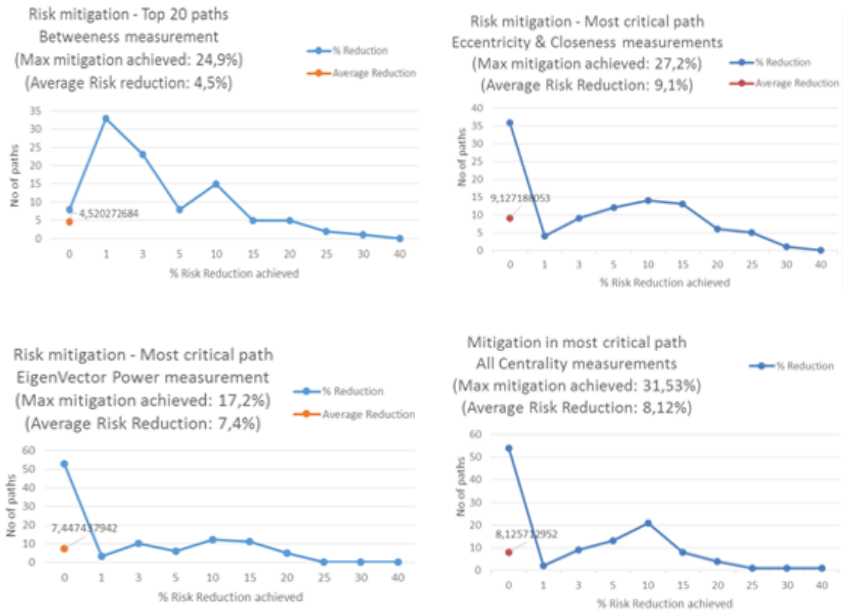

Figure 2: Risk mitigation achieved in Top 20 Critical Paths using 6% of nodes with highest measurements in four different Centrality measurements

ness (highest mitigation achieved: 24,9%, average: 4,5%) and, lastly, using EigenVector (highest mitigation achieved: 16,2%, average: 3,6%).

4.2 An algorithm for an efficient mitigation strategy

The proposed algorithm is based on the previous experimental results. Let U_1 be the subset of the top X% of nodes with the highest centrality measurements from all centrality sets; U_2 be the subset of the top X% of nodes with highest Eccentricity and Closeness centrality; U_3 be the subset of the top X% of the nodes with the highest Degree and Betweeness centrality and U_4 the set of the top X% of nodes with the highest EigenVector centrality. The parameters r_1, r_2, r_3, r_4 depict the average risk reduction for U_1, U_2, U_3, U_4, 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.

Algorithm 1: Mitigation algorithm

Create subset U_1; Create subset S;
if $S \bigcap U_1$ *not empty* **then**
  Implement controls on nodes in $S \bigcap U_1$;
  if *nodes in* $S \bigcap U_1$ *has less nodes than* U_1 **then**
    Implement remaining controls on nodes in U_1
  end
else
  Implement controls on nodes in U_1;
end
if *risk reduction* $< r_1$ **then**
  CONTINUE;
end
Create subset U_2; **if** $S \bigcap U_2$ *not empty* **then**
  Implement controls on nodes in $S \bigcap U_2$;
  if *nodes in* $S \bigcap U_2$ *has less nodes than* U_2 **then**
    Implement remaining controls on nodes in U_2
  end
else
  Implement controls on nodes in U_2;
end
if *risk reduction* $< r_2$ **then**
  CONTINUE;
end
Create subset U_3; **if** $S \bigcap U_3$ *not empty* **then**
  Implement controls on nodes in $S \bigcap U_3$;
  if *nodes in* $S \bigcap U_3$ *has less nodes than* U_3 **then**
    Implement remaining controls on nodes in U_3
  end
else
  Implement controls on nodes in U_3;
end
if *risk reduction* $< r_3$ **then**
  CONTINUE;
end
Create subset U_4; **if** $S \bigcap U_4$ *not empty* **then**
  Implement controls on nodes in $S \bigcap U_4$;
  if *nodes in* $S \bigcap U_4$ *has less nodes than* U_4 **then**
    Implement remaining controls on nodes in U_4
  end
else
  Implement controls on nodes in U_4;
end
if *risk reduction* $< r_4$ **then**
  Implement controls on the highest result from all four algorithms;
end

5 Conclusions

In this paper, we extended our previous method on dependency risk graph analysis with graph centrality measures which we use as additional criteria for the evaluation of alternative risk mitigation strategies. The goal was to identify the nodes that mostly affect the critical risk paths and, thus, are more efficient candidates for the application of risk mitigation controls. We confirmed that the most critical paths in dependency risk graphs tend to involve nodes with high centrality measures. However, there are multiple centrality measures that can be applied and these contribute to overall risk mitigation in various degrees. We performed multiple tests for each centrality measure and for combinations of them, in order to define which combinations are the most efficient. Our results showed that, aggregating all centrality sets to pinpoint nodes with high overall centrality, provides the best mitigation strategy, something to be expected intuitively. Still, this is not always a viable choice, as there might be a dependency graph where no nodes exist in all high centrality sets or there may be contextual reasons that inhibit the application of controls in these nodes.

For this reason, we extended our method to rank different combinations of centrality measures based on experimental results. The results show that, on average, if we combine our previous method for calculating dependency risk chains, together with centrality measures, we can achieve an average risk mitigation of 8.1% average risk reduction in the most critical path by only implementing mitigation controls in 3 out of 50 nodes. Based on our experimental analysis, we proposed an algorithm that seeks the optimum

set of nodes in order to achieve greater than average risk mitigation for the overall network of CIs, as opposed to a single node. The algorithm allows for targeting "important" nodes even if these do not belong to the most critical paths of the risk graph.

Our future work is oriented towards enriching the method with additional parameters, such as the cost of applying the controls or other limitations in mitigation. These could be contextual, such as sector-based characteristics of the node, or constraints posed by legislation, policy and the CI operator.

Acknowledgment

This project has partially received funding from the European Union's Seventh Framework Programme for research, technological development and demonstration under grant agreement no 312450. The European Commission's support is gratefully acknowledged.

References

[1] *NIPP 2013: Partnering for Critical Infrastructure Security and Resilience*. Dept. of Homeland Security, 2013.

[2] Phillip Bonacich. Power and centrality: A family of measures. *American journal of sociology*, pages 1170–1182, 1987.

[3] Phillip Bonacich. Power and centrality: A family of measures. *American journal of sociology*, pages 1170–1182, 1987.

[4] Francesco Cadini, Enrico Zio, and Cristina-Andreea Petrescu. Using centrality measures to rank the importance of the components of a complex network infrastructure. In *Critical information infrastructure security*, pages 155–167. Springer, 2009.

[5] Paul Hines and Seth Blumsack. A centrality measure for electrical networks. In *Hawaii International Conference on System Sciences, Proceedings of the 41st Annual*, pages 185–185. IEEE, 2008.

[6] Panayiotis Kotzanikolaou, Marianthi Theoharidou, and Dimitris Gritzalis. Assessing n-order dependencies between critical infrastructures. *IJCIS*, 9(1/2):93–110, 2013.

[7] Panayiotis Kotzanikolaou, Marianthi Theoharidou, and Dimitris Gritzalis. Cascading effects of common-cause failures in critical infrastructures. In Jonathan Butts and Sujeet Shenoi, editors, *Critical Infrastructure Protection*, volume 417 of *IFIP Advances in Information and Communication Technology*, pages 171–182. Springer, 2013.

[8] Panayiotis Kotzanikolaou, Marianthi Theoharidou, and Dimitris Gritzalis. Interdependencies between critical infrastructures: Analyzing the risk of cascading effects. In *Critical Information Infrastructure Security*, pages 104–115. Springer, 2013.

[9] Eric Luiijf, Albert Nieuwenhuijs, Marieke Klaver, Michel van Eeten, and Edite Cruz. Empirical findings on critical infrastructure dependencies in europe. In *Critical Information Infrastructure Security*, pages 302–310. Springer, 2009.

[10] Dung T Nguyen, Yilin Shen, and My T Thai. Detecting critical nodes in interdependent power networks for vulnerability assessment. *IEEE Trans. Smart Grid*, 4(1):151–159, 2013.

[11] Bin Shao, Haixun Wang, and Yanghua Xiao. Managing and mining large graphs: systems and implementations. In *Proceedings of the 2012 ACM SIGMOD International Conference on Management of Data*, pages 589–592. ACM, 2012.

[12] Marianthi Theoharidou, Panayiotis Kotzanikolaou, and Dimitris Gritzalis. A multi-layer criticality assessment methodology based on interdependencies. *Computers & Security*, 29(6):643–658, 2010.

[13] Marianthi Theoharidou, Panayiotis Kotzanikolaou, and Dimitris Gritzalis. Risk assessment methodology for interdependent critical infrastructures. *International Journal of Risk Assessment and Management*, 15(2):128–148, 2011.

[14] Michel Van Eeten, Albert Nieuwenhuijs, Eric Luiijf, Marieke Klaver, and Edite Cruz. The state and the threat of cascading failure across critical infrastructures: the implications of empirical evidence from media incident reports. *Public Administration*, 89(2):381–400, 2011.

[15] Trivik Verma, Wendy Ellens, and Robert E Kooij. Context-independent centrality measures underestimate the vulnerability of power grids. *arXiv preprint arXiv:1304.5402*, 2013.

[16] Chad Vicknair, Michael Macias, Zhendong Zhao, Xiaofei Nan, Yixin Chen, and Dawn Wilkins. A comparison of a graph database and a

relational database: a data provenance perspective. In *Proceedings of the 48th annual Southeast regional conference*, page 42. ACM, 2010.

[17] Zhifang Wang, Anna Scaglione, and Robert J Thomas. Electrical centrality measures for electric power grid vulnerability analysis. In *Decision and Control (CDC), 2010 49th IEEE Conference on*, pages 5792–5797. IEEE, 2010.

[18] Enrico Zio and Roberta Piccinelli. Randomized flow model and centrality measure for electrical power transmission network analysis. *Reliability Engineering & System Safety*, 95(4):379–385, 2010.