

System Topology Identification with Limited Test Data

Vijitashwa Pandey and Zissimos Mourelatos
Oakland Univ.

ABSTRACT

In this article we present an approach to identify the system topology using simulation for reliability calculations. The system topology provides how all components in a system are functionally connected. Most reliability engineering literature assumes that either the system topology is known and therefore all failure modes can be deduced or if the system topology is not known we are only interested in identifying the dominant failure modes. The authors contend that we should try to extract as much information about the system topology from failure or success information of a system as possible. This will not only identify the dominant failure modes but will also provide an understanding of how the components are functionally connected, allowing for more complicated analyses, if needed. We use an evolutionary approach where system topologies are generated at random and then tested against failure or success data. The topologies evolve based on how consistent they are with test data. We demonstrate the proposed methodology by approximately identifying the topology of an electric motor.

CITATION: Pandey, V. and Mourelatos, Z., "System Topology Identification with Limited Test Data," *SAE Int. J. Mater. Manf.* 5(1):2012, doi:10.4271/2012-01-0064.

1. INTRODUCTION: SYSTEM TOPOLOGY IDENTIFICATION

All systems eventually fail. Reliability measures their ability to accomplish the tasks they are designed for, for a pre-specified amount of time. The reliability of engineering systems is a very important performance measure because it determines their safety record, their lifecycle cost including the maintenance cost and the manufacturer's image. For some sectors such as aerospace and nuclear energy, reliability is very critical and can be the most important metric of performance. For other sectors such as personal computers for example, although reliability may not be as critical, it impacts the overall image of the manufacturer and eventually the demand of the product. For these reasons, a lot of research has been and is currently devoted into developing methods to measure and/or ensure adequate system reliability.

Systems fail when some or all of the components comprising them fail. System topology determines which combinations of components will lead to system failure. As such, it provides an exhaustive repository of failure information. Various methods exist for representing failure succinctly in a visual form. For example, a fault tree takes in different independent component failure/success events and decides whether the entire system will fail using AND/OR logic gates. The reliability block diagram provides similar

information. Details can be found in reliability textbooks (e.g., Kapur and Lamberson, 1977).

In the reliability literature, the system topology is often assumed known. This may not be true because this information is hard to discern and more importantly, the end user may not have complete information about the inner workings of a product because of lack of knowledge about interdependencies among different components, or because the underlying physical or chemical processes are not well understood. If the system topology is unknown, researchers tend to identify only the dominant failure modes instead of approximating the full system topology. The authors believe that test data should be used to learn as much about the system topology as possible. An approximate system topology will not only be more general, it will automatically take into account the dominant failure modes because they are represented in the test data.

This paper presents a simulation-based approach to approximate the system topology using limited available system failure test data. An evolutionary method is used based on a set of rules. Section 2 presents a brief literature survey and discusses different applications where the system topology identification is of value. Sections 3 and 4 describe the proposed evolution-based simulation methodology and Section 5 demonstrates the value of the methodology using an electric motor example. Finally, Section 6 concludes and highlights future work.

2. BACKGROUND AND LITERATURE SURVEY

Reliability engineering accounts for lack of knowledge and inherent variability in engineering design. The field of Reliability-based Design Optimization (RBDO) focuses on finding the best possible design under reliability constraints which ensure feasibility in the presence of uncertainty. Violation of one or more constraints constitutes a system failure. Many publications describe in detail all developments in RBDO. [Agrawal, A. \(2004\)](#) and [Liang et al. \(2007\)](#) provide many relevant references.

Fundamental research in reliability addresses how reliability is defined, calculated or used in system simulation. Complex systems fail under different conditions, each of which is referred to as a failure mode. Understanding how different components affect the overall functioning of a system has been a topic of research for quite some time. [Tang \(2001\)](#) presents a graph theoretic approach to determine system reliability using a properly defined adjacency matrix. [Arunajadai et al. \(2004\)](#) show how dominant failure modes can be detected using clustering analysis. A decomposition based approach to detect failure modes is presented by [Tumer et al. \(2002\)](#). Different failure modes are ranked based on their contribution to the variance of certain performance metrics. Principal Components Analysis (PCA) is then used to identify the more prevalent failure modes. The literature either assumes that the system topology is known or if it is unknown, only the dominant failure modes are of interest.

This research focuses on another class of problems where it is not known how a system fails if some of its components fail. For example, [Bond and Davis \(2004\)](#) talk about adhesive failure modes and how hard it is to pinpoint the cause. The problems become harder when the underlying science (chemistry or physics) of system failure is not well understood. [Ioannou \(2007\)](#) developed failure modes for a Metal Oxide Varistor (MOV) based on its response to operating conditions. We could not find any literature formalizing the approximation of system topology for reliability purposes.

2.1. Topology identification scenarios

Understanding how a system fails can greatly contribute towards design improvements. It is clear however, that scenarios exist where the system topology or even its approximation is not known masking therefore, potentially important failure modes. In the following, we provide some examples.

- **Black box design:** An end user treats a system as a black box when he/she has limited knowledge of how the system functions. After-market repairs of automobiles or other engineering equipment is a classical case. Troubleshooting failures is usually done in an ad-hoc manner using common sense rules and the original manufacturer is usually not in the picture to perform diagnostics or repairs.

- **Composite failure modes not identified:** Complex system failure modes are often due to unaccounted combinations of component failures. In today's distributed design world this is becoming more and more of an issue. Consider a large automobile manufacturer subcontracting individual components to various small manufacturers. There is an enhanced probability of component interactions being overlooked leading to unforeseen failure modes.

- **Operating conditions:** A product may fail due to operation under non-standard or highly variable conditions with many controllable and uncontrollable environmental parameters. When unaccounted for operating conditions lead to a failure, the proposed approach can incorporate them as pseudo components which can be integrated into the system topology.

- **Failure modes not understood:** This is common for many systems where the underlying physics is not well understood or modeled. For example, [Ioannou \(2007\)](#) talks about a new Metal Oxide Varistor for which the failure scenarios are uncertain and discusses how failure modes can be detected.

2.2. Why simulation?

Each system failure or success event provides information about the system topology. In a *deductive approach*, one would analyze the repercussions of each system failure or success event in conjunction with the system state which provides information on which components have failed. For example, if a system fails while only one component has failed, that component can be safely assumed to be in series with the rest of the system components. Similarly if the system is working while a component has failed, that the component is in parallel with at least one other component. Another example where deductions can be made is when two components fail while either one of them is not enough to cause system failure. In such a case, we can deduce that the two components are in parallel with each other while being in series with the rest of the components. Can we make these deductions in practice for large scale systems? Theoretically, yes. While these inferences can lead us eventually to the true system topology, the learning rules to be developed for more complicated failure scenarios are however, highly complex. At the very least, we would need to keep track of many success/failure scenarios as well as easily store and retrieve this information which must be then used to make deductions using the rules. The process becomes progressively intractable.

An alternative approach is to use simulation where different candidate topologies are *scored* based on how similar they are to the true system. The similarity is captured by whether the test failure data is consistent with the reliability behavior of a candidate topology. If the candidate topology fails when the system fails or succeeds when the system also succeeds the candidate topology is consistent with the true system based on that test. If more tests are performed we will obtain better approximate topologies

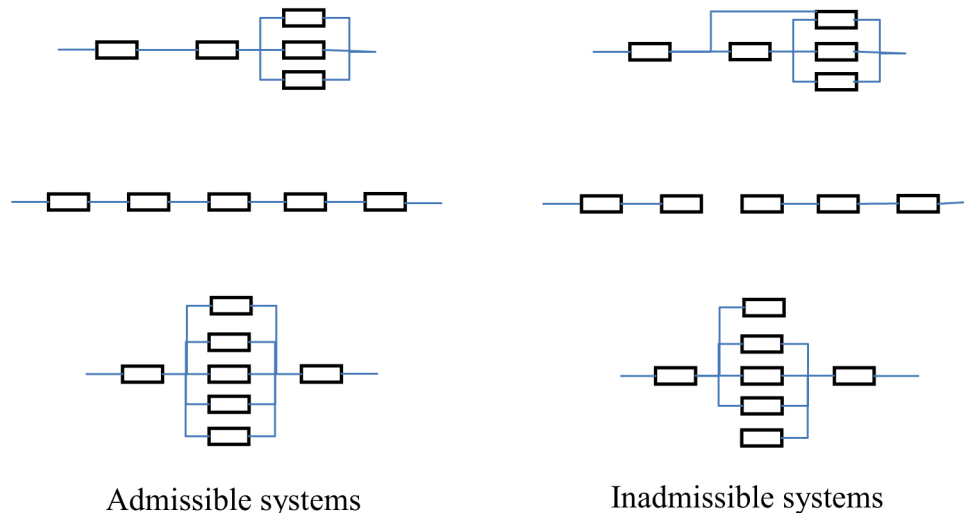


Figure 1. Examples of admissible and inadmissible system topologies

whose reliability approaches that of the true system topology. It is possible that using this method, some incorrect deductions can be made, but the benefits outweigh the disadvantages. The method is far superior to the deductive method because of its ease of application and ability to incorporate large amount of test data easily.

2.3. Dominant failure mode identification

If the system topology is approximated using actual test data, the dominant failure modes are automatically accounted for. As a result, failure modes are encountered based on their frequency of occurrence. The topologies that will be consistent with the test data (i.e. the topologies that fail/work when the system fails/works for a given state of components) will tend to capture these failure modes.

The following two sections present our proposed simulation-based topology identification through evolution using test data.

3. RANDOM TOPOLOGY GENERATION

Generation of a random system topology is a non-trivial task. While certain connections between different components may not represent an actual working system, they should at least be functionally representative of a physically realizable (admissible) system. Figure 1 shows examples of admissible and inadmissible systems. The inadmissible systems have no path from one end to another, or have redundant links or have components that do not form part of the system. The figure does not show all possible admissible or inadmissible systems. The first column of Table 1 lists the system admissibility rules. There are two approaches that can be used to randomly generate an admissible system topology:

(1). Random system topology generation using hit and trial: In the hit and trial approach one generates a system and

then tests whether it is admissible or not, using the rules in Table 1. This is however, computationally inefficient because even for a small number of components, a prohibitively large number of inadmissible systems can be generated. As a result, it will take a long time to generate even a small number of admissible systems.

(2). Random system topology generation using “divide and conquer”: A better alternative to hit and trial is to incrementally generate a system by adding a component, and determine its connection to the partial system generated so far. A component is added only at feasible locations and the feasibility is determined using rules similar to those in the second column of Table 1.

A system topology can be represented using a Directed Acyclic Graph (DAG). A DAG is a collection of nodes (components) and directed edges (connections between components) (Thulasiraman and Swamy, 1992). It only has directed connections between the nodes and it does not exhibit any cycles (hence the term acyclic). This means that there is a source node (all arrows come out of that node) and a sink node (all arrows go into that node). Figure 2 shows an example of a DAG where node 10 is the source node and node 2 is the sink node.

The directedness is necessary to represent system topology using graphs because loops can be identified easily. In the proposed method, we introduce two fictitious components which act as source and sink in the system topology. Arrows are allowed to only come out of the source component and enter only the sink component. The simulation method works with two components at a time. Two components are selected at random with replacement from the pool of components including the source and the sink, and a directed connection is made between them without violating any rule, indicating that one component feeds into another. More components are then selected and tested if a directed connection can be made between them.