Emergent Behavior in Systems of Systems

John S. Osmundson
Department of Information Sciences
Naval Postgraduate School
1 University Circle
Monterey, CA 93943, USA
josmundson@nps.edu

Thomas V. Huynh
Department of Systems Engineering
Naval Postgraduate School
1 University Circle
Monterey, CA 93943, USA
thuynh@nps.edu

Gary O. Langford
Department of Systems Engineering
Naval Postgraduate School
1 University Circle
Monterey, CA 93943
golangfo@nps.edu

Copyright © 2008 by John Osmundson, Thomas Huynh and Gary Langford. Published and used by INCOSE with permission.

Abstract. As the development of systems of systems becomes more important in global ventures, so does the issues of emergent behavior. The challenge for systems engineers is to predict and analyze emergent behavior, especially undesirable behavior, in systems of systems. In this paper we briefly discuss definitions of emergent behavior, describe a large-scale system of system that exhibits emergent behavior, review previous approaches to analyzing emergent behavior, and then present a system modeling and simulation approach that has promise of allowing systems engineers to analyze potential emergent behavior in large scale systems of systems.

Introduction

Recently there has been increased interest in what has been become known as systems of systems (SoS) and SoS engineering. Examples of SoS are military command, control, computer, communications and information (C4I) systems (Pei 2000), intelligence, surveillance and reconnaissance (ISR) systems (Manthrope 1996), intelligence collection management systems (Osmundson et al 2006a), electrical power distribution systems (Casazza and Delea 2000) and food chains (Neutel 2002.), among others.

Much of this current interest in SoS is focused on the desire to integrate existing systems to achieve new capabilities that are unavailable by operation of any of the existing single individual systems. Our focus in this paper follows a similar vein; we are addressing the issue of emergent behavior of an SoS that is formed by combining existing systems into a new larger system to achieve some new operational capability.

System complexity is a challenge to the systems engineering architectural design of many SoS, particularly those that interact with financial systems such as the transportation logistics networks, communications networks, and energy delivery networks (Motter and Lai 2002). These systems are usually comprised of a large number of component systems and subsystems, human operators, and software agents. Developing systems engineering techniques to analyze the effects of system complexity and resulting emergent behavior is essential to understanding
how technologies, policies, and regulations will affect the behavior of complex, evolutionary systems and designing against unwanted behavior.

In this paper we discuss definitions of emergent behavior, discuss an example of a SoS that exhibits emergent behavior, discuss possible approaches to analyzing SoS emergent behavior and the limitations of these approaches, and then present a modeling and simulation approach that is applicable in certain cases to analysis of SoS emergent behavior. We conclude that in cases where the modeling and simulation approach is applicable, it is possible to predict and analyze the effects of SoS emergent behavior.

**Systems of Systems**

A system of systems is described by Maier and Rechtin (Maier and Rechtin 2002) as systems which are operationally independent, managerially independent, evolutionary developed, with emergent behavior and are geographically distributed. Maier and Rechtin go on to define three different types of managerial control: directed, collaborative, and virtual. Directed systems are those in which the integrated SoS is built and managed to fulfill specific purposes. It is centrally managed during long term operation to continue to fulfill those purposes, and any new ones the system owners may wish to address. The component systems maintain an ability to operate independently, but their normal operational mode is subordinated to the central managed purpose. For example, an integrated air defense network is usually centrally managed to defend a region against enemy systems, although its component systems may operate independently.

Collaborative systems are distinct from directed systems in that the central management organization does not have coercive power to run the system. The component systems must, more or less, voluntarily collaborate to fulfill the agreed upon central purposes. The Internet is a collaborative system. Standards are developed for the internet, but there is no power to enforce them. Agreements among the central players on service provision and rejection provide what enforcement mechanism there is to maintain standards. The Internet began as a directed system, controlled by the Advanced Research Projects Agency, to share computer resources. Over time it has evolved from central control through unplanned collaborative mechanisms (Muajumdar 2007).

Virtual systems lack a central management authority. Indeed, they lack a centrally agreed upon purpose for the SoS. Large scale behavior emerges, and may be desirable, but the super system must rely upon relatively invisible mechanisms to maintain it. A virtual system may be deliberate or accidental. A virtual system is controlled by the forces that make cooperation and compliance to the core standards. The standards do not evolve in a controlled way; rather they emerge from the market success of various innovators (Muajumdar 2007).

Boardman and Sauser (Boardman and Sauser, 2006) describe differentiating characteristics of an SoS as: autonomy exercised by the constituent systems in order to fulfill the purpose of the SoS, constituent systems choose to belong to the SoS for greater good, SoS are typically connected dynamically to enhance the SoS performance, and characterized by a diversity of systems. Also, SoS may seem unbounded. The levels of connectivity, platform diversity, and degree of associated interoperability point to the risk of whether the SoS is unbounded. Bounded systems are characterized by centralized data and control, systems and their linkages are known a priori and are specific to the connection and interoperability specified. Unbounded systems are characterized by protocols which provide a loose coupling and are omnipresent and allow for dynamic spontaneous connections (DiMario 2006).
There is not universal agreement on a definition of the term ‘system of systems’, but many definitions have basic thoughts in common. Sage and Cuppan (Sage and Cuppan 2001) describe an SoS as having operational and managerial independence of the individual systems as well as emergent behavior. Other definitions include operational and managerial independence, and geographical separations of the component systems. In this paper we will consider an SoS to consist of separately developed systems that are usually operated by separate organizational entities and are usually geographically dispersed.

**Emergent Behavior**

In addition to many definitions of systems of systems, there are many definitions of emergent behavior. Dyson and George state that: “Emergent behavior is that which cannot be predicted through analysis at any level simpler than that of the system as a whole. Emergent behavior, by definition, is what’s left after everything else has been explained” (Dyson and George 1997). Emergent behavior has also been defined as the action of simple rules combining to produce complex results (Rollings and Adams 2003). While simple rules might be those applied to a single system or to a single system element in an SoS, the resulting collective behavior of the system of systems might turn out to be quite complex and unpredictable.

Parunak and VanderBok (1997) point out that a distributed control approach can produce system of systems behavior that is more complex than the behavior of the individual component systems or system elements. They call such behavior emergent behavior because it emerges from the overall interactions, often in ways not intended by the original designers. Fisher (2006) calls emergent properties “cumulative effects” of the actions and interactions among constituents of complex systems, a constituent being any automated, mechanized, or human participant.

The Wilkipedia (2007) definition states that emergent phenomena can occur due to the pattern of interactions between the elements of a system over time. Emergent phenomena are often unexpected, nontrivial results of relatively simple interactions of relatively simple components. A common theme of these definitions is that emergent behavior is the result of complex interactions. In an SoS the component systems may achieve goals that they couldn’t otherwise by interactions with other component systems. Operators and software agents, considered here as SoS elements, can also interact with component systems as well as with one another. Emergent behavior occurs as a result of these interactions.

The definitions disagree, however, on whether emergent behavior is understandable and, to some degree, predictable. As systems get larger and more complex emergent behavior may become more probable. As systems engineers, can we anticipate emergent behavior, can we design to mitigate undesired effects and encourage desired effects? In this paper we do not attempt to resolve the issue as to whether emergent behavior is always predictable and therefore amenable to systems engineering analysis, but we will restrict ourselves to discussing cases in which emergent behavior may be analyzable.

**An Example of Emergent Behavior in Systems of Systems**

A large scale, complex engineered system of systems is the North American power grid which consists of 680,000 miles of backbone transmission lines and about 2.5 million miles of local transmission lines. An electric power transmission system contains many components such as generators, transmission lines, transformers, and substations. Each component experiences a certain loading each day and when all the components are considered together they experience some pattern of loadings. The pattern of component loadings is determined by the system
operating policy and is driven by the aggregated customer demands at the substations (Chassin et al. 2004). The power system operating policy includes short term actions such as power dispatch. The operating policy seeks to satisfy the customer demands at least cost. Short term customer demand is typically driven by daily and seasonal cycles while long term demand is driven by geographic shifts in population and population changes, development of alternative energy sources, and industry growth and changes.

Events can occur on the power grid that either limit the loading of a component to a maximum or zero the component loading if that component trips or fails. Events that can occur include transformer failure, the probability of which generally increases with loading, and operator re-dispatching to limit power flow on a transmission line to its maximum rating. These events can cause a redistribution of power flow in the network and hence an increase in the loading of other system components, and the events can cascade. If a cascade of events includes limiting or zeroing the load at substations, then a blackout results.

Such systems can undergo non-periodic major cascading disruptions that have serious consequences, such as the collapse of the Canadian system in the Quebec province in 1989 and wide-scale power outages affecting about 50 million people in the eastern US in 1993. The dynamics of North America power grid can create conditions under which major disruptions from a wide variety of sources can occur. Large scale disruptions can be intrinsic to some SoS, especially those that display self-organized criticality (Carreras et al. 2000). A self-organized criticality system is one in which the nonlinear dynamics in the presence of perturbations organize the overall average system state near to, but not at, the state that is marginal to major disruptions.

The requirement of the power grid SoS is to reliably generate and supply electricity, wherever and whenever it is needed on the grid. In order to achieve this, there is a need to continuously balance the generation and load in the power grid, to adjust generation output to match load or reduce load to match available generation. Control actions are limited primarily to adjusting generation output and to opening and closing switches to reconfigure the network. Every action affects all other activities on the grid. The activities of all of the control actions must be coordinated, often across large areas. If not managed properly, failure of a single element can cause the subsequent rapid failure of many additional elements, disrupting interconnected transmission systems over a broad geographic area.

Cooperation between electrical utilities was critical to maintain reliable transmission of electricity. As a result, utility companies formed voluntary control areas which also had the benefit of reducing operating cost by aggregating demand in each of their areas of service. It reduced the possibility of large scale outages because the control areas had better visibility over a larger area of the electricity network.

Deregulation of the electrical power industry in the US was begun in 1992 (FERC 1996) to introduce competition in the market and reduce costs to consumers. This changed the requirement of the industry to not only reliably generate and transmit electricity, but to do so profitably. Under deregulation, profit generation, in addition to reliability requirements, became required in the industry.

Cooperation became more important with deregulation because transmission of electricity between regions increases and requires more careful operation. However, the utilities have different and sometimes, competing, economic interests. This is aggravated by the fact that more than one service provider can be involved in providing electricity service to a customer and, if problems arise, responsibility for the problems might not belong to anyone until a dispute is
resolved. As a result, the electrical power grid SoS that evolved under one set of operating principles, is now adapting to a new set of operating principles, with the emergence of new behaviors that include an increase in grid problems and failures, and unexpected increasing costs of electricity to consumers.

**Approaches to Analyzing Emergent Behavior**

Various approaches have been applied to the analysis of emergent behavior in biological, economic and engineered systems. Reynolds (1987) developed a computer simulation of birds flocking based on simple behavior and interactions of individual birds. The flocking is an example of emergent behavior that could be understood based on simple interactions. Mathematicians at the Santa Fe Institute in New Mexico pioneered techniques that were later used to try to predict movements in trading environments (Waldrop 1993). Industrial equipment manufacturers are applying these techniques to determine how to rapidly adapt manufacturing processes, and telecommunications firms use the techniques to help determine business strategies (Caulkin 2007).

In the field of thermodynamics the macroscopic behavior of complex systems of molecules have been successfully analyzed by considering the form of the system’s entropy function (Chassin et al. 2004). Sergeev, Smith and Foley (2002) have adopted this method to analyze the stability of a complex system by considering the form of the system’s entropy. Success in the field of thermodynamics suggests that equilibrium models can be developed for any complex system. However, it is much more difficult to develop time dependent models of such systems using principles of entropy.

Parunak and VanderBok (1997) consider emergent behavior in distributed control architectures, stating that a population of asynchronously executing processes without central top-down control can exhibit unexpected or “emergent” behavior. Paunak and VanderBok go on to say that nonlinear systems theory can be used to detect and manage such interactions. Thus this type of behavior can be considered to be generated by deterministic interactions among control elements.

If systems engineers are to be successful in analyzing emergent behavior, an approach is needed that allows analysis of dynamic behavior of SoS, with the ability to model and simulate the important aspects of the systems, system elements and, most importantly, the interactions between systems and system elements.

**A System of Systems Modeling Approach**

In order for an SoS to provide more capability than any of the individual systems, the systems have to interact and, therefore, the key to understanding and engineering an SoS, as well as understanding emergent behavior, is clearly defining the system-to-system interfaces. Several recent approaches to modeling complex systems have been developed recently, including that of (Oliver et al. 2007). We have developed an engineering and analysis methodology that aims at providing clear elucidation of interfaces. An SoS systems engineering problem involves analysis of existing and proposed system of systems architectures and analysis of architectures of complex system of systems (Osmundson and Huynh 2005).

A process modeling methodology for performing engineering analyses of an SoS has been developed at the Naval Postgraduate School (Osmundson et al. 2004, Osmundson 2000) over the past several years and has been successfully applied to military systems of systems (Osmundson et al 2006a). This methodology consists of a sequence of analyses, transformations, model
building, and simulations. The steps, which we have applied to a number of US defense systems, are:

- Development of system of systems scenarios and operational architectures
- Identification of system of systems elements and threads
- Representation of operational architectures in a modeling language format (Huynh and Osmundson 2006)
- Identification of system of systems design parameters and factor levels
- Transformation of modeling language representations into executable models
- Application of design of experiments
- Simulation runs and analysis of results

System elements in modeling language representations can be described as executable modeling icons by using modern software applications. This applies to system elements such as people and software agents, as well as hardware elements. These icons can be graphically linked to form models of physically distributed systems of systems. Models of systems of systems can be constructed in a modular manner so that design factors are represented by an association with modeling application objects. System options are represented by rearranging the objects and by varying the object attributes from model to model. Design of experiments guides the development of executable models and the running of simulations.

Interactions that might result in emergent behavior occur at interfaces between systems, between systems and operators and/or between systems and software agents. Following the idea in (Klir 1991) we can consider each of these interface elements as elements that seek to satisfy a goal governed by a set of rules whose inputs are provided through the SoS interactions with the goal seeking element. In some cases the goal-seeking element may have probabilistic behavior and/or may adapt to changing input conditions.

In normal system design the goals and the performance functions of the system and system elements are well defined. In systems engineering we say that the system is designed to meet some functional requirements to specified levels of performance. Now we consider a complex system of systems where SoS elements may include people and software agents whose goals may be the same as, or different from, the original system goals. Unanticipated behavior might then be due to miss-application of the rules by a person. Another source of unanticipated behavior might be due to the fact a hardware element, a person or a software agent correctly applied the rules, but for a set of input conditions that were unanticipated by the system of systems engineers. A third source of unanticipated behavior might be adaptation of a person or a software agent to sets of inputs.

In our approach we model an agent as a rule-based algorithm, with some probability of design error, whose behavior is dependent on its inputs. We model a person as a probabilistic rule-based algorithm whose behavior is dependent on its inputs. We model a system function as a rule-based with some probability of design error, whose behavior is dependent on its inputs.

Figure 1 shows a simple model of a highly abstracted power grid system, consisting of five power generating nodes, three trading agents, three consumer nodes and transmission lines linking the power generating nodes and the consumer nodes. The power generating nodes generate given rates of electricity at initial offering prices unique to each node. The consumer nodes have a rate of demand and initial buying prices unique to each node. The transmission lines have unique capacities and probability of failures that are dependent on their load relative to their capacity. The trading agents seek to maximize their profit by buying from the lowest price generator and selling to the highest bidder. If a generating node is successful in selling an
increment of power, it raises its selling price during the next interval and lowers its price if it was unsuccessful. Consumer nodes lower their buying price if they were successful in purchasing power and raise if they were not successful. The emergent behavior of the power grid can be analyzed by comparing the price and transmission system failure rates to those of the same model in the absence of trading agents, where the system is designed to provide reliable delivery of power.

Figure 1. Representation of a Model of an Abstracted Power Grid System of Systems

In this work only a very limited design of experiments is performed. The power grid is modeled as shown in Figure 1 to represent deregulated operation with trading agents operating normally in order to optimize profit, as well as an excursion case in which one rogue trading agent creates an artificial shortage of electricity at one consumer node in order to artificially raise the cost of electricity at that node and further increase the agent’s profit.

Figure 2 shows the average cost of electricity at node 2 as a function of simulation time when deregulation is in effect. The average cost increases relative to the cost for a regulated power grid when there are no trading agents, and the goal is to supply electricity to consumer nodes in a reliable manner at regulated prices.
Figure 2. Average Cost of Electricity at Node 2 for a Deregulated Power Grid as a Function of Simulation Time

Figure 3 shows the effect of a rouge trading agent. Here a trading agent blocks consumer node 2 for a short time in order to create an artificial shortage of electricity at node 2. The effect is to cause the cost of electricity at node 2 to increase much more than in the case represented by Figure 2.

Figure 3. Average Cost of Electricity at Node 2 for a Deregulated Power Grid as a Function of Simulation Time when a Rouge Trading Agent Blocks Node 3 for a Short Interval

The final measure of grid behavior during regulated operation is shown in Figure 4 where the average number of system failures due to the change in operating policy is shown. Because of the assumptions made in the model, the numerical results do not reflect a realistic percentage but do show a relative increase of about 3% over a regulated grid.
Results show that for even an extremely simple model the results are consistent with the actual behavior of the deregulated North American power grid, and this suggests that such a systems engineering analysis of proposed changes in the grid operating policy would have been valuable in providing quantitative evidence of unwanted emergent behavior in advance of implementing policy changes.

The method described here can be extended to include more realistic and complex agent behavior as well much larger, and more realistic, system architectures. Different types of systems with different emergent behavior issues can be analyzed. For example, the success of failure of web-based systems such as eBay and Napster can be analyzed in terms of attractors that result from number of interactions resulting from such system variables as number of nodes, interaction links, and frequency of interactions.

**Conclusion**

The importance of systems of systems in today’s global endeavors requires that systems engineers develop methods for analyzing emergent behavior, in order to predict favorable and unfavorable consequences and in order to architect SoS to better assure desired results. Modeling SoS in executable form, with emphasis on the interacting elements of SoS, and applying design of experiments to guide the development of executable models and setup of simulation runs shows promise for providing systems engineers with a suitable technique.

**References**


Biographies

**John Osmundson** is an associate research professor with a joint appointment in the Systems Engineering and Information Sciences Departments at the Naval Postgraduate School in Monterey, CA. His research interest is applying systems engineering and computer modeling and simulation methodologies to the development of systems of systems architectures, performance models, and system trades of time-critical information systems. Prior to joining the Naval Postgraduate School in 1995, Dr. Osmundson worked for 23 years at Lockheed Missiles and Space Company (now Lockheed Martin Space Division) in Sunnyvale and Palo Alto, CA, as a systems engineer, systems engineering manager, and manager of advanced studies. Dr. Osmundson received a B.S. in physics from Stanford University and a Ph.D. in physics from the University of Maryland. He is a member of INCOSE.

**Tom Huynh** is an associate professor of systems engineering at the Naval Postgraduate School in Monterey, CA. His research interests include uncertainty management in systems engineering, complex systems and complexity theory, system scaling, simulation-based acquisition, and system-of-systems engineering methodology. Prior to joining the Naval Postgraduate School, Dr. Huynh was a Fellow at the Lockheed Martin Advanced Technology Center, where he engaged in research in computer network performance, computer timing control, bandwidth allocation, heuristic algorithms, nonlinear estimation, perturbation theory, differential equations, and optimization. He was also a lecturer in the Mathematics Department at San Jose State University. Dr. Huynh obtained simultaneously a B.S. in Chemical Engineering and a B.A. in Applied Mathematics from UC Berkeley and an M.S. and a Ph.D. in Physics from UCLA. He is a member of INCOSE.

**Gary Langford** is a lecturer in the Systems Engineering Department at the Naval Postgraduate School in Monterey, California. His research interests include the theory of systems engineering and its application to commercial and military competitiveness. Mr. Langford founded and ran five corporations – one NASDAQ listed. He was a NASA Ames Fellow. He received an A.B. in
astronomy from UC Berkeley, and an M.S. in physics from Cal State Hayward. He is a member of INCOSE.