MEASURING FUNCTIONAL ROBUSTNESS WITH NETWORK TOPOLOGICAL ROBUSTNESS METRICS

Haley, Brandon (1); Dong, Andy (2); Tumer, Irem (1)
1: Oregon State University, United States of America; 2: The University of Sydney, Australia

Abstract

This paper describes a study on the use of network topological robustness metrics to measure the functional robustness of complex engineered systems. The goal of the research is to identify network metrics capable of accounting for the functional robustness of a system without the need for a behavioral simulation. Three network topology metrics, average shortest path length, network diameter, and a robustness coefficient, were evaluated given their prominence in characterizing the topological robustness of networks but lack of evidence of their correlation to functional robustness in engineered systems. A bipartite 'behavioral' network of a drivetrain was modeled and simulated for failure using a model-based simulation package, Modelica, and a network attack approach. Average shortest path length and the robustness coefficient showed consistent topological disintegration, revealing the effect of a failure on system performance. Network diameter does not show topology changes when the failure is located outside of the cluster containing the failure. The research demonstrates the plausibility of certain network metrics to characterize the functional robustness of systems.

Keywords: Complexity, Robust design, Systems engineering (SE)

Contact:
Prof. Andy Dong
The University of Sydney
Faculty of Engineering and Information Technologies
Australia
andy.dong@sydney.edu.au

Please cite this paper as:
Surnames, Initials: *Title of paper*. In: Proceedings of the 20th International Conference on Engineering Design (ICED15), Vol. nn: Title of Volume, Milan, Italy, 27.-30.07.2015

1 INTRODUCTION

The study of failures in engineered systems is heavily focused on the reliability of individual components and on performance simulations conducted under variable input conditions so that an understanding of the system-wide effects of input variability could be gathered (Arvidsson and Gremyr, 2008). To avoid failures of important components, setting aside the problem of identifying these important components, the engineering design literature recommends techniques such as failure-modes and effects analysis (FMEA) (Automotive Industry Action Group 2008) or a functional-failure identification and propagation (FFIP) analysis (Stone, Tumer and Wie, 2005; Kurtoglu and Tumer, 2008). They are not without their limitations, though. FMEA requires significant expertise and a knowledge base of operational and historical systems and failures. While FFIP can identify fault propagation paths that cross disciplinary boundaries (Sierla et al., 2012), it cannot readily handle the interaction effects of failures. FFIP is also not effective for the identification of the most vulnerable components without significant historical knowledge of the system (Stone, Tumer and Stock, 2005) or the failure mode of each component (O'Halloran et al., 2014).

Current methods for the failure analysis of complex engineered systems do not readily scale as the number of components and interactions increase because of the super-linear increase in compute time for behavioral analysis relative to the number of components. As the physical architectures of engineered systems progress from complicated to complex, their failure properties become increasingly dependent upon the pattern of connections between components. It is the pattern of connections, rather than the connections themselves, that have a significant effect on the behavior of the system (Newman, 2010). Therefore, understanding failure properties requires an understanding of both the constituent components and the patterns of connections between those components.

In order to understand the behavior of systems, and in particular the relation between structure and fault behavior, researchers have modeled them as networks (Albert and Barabási, 2002), leading some to adopt the term complex engineered systems (Bloebaum and McGowan, 2010). In engineering design, researchers have applied social networks analysis and complex networks theory to analyze the statistical properties of very large scale design and engineering projects, and have shown that their network models have structural properties that are like those of other biological, social, and technological networks (Braha and Bar-Yam, 2007; Braha and Bar-Yam, 2004). Given this finding, the physical architecture of engineered systems have also been modelled and studied as networks (Sarkar et al., 2014; Sharman and Yassine, 2004; Sosa, Eppinger and Rowles, 2007). The objective is to minimize the number of defects (Sosa, Mihm and Browning, 2011) by re-architecting engineered systems and thereby create increasingly robust systems. Even if engineered systems such as aircraft carriers, jet engines, and nuclear power plants do not fully satisfy the definition of a complex system (Minai, Braha and Bar-Yam, 2006), methods from complex network theory have provided productive tools to evaluate the failure properties of complex engineered systems as a consequence of their structural configuration. For example, network radius measures, which had previously been used to determine the vulnerability of software architecture to virus propagation (Jamakovic et al., 2006), have been applied to understand the failure tolerance of engineered systems due to the propagation of a fault, such as the loss of flow or energy (Mehrpouyan et al., 2013).

Despite a growing number of metrics to quantify the failure of networks by characterizing their topological disintegration (Piraveenan et al., 2013), two significant challenges exist in directly applying these metrics to engineered systems. First, one under-investigated assumption is the adequacy of topological disintegration as an indicator of performance degradation in engineered systems. Stated another way, if there were a 25% degradation in system performance, is there an equivalently granular change in (any of) the network metrics of topological disintegration? Is the correlation between performance degradation and topological disintegration linear, quadratic, or something else? It has been hypothesized that the degree of topological disintegration correlates with degree of system failure, but no study has examined this correlation or whether the metrics are suitably sensitive and appropriately granular in detecting performance failures except at the limit of the total loss of an edge or node. The total loss of a component would be exceptional and relatively rare for an engineered system, tantamount to losing, e.g., an engine of an airplane. For the complex network metrics to be useful to engineered system design, the value of the metric(s) should have congruence to the actual degradation in system performance. If they do, then they will become useful in understanding the failure properties of the complex engineered systems without having to perform a full behavioral simulation.

Second, while the behavior of an engineered system is dependent upon inter-connectivity between components, it is not sufficient to describe the behavior of the system in terms of physical topology alone. Generally, the behaviour and performance of engineered systems must be simulated with techniques such as a response surface method or model-based design simulations when many parameters change (Kim and Choi, 2008; Arvidsson and Gremyr, 2008). Analyzing the failure tolerance of engineered systems relying solely on physical and functional dependencies between components may inadvertently produce systems having underlying behavioral coupling between design variables, parameters, and performance, which make the systems vulnerable to failure. It is the interaction between design variables and parameters to performance across the physical architecture that actually determines the system behavior.

In an effort to expand the application of complex network theory to the evaluation of engineered systems, this paper studies the relationship between three commonly used network metrics to quantify topological robustness and simulated system performance under a given degradation. In this paper, we investigate whether these metrics, average shortest path length, network diameter, and robustness coefficient, are appropriate for the characterization of failures. We limit the investigation to the characterization of the sensitivity of the metrics, that is, that they are sensitive to system performance degradation and that they neither over-magnify nor under-represent the extent of a failure.

2 METHODOLOGY

First, the methodology for the analysis of the suitability of network topology disintegration metrics to characterize the degradation in performance of an engineered system is presented. Accordingly, a discussion is presented of how to describe the behavior of an engineered system in a bipartite 'behavioral' network. Then, the calculation of three network metrics are described, average shortest path length, network diameter, and robustness coefficient, to characterize the degree of topological disintegration. When network topology disintegrates due to an attack, that is, the removal of a node or edge from the network, changes in these metrics may describe the severity of the failure through the amount of topological change. The degree of topological change is tested for each of the three metrics.

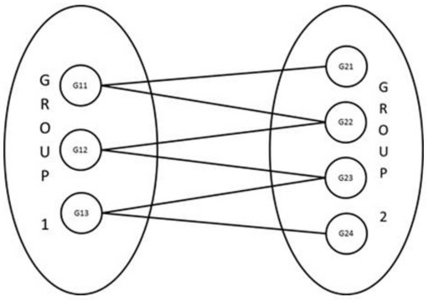
In a case study, the performance characteristics of a drivetrain system model operating nominally and failed are shown. The behavior of the drivetrain is broken down into a set of constituent equations, which are represented as a bipartite network. This network is then subjected to an attack to simulate a fault in a variable or parameter that could degrade the system performance. The same failure in the drivetrain model is then simulated to determine the effect on system performance. Finally, changes to these metrics are compared to simulated system performance using OpenModelica to determine whether changes in these metrics relate to degraded system performance.

For the forthcoming analysis and discussion, we have utilized the following definitions:

1. A Function is the transformation of a flow (Stone, Tumer and Wie, 2005). A Function can be mathematically represented as a relation between design parameters.
2. Behavior is how a Function is achieved. For instance, system Behavior might be the rotation of a shaft to serve the Function of transferring rotational mechanical energy.
3. Performance is the manifestation of Behavior. For instance, a shaft rotating at 100 revolutions per minute (RPM) is its Performance.
4. A Failure is the degradation of Performance. For instance, a shaft rotating at 80 RPM is in a Failure state when it should be rotating at 100 RPM.
5. A Robust system is one that is able to continue operating as intended given the presence of variations to internal or external operating conditions. This includes unexpected component failures and system-level effects.

2.1 Bipartite Network Representation

Network representations involving relationships between two distinct node types are represented with a bipartite network. In this paper, the bipartite 'behavioral' network represents the relations between *Functions* and *Parameters* (design variables and quantifiable noise factors) that influence the *Function* of each component given an engineering choice as to the manner by which the *Functions* will be achieved. The values of the *Parameters* determine the *Performance* of the system.



1	1	0	0
0	1	1	0
0	0	1	1

Figure 1. Bipartite network. Three sub-nodes within type-1 and four sub-nodes within type-2. (Left) Network Graph. (Right) Network Adjacency Matrix.

A bipartite behavioral network is utilized instead of a physical architecture network because the physical architecture model does not contain sufficient information to characterize performance failures in complex engineered systems (Haley, Dong and Tumer, 2014). A physical architecture model lacks a formal (mathematical) description of the functions and the expected (nominal) performance of the system. It does not map the relation between function and design variables and parameters. An example of a bipartite network is displayed in Figure 1. The example has two node types with three sub-nodes within type one (Group 1) and four sub-nodes within type two (Group 2). Each node type can be referred to as a mode. The individual sub-nodes are then connected with edges across each mode. The following rules are applicable when using a bipartite network:

1. No sub-node can be connected to any other sub-node from the same mode.
2. A sub-node from one mode must be connected to at least one sub-node from the other mode.

In determining the bipartite adjacency matrix, a rectangular matrix is used with size $N \times M$.

$$N = \text{Number of Sub Nodes in Type-1} \tag{1}$$

$$M = \text{Number of Sub Nodes in Type-2} \tag{2}$$

Bipartite networks may be represented with rectangular adjacency matrices. The matrices can be weighted or un-weighted. An adjacency matrix for the example bipartite network displayed in Figure 1 (Left) has been provided in Figure 1 (Right). The adjacency matrix, **A**, is defined as follows:

$$\mathrm{A_{ij}} = \begin{cases} 1 \text{ edge connection between sub node i and j} \\ \quad 0 \text{ otherwise} \end{cases} \tag{3}$$

For the weighted case, **A** is defined as follows:

$$\mathrm{A_{ij}} = \begin{cases} \mathrm{w_{ij}} \text{ edge connection between sub node i and j} \\ \quad 0 \text{ otherwise} \end{cases} \tag{4}$$

Often it is necessary to represent a bipartite rectangular matrix as a block, $(N + M) \times (M + N)$, adjacency matrix. This way, all of the information required for the analysis is preserved, yet the math required to be performed is simplified for use on a 'unipartite' like network representation (Grassi, Stefani and Torriero, 2011). The following adjacency matrix, **B**, is represented as:

$$\mathbf{B} = \begin{bmatrix} 0 & \mathrm{A} \\ \mathrm{A^T} & 0 \end{bmatrix} \tag{5}$$

where **A** is the adjacency matrix from Eq. 3 and A^T is the transpose of Eq. 3. **B** retains the correlation between both modes with **A** representing an existing correlation and a 0 showing a lack thereof.

2.2 Metrics of Topological Disintegration

Existing metrics for the topological robustness of networks generally assess the connectivity of the network after a node or edge has been removed. There are number of such metrics (see (Piraveenan et al., 2013) for a review). For this research, we utilize three metrics that are considered fundamental to the analysis of the connectivity of networks (Albert, Jeong and Barabási, 2000; Piraveenan et al., 2013).

2.2.1 Average Shortest Path Length

The average shortest path length, sometimes called average path length, is a measure of the topological connectedness of a network (Wilson, 1996). The average shortest path length is generally used to describe the relative efficiency for a flow to travel throughout a network. In this context, the average shortest path length is a measure of topological disintegration. The average shortest path length will always decrease when a node or edge is removed because nodes become isolated. Formally, when there is no path between nodes, the shortest path length is defined as infinity, but the calculation of the average shortest path length ignores disconnected nodes. Thus, the average shortest path length approaches 0.

Consider the block network representation presented in the previous section (Eq. 5). Assuming $\mathbf{B}$ is a graph with a set of nodes, K, the shortest distance d between any two nodes is $d(k_i, k_j)$, where $k_i, k_j \in K$. The average shortest path length L_{ASP} is defined as:

$$L_{ASP} = \frac{1}{(N+M)(N+M-1)} \sum_{i \neq j} d(k_i, k_j) \quad (6)$$

where N is the number of nodes in mode 1 and M is the number of nodes in mode 2. Eq. 6 has been normalized with the size of the network in order to provide scale and comparability between networks of different sizes (Bounova and de Weck, 2012).

2.2.2 Network Diameter

Network diameter is defined as the longest shortest path between any two nodes within the network and is similar to average shortest path length in its calculation (Wilson, 1996). The network diameter is the classical measure of network tolerance to failure because the diameter characterizes the ability of two nodes to communicate with each other (Albert, Jeong and Barabási, 2000). While useful as a metric when a network is subject to repeated attacks on separate nodes, it may not carry much significance on single node or edge attacks, which is more plausible for engineered systems. In engineered systems, if multiple components fail within a relatively short time span, this would be considered a rare and catastrophic event. Geodesic distances may not change unless network wide effects are observable. We calculate this metric to compare its sensitivity relative to the average shortest path length metric.

2.2.3 Robustness Coefficient

The robustness coefficient is a measure of the topological robustness of a network to node or edge removal (attack) (Piraveenan et al., 2013). The calculation of the metric is based upon measuring the size of the largest connected component after an attack. A component is a set of nodes that are topologically connected; the size of the component is simply the number of nodes in the component. The largest component is also referred to as a giant cluster. When a node is removed, this may alter the size of the largest remaining component. When a network is correctly connected (all the nodes are connected to one another), the coefficient is defined as 100% because the size of the giant cluster is equal to the number of nodes. As the network disintegrates, the robustness coefficient reduces to 0%. Therefore, when evaluated over the total number of nodes, the metric approximates system connectedness when subject to attacks. Because the metric relies only on a count of nodes, it is computationally efficient, a useful feature as network diameter and radius-based metrics are computationally expensive (Aingworth et al., 1999).

The robustness coefficient is valid for use on unipartite representations of networks. As this paper analyzes a complex engineered system with a bipartite network, an alteration must be performed for the metric to remain valid. The specific derivation of the metric is identical to the derivation for unipartite networks; however, the premise behind the metric must change slightly. The robustness coefficient is a ratio of the areas beneath two network disintegration profiles.

$$R = \frac{A_1}{A_2} \quad (7)$$

where A_1 is the area beneath the profile of the target network and A_2 is the area beneath the profile of an idealized network. Rather than evaluating the network for the size of the largest connected component, as has been done previously, the presented metric is evaluated with the average size of a component belonging to a specific mode. Average component size is commonly utilized in network analysis for evaluating the general size of a network. This allows the idealized component size to decrease by unity with the removal of each node belonging to the other mode. The derivation continues similarly to the literature. Eq. 8 through Eq. 10 describes the calculation of the area beneath each profile.

$$A_1 = \frac{1}{2}(S_0 + S_1) + \frac{1}{2}(S_1 + S_2) + \cdots + \frac{1}{2}(S_{N-1} + S_N) \tag{8}$$

$$A_1 = \sum_{k=0}^{N} S_k - \frac{1}{2} S_0 \tag{9}$$

$$A_2 = \frac{1}{2} N^2 \tag{10}$$

Because a bipartite network is represented by an $N \times M$ adjacency matrix, the robustness coefficient is calculated with respect to the average size of one mode while the other is subjected to sustained failures. Either mode could be chosen for application. For the purpose of this paper, the performed analysis follows the equations as they are presented here.

The robustness coefficient, normalized to a 0-100 scale, is defined as the following:

$$R = \frac{200\sum_{k=0}^{N} S_k - 100 S_0}{N^2} \tag{11}$$

where S_k is the average size of the component in one mode after k nodes from the other mode have been removed. S_o is the initial average size. N is the number of nodes in an $N \times M$ network. It can be verified that the equation has been normalized to result in 100% if the network is fully connected and 0% if it is completely disconnected.

3 CASE STUDY

This section presents a case study of the failure of a drivetrain modelled with OpenModelica, a Modelica software platform. The model is used to show how failures related to the parameters that define system behavior and performance can impact network topology. Following the analysis will be a discussion of the relevant results and an evaluation of whether or not failures to engineered systems can be adequately modelled with a network at this level of topological granularity.

For both the system and network evaluations, a torque degradation failure was implemented in the clutch. The failure is associated with a degradation of the coefficient of friction, *mu*, between rotating clutch disks. This results in reduced torque output to the rest of the drivetrain.

3.1 The Drivetrain Model Simulation

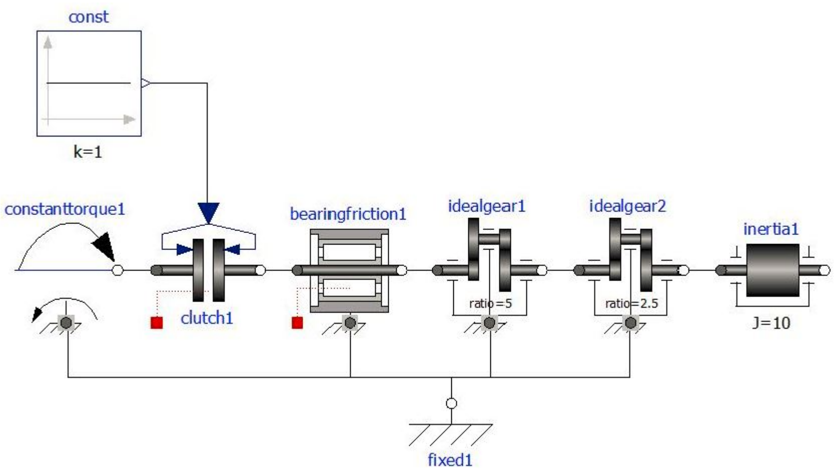


Figure 2. Drivetrain model constructed in OpenModelica (Tiller, 2001).

The model depicted in Figure 2 depicts a drivetrain that accepts a constant torque and clutch state input (clutch engaged or disengaged). This torque is passed through a clutch, bearing, and two gearing manipulations under an inertial load. The equations governing model performance have been presented in Eq. 12-18. Table 1 provides the definitions for the parameters used in each equation.

$$\mathrm{F1: Clutch_{fric} = mu * cgeo * Fn} \tag{12}$$

$$\mathrm{F2: cgeo = N * \frac{1}{2} * (r_o + r_i)} \tag{13}$$

$$\mathrm{F3: Clutch_{out} = Torque_{In} - Clutch_{fric}} \tag{14}$$

$$\mathrm{F4: 0 = Clutch_{out} + bearing_{out} - tau - fric_{viscous}} \tag{15}$$

$$\mathrm{F5: fric_{viscous} = 10^{-7} * f_o * (nu * RPM)^{\frac{2}{3}} * dm^3} \tag{16}$$

$$\mathrm{F6: 0 = ratio1 * bearing_{out} + gear1_{out}} \tag{17}$$

$$\mathrm{F7: 0 = ratio2 * gear1_{out} + gear2_{out}} \tag{18}$$

The equations represent the drivetrain's fallible behavior. That is, changes in the values of the parameters may lead to incorrect performance of the drivetrain. This is true for each equation aside from Eq. 13. In this equation, a parameter defining the geometric properties of a clutch, *cgeo*, has been included. While not strictly behavioral information, this equation defines clutch properties that make a clutch subject to internal contact mechanism failures, such as clutch slipping. This is an important failure related to clutch performance not directly captured with typical network modelling techniques (O'Halloran et al., 2013).

Table 1. Variable Value Definitions (Tiller, 2001; Harris and Kotzalas, 2007)

Number	Variable	Related Component	Definition
1	$Clutch_{fric}$	Clutch	Amount of Torque from Friction
2	mu	Clutch	Static Coefficient of Friction
3	cgeo	Clutch	Geometric Constant for Clutch Surface
4	Fn	Clutch	Normal Force Between Clutch Plates
5	N	Clutch	Number of Frictional Surfaces
6	r_o	Clutch	Outer Surface Diameter
7	r_i	Clutch	Inner Surface Diameter
8	$Clutch_{out}$	Clutch	Torque Output from Clutch
9	$Torque_{In}$	Clutch	Torque Input to Clutch
10	$Bearing_{out}$	Bearing	Torque Output from Bearing
11	tau	Bearing	Bearing Torque Loss
12	$fric_{viscous}$	Bearing	Bearing Torque from Viscous Friction
13	f_o	Bearing	Bearing Type Factor
14	nu	Bearing	Kinematic Viscosity of Lubricant
15	RPM	Bearing	Shaft Speed
16	dm	Bearing	Bearing Diameter
17	ratio1	Gear 1	Gear Ratio for Gear 1
18	$gear1_{out}$	Gear 1	Torque Output from Gear 1
19	ratio2	Gear 2	Gear Ratio for Gear 2
20	$gear2_{out}$	Gear 2	Torque Output from Gear 2

Research has been performed on how to model fault behaviors within engineered systems. Much of the work centers on the implementation of fault variables that degrade system performance (Joshi and Heimdahl, 2007; O'Halloran et al., 2013). For this analysis, the edges in the bipartite behavioral network were weighted with a value between 0 and 1: 1 if no failure exists within the parameter, between 0 and 1 if a degradation failure is present, and 0 if there is a complete failure. This scale ensured the applicability of each network metric. Thus, the edge weights could be thought of as a fault variable (FV). Eq. 19 displays the implementation of a fault variable in *F1* for use in a traditional system model. The implementation process would be identical for each additional function. In a network, each fault variable, denoted *FV1* through *FVM* (*M* is the number of mode 2 nodes), represents the failed state of the parameter for which it is associated, and is applied to an edge (Figure 5). This allows for the alteration of topology according to the current state of each relationship.

$$\mathrm{F1_f: Clutch_{fric}(FV_1) = mu(FV_2) * cgeo(FV_3) * Fn(FV_4)} \tag{19}$$

To simulate a failure in the torque transferability of a clutch, a fault variable, *FV2*, was applied. The model works by setting a clutch state input, disengaged, partially engaged, or fully engaged. For this simulation, the clutch was set to partially engage at 75% of an arbitrary set maximum value.

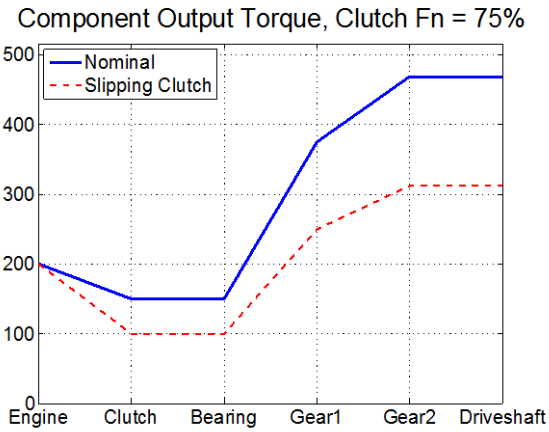


Figure 3. Torque Simulation Results of a Drivetrain. Nominal (Blue). Slipping Clutch (Red).

Figure 3 shows the output torque at each component under the condition of a slipping clutch. The failure is implemented within the torque transfer capability of the clutch. A slipping clutch occurs when the input torque from the engine is greater than the frictional torque resisting relative motion between clutch disks (O'Halloran et al., 2013). In this case, the clutch outputs less torque than the supplied engine torque but equal to the frictional resistance torque of the rotating clutch disks. In the event of a slipping clutch condition, there would be no relative motion between rotating disks, only a reduced torque output according to the severity of the failure. For this analysis, a fault variable value of 0.5 was implemented into Eq. 12 for *FV2*. The fault will be implemented into a bipartite network at the same magnitude.

3.2 Modelling the Drivetrain and Failure with a Bipartite Network

In the case of this drivetrain, the relationships defining system performance have been reproduced in Eq. 12-18 and were extracted from the system modelled in OpenModelica. Each equation is a function belonging in mode 1, *F1* through *F7*. Each function's constituent parameters belong in mode 2. Therefore, to construct the bipartite network for this drivetrain, there are 7 sub-nodes within mode 1 (functions, *N*) and 20 sub-nodes within mode 2 (parameters, *M*). A connection between modes will occur if a mode 2 system parameter is present within a mode 1 function. To model a failure in the adjacency matrix, the edge weight in the adjacency matrix is changed. For example, if the static coefficient of friction drops due to wear, each existing edge associated with the failed node will have its edge weight modified to reflect the failure. For each network metric described in Section 2.2, two simulations were conducted, one for a nominal drivetrain and one for a slipping clutch.

Figure 4 shows the results from the failure analysis. The value for each fault variable was set to FVi = 0.5 and was failed independently of all other parameter nodes (i.e., only one node is in a failure state at a time). When the drivetrain clutch begins to slip due to a 50% loss in the clutch frictional coefficient, the average shortest path length decreases between approximately 0.001% and 9%, the network diameter between 0% and 8%, and the robustness coefficient by 2% to 5%. The results show that the network diameter is not always sensitive to a fault, and, therefore, is not suitable when the number of faults is low. The average shortest path length produced almost no discernible change for 16 nodes whereas the robustness coefficient produced discernible change for all nodes. What is interesting to note is that there are essentially three failure groups, that is, nodes for which failure produced the same decrease in the robustness coefficient. One group consists of variables 1, 3, 10, and 18, another group consists of nodes 2, 4-7, 9, 11, 17, and 19-20, and one group consists of variables 13-16. Identifying the designation of these variables in Table 1, these groups suggest that failure to the bearing has the most critical impact on system performance whereas frictional losses and geometric faults have a lower effect. In other simulations of power losses in gearboxes, it has been identified that the main amount of losses is caused by the bearings (Schlegel, Hösl and Diel, 2009). Thus, this relatively simple analysis of failure using a network topology approach accords with known performance issues in vehicle drivetrains. For comparison, note that the physical architecture network of the drivetrain would show that each component is equally important in a failure scenario as they are connected in series; each node is connected to one other node. A loss of any node would result in a disintegration of the network. These results highlight the importance of the design variables and parameters associated with the bearings, and that it is these parameters rather than the component that influence failure.

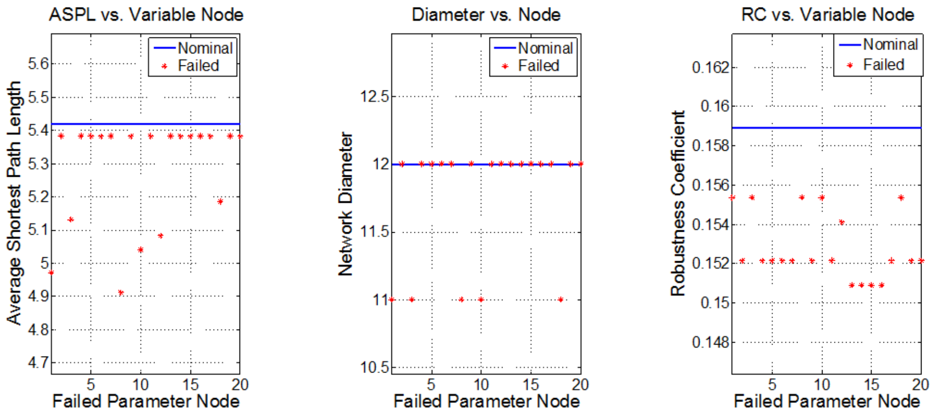


Figure 4. Topology metrics for drivetrain behavioral network failure simulation. ASPL=average shortest path length; RC=robustness coefficient

4 CONCLUSION

This paper presented a study of the relation between three metrics used for network topology evaluation and engineered system failure, which is characterized by performance degradation. Average shortest path length, network diameter, and robustness coefficient were evaluated on the behavioral network of a drivetrain model to determine whether the metrics change when system performance degrades. The values of the metrics under nominal and failed states of the drivetrain were compared to a simulation of system torque response. Two of the metrics, average shortest path length and the robustness coefficient, showed topological disintegration patterns that changed between nominal and failed cases. The network diameter is not sufficiently dependent on local topology to consistently show topology changes. Therefore, it lacks a clear relation to system failure. Between average shortest path length and the robustness coefficient, while a failure is shown in the evaluation of each metric, they are currently most useful in identifying failure origins rather than the impact of failure. With this information, topology can be optimized so that failures related to one parameter do not cripple a system.

While the drivetrain model used in this paper is useful for explaining the approach and to show the correlation between network topology disintegration and system failure, in future work, a large scale system will be analyzed with a wide array of failure modes to determine if the approach is scalable. Additionally, research is required to map simulation results of complex engineered system models and network topology metrics to determine the relevant correlations and manifestations of system performance within network analysis tools.

REFERENCES

Aingworth, D., Chekuri, C., Indyk, P. and Motwani, R. (1999) Fast Estimation of Diameter and Shortest Paths (Without Matrix Multiplication), *SIAM Journal on Computing*, Vol. 28, No. 4, pp. 1167-1181.

Albert, R. and Barabási, A.-L. (2002) Statistical mechanics of complex networks, *Reviews of Modern Physics*, Vol. 74, No. 1, pp. 47-97.

Albert, R., Jeong, H. and Barabási, A.-L. (2000) Error and attack tolerance of complex networks, *Nature*, Vol. 406, No. 6794, pp. 378-382.

Arvidsson, M. and Gremyr, I. (2008) Principles of Robust Design Methodology, *Quality and Reliability Engineering International*, Vol. 24, No. 1, pp. 23-35.

Automotive Industry Action Group (ed.) 2008, 'Potential Failure Mode and Effects Analysis', Automotive Industry Action Group, Southfield.

Bloebaum, C. and McGowan, A. (2010) Design of Complex Engineered Systems, *Journal of Mechanical Design*, Vol. 132, No. 12, pp. 120301-120301.

Bounova, G. and de Weck, O. (2012) Overview of metrics and their correlation patterns for multiple-metric topology analysis on heterogenous graph ensembles, *Physical review E*, Vol. 85, p. 016117.
Braha, D. and Bar-Yam, Y. (2004) Topology of large-scale engineered problem-solving networks, *Physical Review E*, Vol. 69, No. 1, p. 016113.
Braha, D. and Bar-Yam, Y. (2007) The statistical mechanics of complex product devlopment: empirical and analytical results, *Management Science*, Vol. 53, No. 7, pp. 1127-1145.
Grassi, R., Stefani, S. and Torriero, A. (2011) Using Bipartite Graphs to assess power in organizational networks: a case study, *Dynamics of Socioeconomic Systems*, pp. 199-216.
Haley, B., Dong, A. and Tumer, I.Y. (2014) Creating Faultable Network Models of Complex Engineered Systems, ASME 2014 International Design and Engineering Technical Conferences & Computers and Information in Engineering Conference (IDETC/CIE 2014), New York, DETC2014-34407.
Harris, T. and Kotzalas, M. (2007) Kinematic Speeds, Friction Torque, and Power Loss, in *Essential Concepts of Bearing Technology*, Taylor and Francis Group.
Jamakovic, A., Kooij, R., Van Mieghem, P. and Van Dam, E. (2006) Robustness of networks against viruses: the role of the spectral radius, Communications and Vehicular Technology, 2006 Symposium on, New York, 35-38.
Joshi, A. and Heimdahl, M. (2007) Behavioral Fault Modeling for Model-bsed Safety Analysis, *High Assurance Systems Engineering Symposium*, Vol. 199, No. 208, pp. 14-16.
Kim, C. and Choi, K. (2008) Reliability based design optimization using response surface method with prediction interval estimation, *Journal of Mechanical Design*, Vol. 130, No. 12, p. 121401.
Kurtoglu, T. and Tumer, I. (2008) A graph-based fault identification and propagation framework for functional design of complex systemsd, *Journal of Mechanical Design*, Vol. 130, No. 5, pp. 051401-8.
Mehrpouyan, H., Haley, B., Dong, A., Tumer, I.Y. and Hoyle, C. (2013) Resilient design of complex engineered systems, ASME 2013 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference (IDETC/CIE2013), Portland, V03AT03A048.
Minai, A.A., Braha, D. and Bar-Yam, Y. (2006) Complex engineered systems: A new paradigm, in *Complex Engineered Systems: Science Meets Technology*, Berlin: Springer.
Newman, M. (2010) *Networks: An Introduction*, New York: Oxford University Press.
O'Halloran, B., Jensen, D., Tumer, I., Kurtoglu, T. and Stone, R. (2013) A framework to generate fault-based behavior models for complex systems design, Reliability and Maintainability Symposium (RAMS), 1-6.
O'Halloran, B.M., Haley, B., Jensen, D.C., Arlitt, R., Tumer, I.Y. and Stone, R.B. (2014) The early implementation of failure modes into existing component model libraries, *Research in Engineering Design*, Vol. 25, No. 3, May, pp. 203-221.
Piraveenan, M., Thedchanamoorthy, G., Uddin, S. and Chung, K. (2013) Quantifying topological robustness of networks under sustained attack, *Social Network Analysis and Mining*, pp. 1-14.
Sarkar, S., Dong, A., Henderson, J. and Robinson, P. (2014) Spectral characterization of hierarchical modularity in product architectures, *Journal of Mechanical Design*, Vol. 136, No. 1, p. 13.
Schlegel, C., Hösl, A. and Diel, S. (2009) Detailed Loss Modelling of Vehicle Gearboxes, Proceedings 7th Modelica Conference, Linköping, 434-443.
Sharman, D.M. and Yassine, A.A. (2004) Characterizing complex product architectures, *Systems Engineering*, Vol. 7, No. 1, pp. 35-60.
Sierla, S., Tumer, I.Y., Papakonstantinou, N., Koskinen, K. and Jensen, D.C. (2012) Early integration of safety to the mechatronic system design process by the functional failure identification and propagation framework, *Mechatronics*, Vol. 22, No. 2, March, pp. 137–151.
Sosa, M., Eppinger, S.D. and Rowles, C.M. (2007) A Network Approach to Define Modularity of Components in Complex Products, *Journal of Mechanical Design*, Vol. 129, No. 11, pp. 1118-1129.
Sosa, M., Mihm, J. and Browning, T. (2011) Degree distribution and quality in complex engineered systems, *Journal of Mechanical Design*, Vol. 133, No. 10, p. 101008.
Stone, R.B., Tumer, I.Y. and Stock, M.E. (2005) Linking product functionality to historic failures to improve failure analysis in design, *Research in Engineering Design*, Vol. 16, No. 1-2, November, pp. 96-108.
Stone, R., Tumer, I. and Wie, M. (2005) The function-failure design method, *Journal of Mechanical Design*, Vol. 127, No. 3, pp. 397-407.
Tiller, M. (2001) *Introduction to Physical Modeling with Modelica*, Springer-Science+Business Media, LLC.
Wilson, R.J. (1996) *Introduction to graph theory*, 4th edition, Harlow: Pearson.