Design for Fault Analysis Using Multi-partite, Multi-attribute Betweenness Centrality Measures

Tsai-Ching Lu[1], Yilu Zhang[2], David L. Allen[1], and Mutasim A. Salman[2]

[1]*HRL Laboratories LLC, Malibu, California, 91360, U.S.A*
tlu@hrl.com
dlallen@hrl.com

[2]*General Motors, Warren, Michigan, 48090, U.S.A*
yilu.zhang@gm.com
mutasim.a.salman@gm.com

ABSTRACT

As electrical and electronic systems (EES) steadfastly increase their functional complexity and connectedness, they pose ever-growing challenges in fault analysis and prevention. Many EES faults are intermittent, emerging (new faults), or cascading, and cannot be addressed by the traditional component-level diagnostic design. Leveraging the latest advancements in Network Science, we take the holistic approach to model and analyze the highly interrelated in-vehicle EES as layered sub-networks of hardware components, software components, and communication links. We develop multi-partite, multi-attribute betweenness centrality measures to quantify the complexity and maintainability of the layered EES network. We then use the betweenness centrality distribution to identify fault analysis monitoring points and fault-mitigation strategies. The promising results obtained by our initial empirical study of an example in-vehicle EES presents a first step toward network-theory based IVHM.

1. INTRODUCTION

The complexity of the electrical and electronic system (EES) in vehicles has evolved over the years in response to continuously increasing demand for incorporating new electronic control units (ECUs) onto vehicles. These allow for advanced safety, convenient and comfort features, as well as meeting new emission and fuel-economy standards. However, the fast growing number of ECUs and their peripherals has led to complex interactions which can lead to unexpected emerging or cascading failures.

Current state-of-the-art diagnosis and prognosis algorithms typically focus on one aspect of the system which makes it difficult to capture problems originating from the interaction between and across different system layers: physical level (power or communication), functional level and communication level. Such multi-layer problems are typically addressed after the fact with tedious and error prone manual analysis.

In this paper, we consider in-vehicle EES as an embedded and distributed complex system, subject to the design for fault detection, isolation, and mitigation. Based on recent advancements in Network Science, we develop the layered EES network modeling methodology to capture highly inter-related in-vehicle EES. We develop novel multi-partite and multi-attribute betweenness centrality measures to quantify the importance to which a node has control over pair-wise connections between other nodes in the layered EES network model. We apply multi-partite and multi-attribute betweenness centrality measures to rank and recommend fault detection and isolation monitoring points that cannot be discovered by single layered analysis techniques and conventional betweenness centrality measures. We provide usage-based and random failure simulation strategies for recommending fault isolation and mitigations points for desired diagnostic coverage. We present our initial empirical study toward this network-based approach of IVHM.

We discuss related work in Section 2 and introduce our layered network modeling methodology in Section 3. In Sections 4-6, we describe our multi-partite and multi-attribute betweenness centrality, and their application to fault analysis monitoring. Section 7 provides an example study. We conclude our papers with future research direction in Section 8.

2. RELATED RESEARCH

Our work is related to embedded system, complex system diagnosis, and network science. Struss et. al. (2010)

compiled a special issue on the recent advancements of model-based diagnosis in which (Wang & Provan, 2010) describes the automated benchmark diagnostic model generator, with various domain, topology and system-level behaviors, based on the graphical model approach of network science. The benchmark models generated in (Wang & Provan, 2010) can be provided as the input to our methodology for fault detection, isolation and mitigation analysis.

Simonot-Lion (2009) is another special issue compiling recent advancements in the area of in-vehicle embedded system. Zeng et al., (2009) describes a stochastic analysis framework for the end-to-end latency of distributed real-time systems and demonstrated the experimental results on Controller Area Network (CAN). This work focuses on simulation and analysis of probability distribution for end-to-end latency analysis of active safety functions on vehicles. Our work, on the other hand, focuses on design and diagnosis.

Our proposed new measures for quantifying EES complexity and maintainability is based on betweenness centrality measures in network science. Brandes (2008) gives a comprehensive survey and contrasts most recent variants of betweenness centrality. Our proposed new measures are inspired from our layered EES network; therefore, there is no compatible measures in the state-of-the-art as surveyed in (Brandes, 2008). The measures closest to ours are those described in (Borgatti, 2005; Flom et. al., 2004). However, their works do not consider multi-partite, multi-attributes layered networks. In general, these works focus on social network analysis and has no mentioning of fault-isolation and fault-mitigation analysis.

3. Layered Network Modeling

By taking the holistic approach to model in-vehicle EES, we make the following modeling assumptions to construct the layered, multi-partite, multi-attribute network for analyzing an EES system.

1. Each network layer models one aspect of EES; for example, physical network layer represents physical wiring connections of ECUs, functional network layer represents relations of software functions among ECUs, message network layers models message flows among ECUs, and so on.
2. Nodes can be annotated with node types. Designation of node type leads to partitions of nodes where nodes in the same partitions do not have edges; for example, one ECU node is not directly linked to another ECU node, but via Message nodes in a message network layer.
3. Nodes can be annotated with node attributes to represent their special characteristics. Node attributes are usually defined orthogonally to node types; nodes with the same node type may have different node attributes, and similarly nodes with different types may have the same node attribute. For example, node attributes {Sending, Receiving} can be used to annotate nodes across node types {ECU, Message}.
4. Edges within the layered network can be annotated with edge attributes where the value of an attribute typically represents the types of information flowing between nodes. For example, a feature node may have an edge to another feature node with edge attributes {data, frequency} in the dataflow network.
5. Edges across different layers typically represent dependency or identity relations. For example, the same hardware ECU node may appear in both electrical and physical sub-networks which warrant across layer edges.

Formally, we consider a graph $G=(N,E)$ consists of a nonempty countable set of nodes N and a set of directed or undirected edges $E \subseteq N\times N$. A multipartite graph is a graph where N is divided into nonempty disjoint subsets (called Parts) and no two nodes in the same subset have an edge connecting them. Nodes can be associated with a vector of node attributes N_A; similarly, edges can be associated with a vector of edge attributes E_A. Part is imposed by topological structure, whereas attribute is primarily augmented for the semantic aspect of a node. A layered, multi-partite, multi-attribute EES network consists of layers of multi-partite, multi-attribute graphs where node types correspond to parts, and edges across layers represent dependency or identity relations for entities in different layers. Figure 1 shows an example layered network of in-vehicle EES layered network.

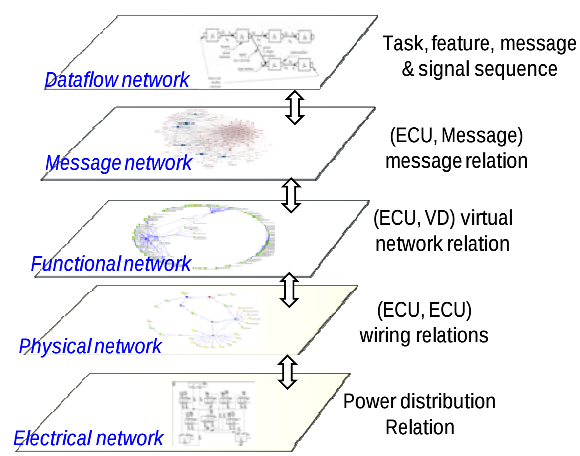

Figure 1: An example layered EES network consists of layers of electrical, physical, functional, message, and dataflow sub-networks; relation within each layers are shown to the right; dependency and identify relations across layers are summarized into double arrows across layers of sub-networks. Note that across layer links are not restricted to neighboring layers only.

4. BETWEENNESS CENTRALITY

Betweenness centrality is defined in social network analysis to quantify the importance to which a node has control over pair-wise connections between other nodes, based on the assumption that the importance of connections is equally divided among all shortest paths for each pair (Freeman, 1978). The *betweenness centrality* $BC(n_i)$ for a node $i \in N$ is defined as follows.

$$BC(i) = \sum_{h \neq i \neq j} \frac{\sigma_{hj}(i)}{\sigma_{hj}},$$

where σ_{hj} is the total number of shortest paths between h and j, and $\sigma_{hj}(i)$ is the number of such shortest paths that pass through the node i. The $BC(i)$ can be scaled between 0 and 1 using $\frac{BC(i)}{(|N|-1)}$ where $|N|$ is the number of nodes in the graph. Correspondingly, the *betweenness centrality* $BC(e)$ for an edge $e \in E$ is defined as the number of shortest paths passing through the edge, i.e., $BC(e) = \sum_{h \neq i \neq j} \frac{\sigma_{hj}(e)}{\sigma_{hj}}$. The $BC(e)$ could be normalized between 0 and 1 using $\frac{BC(e)}{[(|N|-1)(|N|-2)]/2}$.

Recognizing the rich semantics in the layered EES network, we develop novel multi-partite, multi-attribute betweenness centrality to account for node types and attributes in the layered in-vehicle EES network.

4.1. Multi-partite Betweenness Centrality

In the layered EES network, each node and edge can have different types and attributes which warrant further constraints on how betweenness centrality can be defined when considering different semantic meaning of shortest paths in the layered EES network. We propose three different multipartite betweenness centrality measures based on the constraints on node types (parts) in the network.

We first define the *homogeneous* multipartite betweenness centrality $BC_P(i)$ for a node $i \in N_P$, where N_P is a part $N_P \subset N$, is defined as follows:

$$BC_P(i) = \sum_{h \neq i \neq j \in N_P} \frac{\sigma_{hj}(i)}{\sigma_{hj}},$$

where σ_{hj} is the total number of shortest paths between h and j, given that nodes h, i, and j are all in the same part N_P, and $\sigma_{hj}(i)$ is the number of such shortest paths that pass through the node i. This is to constrain the shortest paths such that the starting and ending nodes are the same node types (in the same part) as the one of the intermediate node. For example, an ECU node linked to another ECU node via a gateway ECU with some message nodes along the path.

Next, we define the *bi-mode* multipartite betweenness centrality where the starting and ending nodes are the same part but different from the part of the intermediate node. The *bi-mode* multipartite betweenness centrality $BC_{\bar{P}}(i)$ for a node i is:

$$BC_{\bar{P}}(i) = \sum_{h,j \in N_Q \neq N_P, i \in N_P} \frac{\sigma_{hj}(i)}{\sigma_{hj}},$$

where σ_{hj} is the total number of shortest paths between h and j that are in the same part, but are different from the part of the node i, and $\sigma_{hj}(i)$ is the number of such shortest paths that pass through i. One example use of this measure is to consider a message node i sitting on the paths of communications between two different ECU nodes.

We define the *heterogeneous* multipartite betweenness centrality for a node $BC_{\hat{P}}(i)$ for a node i as follows:

$$BC_{\hat{P}}(i) = \sum_{h \in N_O, i \in N_P, j \in N_Q} \frac{\sigma_{hj}(i)}{\sigma_{hj}},$$

where σ_{hj} is the total number of shortest paths between h and j that are in different parts and not in the same part as the node i ($N_O \neq N_P \neq N_Q$), and $\sigma_{hj}(i)$ is the number of such shortest paths that pass through i. This measure assumes that there are at least three parts defined in the network. One example use of such measure could be finding out the betweenness for a node in functional layer and starting and ending nodes are in the layers of message and physical networks.

4.2. Multi-attribute Betweenness Centrality

To account for attributes orthogonal to topological definition of parts, we define *homogeneous* multi-attribute betweenness centrality $BC_A(i, a)$ and *negated* multi-attribute betweenness centrality $BC_{\bar{A}}(i, a)$ for a node $i \in N$ and an attribute $a \in A_N$ as follows:

$$BC_A(i, a) = \sum_{h \neq i \neq j, a(h)=a(i)=a(j)} \frac{\sigma_{hj}(i)}{\sigma_{hj}},$$

where σ_{hj} is the total number of shortest paths between h and j, given that nodes h, i, and j has the same values for the attribute a (i.e., $a(h) = a(i) = a(j)$), and $\sigma_{hj}(i)$ is the number of such shortest paths that pass through i ; and

$$BC_{\bar{A}}(i, a) = \sum_{h \neq i \neq j, a(h)=a(j), a(i) \neq a(j)} \frac{\sigma_{hj}(i)}{\sigma_{hj}},$$

where σ_{hj} is the total number of shortest paths between h and j, given that nodes h and j has the same values for the attribute a (i.e., $a(h) = a(j)$) but they have different values from node i ($i.e., a(i) \neq a(j)$), and $\sigma_{hj}(i)$ is the number of such shortest paths that pass through i.

Similarly, the multi-attribute betweenness centrality, $BC_A(e, a)$ and $BC_{\bar{A}}(e, a)$ for an edge $e \in E$ and an attribute $a \in A_E$, can be defined as those for the nodes. One example use of multi-attribute betweenness centrality is the attributes of an ECU such as the "role" which can have the attribute value "receiving" or "sending" for different messages.

4.3. Betweenness Centrality Distribution

In addition to quantifying the importance of a target (a node or an edge) in the network, we can compute the betweenness centrality for every target in the network to derive a distribution of betweenness centrality. We can then compute descriptive statics (e.g., average, percentile, variance, skewness, etc.) to characterize such betweenness centrality distribution. In Figure 2, we show an example of truncated homogeneous betweenness centrality distribution for a functional network.

The betweenness centrality distribution can be used to quantify the complexity, as well as maintainability of a layered EES system. For example, a centralized design of EES may have a more skewed betweenness centrality distribution than the one with distributed design. In Figure 2, we see that 69/584=11.81% nodes have above average betweenness centrality, which give us a quite skewed distribution from the functional network point of view.

To improve system maintainability, more resources can potentially be put into the system to improve the reliability of the targets with high betweenness centrality metric, or to increase diagnostic coverage for targets with low betweenness centrality metric. The betweenness centrality distribution can enable such a trade-off analysis for improving the design of maintainability.

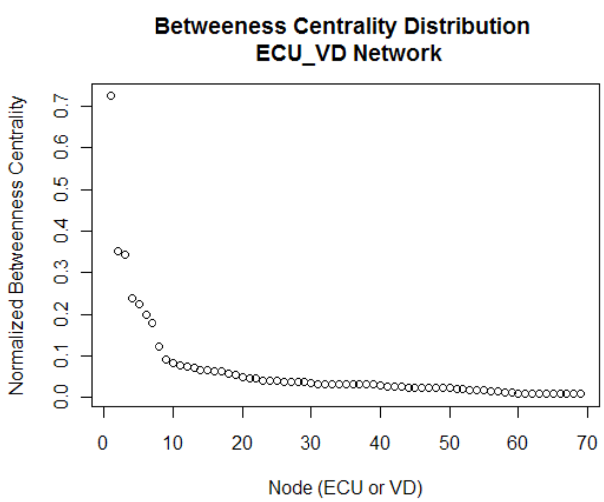

Figure 2: An example distribution of betweenness centrality: the top 69 nodes (out of 584 nodes) have above average betweenness centrality in a functional network.

5. Fault Analysis Monitoring Points

Fault-detection and fault-isolation requires actively monitoring a system in operation. The layered EES network models diverse aspects of the system and the operational status can be thought of as signals and information flow over the network. One may consider using the network model as a platform to simulate the operations of EES; however, it is unlikely to simulate all possible combinations of inputs, especially in the wide ranges of different and unforeseeable operational environments.

Betweenness centrality, as a measure of quantifying the node importance, provides a good basis for ranking where to include fault analysis monitoring points, assuming due to resource constraints not all parts can be monitored. In Figure 3, we show an example network to illustrate this point. The upper panel shows that node *G* has four immediate neighboring nodes whereas node *I* has only three immediate neighbors; however, node *I* is more important than node *G* with respect to betweenness centrality measure ($BC(I)$=14 > $BC(G)$=0.67). The bottom left panel shows that if node *G* fails, node *I* can still monitor all traffics on the network; however, if node *I* fails, the network is fragmented as shown in the bottom right panel. This warrants the claim that the high betweenness centrality node serves as a better fault monitoring point.

We propose to use the betweenness centrality distribution, in conjunction with the measure of degree neighbors, as the basis for setting up monitoring points for fault detection and isolation with respect to desired fault coverage. The following steps can generate recommended fault analysis monitoring points:

1. Compute betweenness measures (multi-partite, multi-attribute betweenness centrality) to quantify the importance of the nodes in EES;

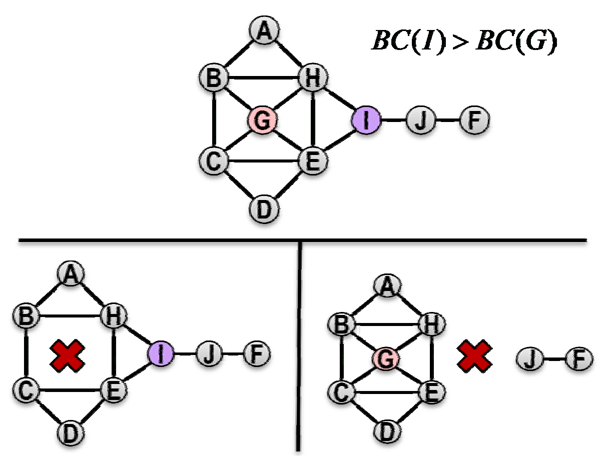

Figure 3: An example to illustrate the usefulness of betweenness centrality for fault analysis monitoring.

2. Apply an adjustable threshold to select nodes as candidate monitoring points (e.g., select nodes with BC measure above *x*-percentile of BC distribution).
3. Check whether the degree neighbors (e.g. 2nd degree neighbor) of all selected nodes provide the desired coverage of the whole network;
4. If no, go back to Step 2 and adjust the threshold;
5. If yes, recommend the selected nodes as the monitoring points.

Diagnostic coverage for a given monitoring point is computed via the nodes' degree neighbors, which is system dependent and subject to the observability of failure mechanisms. Figure 4 shows an example of diagnostic coverage for node I as the monitoring point.

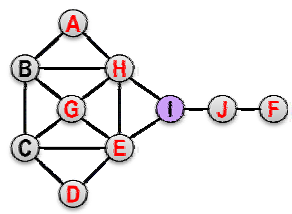

Figure 4: Example diagnostic coverage with a monitoring point at node *I* and coverage of 2nd degree neighbors (nodes in red fonts).

The diagnostic coverage of the whole network is the union of the diagnostic coverage of all selected monitoring points. It is advised to trade-off between cost and diagnostic coverage in selecting monitoring points for a given system.

6. Fault Mitigation Analysis

The purpose of fault-mitigation analysis is (1) to quantify how robust the EES is with respect to different failures, and (2) to identify which surviving nodes can potentially take over the functionality of nodes which have failed.

To support fault-mitigation analysis, we introduce two sequential failure strategies: random failure and usage-based failure strategies to simulate the effects of failures. For usage-based failure strategy, we assume that the usage of a node is in proportion to its betweenness centrality. We can either deterministically fail the node with the largest betweenness centrality by assuming that the most used node is more likely to fail, or randomly select a node to fail.

The steps for fault-mitigation simulation are summarized as follows:

1. Compute betweenness centrality measures (multi-partite, multi-attribute betweenness centrality) for all nodes.
2. Select the next node to fail according to the selected failure strategy (random or usage-based failure).
3. Simulate the effect of failures by removing the edges of the selected node.
4. Check whether the sequential failure simulation has reached the completion criterion (e.g., stop simulation when all edges are removed from the network; or when a certain percentage of survival nodes is remained in the network).
5. If no, go to Step 1;
6. If yes, output the effects of sequential failure for mitigation analysis.

The output of sequential failure simulation consists of a sequence of failure nodes and their effects in the form of updated betweenness centrality distributions.

We propose two measures to quantify the robustness of EES. First, we propose to quantify network fragmentations that may result in the loss of the ability in executing fault-mitigation operation, using the threshold for dissolving the giant component. Second, we propose to quantify the gradual changes of sequential failures using the mean of the normalized betweenness centrality.

A giant component is a connected sub-network that contains a majority of the entire network nodes. Since nodes in the giant component can all reach each other, this warrants the potential of executing fault-mitigation operations. However, when failures are induced, edges are removed from failed nodes. This may lead to network fragmentation which in turn dissolves the giant component; consequently, fault-mitigation operations may not be able to reach all nodes in the network. Hence we can evaluate the robustness of a layered EES network by considering how many failures are needed for a given failure strategy (usage-based or random) to reach a threshold value of nodes remaining in the giant component.

To quantify the gradual effect of sequential failures before the giant component reaches its dissolving threshold, we propose to use the changes in the means of the normalized betweenness centralities. By definition, nodes in two different fragmented subnets will not have shortest paths between them. This will lead to the decreasing of the mean of the normalized betweenness centrality for the whole network as sequential failures progress.

Since the output of a failure simulation records the effect of each simulated sequential failure, we can make recommendations on which nodes may potentially be burdened to implement fault-mitigation functions of the failed nodes. A simple heuristic is to use the neighbors of the failed node to carry out the function. Such heuristic may not be viable for usage-based strategy, as the failing node is the one that has the highest importance for pair-wised connections. Another heuristic is having every second highest importance node of the survival network fragments carry out the function of the failed node. This heuristic avoids immediate nearest-neighbor failing and at the same time carries out the failed node that needs to sit on many shortest paths.

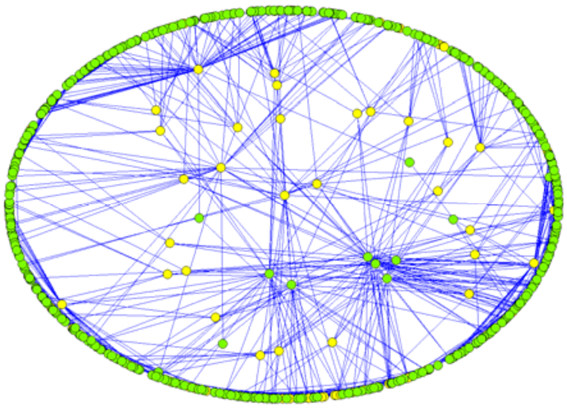

Figure 5: An example functional network depicting relations between virtual devices (green nodes) and ECUs (yellow nodes).

7. AN EXAMPLE STUDY

To demonstrate the values of proposed methods, we show our analysis on an example layered EES network. We first show how consideration of different node types may lead to different views of the importance of a node. We next show the effect of usage-based node failures based on different node types. Finally, we show simulation of sequential failures for fault-mitigation analysis.

We apply multi-partite betweenness centrality on the network depicted in Figure 5. We show the distributions of betweenness centrality for each part in Figure 6 and Figure 7.

We simulate the failures and inspect the changes of betweenness centrality measures. Figure 8 show an example of changes in the distribution of betweenness centrality for failing the top three ECUs. Nodes with increasing betweenness centrality after the failures can be considered as survival nodes that can carry out functions of failed nodes (e.g., Node9 and Node17 in Figure 8).

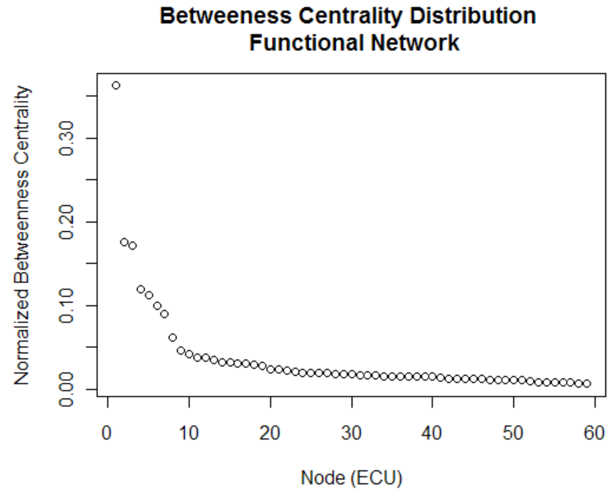

Figure 6: An example distribution of betweenness centrality in functional network. The distribution shows the top 59 ECU (out of 102) with above average betweenness centrality.

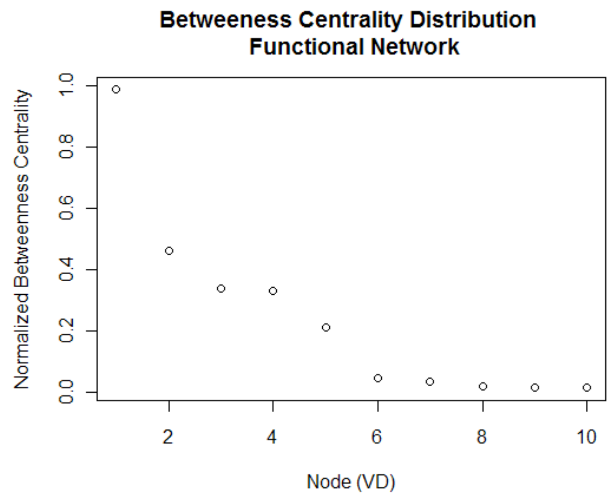

Figure 7: An example distribution of betweenness centrality for VD part in functional network. The distribution shows the top 10 VDs (out of 482) with above betweenness centrality.

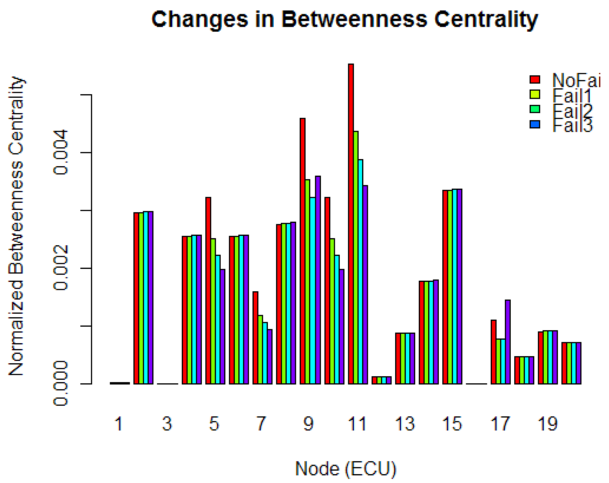

Figure 8: An example of changes in the distribution of betweenness centrality by failing the top 3 ranking ECU. The original ECU betweenness centrality is charted along with their new distribution after each simulated failures (Fail1, Fail2, and Fail3).

8. CONCLUSION

The network-theory based approach reported in this paper provides a first step toward integrated fault detection, isolation, and mitigation analysis capabilities for in-vehicle embedded electrical and electronic systems (EES). We apply layered network modeling over EES to build a layered multi-partite, multi-attribute network which represents physical, structural, functional, and data-flow aspects of in-vehicle EES. We employ two failure strategies to simulate failures and analyze the effects using betweenness centrality measures. We develop novel multi-partite, multi-attribute betweenness centrality to account for the effects of failures and to quantify complexity, maintainability, and robustness of EES. We provided an example to demonstrate our proposed methodology.

ACKNOWLEDGEMENT

The authors would like to thank Thomas Fuhrman, Mark Howell, and Shengbing Jiang for many valuable technical discussions. Special thanks also go to many of our colleagues in Engineering, including Doug Duddles, Sandeep Menon, Ken Orlando, Bob Schwabel, Lars Soderlund, Mike Sowa, and Natalie Wienckowski, for sharing the domain knowledge and shaping the research directions.

REFERENCES

Brandes, U. (2008). On Variants of Shortest-Path Betweenness Centrality and their Generic Computation, *Social Networks* 30(2): 136-145.

Borgatti, S.P. (2005). Centrality and Network Flow, *Social Networks*: 27, 55-71.

Flom, P.L., Friedman, S.R., Strauss, S. and Neaigus, A. (2004). A New Measure of Linkage Between Two Sub-networks, *Connections* 26 (1): 62-70.

Freeman, L.C. (1978). Centrality in Social Networks: Conceptual Clarification, *Social Networks*: 1, 215-239.

Simonot-Lion, F. (2009). Special Section on In-Vehcile Embedded Systems, *IEEE Transaction on Industrial Informatics*, Vol. 5, No. 4.

Struss, P., Provan, G., de Kleer, J., and Biswas, G. (2010). Introduction to Special Issue on Model-Based Diagnosis, *IEEE Transaction on System, Man, and Cybernetics – Part A: Systems and Humans*, Vol 40, No 5.

Wang, J. and Provan, G. (2010). A Benchmark Diagnostic Model Generation, *IEEE Transaction on System, Man, and Cybernetics – Part A: Systems and Humans*, Vol 40, No 5.

Zeng, H., Di Natale, M., Giusto, P. and Sangiovanni-Vincentelli, A. (2009). Stochastic Analysis of CAN-based Real-Time Automotive Systems, *IEEE Transaction on Industrial Informatics*, Vol. 5, No. 4.

Tsai-Ching Lu received his M.S. degree in Computer Sciences from NYU in 1996, and his M.S. and Ph.D. degrees in Intelligent Systems from University of Pittsburgh in 2000 and 2003 respectively. He joined HRL Laboratories, LLC, in 2003 and currently holds the position of Senior Research Staff Scientist. His research interests include decision making under uncertainty, complex electronic diagnosis and prognosis, and behavior modeling and predictions. He received HRL's Distinguished Inventor Award in 2010.

Yilu Zhang received his B.S., and M.S. degrees in Electrical Engineering from Zhejiang University, China, in 1994, and 1997, respectively; and his Ph.D. degree in Computer Science from Michigan State University in 2002. He joined the R&D center of General Motors Co. at Warren, Michigan in 2002, and currently holds a position of Staff Researcher. Dr. Zhang's research interests include statistical pattern recognition, machine learning, signal processing, and their applications such as integrated system health management and human machine interactions. Dr. Zhang is a Senior Member of IEEE. He is a winner of 2008 "Boss" Kettering Award – the highest technology award in GM – for his contribution to Connected Vehicle Battery Monitor, a remote vehicle diagnostics technology.

David Allen received his M.S. and Ph.D. degrees in Computer Science from the University of California, Los Angeles (UCLA) in 2001 and 2005 respectively. He joined HRL Laboratories, LLC, in 2006 and currently holds the position of Research Staff Scientist. His research interests include artificial intelligence, probabilistic reasoning, decision making under uncertainty, complex systems analysis, and network science; he has applied them to many domains including integrated system health management, cybersecurity, social network analysis, and behavior modeling. In 2010 he received HRL's Distinguished Inventor Award.

Mutasim Salman is a Lab. Group manager and a Technical Fellow in the Electrical, Controls and Integration Lab. of GM Research and Development Center. He has the responsibility of development and validation of algorithms for state of health monitoring, diagnosis, prognosis and fault tolerant control of vehicle critical systems. He pioneered the work on integrated chassis control in the late eighties that led to the production of GM "Industry First" Stabilitrak1 and then to Stabilitrak3. He had an extensive experience in hybrid vehicle, modeling, control and energy management strategies. He has several GM awards that includes 3 GM prestigious Boss Kettering, 3 McCuen, and 2 President and Chairman Awards. Mutasim received his bachelor's degree in Electrical Engineering from University of Texas at Austin; M.S. and PhD in Electrical Engineering with specialization in Systems and control from University of Illinois at Urbana- Champaign. He also has an Executive MBA. He is IEEE senior member. He holds 21 patents and has coauthored more than 42 refereed technical publications and a book. He joined the GM R&D staff in 1984.