# Fault Propagation and Sensitivity Analysis to Support Counterterrorism Activities

Robert L. Nagel
Missouri Univ. of Science & Technology
G4B Interdisciplinary Eng. Building
Rolla, MO 65409, USA
Robert.Nagel@mst.edu

Robert B. Stone
Missouri Univ. of Science & Technology
102A Interdisciplinary Eng. Building
Rolla, MO 65409, USA
rstone@mst.edu

James L. (Monty) Greer
United States Air Force Academy
2354 Fairchild Hall – Room 6H140
USAFA, CO 80840, USA
Monty.Greer@usafa.edu

Daniel A. McAdams
Texas A&M University
Engineering/Physics Building 108, MS 3123
College Station, TX 77843, USA
dmcadams@tamu.edu

**Abstract.** This paper discusses research on fault propagation analysis and two associated fault sensitivity measurement approaches for the investigation of faults in human-centric processes. The techniques presented grew out of a joint project between the U.S. Air Force Academy and Missouri University of Science and Technology to investigate terrorist activities involving improvised explosive devices. Two potential counterterrorism activities related to improvised explosive devices are developed through the paper to demonstrate the application and usefulness of the methodologies. Provided with the examples is a discussion of fault propagation, the sensitivity analysis performed, an interpretation of the results, and an overview of anticipated future applications in system design.

## Introduction

Traditionally, process analysis techniques are used in conjunction with fault propagation and sensitivity measurement tools to understand why a failure occurred or how to prevent future failures. This research, however, has developed fault propagation and sensitivity measurement techniques to be used in conjunction with process analysis such that weaknesses in a system might be exploited to aid counterterrorism activities. Process analysis is applied to gain a better understanding of the actions common to the prelude, execution and conclusion of an Improvised Explosive Device (IED) event. Faults are propagated through the process models to indicate where possibilities exist to reduce soldier risk through preemptive termination of IED incidents.

In this paper, three process-based failure analysis tools are presented: (1) propagated fault analysis, (2) process fault levels, and (3) system to process sensitivity. Propagated fault analysis investigates system failures from the perspective of the actions taken to develop, construct and implement a system. To gauge the sensitivity of the process to the faults, process fault levels and system to process sensitivity are presented. Process fault level is a qualitative assessment tool to rate potential effects of propagated faults, and system to process sensitivity is a quantitative measure of the system sensitivity to the loss of flow paths. Process fault level and system to process sensitivity, taken together, provide sensitivity measures to gauge the scope of projected faults. The techniques presented are based on representations of the system, and not the components comprising the system, thus insight into how a system might react to unexpected faults occurring during systems' applications is provided without needing to know the exact system components. This proves particularly valuable to IED deterrence where the components that comprise an IED are not often known.

Propagated Fault Analysis (PFA), Process Fault Levels (PFL) and the associated sensitivity metric, System to Process Sensitivity (SPS), grew out of a joint project between the U.S. Air Force Academy (USAFA) and Missouri University of Science and Technology (Missouri S&T) to investigate terrorist activities involving Improvised Explosive Devices (IEDs). The researchers applied integrated process and functional modeling (Nagel et al. 2008) to the IED system to study IED development and deployment from an engineering design perspective to gain insight into potential components, common actions and potential weaknesses. The paper is organized as follows: The background section provides a brief description of current system-based counterterrorism research as well as design-based failure analysis techniques. Next, the process analysis methodology used in this research is presented, and is followed by the approach taken in this research and the resultant methodologies. Through these sections, the approaches are applied to two counterterrorism examples. The paper concludes with a discussion of the results, concluding remarks and anticipated future work.

## Background

**Faulting Terrorist Activities.** Modeling terrorist actions, whether during planning or execution, is fundamental to counterterrorism activities. Active modeling of terrorist scenarios provides insight leading to the discovery of new information about terrorist activities to increase efficiency of current counterterrorism measures and more effectively intercept current terrorist actions (Harowitz and Haimes 2003). Modeling efforts and techniques have increased since 9/11 (Goldstein 2006) and include a number of different approaches. A few approaches are reviewed here, however, a more complete review of modeling activities relating to counterterrorism activities can be found in (Haimes and Harowitz 2004).

In (Goldstein 2006), Goldstein provides an overview of two mathematical approaches to modeling the social aspects of terrorism with the research of Silverman and of Carley. Silverman’s research focuses on mathematically simulating the social aspects (i.e., values, culture, emotions, etc.) of individual agents to provide insight into the minds and motives of terrorists, while Carley’s research focuses on modeling and analyzing the formation, morphing and leadership of terrorist social networks. Other modeling approaches, such as in the research of Kujawski and Miller (Kujawski and Miller 2007) and Horowitz and Haimes (Harowitz and Haimes 2003) focus on the reduction of risk by modeling terrorist attack scenarios. The research of Kujawski and Miller uses decision-attack event trees to model terrorist attack scenarios in conjunction with quantitative probabilistic risk assessment matrices to assess the risk associated with identified outcomes, while Horowitz and Haimes utilize Hierarchical Holographic Modeling to develop attack scenarios and employ Risk, Filtering, Ranking and Management (RFRM) to filter and rank known intelligence. Horowitz and Haimes perform Bayesian Analysis to add credibility to known intelligence, and utilize multi-objective decision trees to facilitate decision-making under uncertainty and risk.

**Product Design Failure Analysis Techniques.** While the majority of this paper researches the deliberate faulting of an IED, it is still necessary to understand techniques used to investigate and predict system failures during the product design process since the IED process, through abstraction, can be formulated and approached as an engineering design problem. This is the approach often taken with Red Team studies where an entity will put on the Red Hat of the adversary and conduct planning activities to determine the vulnerability of its own defensive or offensive systems (Defense Science Board Task Force Sept 2003). Red teams are established by an enterprise to challenge aspects of that very enterprise's plans, programs, assumptions, etc. It is this aspect of deliberate challenge that distinguishes red teaming from other management tools although the boundary is not a sharp one. Red teaming can be used at multiple levels within the enterprise; for example, at the: (1) strategic level to challenge assumptions and visions, (2) operational level to challenge force postures, a commander's war plan and acquisition portfolios and (3) tactical level to challenge military units in training or programs in development (Defense Science Board Task Force Sept 2003).

Failure Modes and Effects Analysis (FMEA) is considered the industry standard methodology for failure analysis. FMEA was originally developed from the failure modes and effects criticality analysis (FMECA) defined in MIL-P-1629A (Kumamoto and Henley 1996; MIL-P-1629A 1980). An effort was later made by Ford, Chrysler, General Motors and the Automotive Industry Action Group to standardize FMEA, and a reference manual was published (Automotive Industry Action Group (AIAG) 1993). FMEA provides the methodology for the identification of potential modes of failure for each component in a system and the consequences of each failure. FMEA tends to rely heavily on component and expert knowledge to identify failures and possible consequences making FMEA well suited for product redesign. To move failure analysis into conceptual design (where system components are rarely known), a number of alternative FMEA-based approaches utilizing intended system functionality instead of component specific knowledge have been developed. Russomanno has proposed an Expert System for FMEA (XFMEA) (Russomanno, Bonnell, and Bowles 1993). XFMEA automates FMEA activities and utilizes behavioral, functional and structural representations that allow failure analysis activities to be performed before component assignment has been performed. Advanced FMEA also tries to bring the failure analysis activities into the conceptual design phase by applying behavior models mapping control-based functionality to system components (Kmenta and Ishii 1998; Kmenta, Fitch, and Ishii 1999). This method identifies deviations from intended functionality that result from system failures. With Function Hazard Analysis (FHA), experts determine potential failures based on the behaviors of the system's functions to determine function-failure combinations. These function-failure mappings are based on system and subsystem functional decompositions (Wilkinson and Kelly 1998). The Function Failure Design Method (FFDM) utilizes functional modeling with the Functional Basis (Hirtz et al. 2002) to apply failure analysis during the conceptual design phase and overcome the shortfalls of expert-based systems through the application of a knowledge-based repository of failure data (Stone, Tumer, and Van Wie 2005; Roberts, Stone, and Tumer 2002).

The aforementioned FMEA-based approaches fail to identify the cascading or propagation of failures through a system. Fault tree analysis and event tree analysis are specifically designed to investigate failure propagation. Fault Tree Analysis (FTA) applies backward logic to develop a top-down chain of events which have the potential to lead back to a single negative event (Vesely and Goldberg 1981; Blanchard and Fabrycky 2006; Voland 2004). Events propagating to the negative event are modeled using Boolean logic to create chains of potential propagated failures. Event Tree Analysis (ETA), conversely, uses forward logic to investigate a single initiating event. From the single initiating event, probable failures of events, which can occur in sequence, are analyzed to determine the likelihood of success or failure (Kumamoto and Henley 1996; Bedford and Cooke 2001). Both ETA and FTA can be performed on a system during conceptual design and can address internal and external failures. Probabilistic Risk Analysis (PRA) combines the failure propagation techniques of FTA and ETA with failure effects identification techniques such as FMEA to answer three questions: what can go wrong, how severe will the failure be, and how likely is the failure to occur (Stamatelatos 2000; Kumamoto and Henley 1996; Bedford and Cooke 2001). PRA can be performed on internal and external initiating events through all phases of a system's life cycle. PRA, like FMEA, FTA and ETA, tends to rely on expert knowledge and does not typically employ structured modeling for system abstraction.

Function-based Failure Propagation and Functional Failure Identification and Propagation (FFIP) have been proposed to analyze the propagation of faults through function-based system models utilizing the Functional Basis. Function-based failure propagation identifies two failure modes, "No Flow" and "No Failure," which occur due to the propagation of failures along flows (Krus and Grantham Lough 2007), and likelihoods are calculated based on information stored within a failure knowledge base. FFIP identifies failures in a system occurring from the loss of functionality (Kurtoglu and Tumer 2007). Functional failures occur from negative events in a behavior model and are propagated through the functional model to determine the impact to the system.

Each of the previously mentioned techniques for identifying failures and their resultant faults deal primarily with the workings of the system. Fault propagation graphs, however, are based on structured hierarchical process and sub-process models and apply causal relationships to a set of possible failure modes (Padalkar et al. 1991). Fault propagation graphs applied to process models can be used to analyze failure propagation between the states, where states are defined as the different phases of an entire process. Causal relationships model the propagation of failures from a process to its sub-processes. In this same vein, graph theory-based approaches may be utilized to analyze the time and resource requirements of processes via Program Evaluation and Review Technique (PERT) charts where processes are modeled as network diagrams with the nodes representing events and the arcs representing constraints (Marshall 1971). To measure the effect of unexpected events modeled via PERT charts, Bowman presents a sensitivity analysis to estimate resultant probability distributions for program performance measures based on changes to project constraints (Bowman 2007).

**Process Modeling.** A number of methodologies have been developed to allow for the modeling of processes, which could be utilized in fault propagation studies. A few of these methods include: PERT charts (Ulrich and Eppinger 2004; Blanchard and Fabrycky 2006), activity diagrams (Ulrich and Eppinger 2004), Integrated Definition (IDEF) modeling such as IDEF#3 (Mayer et al. 1995) and SysML (Friedenthal, Moore, and Steiner 2008). This work applies the process modeling techniques developed in (Nagel, Stone, and McAdams 2006; Hutcheson et al. 2006) and integrated with functional modeling in (Nagel et al. 2008) for its flexible, flow-based, hierarchical structure and integration with a standard modeling lexicon, the Functional Basis (Hirtz et al. 2002).

The process modeling methodology applied in this research considers two levels of models: event level model and configuration level model. Events are further modeled at two levels of fidelity: a high-level black box model and a more detailed decomposition of the black box. Flows, which enter higher-level models, must be the sum of the flows required by lower-level models. Event models provide a hierarchical decomposition of the process and are comprised of multiple sub-events. A configuration model is a decomposition of a single event and represents the discrete functional changes, which must occur to a system, for the successful operation of an event. Configuration models, as modeled in this research, are labeled using the lexicon provided by the Functional Basis (Hirtz et al. 2002). The terms system, process, event and configuration may be defined as:

- **System** - A structured arrangement of functional elements tied together via material, energy and signal flows describing an artifact or collection of artifacts. Functional models are often used for system representation.

- **Process** - Systematic and continuous actions, operations or changes occurring in a definite manner toward a particular goal (Process  2001) and tied together via the product and material, energy and signal flows.
- **Event** - A specific action or operation of a system, which may relate to the environment, configuration, specific application, or sequencing of operations.  The sum of events where the system is employed results in a process.
- **Configuration** - A specific discrete instance of the overall functionality of the product occurring as a part of an event.  Collectively many configurations define an event of the product.  The configuration of a product is modeled functionally.

A process model is generated following six steps:  (1) identify requirements necessary to complete the process, (2) formulate a black box event model for the process, (3) identify events, input/output flows and start/stop times based on the process requirements, (4) decompose the black box model into an event model consisting of chains of events, (5) decompose individual events into more detailed configuration models (6) verify that each process requirement is addressed by at least one configuration (Nagel, Stone, and McAdams 2006; Hutcheson et al. 2006).

As an example of the generation of process models, consider the case studied for the Joint Improvised Explosive Device Organization (JIEDO) where the planning, construction and deployment of an Improvised Explosive Device by a managed terrorist cell is investigated.

The event model, shown in Fig. 1, provides an abstraction covering the three phases of an IED incident:  prelude, development and execution.  Through the open-source research performed during the contract project, twelve key events have been identified as a part of an IED incident.  These events are each modeled as individual boxes in Fig. 1 and are:

0.  Manage & Recruit Terrorists
1. Train Terrorists
2. Wait Terrorists
3. Determine Target
4. Gather Intelligence
5. Acquire Materials
6. Build IED
7. Store IED
8. Deploy IED
9. Target Target
10. Detonate IED
11. Evade Detection

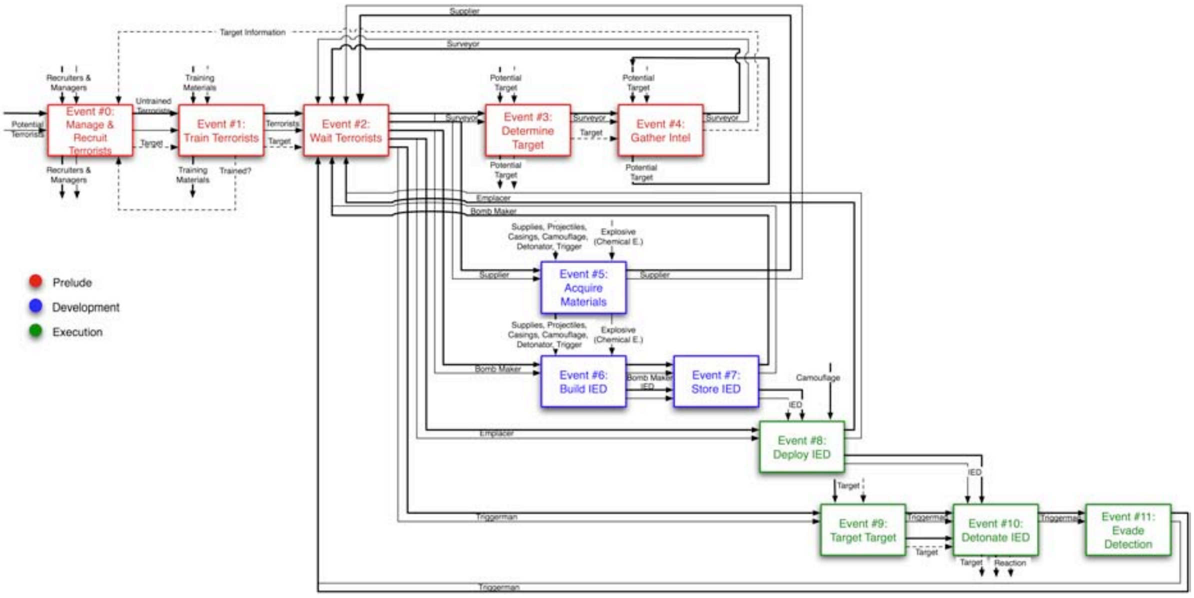

Figure 1:  Event model covering the three phases of an IED incident

The event model begins with an event for recruiting and managing the terrorist cell.  Potential terrorists are brought together with recruitment agents.  Newly recruited terrorists are provided with training materials and taught to fill specific tasks; once trained, terrorists wait for their assignments. Since terrorist cells are often highly complex, managed groups, the event model branches into five branches to demonstrate distribution of tasks with each branch having a specific, specialized agent to perform each task. The first branch of the event model, colored red, is the prelude stage of an IED incident. During the prelude stage, surveyors identify potential targets and gather intelligence. The process of gathering intelligence continues over a period of days or weeks, which is represented with the bold face arrow leaving and reentering Event #4, Gather Intel. Gathered target information is returned to the managers and recruiters in Event #0, Manage & Recruit Terrorists; this information feedback is represented with the dash arrow titled, *Target Information*. Managers determine the type of IED to be developed, built, and deployed and this information is fed forward to the agents assigned or trained for specific tasks. A supplier acquires the appropriate materials for the IED; a bomb maker constructs and stores the IED; an emplacer retrieves the IED from its storage location and deploys it; and if required, a triggerman targets the target and detonates the IED before trying to evade detection. This process occurs continually with potential targets being identified, intelligence being gathered and processed, and IEDs being built and deployed. Because of the continual nature of the process, the prelude (red) events can occur while the development (blue) and execution (green) events occur. The modeling of the three stages is represented with all of the specifically trained terrorists feeding back to a wait event to represent the repetition of the process.

Each of the events in the event model of the IED incident (Fig. 1) can be decomposed into a configuration model to detail the specific actions terrorists are required to perform to successfully complete an event. A configuration model is generated similarly to a functional model using function-flow pairs; it is important to note, however, that unlike functional models, configuration models include the entire system as a flow to capture interactions such as construction and transportation.

The configuration model for the event, *Gather Intelligence*, provides an abstraction representing the act of a surveyor monitoring the targets' habits, interpreting the habits, collecting information and then supplying the information to the managers. The model, provided as Fig. 2, represents the surveyor entering the event as two flows, one thin, representing energy and one bold, representing physical form. The surveyor is transferred and guided by the notion of a selected target–represented as a dashed arrow–to detect the potential target. Intel–represented as a dashed arrow–is processed, collected and stored by the surveyor. A feedback loop–represented as the bold and thin arrows of the surveyor–signifies the repetition of the surveyor watching the actions of interest before the intelligence is supplied to the managers. Once the surveyor has completed the assigned task, the surveyor is exported from the gather intelligence event.

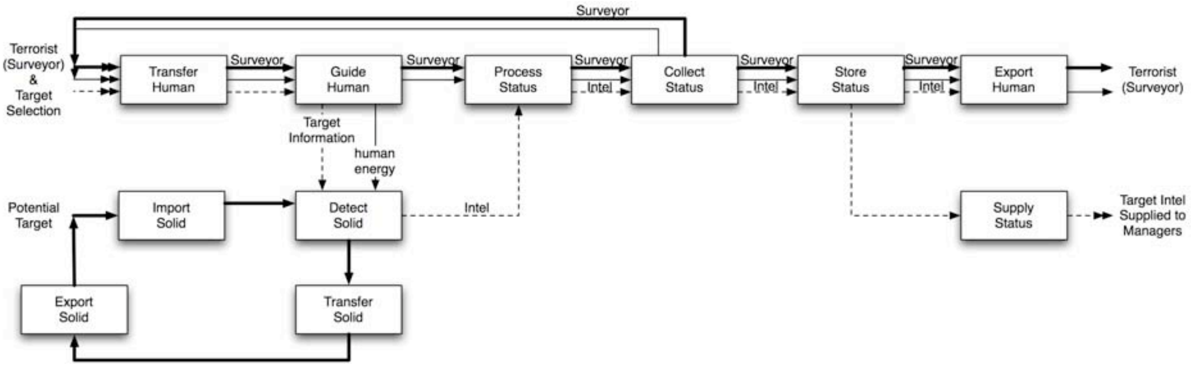

Figure 2: Configuration model for Event #5, Gather Intelligence

Configuration models for each event in an IED incident are generated similarly to the example configuration model for Event #5, Gather Intelligence, by considering each of the operations, which must occur on the flows entering the event.

## Approach and Methodology

The general research approach is summarized in Fig. 3. Placing the methodology in context of the IED application, the following specific approach was followed: Open-source research was conducted on terrorist organizations and their actions involving IED development and deployment. From the research collected, the IED process models were generated following the aforementioned methodology. Using the research on IED incidents and the interactions made clear through the process representations, possible fault points were analyzed to find potential weaknesses that might be exploited in efforts to prevent future IED events. This methodology used to identify potential fault points is PFA and is represented by Steps 2-4 in Fig. 3. PFL, Step 5, and SPS, Steps 6-8, were developed as sensitivity metrics to indicate the impact of the identified faults on the IED process and to assess the value of affecting the identified fault points. PFL provides a rating indicating the level to which a process is terminated for potential scenarios, which result from the propagation of a given fault. SPS provides a percentage of function-flow pairs faulted by the loss of flows.

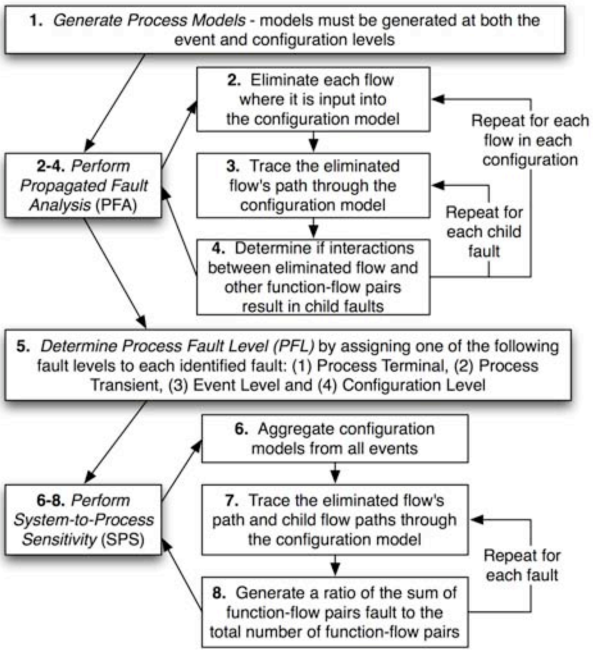
1. Generate Process Models - models must be generated at both the event and configuration levels
2. Eliminate each flow where it is input into the configuration model
Repeat for each flow in each configuration
2-4. Perform Propagated Fault Analysis (PFA)
3. Trace the eliminated flow's path through the configuration model
Repeat for each child fault
4. Determine if interactions between eliminated flow and other function-flow pairs result in child faults
5. Determine Process Fault Level (PFL) by assigning one of the following fault levels to each identified fault: (1) Process Terminal, (2) Process Transient, (3) Event Level and (4) Configuration Level
6. Aggregate configuration models from all events
6-8. Perform System-to-Process Sensitivity (SPS)
7. Trace the eliminated flow's path and child flow paths through the configuration model
Repeat for each fault
8. Generate a ratio of the sum of function-flow pairs fault to the total number of function-flow pairs

Figure 3: Methodology for analyzing propagated configuration faults

**Steps 2 - 4, Propagated Fault Analysis (PFA).** First, process models of the system are generated. Once process models are generated, propagated fault analysis is applied to the models by considering the elimination of flows at three locations: (1) flow importation into a model, (2) flow crossing in a model, and (3) flow branching in a model. PFA investigates how changes to these flows, vital to an overall process, affect the desired final outcome. Thus, PFA is based upon each flow in a process model having a key role in the successful completion of the overall objective. If a function or flow is disrupted, the final desired outcome is also disrupted. The elimination of flows is propagated through configuration models; the eliminated flows are denoted with an X in the model. As faulted flows are propagated, potential interactions, which result in other flows being eliminated, should be considered. Flows eliminated by other faulted flows are termed *child faults*, and they too are traced through the system considering new interactions.

PFA utilizes two failure methods to show failure propagation through a process model. First is a “No Flow” failure, which has been taken from function-based failure propagation (Krus and Grantham Lough 2007) and occurs when a function (or configuration change) fails. This failure results in the termination of the flow on which the function acts allowing the failure to propagate along the flow path. The “No Flow” failure is demonstrated as the initiating failure in the diagram provided in Fig. 4. The second failure method is a “No Function” failure where a specific function (or configuration change) fails, but flow through the configuration is not affected and the remainder of the chain continues to be operational. The configuration faulted by the “No Function” failure is marked by a circle with a slash as well as by the continuation of the flow arrow over the faulted configuration block. The “No Function” failure is demonstrated as the second failure in the diagram provided in Fig. 4.

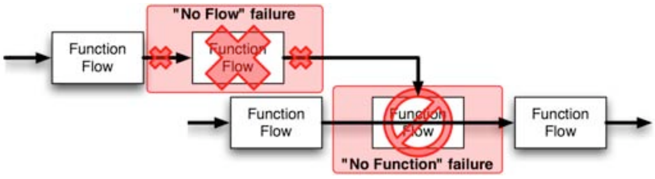

Figure 4: "No Flow" and "No Function" failure modes

Consider again, as an example, the model for a surveyor gathering intelligence on the selected target modeled in Fig. 2. The model for this event includes elements for the surveyor monitoring the routines and actions of a potential target as well as the elements for the potential target being detected by the surveyor. If the surveyor is stopped while monitoring the actions of the potential target, then the target cannot be detected, intelligence cannot be collected, and there is no intelligence to transfer to the leaders of the terrorist cell. The initiating fault, stopping the surveyor, is represented by a shaded X on the flows and the configuration, *transfer human*, in Fig. 5, and its fault propagation is represented by an unshaded X. The failure results in “No Flow” faults for the surveyor since the surveyor is no longer present to perform the assigned task; however, since the potential target is still carrying out its standard routines and actions, a “No Function” fault occurs with the *detect solid* configuration. The circle with a slash over the faulted configuration, *detect solid*, as well as the extension of the potential target flow arrow represents the “No Function” fault. A shaded X is used to denote the child fault occurring with the flow of intelligence at the configuration, *detect solid*.

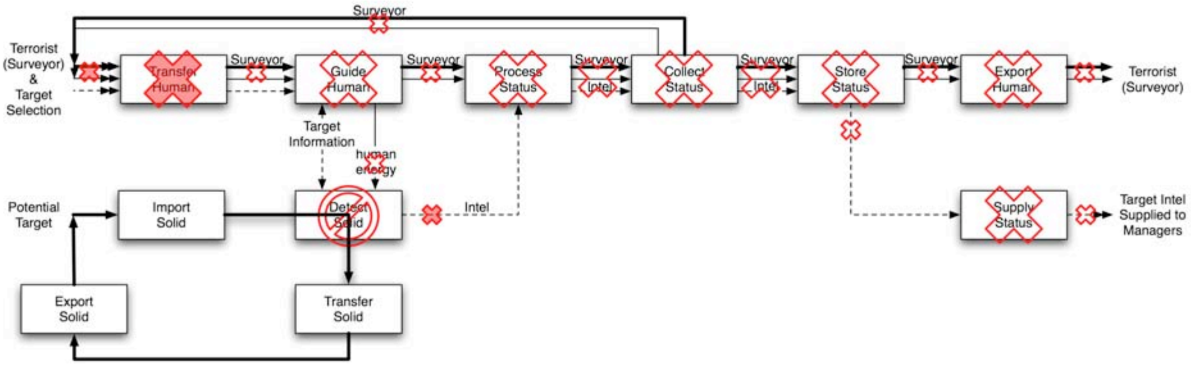

Figure 5: Configuration model with a fault on the terrorist (surveyor) flow

If, however, the observability of the potential target were to be reduced such that the surveyor were no longer able to gather intelligence by monitoring the potential target’s actions, then an effective fault would be created on the potential target flow as modeled in Fig. 6. Again, the initially faulted flow and function block are represented with a shaded X, and the propagation is represented with an unshaded X. This faulting scenario would result in a “No Flow” fault for the configurations of the potential target–*import solid, detect solid, transfer solid and export solid*–since the configurations are no longer performed. Those configurations dealing with processing, collection and storage of intelligence, however, would each have “No Function” faults since the surveyor is still present and able to perform the assigned functions; a circle with a slash represents the “No Function” faults and the surveyor flow passes over the faulted configurations.

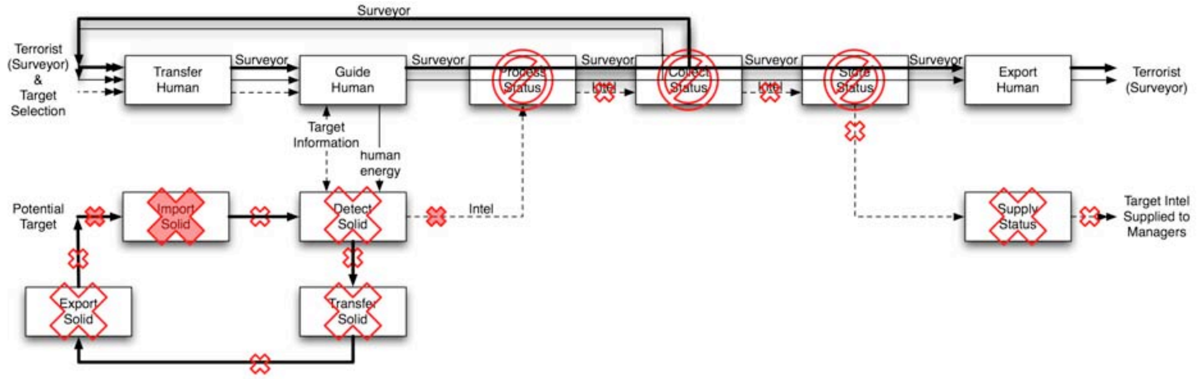

Figure 6: Configuration model with a fault on the potential target flow

**Step 5, Propagated Fault Level (PFL).** At the completion of PFA, configuration models are marked with a number of faulted configurations and flow paths. Each fault is now rated with a PFL considering whether the fault affects a single configuration, multiple configurations and consequently an entire event, or the entire process. PFL is a qualitative rating of the impact a fault has on the overall process based on projected scenarios for the continued operation of the process, and PFL provides a way for an analyst to denote those faults having the potential to be more devastating to the system as a whole.

From the PFA performed on the modeled IED processes, four distinct types of faults were identified which has lead to four PFL ratings: (1) process terminal, (2) process transient, (3) event level and (4) configuration level. The fault level is assigned based on how far a fault propagates through a process; for instance, a configuration level fault affects only the system’s configuration during a single event. An event level fault, however, affects the operation of an entire event. The final two types of faults, process terminal and process transient, affect the entire process. Process transient faults stop the process but the process may be restarted once the fault is repaired. Process terminal faults, however, leave the process irreparably faulted. The fault levels are:

- **Process Terminal Fault** - ends a process completely. The process cannot be restarted. Process Terminal is the most severe fault.
- **Process Transient Fault** - ends a current instantiation of a process. The process can be restarted at a future time.
- **Event Fault** - occurs when the process is not stopped. Instead, an event within a process does not function properly.
- **Configuration Fault** - occurs when a single configuration change within an event can no longer occur. Configuration faults do not cause the entire event to fault, yet may degrade the performance of the event.

Since failures propagate through an entire process, a minor failure during one event might result in a more significant failure at a later event. Thus, a single failure might result in more than one PFL. Also, since the PFL is a tool to assess the impact of various scenarios for how the remainder of a process will operate, different scenarios are often assigned a different PFL.

For the IED incident, consider the faulted surveyor flow shown in Fig. 5. A number of different scenarios can be considered for how the remainder of the process will be performed with a faulted surveyor flow. Consider, first, that the surveyor is irreplaceable and the managers of the terrorist cell refuse to take action without proper intelligence; then faulting the surveyor would be a **process terminal fault**. If, however, the surveyor can be replaced, which is the likely scenario since the process is continuous, but the managers of the terrorist cell still refuse to take action without proper intelligence, then the fault would be a **process transient fault**. It is probable, however, for the managers to act on inadequate intelligence, miscalculate the actions of the target and proceed with a terrorist act that has a reduced likelihood of success; in this case, an **event fault** has occurred. For the effective fault of the potential target flow shown in Fig. 6, a reduction in the repeatability of the potential target (i.e. convoys, troops, etc. vary their routes and transit times) results in **configuration fault** on the intelligence flow where it is still collected, but the predictability is reduced.

**Steps 6 - 8, System to Process Sensitivity (SPS).** Last, SPS is calculated for each fault. SPS is a percentage of flow paths faulted in a combined configuration model due to single initiating failure. The sensitivity metric requires that configuration level models from all events in a process be compiled into a single aggregated configuration model and the faults identified during the PFA be traced through the aggregated model. A summation of faulted configurations is used to generate a ratio of faulted configurations to the sum of all configurations in the combined model following Eq. (1).

$$Sensitivity = \frac{\sum ConfigurationsFaulted}{\sum Configurations} \cdot 100\% \tag{1}$$

Consider, from the IED example, two faults: (1) stopping the surveyor from monitoring our troops and convoys and (2) reducing the predictability of our troop and convoy movements. Each of these faults occur during Event #5, Gather Intelligence, shown in Fig. 2. To determine the SPS for each of these two faults, their propagation must be traced through the combined configuration model of the entire IED process. The combined configuration model connects the decompositions of each event into a single model based on the flow connectivity originally modeled at the event level shown in Fig. 1 and contains 105 unique configurations. Table 1 provides the calculated sensitivities for the two faults.

Table 1: SPS calculations

| Faults | Sum Faulted Configurations | Total Number Configurations | Sensitivity |
|---|---|---|---|
| (1) Stopping the surveyor from monitoring troops and convoys | 20 | 105 | $\frac{20}{105} \cdot 100\% = 19.0\%$ |
| (2) Reducing the predictability of troop and convoy movements | 12 | 105 | $\frac{12}{105} \cdot 100\% = 11.4\%$ |

The use of configuration models for each event ensures that the percent sensitivities can be compared between models with different numbers of events and configurations. Also, it can and was used to compare multiple potential configuration mappings for different potential IED processes to identify where various IED processes are more or less sensitive to faults.

## Discussion

The preceding example, based on IED deterrence, demonstrates the application of PFA, PFL and SPS toward understanding fault propagation and sensitivity at a configuration level. The SPS determined for the faults, stopping the surveyor from monitoring our troops and convoys and reducing the predictability of our troop and convoy movements, show insight into the differences between the SPS and the PFL. Stopping the consistency of the troop and convoy movements has a sensitivity of 11.4% indicating that 11.4% of the flow paths through the combined model are faulted. This fault was assigned a configuration level PFL since the event can still be performed with limited reliability once faulted, which logically should indicate that the fault is limited to affect only configuration changes within Event #5 where the fault occurs. However, had the configuration level fault been isolated to Event #5 then it would have a sensitivity of less than 8.3% for 1 event out of 12 total events; this was not the case. Instead, the fault, which still allows all of the events to take place, propagates back to affect the decision makers of the terrorist cell and the assignment of tasks.

A unique PFL was assigned to a number of faults on the surveyor flow depending on the leadership of the terrorist cell. Stopping the surveyor flow, if assigned an event level PFL to indicate that the managers operate on reduced intelligence, has the potential to fault 2 events out of the 12 total or 16.6% of the process. This fault to the surveyor flow is consistent with the SPS finding that 19.0% of the flow paths in the combined model are faulted. A number of different possible scenarios, however, are identified for faulting the surveyor flow because the PFL is meant to be a qualitative impact assessment to classify potential scenarios resulting from identified faults, while there is a single SPS since it is a mathematical assessment of a single fault's propagation.

Other faults identified during the study are provided in Fig. 7 with their sensitivities in the form of a Pareto chart (Blanchard and Fabrycky 2006), the benefit of which is to identify, in rank order, the potential faults that will have the greatest effect on the system under study.

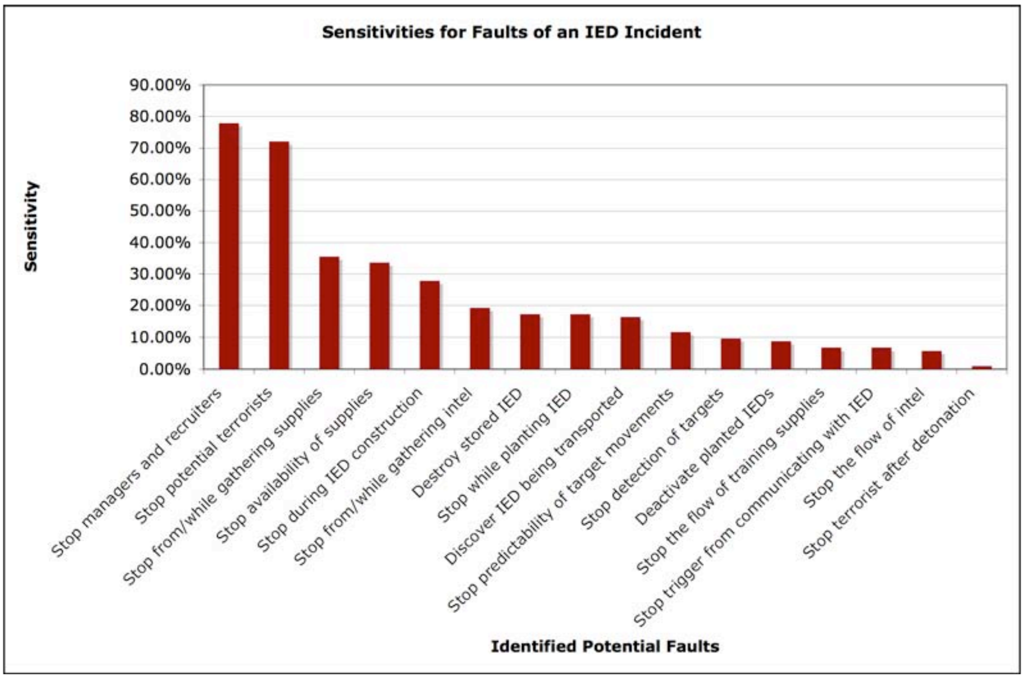

Figure 7: Sensitivities for some of the identified potential fault points in a single IED incident displayed in a Pareto chart

Some of the faults identified during this study, such as stopping potential terrorists before training and stopping the managers and recruiters of a terrorist cell, have process terminal fault levels indicating termination of the IED process. These faults, while being rated as process terminal, result, however, in SPS values of only 72.1% and 77.9%, respectively. Since SPS is an indicator of the configurations faulted along flow paths, flows that are still available as inputs to the system, but are no longer utilized, are considered as resources still available. These available configurations are not counted as a part of the faulted configurations making SPS values of 100% unlikely. It should also be noted that the sensitivities are for a single IED incident. If they handled the repetition of IED incidents, it would not matter if a specific agent were stopped at any point in the process because of continual redundancy. Thus, these sensitivities are **only** applicable for a **single** non-repeating IED incident.

## Conclusions and Future Work

Modeling the actions, social networks and operations of terrorists is key to understanding how terrorist cells operate and mitigating the terrorist threat. This work has utilized process modeling with the Functional Basis to model IED incidents in an effort to better understand potential fault points such that soldier risk might be reduced through more preemptive termination of IED incidents.

The methodologies, PFA, PFL and SPS, have been developed through this research to provide a suite of analysis techniques allowing investigation of faults occurring in a human-centric process. Faulted flows are propagated through models of system configuration changes to investigate their effect. Faults can be investigated without reliance on components due to the application of process modeling. For complex systems, qualitative (PFL) and quantitative (SPS) sensitivity metrics provide the ability to focus efforts on faults whose propagations are most catastrophic to the system. Qualitative sensitivities for various fault scenarios can be presented, and alternative system configurations can be analyzed. Multiple applications of PFA, PFL and SPS to various configuration models for the system allow alternative scenarios to be compared to determine the highest sensitivity to configuration changes and faults.

These tools' lack of reliance on components makes them well suited for use in the product design process during the determination of customer needs and concept identification when components are not likely to be known. Further research will investigate their integration into the conceptual design processes. The application of process models to the collection of customers' needs has the potential to allow designers to investigate design problems following an outcome-driven design paradigm (Ulwick 2005). Once process models of the customers' needs are generated, the methodologies presented in this work will allow a designer to investigate the sensitivity of the current customer actions to the desired customer outcome directing the designer to potential weaknesses in the customers' current actions. Places where weaknesses exist in the customers' current actions represent ideal locations where assistive technologies could be utilized to allow the customer to more effectively reach their desired outcome. As the product is developed, the integration of process and functional models will allow existing functional modeling based failure and risk tools such as function-based failure propagation (Krus and Grantham Lough 2007), RED (Grantham Lough, Stone, and Tumer 2007), FFIP (Kurtoglu and Tumer 2007), and FFDM (Stone, Tumer, and Van Wie 2005) to be paired with the methods presented in this work.

PFL will be further investigated as a technique to identify event and configuration redundancy, where if a failure occurs at only the configuration or event level and the output seems to remain unchanged then redundancy may exist in the system. This analysis could be useful for verifying redundancy in critical systems or for removing redundancy in disposable systems. Also, with complex systems, as configuration models increase in complexity, manual calculation becomes more time consuming, thus a computational tool will be developed to interface with a purpose-specific functional model drawing tool to assist a designer through the propagation of faults along flow paths following the connectivity

◀ ● ▶

between configurations. The hierarchy of the functional and process models will be built into the tool to (1) provide active feedback on a fault's PFL and SPS and (2) provide linking to a system's functional model at faulted configurations so that existing function-based failure propagation and risk tools can be applied.

## Acknowledgements

This work has been performed in conjunction with the United States Air Force Academy and has been funded by the Joint IED Defeat Organization.

## References

Automotive Industry Action Group (AIAG). 1993. Potential Failure Mode and Effects Analysis (FMEA) Reference Manual. Automotive Industry Action Group.

Bedford, T., and R. Cooke. 2001. Probabilistic Risk Analysis: Foundations and Methods. Cambridge: Cambridge University Press.

Blanchard, B.S., and W.J. Fabrycky. 2006. Systems Engineering and Analysis. 4th ed. Upper Saddle River, NJ: Prentice-Hall.

Bowman, R.A. 2007. Efficient sensitivity analysis of PERT network performance measures to significant changes in activity time parameters. Journal of Operational Research Society 58:1354-1360.

Defense Science Board Task Force. Sept 2003. The Role and Status of DoD Red Teaming Activities. edited by Office of the Under Secretary of Defense.

Friedenthal, S., A. Moore, and R. Steiner. 2008. A Practical Guide to SysML: The Systems Modeling Language: Morgan Kaufmann.

Goldstein, H. 2006. Managing Terrorists. IEEE Spectrum 43 (9).

Grantham Lough, K., R.B. Stone, and I.Y. Tumer. 2007. The Risk in Early Design Method (RED). Journal of Engineering Design 18 (1).

Haimes, Y.Y., and B.M. Harowitz. 2004. Modeling Interdependent Infrastructures for Sustainable Counterterrorism. Journal of Infrastructure Systems 10 (2):33-42.

Harowitz, B.M., and Y.Y. Haimes. 2003. Risk-Based Methodology for Scenario Tracking, Intelligence Gathering, and Analysis for Countering Terrorism. Systems Engineering 6 (3):152-169.

Hirtz, J., R. Stone, D. McAdams, S. Szykman, and K. Wood. 2002. A Functional Basis for Engineering Design: Reconciling and Evolving Previous Efforts. Research in Engineering Design 13 (2):65-82.

Hutcheson, R.S., D.A. McAdams, R.B. Stone, and I.Y. Tumer. 2006. A Function-Based Methodology for Analyzing Critical Events. Paper read at ASME International Design Engineering Technical Conferences, at Philadelphia, Pennsylvania.

Kmenta, S., P. Fitch, and K. Ishii. 1999. Advanced Failure Modes and Effects Analysis of Complex Processes. Paper read at ASME Design Engineering Technical Conference, Design for Manufacturing Conference, at Las Vegas, NV.

Kmenta, S., and K. Ishii. 1998. Advanced FMEA using Meta Behavior Modeling for Concurrent Design of Products and Controls. Paper read at ASME Design Engineering Technical Conference, at Atlanta, GA.

Krus, D., and K. Grantham Lough. 2007. Applying Function-Based Failure Propagation in Conceptual Design. Paper read at ASME International Design Engineering Technical Conference, Sept 4-7, at Las Vegas, NV.

Kujawski, E., and G.A. Miller. 2007. Quantitative Risk-Based Analysis for Military Counterterrorism Systems. Systems Engineering 10 (4):273-289.

Kumamoto, H., and E.J. Henley. 1996. Probabilistic Risk Assessment and Management for Engineers and Scientists. New York: IEEE Press.

Kurtoglu, T., and I.Y. Tumer. 2007. A Graph-Based Framework for Early Assessment of Functional Failures in Complex Systems. Paper read at ASME International Design Engineering Technical Conference, at Las Vegas, NV.

Marshall, C.W. 1971. Applied Graph Theory. New York: Wiley-Interscience.

Mayer, R.J., C.P. Menzel, M.K. Painter, P.S. deWitte, T. Blinn, and B. Perakath. 1995. Information Integration for Concurrent Engineering (IICE) IDEF3 Process Description Capture Method Report. Knowledge Based Systems, Incorporated.

MIL-P-1629A. 1980. Procedures for Performing a Failure Mode, Effects and Criticality Analysis. Department of Defense.

Nagel, R.L., R.S. Hutcheson, R. Stone, D. McAdams, and J Donndelinger. 2008. Function Design Framework (FDF): Integrated Process and Function Modeling for Complex System Design. Paper read at ASME International Design Engineering Technical Conference, at New York.

Nagel, R.L., R.B. Stone, and D.A. McAdams. 2006. A Process Modeling Methodology for Automation of Manual and Time Dependent Processes. Paper read at ASME International Design Engineering Technical Conferences, at Philadelphia, PA.

Padalkar, S., G. Karsai, C. Biegl, J. Sztipanovits, K. Okuda, and N. Miyasaka. 1991. Real-Time Fault Diagnostics. IEEE Expert: Intelligent Systems and Their Applications 6 (3):75-85.

Process. 2001. In Webster's Encyclopedic Unabridged Dictionary of the English Language. New York, NY: Random House Value Publishing, Inc.

Roberts, R., R Stone, and I. Y. Tumer. 2002. Application of Function-Failure Similarity Method to Rotorcraft Component Design. Submitted to Journal of Engineering Design.

Russomanno, D.J., R.D. Bonnell, and J.B. Bowles. 1993. Functional Reasoning in Failure Modes and Effects Analysis (FMEA) Expert System. Paper read at Reliability and Maintainability Symposium.

Stamatelatos, M. 2000. Probabilistic Risk Assessment: What is it and Why is it Worth it? edited by NASA Office of Safety and Mission Assurance.

Stone, R., K. Wood, and R. Crawford. 2000. A Heuristic Method for Identifying Modules for Product Architectures. Design Studies 21 (1):5-31.

Stone, R.B., I.Y. Tumer, and M. Van Wie. 2005. The Function-Failure Design Method. Journal of Mechanical Design 127 (3):397-407.

Ulrich, K.T., and S.D. Eppinger. 2004. Product Design and Development. 3rd ed. Boston, MA: McGraw-Hill/Irwin.

Ulwick, A.W. 2005. What Customers Want: Using Outcome-Driven Innovation to Create Breakthrough Products and Services. New York: McGraw-Hill.

Vesely, W.E., and F.F. Goldberg. 1981. Fault Tree Handbook. edited by US Nuclear Regulatory Commission.

Voland, G. 2004. Engineering By Design. 2nd ed. Upper Saddle River, NJ: Pearson Prentice Hall.

Wilkinson, P.J., and T.P. Kelly. 1998. Functional Hazard Analysis for Highly Integrated Aerospace Systems. Paper read at Certification of Ground/Air Systems Seminar.

## Biography

**Robert L. Nagel** is a Doctoral Candidate in Mechanical Engineering at Missouri University of Science and Technology working in the Design Engineering Lab. Robert has a BSME from Tri-State University (known now as Trine University) in Angola, IN and a MSME from Missouri University of Science and Technology in Rolla, MO. Robert has been an intern at General Motors in the Vehicle Development Research Lab and has worked on contract projects with both the United States Army and United States Air Force. His research interests include functional and process analysis and their application to mechatronic systems, complex system design, and specification of customer needs.

**Robert B. Stone** is a Professor in the Interdisciplinary Engineering Department at Missouri University of Science and Technology. Stone has utilized his design background to assist in creating the department's design-focused Interdisciplinary Engineering degree programs. His research interests include design theories and methodologies, specifically product architectures, functional representations and automated conceptual design techniques, and he leads the Design Engineering Lab at Missouri University of Science and Technology (designengineeringlab.org). He has authored chapters on product architecture and reverse engineering techniques in product design texts. Prior to initiating his graduate work, Dr. Stone worked in the Missions Operation Directorate of NASA-Johnson Space Center as a Space Shuttle Flight Controller for the Guidance, Navigation and Control Section.

**Lieutenant Colonel James L. (Monty) Greer** is currently Director of Systems Engineering at the United States Air Force Academy. His education includes Bachelors, Masters, and Doctoral degrees in Mechanical Engineering and he is a registered professional engineer in Colorado. LtCol Greer worked as a construction project engineer and then entered the US Air Force as a commissioned officer. His assignments include serving as a Global Positioning System–satellite subsystems engineer, USAF Academy faculty member, C-17 Flight Test Program Manager, Director of the Applied Mechanics Laboratory and his current systems engineering position. His research interests are in product development and homeland defense.

**Daniel A. McAdams** is currently Associate Professor of Mechanical Engineering at Texas A&M University. McAdams holds a Bachelors and Doctoral degree from The University of Texas at Austin and a Masters degree from the California Institute of Technology. McAdams has held the position of Associate Professor and Associate Chair of Graduate Affairs for Mechanical Engineering at Missouri University of Science and Technology. His research interests are in the area of design theory and methodology with specific focus on functional modeling; innovation in concept synthesis; biomimetic design methods; design of innovative automated products through process modeling; model construction and selection for design; and failure avoidance as applied to product design.