Examining Survivability of Systems of Systems

Examining Survivability of Systems of Systems

Brian Mekdeci, Adam M. Ross, Donna H. Rhodes, and Daniel E. Hastings
Massachusetts Institute of Technology
77 Massachusetts Avenue, E38-576
Cambridge, MA 02139-4307
617-324-0473

Abstract. Previous research has identified design principles that enable survivability for systems, but it is unclear if these principles are appropriate and sufficient for systems of systems as well. This paper presents a preliminary examination of how some of the characteristic properties of systems of systems may enable or hinder survivability, based on existing design principles and a newly proposed taxonomy of disturbances. Two new design principles, defensive posture and adaptation, are introduced. The next phase of research will be to conduct empirical studies to validate the design principles against some of the characteristic properties of systems of systems, and test hypotheses about how survivability will be affected.

Introduction

As systems complexity grows, traditional systems are being interconnected to form larger, more capable systems of systems (SoS). In many circumstances, systems of systems are operated in contexts that are subject to disturbances which may impact the ability of the SoS to deliver value. Increasing the survivability of systems can be expensive, and typically involves tradeoffs. Decision makers are forced to select options that balance value, cost and risk according to their needs, but in systems of systems, the problem is often compounded due to diverse stakeholders and conflicting risk mitigation strategies (Ellison and Woody 2007). Systems engineering design principles to aid designing systems for enhanced survivability were developed in a previous research effort (Richards 2009), but the case studies upon which they were developed and validated, involved traditional systems, such as satellite radar. There has been some debate as to whether traditional systems engineering methods and practices are still valid at the SoS level (Dickerson 2009) . The literature is unclear as to the definition of a SoS, and how it is distinct from a traditional system (Chattopadhyay 2008). This is not surprising, since the definitions of a "system" itself is also ambiguous (Backlund 2000). However, systems of systems can be thought of as a special case of systems, and thus it is important to highlight the characteristic properties of a SoS, and determine how they might affect its survivability. Unfortunately, the concept of survivability upon which the original design principles were generated, was based upon a definition of disturbances that was insufficient for many of the types of problems a SoS may face. Since systems of systems tend to be larger, more complex and operate under more varied contexts than traditional systems, a broader definition of disturbances is needed.

This paper has two goals; (1) To point out deficiencies in the existing classification of disturbances and propose a new taxonomy, and (2) generate hypotheses as to whether or not some of the characteristic properties of systems of systems affect its survivability. The paper begins with the existing definition of survivability and original design principles that enable it for systems.

Survivability

Survivability is defined as the ability of a system to minimize the impact of a finite-duration disturbance on value delivery (Richards et al. 2007; Richards 2009) (Figure 1). Value can be thought of as the net utility (benefit) a system provides to its stakeholders (Keeney 1996). Systems can achieve survivability in three ways (Westrum 2006): (1) reducing the probability that a disturbance will impact the system, known as a system *susceptibility* (2) reducing the amount of value lost directly as a result of a disturbance occurring, known as a system *vulnerability*, and (3) increasing the system's ability to make a timely recovery from a disturbance, known as system *resilience*.

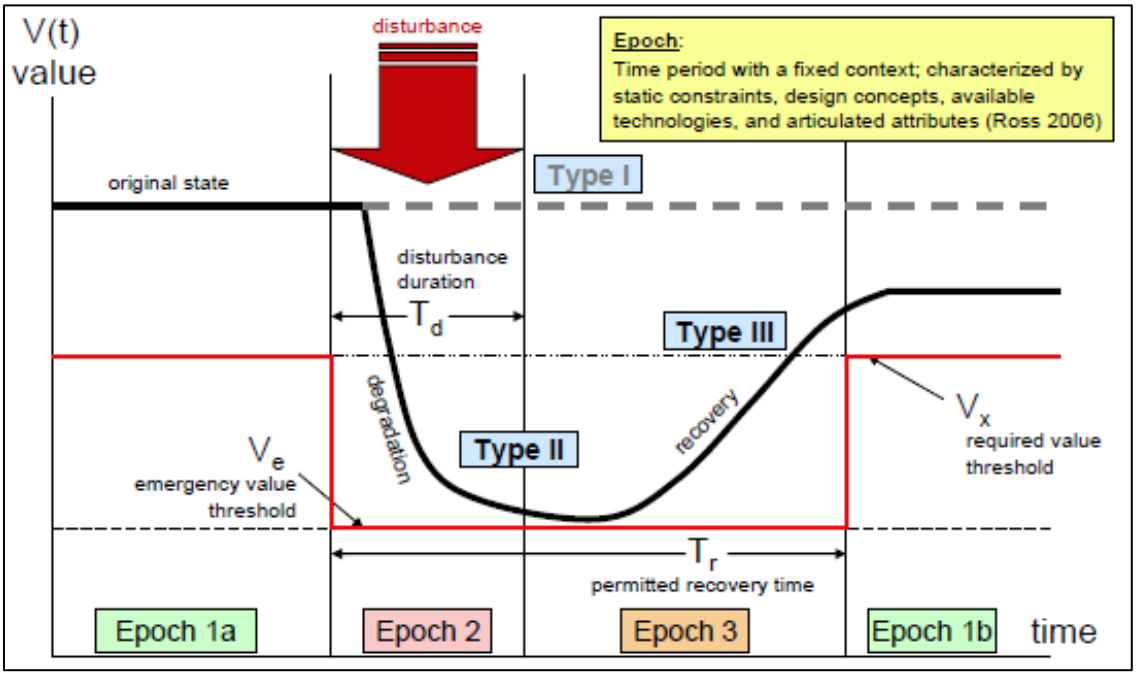

Figure 1: Definition of Survivability (Richards, 2009)

Richards (2009) generated a set of 17 design principles that can enhance the survivability of systems (Table 1). Most of these design principles are examined against the properties that distinguish a SoS from a traditional system and hypotheses about their impact on system survivability are made. However, since survivability is about avoiding, mitigating and recovering from disturbances, it is essential to define and characterize exactly what is and isn't a disturbance.

Table 1: Design Principles for Survivability (from Richards 2009)

Design Principle	Phase of Survivability	Definition	Example
Prevention	Reduce Susceptibility	Suppression of a future or potential future disturbance	Destroying the weapons manufacturing capability of an enemy
Mobility	Reduce Susceptibility	Relocation to avoid detection by an external change agent	Iraqi Scud missile launchers moving during the Gulf war to avoid detection by U.S. forces
Concealment	Reduce Susceptibility	Reduction of the visibility of a system from an external change agent	Stealth technology on the F=117 Nighthawk

Deterrence	Reduce Susceptibility	Dissuasion of a rational external change agent from committing a disturbance	Mutual Assured Destruction during the Cold War
Preemption	Reduce Susceptibility	Suppression of an imminent disturbance	Using Patriot missiles to shoot down Scud missiles during Gulf War
Avoidance	Reduce Susceptibility	Maneuverability away from an ongoing disturbance	Changing flight path to fly around a thunderstorm
Hardness	Reduce Vulnerability	Resistance of a system to deformation	M1 Abrams tank armor
Redundancy	Reduce Vulnerability	Duplication of Critical System Functions	Back-up GEO communications satellites
Margin	Reduce Vulnerability	The allowance of extra capability for maintaining value delivery despite losses.	Long, low-set wings on the A-10 that are able to fly even if half of it is missing (lift margin)
Heterogeneity	Reduce Vulnerability	Variation in system elements to mitigate homogeneous disturbances	Nuclear "triad" of ICBMs, airborne bombers and nuclear submarines
Distribution	Reduce Vulnerability	Separation of critical system elements to mitigate local disturbances	Two mechanical assemblies functionally and spatially separated on A-10
Failure Mode Reduction	Reduce Vulnerability	Elimination of system hazards through intrinsic design	Replacement of Teflon insulation in the oxygen tank with stainless steel following Apollo 13
Fail-safe	Reduce Vulnerability	Prevention or delay of system degradation by leveraging the physic of incipient failure	Autorotation of rotor blade in the Blackhawk
Evolution	Reduce Vulnerability	Alteration of system elements to reduce disturbance effectiveness	B-17 design and tactics evolving during WWII
Replacement	Increase Resilience	Substitution of system elements to improve value delivery	XM-3 and XM-4 satellites replacing XM-1 and XM-2.
Repair	Increase Resilience	Restoration of a system to an improved state of value delivery	STS-61 mission placing COSTAR on the Hubble Space Telescope

Disturbances

Intuitively, a disturbance is something *bad* (Jackson 2010), that may negatively impact a system's ability to deliver value. To distinguish survivability from other related "ilities", Richards (2009) classified disturbances along two axes; (1) whether the origin was internal or external to the system , and (2) whether the disturbance was natural / accidental or malevolent. Survivability of systems was defined to be unique among the "-ilities", as it was only concerned with disturbances that were external to the system, regardless of whether they were natural / accidental or malevolent (`Figure 2`). These distinctions are important, because system designers have to select the appropriate design principles corresponding to the disturbance. However, disturbances have other important characteristics beyond intent and place of origin that should be considered as well. In this section, a taxonomy of disturbances in introduced, which distinguishes disturbances based on their origin, nature, intent, duration, and effect on context.

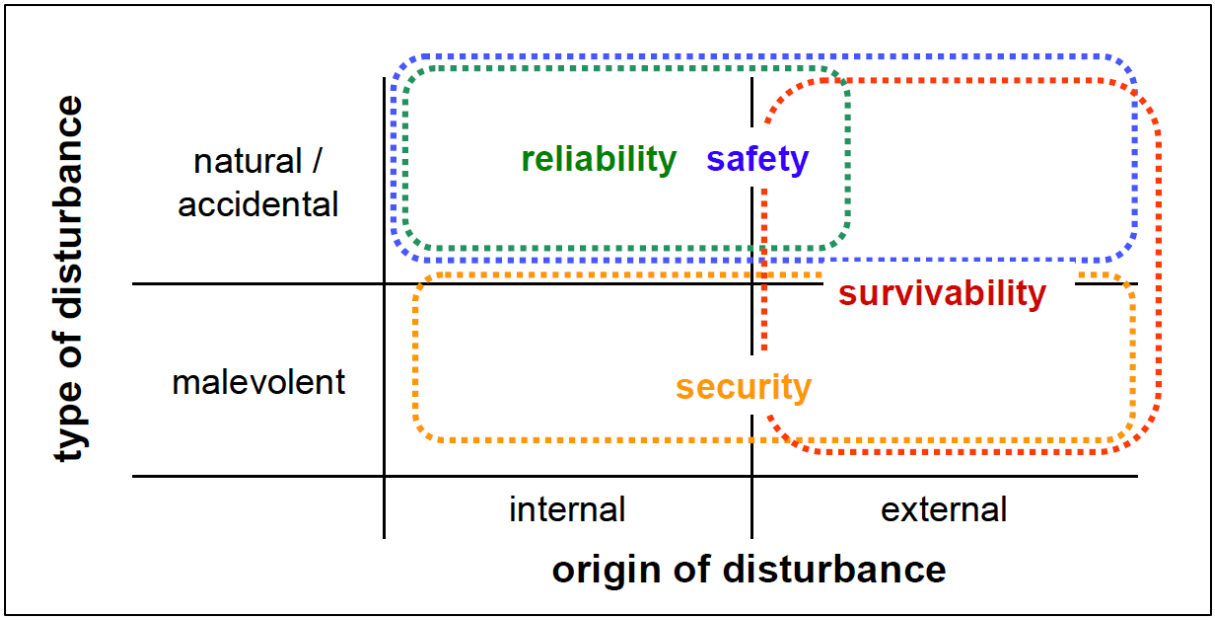

Figure 2: Difference between Survivability and other "-ilities" (Richards, 2009)

Origin of Disturbances. Disturbances can be either internal (endogenous) or external (exogenous) to the system. It is interesting to note, that disturbances between systems within a SoS, are considered internal to the SoS overall. However, these same disturbances would likely be considered exogenous to the individual constituent systems.

Nature of Disturbance. Disturbances can be either natural or artificial. Natural disturbances are those that arise from the interaction of the system with the natural environment. Floating debris and thunderstorms are examples of natural disturbances that could affect the value delivered by a SoS under consideration. Even something as seemingly trivial as a bird striking an aircraft can have serious consequences (Kelly 2009). Artificial disturbances, like missile attacks or policy changes, arise from the actions of external agents.

Intent. (Ellison et al. 1997) classifies disturbances as attacks, failures and accidents. When external agents are involved, it is necessary to consider their intent, in order for the appropriate survivability principles to be applied. Attacks are events, such as a missile attack, caused by intelligent adversaries with malevolent intent. Failures are events caused by deficiencies in the system or in an external entity upon which the system depends whereas accidents are randomly generated events outside the system. However, this classification does not include disturbances generated intentionally by external agents without malevolent intent. An example of this type of disturbance would be government agency raising the threat level. Even if this entity realizes the harm that this disturbance could cause (the DHS, for instance, acknowledges the negative impact of raising the threat level stating that it has "economic, physical, and psychological effects on the nation" (DHS 2010)), it is not done with malevolent intent. Instead, the external agency chooses to exercise this option, when considering objectives and attributes beyond those of the system itself.

Disturbance Duration. Disturbances can vary in duration, from instantaneous events to very lengthy disturbances.

Context Change. While a disturbance is occurring, the context in which the system operates is changes. However, after the disturbance is over, the context may or may not return to what it

was previously. Sometimes, a finite duration disturbance can cause a permanent context change. For instance, while the hijackings of 9/11 were finite in duration, the changes they caused in airline security are still there 10 years later.
The taxonomy of disturbances discussed above is applied to a set of example disturbances in Table 2.

Table 2: Classifying Example Disturbances using Disturbance Taxonomy

Example Disturbance	Origin	Nature	Duration	Context Change	Intent
Lightning strike	External	Natural	Short	Temporary	Accident
Missile attack	External	Artificial	Short	Temporary	Attack
Policy change	External	Artificial	Short	Permanent	Intentional
Sudden increase in boats arriving	External	Artificial	Short	Temporary	Accident
Component failure in vehicle	Internal	Natural	Short	Temporary	Accident
Climate change	External	Natural	Long	Permanent	Accident
Obstacle in path of vehicles	External	Either	Short	Temporary	Accident
Operator error	Internal	Artificial	Short	Temporary	Accident
Biological virus	External	Natural	Short	Temporary / Permanent	Intentional
Changes in system form	Internal	Artificial	Short / Long	Temporary / Permanent	Intentional / Accident
Fuel prices increase	External	Artificial	Long	Permanent	Intentional
Technology improvement	External	Artificial	Short	Permanent	Intentional
Bad communications	Internal	Artificial	Short	Permanent	Accident

Properties that Distinguish Systems of Systems from Traditional Systems

The literature has identified several characteristics of systems of systems that tend to set them apart from traditional, 'monolithic' systems (Jamshidi 2009). Having one or more of these characteristics does not necessarily make a system a SoS. In fact, debating whether a particular system is actually SoS or not, may be moot; what is important is that system designers, architects and analysts recognize whether or not the system under investigation has some of these SoS-like properties, and apply design principles and methodologies accordingly. Some of these characteristics include operational independence, managerial independence, geographical distribution of components, evolutionary development (Maier 1998), multi-functionality (Eisner, Marciniak, and McMillan 1991), distributed authority, abstruse emergence (Boardman and Sauser 2006), internal interoperability and dubious validation (Ellison and Woody 2007).

Component Independence. A component of a system is any entity within a system, whether it is a system itself (referred to as a constituent system) or some other supporting element (such connecting wires). Components have *operational independence* if they can operate outside the system and still produce value, whereas they have *managerial independence* if they actually do operate independently from the other components. Components that operate independently of other components and produce value on their own, would typically be considered constituent systems of the overall SoS.

Distributed Authority. In order for systems to have managerial independence, they need to be able to make decisions for themselves. Thus, many systems of systems tend to have distributed authority, whereas traditional systems are more likely to have central authority. However, distributed authority does not guarantee managerial independence, if decisions are made collectively.

Geographic Separation. Components in traditional systems tend to be more co-located that those in systems of systems. Operational and managerial independence of systems of systems facilitates geographically separated components, where decisions based on local context can occur.

Multi-Functionality. A simple, traditional system is more likely to have a single function or purpose, whereas a system of system is more likely to be multi-functional. For instance, UAV A may be designed to detect targets, UAV B may be designed to take pictures of targets and UAV C may be designed to identify targets. The function of the SoS is to provide situational awareness, which is the aggregate of all functions of its components (detect, photograph and identify targets).

Increased Contextual Diversity. Since components in systems of systems are more likely to be physically separated than those in traditional systems, it follows that they will be more likely to be operating under different environmental conditions. Furthermore, because of managerial independence, components in a SoS are also more likely to be operated with different stakeholder needs and expectations. Therefore, components within a SoS are more likely to be operating under heterogeneous contexts, than components within a traditional system (`Figure 3`).

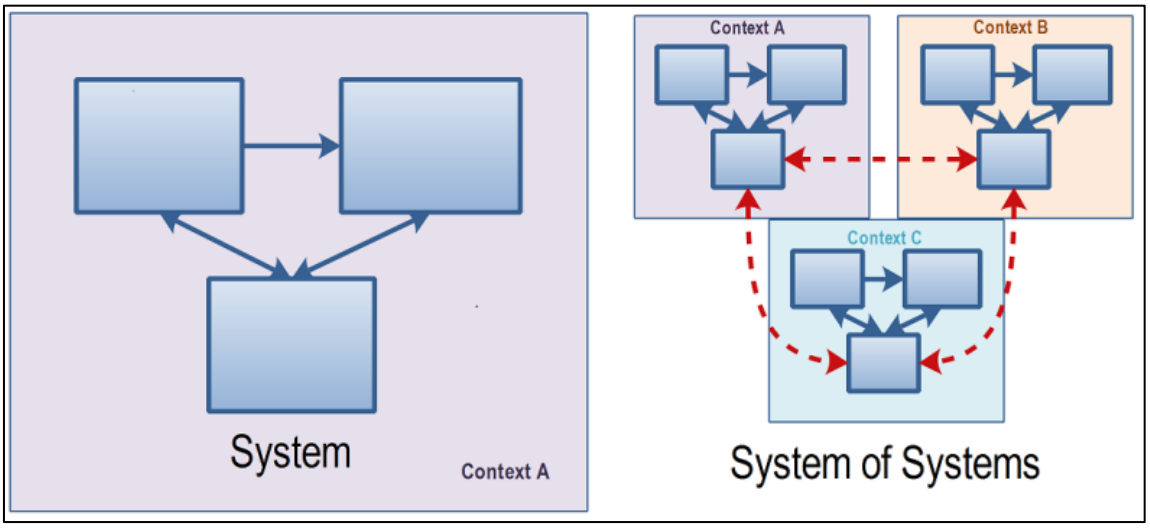

Figure 3: Traditional Systems and Systems of Systems

Decreased System Awareness. Since components in a system of system are often operating under different contexts, they must share the contextual information with each other in a timely manner, for all of the components to have the same system awareness at any given time. In order for that to happen, three things must occur; (1) the important differences in context must be apparent, (2) stakeholders must be willing to share this information (not always the case, particularly if the contextual differences are the stakeholder preferences and policies), and (3) mechanisms must exist for this information to be shared in a timely manner. For these reasons, components within systems of systems that operate under different contexts may be operating

under incorrect or incomplete information about the system itself, than components within traditional systems operating under the same context.

Evolutionary Development. Traditional systems are typically assembled during implementation, before the system is operated. Components of systems of systems, are often added or removed dynamically, during the operation of the SoS, and are considered to be constantly evolving.

Abstruse Emergence. In traditional systems, emergent behavior is often part of the design (or at least expected), and as such, is usually a benefit overall. In systems of systems, particularly those with evolutionary development, emergent behaviors are more difficult to predict and often end up being problematic.

Internal Interoperability. Since traditional systems tend to have more holistic designs with specialized components, interoperability is usually only an issue when interfacing with external systems. Components within a SoS, however, are often constituent systems that must interoperate with each other. With evolutionary development, these constituent systems are often designed and operated independently and newer constituent systems from one supplier must often interface with legacy constituent systems form another supplier. While standards often exist, they are not always strictly enforced and interoperability within the SoS can become a major concern.

Dubious Validation. Due primarily to the evolutionary development of systems of systems, testing and validation becomes increasingly difficult. Components often change and it is not practical to validate each change with every possible permutation of components past, present and future, particularly when the components are designed and operated by different stakeholders. While interoperability standards typically exist, systems of systems are less likely to be held to the same rigorous testing and validation procedures as traditional systems.

Discussion on SoS Properties and Their Impact on Survivability

Reducing Susceptibility. It is important to distinguish intentional disturbances from unintentional, because there are several survivability design principles systems can take to reduce susceptibility by preventing attacks, such as *deterrence*, *prevention* and *preemption*. These principles will be ineffective, however, if the disturbances are unintentional. Since many systems of systems involve a large sociotechnical component (Bjelkemyr, Semere, and Lindberg 2007), they will more likely be impacted by disturbances which are side effects of polices. For instance, a raise in taxes on gasoline may impact a transportation systems ability to survive. A new survivability design principle may be needed where the system somehow influences policy makers away from creating disturbances that will impact its survivability.

A system that has components located in geographically disparate areas and operating under multiple contexts is more likely to encounter some disturbance than one that is co-located and operating under a single context. Suppose the probability of a particular disturbance *d*, over a period of time *t*, for a particular environment (context) *c* is given by p_{dc}. For a traditional system, with all of its components collocated, then the probability p_d of this particular disturbance impacting the system over time period *t* will be p_{dc}. However, for a SoS that has components in *n* separate contexts that are different enough for the risk of disturbance to be independent, then the probability of that disturbance affecting at least one component is given by

$$p_d = 1 - \left(\prod_{c=1}^{n} (1 - p_{dc}) \right)$$

As an example, suppose every hour there is a 1% chance that a thunderstorm will affect any particular location in the world. If two ground control stations are located far away from each other, then the probability that at least one of them is affected by a thunderstorm becomes $(1 - (1 - 0.01)^2) = 0.02$ or twice the probability of a co-located traditional system. Thus, without considering anything else, geographical distribution may make systems more susceptible to disturbances simply because they are more likely to be exposed to multiple contexts simultaneously. On the other hand, geographical distribution reduces susceptibility in a number of ways. By physically separating components, critical components can be located in safer environments. For instance, UAV operators can be located far away from hostile environments where the UAVs themselves operate, following the *avoidance* design principle. Also, operational and managerial independence of the components allows systems to act and react to their environments dynamically. This, coupled with geographical separation of critical and expendable components, facilitates prevention, preemption and deterrence design principles. For instance, an expendable UAV may discover a hostile boat while on patrol and be able to preemptively attack it without waiting for a central authority figure (who may be overwhelmed) to give approval, or allowing it to get into range of the non-expendable ground control stations.

Reducing Vulnerability. Obviously, geographical separation fully adheres to the survivability design principles of *distribution* and *containment*, thereby decreasing vulnerability to local disturbances. If the two ground control stations are separated geographically, then a disturbance that affects one, such as missile strike, may not impact the other.

Endogenous disturbances (i.e. internal to the system), such as component failure, are not survivability issues as they are reliability issues. Naturally, an exogenous disturbance, such as a missile strike, can cause a chain of events to occur that causes failure within a system. Perhaps the biggest drawback of a highly connected, interdependent system is a cascading-failure, where the failure of one component, causes failure in the next, and so on, similar to dominoes. This type of failure is responsible for some of the biggest systems of systems failures, such as the Northeast Blackout of 2003 (Andersson et al. 2005). Although it is often reliability issues that cause the cascading failure to propagate through the system, this is still a survivability issue since it was instigated by an external disturbance. Thus, reliability of a system is part of survivability. However, in systems of systems, the distinction between endogenous and exogenous disturbances is blurred. In traditional systems, most individual components tend not to have an external interface. Thus, the environment only interacts with a limited set of components, or the entire system as a whole. In a SoS, the components are themselves systems and interact with each other through interfaces that are external to the component systems, but internal to the SoS. Thus, the inputs to a system within a SoS are exogenous and therefore related to the survivability of that system, even if the inputs come from other components within the SoS. Thus, the survivability of constituent components is almost a pre-requisite for SoS reliability (unless the system has redundancy built-in)..
The *hardness* of highly-connected systems will likely be reduced, since outputs of a system become inputs to another and therefore potential disturbances. Furthermore, overall reliability within the system may be reduced due to the fact that the connections of systems that have been likely engineered by different companies may not be as interoperable as a traditional, cohesive

system. This means that the *fail-safe* and *failure mode reduction* survivability design principles will likely be harder to reach in a SoS, thereby making it more vulnerable to disturbances.

Multi-function systems support the design principle of *heterogeneity*, since alternate methods of providing value to the stakeholders are possible in the event that a disturbance impacts ones of more of the functions that the SoS provides.
Intuitively, evolutionary development would likely make a system less vulnerable to disturbances, since the system has (in its past) functioned with or without certain components or capabilities. Therefore, it is more likely that a SoS will be able to survive if a disturbance impacts it in such a way that the system still resembles one of its intermediary stages. An example would be that a couple of UAVs can be added or removed from the SoS under consideration at any time and it would still function (although at different performance levels). Thus, if a missile attacks shoots down one of the UAVs, the SoS will likely survive. This most closely follows the principle of *margin*, since there would more likely be extra capability in a SoS than in a traditional system, due to evolutionary development and functional intermediate stages. To deal with the important issue of interoperability, perhaps the success of the Internet is a good place for inspiration. For instance, the robustness principle, known as one of the essential design principles of the Internet (Rosenthal 2010) might be a worthy addition to the set of system design principle for survivability, particularly when applied to systems of systems. The robustness principle, first introduced by Jon Postel in 1981 and also sometimes known as *Postel's Law*, can be summarized as the following:

Be conservative in what you send, liberal in what you receive

In other words, when sending information to other components, always ensure that the component is strictly following standards and protocols. However, when receiving information from other components, always assume that there are errors and try to handle them as well as possible. This principle is referred to as a *defensive posture* by Ellison and Woody (2007).

Increasing System Resilience. Of the original 17 system design principles for survivability, only *repair* and r*eplace* increased the third type of survivability; system resilience. Independence of components and evolutionary development facilitates both of these principles, since the system tends to keep operating independently while the number and configuration of components change. While repairing and replacing components will likely allow systems to recover from finite duration disturbances, they are less likely to restore value delivery in the presence of a permanent context change. For this, a new design principle *adaptation* could be useful. Given enough time, systems that have an evolutionary nature and the ability to operate with a dynamic set of components, will be more likely to be able to change in form and/or operation, in order continue to deliver acceptable value to the stakeholders (Sage and Cuppan 2001).

Perhaps one of the greatest drawbacks to some of the key SoS characteristics, such as managerial independence and geographical separation, is reduced system awareness. Although this problem potentially affects all three types of survivability, it particularly increases system vulnerability by creating internal disruptions, and reduces system resilience, by hindering a timely and coordinated response to an unanticipated disturbance.

Conclusions

When comparing the set of properties that distinguish a SoS from a traditional system and applying the set of design principles for survivability, it appears that systems of systems are more likely to be more survivable, due primarily to the properties of operational and managerial independence, geographical separation, and evolutionary development. This would concur with the notion that loosely coupled systems are inherently more survivable (Jackson 2010). However, reduced system awareness can increase vulnerability and decrease resilience, to the point where overall SoS survivability may actually be worse than a traditional system. Making generalizations about systems of systems is dangerous, as many systems exhibit some properties of both traditional systems and systems of systems. It is not recommended that a new set of design principles for systems of systems be created; rather, the existing system survivability design principles should be modified and augmented to address the differences between various systems. In addition to the two new design principles introduced in this paper, more design principles should be explored. For instance, Huynh et al. (2009) suggest that systems are more likely to be robust if the components are similar. This assertion, which needs to be tested in a systems context, is based on a *similarity principle* in chemistry that which states that mixtures of similar components will have a higher entropy and be more stable than mixtures of dissimilar components (Lin 2008). The next phase of this research is to conduct simulations of systems of systems that are subjected to various disturbances of the proposed taxonomy. The goal will be to compare designs that have or have not used the proposed design principles, and validate some of the hypotheses made in this paper. Following the analysis of the results, and using historical case studies as a reference, existing and new system survivability design principles will be validated and updated accordingly for some of the characteristic properties of systems of systems.

Acknowledgement

The authors gratefully acknowledge funding for this research provided through MIT Systems Engineering Advancement Research Initiative (SEAri, http://seari.mit.edu) and its sponsors.

References

1. Andersson, G., et al. 2005. Causes of the 2003 major grid blackouts in North America and Europe, and recommended means to improve system dynamic performance. In *Power Systems, IEEE Transactions on,* 20 (4): 1922-1928.
2. Backlund, A. 2000. The definition of system. *Kybernetes* 29 (4): 444-451.
3. Bjelkemyr, M., Semere, D., and Lindberg, B. 2007. An engineering systems perspective on system of systems methodology. In *Proceedings of the 1st Annual IEEE Systems Conference 2007*, Honolulu, HI.
4. Boardman, J., and Sauser, B. 2006. System of Systems-the meaning of. Presented at *IEEE / SMC International Conference on System of Systems Engineering 2006*, Los Angeles, CA.
5. Chattopadhyay, D. 2008. A Framework for Tradespace Exploration of Systems of Systems, S.M. Thesis, Department of Aeronautical and Astronautical Engineering, Massachusetts Institute of Technology, Cambridge.
6. Department of Homeland Security. 2010. Homeland Security Advisor System. http://www.dhs.gov/files/programs/Copy_of_press_release_0046.shtm. Accessed March 18, 2011.

7. Dickerson, C.E. 2009. Defense applications of SoS. *Systems of Systems Engineering: Principles and Applications*, Boca Raton, FL: CRC Press.
8. Eisner, H., Marciniak, J., and McMillan, R. 1991. Computer-aided system of systems engineering. Presented *at IEEE Conference on Systems, Man, and Cybernetics*, Charlottesville, VA.
9. Ellison, R., et al. 1997. Survivable network systems: An emerging discipline. Pittsburgh: Carnegie Mellon.
10. Ellison, R., and Woody, C. 2007. Survivability Challenges for Systems of Systems. *News at SEI*, http://www.sei.cmu.edu/library/abstracts/news-at-sei/securitymatters200706.cfm. Accessed March 18, 2011.
11. Huynh, T.V., Tran, X.L., and Osmundson, J.S. 2009. Architecting of Systems of Systems for Delivery of Sustainable Value. Presented at *Second International Symposium on Engineering Systems,* Cambridge, Massachusetts.
12. Jackson, S. 2010. *Architecting Resilient Systems: Accident Avoidance and Survival and Recovery from Disruptions*, Hoboken, N.J.: John Wiley & Sons.
13. Jamshidi, M. 2009. *Systems of systems engineering: principles and applications*, Boca Raton, FL: CRC Press.
14. Keeney, R.L. 1996. *Value-focused thinking: A path to creative decision making*: Harvard Univ Press.
15. Kelly, T.A. 2009. Beware of the birds. In *Potentials, IEEE* 28 (5): 31-35.
16. Lin, S.K. 2008. Gibbs Paradox and Similarity Principle. In *Bayesian Inference and Maximum Entropy Methods in Science and Engineering*, AIP Conference Proceedings 1073. 2008
17. Maier, M. W. 1998. Architecting Principles for System of Systems. In *Systems Engineering* 1 (4): 267-284.
18. Richards, M.G., Hastings, D.E, Rhodes, D.H., and Weigel, A. 2007. Defining Survivability for Engineering Systems. Presented at *Conference on Systems Engineering Research*, March 2007, Hoboken, NJ.
19. Richards, M.G., Ross, A., Hastings, D.E. and Rhodes, D.H. 2009. Survivability Design Principles for Enhanced Concept Generation and Evaluation. Presented at *19th Annual INCOSE International Symposium*, Singapore.
20. Richards, M.G. 2009. Multi-Attribute Tradespace Exploration for Survivability, Engineering Systems Division, Ph.D. Thesis, Engineering Systems Division, Massachusetts Institute of Technology, Cambridge, MA.
21. Rosenthal, D.S.H. 2010. Format obsolescence: assessing the threat and the defenses. In *Library High Tech,* 28 (2):195-210.
22. Sage, A.P., and Cuppan, C. 2001. On the systems engineering and management of systems of systems and federations of systems. In *Information, Knowledge, Systems Management,* (4): 325-345.
23. Westrum, R. 2006. A Typology of Resilience Situations. In *Resilience engineering: concepts and precepts*, Aldershot, England.

Biography

Brian Mekdeci completed a B.A.Sc. (2002) and a M.A.Sc. (2005) in systems design engineering at the University of Waterloo in Canada. Afterwards, he worked at CDL Systems Ltd in Calgary, Alberta as a systems engineer in charge of designing ground control station software for unmanned aerial vehicles. Currently, Brian is researching survivability and methods of operations for systems of systems as part of his doctoral studies at the Massachusetts Institute of Technology.

Adam M. Ross is a Research Scientist in the MIT Engineering Systems Division (ESD) and a cofounder of SEAri. His research focuses on managing unarticulated value, designing for changeability, and dynamic tradespace exploration for complex systems. Dr. Ross received his Ph.D. from ESD in June 2006 and has published papers in the area of space systems design. He has work experience with government, industry, and academia including NASA Goddard; JPL; the Smithsonian Astrophysical Observatory; Boeing Satellite Systems; MIT; and Harvard and Florida State Universities; performing both science and engineering research.

Daniel E. Hastings is a Professor of Aeronautics and Astronautics and Engineering Systems at MIT. Dr. Hastings has taught courses and seminars in plasma physics, rocket propulsion, advanced space power and propulsion systems, aerospace policy, technology and policy, and space systems engineering. He served as chief scientist to the U.S. Air Force from 1997 to 1999, as director of MIT's Engineering Systems Division from 2004 to 2005, and is a former chair of the Air Force Scientific Advisory Board. Dr. Hastings was elected a Fellow of the International Council on Systems Engineering (INCOSE) in June 2007.

Donna H. Rhodes is the director of SEAri and a Senior Lecturer in Engineering Systems at MIT, where she is also a principal research scientist. Her research interests are focused on systems engineering, systems management, and enterprise architecting. Dr. Rhodes has 20 years of experience in the aerospace, defense systems, systems integration, and commercial product industries. Prior to joining MIT, she held senior level management positions at IBM Federal Systems, Lockheed Martin, and Lucent Technologies in the areas of systems engineering and enterprise transformation. Dr. Rhodes is a past president and Fellow of INCOSE, and the 2005 recipient of the INCOSE Founders Award.