December 2012 | Volume 15 Issue 4

A PUBLICATION OF THE INTERNATIONAL COUNCIL ON SYSTEMS ENGINEERING

INCOSE
International Council on Systems Engineering

INSIGHT

What's Inside

What's Inside

SPECIAL FEATURE

Engineering Health-Care Systems

Syntell welcomes you to the:

Scandinavian Winter School Week

Imagine spending a week in a first class educational environment together with internationally recognized lecturers from both Industry and Academia and highly motivated fellow students.

We promise you an intense learning experience mixing interactive and challenging studies with networking activities in an inspiring Nordic winter environment. Join more than 500 fellow students who have sharpened their competencies over the last 14 years and become a Syntell Training Alumni!

January 13-18, 2013
Tällberg, Sweden

Last day for registration is December 23, 2012!

During this exceptional course week we will help you to improve your professional skills and to extend your network! Together with us you will meet fellow students from industrial sectors such as aerospace, automotive, defense, transport, and utilities.

The Scandinavian Winter School week is a good opportunity to establish a shared view within your project team, your department, or your customer-supplier relationships. Both courses combine solid theory with proven engineering practice explained using real-world examples.

The venue for the Scandinavian Winter School Week is the authentic village of Tällberg in Dalecarlia county in central Sweden!

- **Systems Engineering and Project Management**

  This course gives you the essentials in the integration of project and systems engineering management. Focal points are planning, management and the execution of complex industrial projects. Head lecturer is Dr Kevin Forsberg, INCOSE fellow, ESEP, and a leading force behind the Vee model. The concepts presented in the course are consistent with ISO/IEC 15288, and the INCOSE Systems Engineering Handbook 3.2.2, and INCOSE's Systems Engineering Professional (SEP) Certification program

- **System Safety Management**

  Our society increasingly depends on safe technology. This course provides you with an in-depth insight into techniques and methods applicable for the development of safety critical systems comprising hardware, electronics, and software. Examples and case studies come from domains such as aerospace, rail, automotive, defence, offshore, medical technology, and nuclear industry. Head lecturer is Dr David Pumfrey from HISE, High Integrity Systems Engineering in the University of York.

More information
and registration
training@syntell.se
www.syntell.se.

Syntell AB is a leading Scandinavian provider of consulting and training, supporting clients in the design, acquisition, and life cycle management of complex systems. Syntell Training is a business area of Syntell offering clients advanced competence development for professionals within Engineering Management, Systems Engineering, Logistics Engineering and System Safety.

Syntell AB. PO Box 100 22, SE-100 55 Stockholm, Sweden. Tel +46(0)8 660 02 80 Fax +46(0)8 660 09 65. info@syntell.se, training@syntell.se. www.syntell.se

INSIGHT

Publication of the International Council on Systems Engineering

Chief Editor	Bob Kenley
insight@incose.org	+1 260 460 0054
Assistant Editor	Andrew Cashner
andrew.cashner@incose.org	
Theme Editors	
bradley.peck@incose.org	Brad Peack
meaghan.oneil@incose.org	Meaghan O'Neil
Advertising Editor	Christine Kowalski
advertising@incose.org	+1 858 541 1725
Layout and Design	Chuck Eng
chuck.eng@comcast.net	+1 206 364 8696
Member Services	INCOSE Administrative Office
info@incose.org	+1 858 541-1725
On the Web	http://www.incose.org
Article Submission	INSIGHT@incose.org

Publication Schedule. ***INSIGHT*** is published four times per year. Issue and article/advertisement submission deadlines are as follows: ***April 2013*** issue – 15 February; ***July 2013*** issue – 15 May; ***September 2013*** issue – 8 July; ***December 2013*** issue – 15 October
For further information on submissions and issue themes, visit the INCOSE website as listed above.

Advertising in *INSIGHT*. Please see http://www.incose.org/ProductsPubs/periodicals/insight.aspx or e-mail advertising@incose.org.

Subscriptions to *INSIGHT* are available to INCOSE members as part of their membership. Complimentary copies are available on a limited basis. Back issues are available on the INCOSE website. To inquire about membership or to order a copy, contact Member Services.

 ISSN 2156-485X (print) ISSN 2156-4868 (online)

Who are we? INCOSE is a 8000+ member organization of systems engineers and others interested in systems engineering. Its mission is to share, promote, and advance the best of systems engineering from across the globe for the benefit of humanity and the planet. INCOSE charters chapters worldwide, is sponsored by a corporate advisory board, and is led by elected officers and directors.

From the President

John A. Thomas, ESEP, john.thomas@incose.org

Greetings to my fellow INCOSE colleagues. I have received multiple requests to share the key themes from my closing speech at the 2012 International Symposium. So, I thought I would do just that in my note to you today. I focused on two major themes during that speech. One was key business imperatives for our organization. The second was key messages from the executive-summit meeting that I hosted as president.

I believe there are three key business imperatives for our organization. My focus has been laser-sharp on these imperatives, first as our president-elect, and now as president of INCOSE.

Business Imperative no. 1. I believe it is in our best interests to advocate and promote:

- The system engineer as a multidisciplinary leader
- The systems engineering discipline as a critical tool in the toolbox of a systems engineer and of those who have systems problems
- The value of the well-trained system engineer—skilled in both the science of system engineering and the art of leadership

Business Imperative no. 2. To achieve our mission we must increase INCOSE's influence on worldwide systems issues. To increase our influence, we must deepen our leadership connections and form partnerships with sister organizations. INCOSE's relationships with these sister organizations mirror the relationships we have as system engineers with those whom we work with on a daily basis. There are too many examples to share a complete list. Additionally, the list will have to be prioritized. But a few include organizations involved with these domains:

- Program management
- The engineering subdisciplines: mechanical, electrical, civil, chemical, computer science, software, and others
- Safety and cyber security
- Reliability and human factors
- Test and evaluation; costing; and acquisition

Business Imperative no. 3. To achieve the first and second imperatives we require additional resources to implement the thought-leadership agenda of our mostly volunteer organization. The breadth of increased resources includes these changes:

- Modernization of our information technology to connect our distributed membership base and enable them to communicate and collaborate
- The addition of professionals to support the operation and planning within our organization—and maintain day-to-day relationships and execution of joint agendas with sister organizations

The second major theme of my speech was a summary of the executive summit. The summit is a forum each INCOSE president has hosted since its inception under Past President Heinz Stoewer. The purpose of the summit is to (1) raise the awareness and value of INCOSE to government and industry, (2) expand INCOSE relationships with senior members of those organizations, and (3) tap the diversity of powerful minds and wisdom of different experiences.

There was a remarkable level of senior perspective sitting in on this day-long dialogue in Rome. Our guests included the

Honorable Michael Chertoff, former United States Secretary of Homeland Security; Dr. Terry Cooke-Davies, chair of Human Systems International Ltd; Professor Andrew McNaughton, technical director of High Speed 2 (HS2) Limited; and Meg Selfe, IBM's vice president of complex and embedded systems. Additionally, President-Elect David Wright; Technical Director Jean Claude Roussel; Director for Strategy Ralf Hartman; and Managing Director Holly Witte were with me representing INCOSE.

The 16-page document of the executive summit discussions, "Pathways to Influence," can be found on our INCOSE website. I emphasized three points at the closing plenary from this meeting. First, the most important skill of a system engineer is the skill to influence decisions. The metric for measuring the power of this skill is a question: "Can my system engineer be put in front of the corporate board of directors?" Second, the world's greatest challenges need the power of systems thinking and engineering. The engineer must possess a systems perspective that goes beyond the technical dimension of the problem. Third, we need to work to evolve technologists' mindset of their career options. That mindset needs to be shifted from a trade between the "technical track" versus the "management track" to a journey of learning both functional and leadership skills that support technical and management roles as assignments evolve.

INCOSE
International Council on Systems Engineering

INCOSE PAST PRESIDENTS

Samantha Robitaille, 2010 – 11
Pat Hale, 2008 – 09
Paul Robitaille, 2006 – 07
Heinz Stoewer, 2004 – 05
John Snoderly, 2002 – 03
John Clouet, 2001
Donna H. Rhodes, 2000
Ken Ptack, 1999
William W. Schoening, 1998
Eric C. Honour, 1997
V. A. (Ginny) Lentz, 1996
James Brill, 1995
George Friedman, 1994
Brian Mar, 1993
Jerome Lake, 1992

Promote INCOSE

To obtain materials to promote INCOSE in the workplace and at events, such as regional conferences and chapter meetings, contact the INCOSE Administrative Office:

info@incose.com
+1 858 541-1725, or access the INCOSE website at www.incose.org

7670 Opportunity Road
Suite 220
San Diego, CA 92111-2222

We supply INCOSE table signs, promotional items, and informational materials.

Corporate Advisory Board – Member Companies

Air Force Center for Systems Engineering
Airservices Australia
Alliant Techsystems
Analytic Services-Applied Systems Thinking Institute
Astrium, an EADS Company
ATKINS
BAE Systems
Beihang University
Boeing Commercial Airplane Co.
Boeing Defense, Space & Security
Boeing Defense, Space & Security – East
Booz Allen Hamilton Inc.
C.S. Draper Laboratory, Inc.
Carnegie Mellon University Software Engineering Institute
Cassidian
Cranfield University
Cummins, Inc.
Defense Acquisition University
Deloitte
Deputy Assistant Secretary of Defense for Systems Engineering, US Department of Defense
DRS Technologies, Inc.
EADS N.V.
Exelis
Federal Aviation Administration US
Ford Motor Company
General Dynamics
General Electric
George Mason University
Honeywell International
IBM Corporation
JAXA (Japan Aerospace Exploration Agency)
Jet Propulsion Laboratory
Johns Hopkins University
Keio University
L-3 Communications
Lockheed Martin Corporation
Los Alamos National Laboratory
ManTech International Corporation
MAP systeme
Massachusetts Institute of Technology
Medtronic, Inc.
Missouri University of Science and Technology
Mitsubishi Electric Corporation
Nanyang Technological University
National Aeronautics and Space Administration
National Geospatial – Intelligence Agency
National Reconnaissance Office
National University of Singapore
Naval Postgraduate School
Naval Surface Warfare Center – Dahlgren Division
Northrop Grumman Corporation
Orbital Sciences Corporation
Pacific Northwest National Laboratory
Proctor & Gamble
Project Performance International
Raytheon Corporation
Rockwell Collins, Inc.
Rolls-Royce
Saab AB
SAIC
Sandia National Laboratories
Scitor Corporation
SELEX Sistemi Integrati SpA
Serco – NA
Siemens
Singapore University of Technology and Design
SRA International
Stevens Institute of Technology
Swedish Defence Materiel Administration
TASC, Inc.
Tectura Corporation
Thales
The Aerospace Corporation
The MITRE Corporation
The SI
The University of New South Wales, Canberra
UK MoD
United Technologies Corporation
University of Maryland
University of South Australia Defense and Systems Institute
University of Southern California
University of Texas at El Paso
US Army ARDEC
US Army TARDEC
Vitech Corporation
Volvo Construction Equipment
Woodward Inc
Worcester Polytechnic Institute

Theme Editors' Introduction

Engineering Health-Care Systems

Brad Peck, bradley.peck@incose.org; **and Meaghan O'Neil**, meaghan.oneil@incose.org

Each article in this issue provides insight into the challenge and opportunity of engineering health-care systems, as we move from the biological interface up to the societal level.

This issue of ***INSIGHT*** takes us on a journey across the full scale of the systems engineering application in health care. The topics covered range from the microscopic biological interface where the medical device interfaces directly with the body, to the point of care where the patient, health-care professional, and medical-device technology combine in acute care. They take us from the facility level where health-care staff treat many patients over time, to the regional level where a multiple-facility system treats a population, and finally to the societal level where health care must comply with the laws and regulations of the land.

The application of system engineering in health care is progressing on each of these levels, driven by the forces of medical science and aided by innovative technology: there are new developments in sensor technology, integration of digital technology (such as mobile computing and wireless connectivity), system integration and interoperability, and the automation of routine tasks and those prone to human error. These advances are driven by the increased need for economic reform and, most importantly, by ever-rising expectations for safety, quality, and effectiveness.

Each article in this issue provides insight into the challenge and opportunity of engineering health-care systems, as we move from the biological interface up to the societal level. Our contributors demonstrate that the complete solution for continued progression of health care will require system integration by multiple parties: scientists, medical-device manufacturers, health-care providers, and regulatory bodies.

Platforms for Engineering Experimental Biomedical Systems

We begin by zooming in to the microscopic level, to look at how systems engineering is affecting the biological interface of the health-care system. The authors are Matthew Wosteller (a graduate student in systems engineering at the Institute for Systems Research at the University of Maryland), Mark Austin (an associate professor in the Department of Civil and Environmental Engineering at the University of Maryland), Reza Ghodssi (the director of the Institute for Systems Research and director of the Microelectromechanical Systems Sensors and Actuators Lab in the Department of Electrical and Computer Engineering at the University of Maryland), and Shah-An Yang (a postdoctoral associate in systems engineering at the Institute for Systems Research at the University of Maryland). This paper is a great reminder that whereas there is much attention these days on the interoperability and usability of the health-care system as evidenced in the next several papers in this issue, the biological interface continues to be a relevant and fundamental component of systems engineering application in health care. This paper offers a method for modeling and integrating the stochastic nature of biological components into a larger system of medical or experimental devices.

Clinical Engineering: A Systems Focus on the Point-of-Care

Rick Schrenker, systems engineering manager for the Department of Biomedical Engineering at Massachusetts General Hospital in Boston, Massachusetts (US), brings our attention to the point of care in a modern health-care environment. His interesting account of the recent history of clinical engineering describes how systems engineering has evolved: at the beginning engineers designed relatively simple medical devices, which were implicitly integrated in the mind of the clinician, whereas now engineers are integrating the systems explicitly through communication and information technology. Rick demonstrates that health-care systems have become so complex that system solutions will require collaboration across the medical-device industry.

Applying Systems Engineering to Improve Extracorporeal Membrane Oxygenation Therapy

Drew Pihera and his coathors offer a real-world example of systems engineering in Extracorporeal Membrane Oxygenation Therapy (ECMO). Drew is a research scientist at the Georgia Tech

Research Institute in Atlanta, Georgia (US), and writes in collaboration with Matt Paden (clinical director of apheresis and associate director of pediatric and adult ECMO at Children's Healthcare of Atlanta, Georgia, US, and with Tommer Ender (senior research engineer at Georgia Tech Research Institute), Brian Taylor (a graduate research assitant at Georgia Tech Research Institute), Andrew Lopez (lead electrical engineer for a United States Department of Defense contractor in Huntsville, Alabama, US), Nicholas Bollweg (research scientist at Georgia Tech Research Institute), and Scott King (project manager for The Home Depot). They provide a compelling example of how modern medical-device systems have become individually complex and together integrate into a situation that demands a more elegant system solution.

Human Systems Integration in Next-Generation Expeditionary Medical-Treatment Facilities

One great way to gain insight into the fundamental workings and challenges of a system is to study it under stress. Expeditionary medical-treatment facilities offer such a case study for the health-care system. They are deployed rapidly in austere conditions and in an emergency or crisis situation. Dennis Folds, chief scientist at the Georgia Tech Research Institute, provides an in-depth analysis of the human–system interface for expeditionary medical-treatment facilities. He identifies the most pressing issues and recommends system-design attributes for potential solutions.

The Facility Location for Emergency Response: A Multi-Objective Approach

Ivan Hernandez (a graduate student in systems engineering at the Stevens Institute of Technology), and Jose Ramirez-Marquez (an associate professor at the Stevens Institute of Technology), model strategic deployment of temporary emergency units in the event of an urban biohazard outbreak. The goal of these units is to dispense medication as efficiently and effectively as possible. The authors model the number, capacity, and location of temporary emergency units against population density to analyze the overall cost and effectiveness of the response. Ivan and Jose use multi-objective optimization techniques to simulate the effects of different system-design solutions on cost (minimization of number of facilities, minimization of unused excess capacity) and effectiveness (unmet demand). This optimization is becoming increasingly important for all aspects of health care.

A Systems Approach to Medical-Device Compliance with IEC 60601-1:2005

Finally, Chad Gibson, a systems engineer with the Battelle Memorial Institute's Health and Life Sciences Medical Device Solutions group, provides an overview of the recent changes contained in the third edition of the standard IEC 60601-1:2005, *Medical electrical equipment—General requirements for basic safety and essential performance*. He shows how systems engineering can play a central role in enabling medical-device manufacturers to comply with the standard.

These articles are intended to provide a real-world sample of the current challenges for systems engineering in health care at every level of the system. This domain provides ample opportunities for systems engineering practitioners to address problems that are relevant to everyday life, and are challenges vital for the well-being of our societies. The Biomedical Working group of INCOSE will be meeting in Jacksonville, Florida (US), 26–29 January 2013, as part of the International Workshop. At this meeting we will continue to work on current projects as well as review and update the working group's charter. We will further develop our project list to maintain our focus on today's system engineering challenges in health care. All INCOSE members who would like to participate are invited. In addition to IW 2013 the workshop, we are also planning programming on the biomedical domain for the 2013 International Symposium, and we invite all interested INCOSE members to attend and participate in these sessions as well.

Platforms for Engineering Experimental Biomedical Systems

Matthew Mosteller, matthew.mosteller@incose.org**; Mark Austin,** mark.austin@incose.org**; Reza Ghodssi, and Shah-An Yang**

We have shown that the implementation of a platform for engineering experimental biomedical systems can bridge the knowledge gap between biologists and engineers and ensure more successful system development.

Biomedical systems designed for experimental purposes are a vital aspect of today's medical field, from bench-top systems driving advances in biological science to bedside point-of-care devices in the clinical realm. Devices aiding medical researchers in advancing the science and knowledge of physiological processes allow for the continued development of new medicines and treatment methods. Similarly, devices that are capable of providing accurate diagnoses and prognoses of patients are necessary if this developing knowledge is to help clinicians improve the health and safety of future generations.

The difficulty in developing systems for biomedical assays is complicated immensely by the variant nature of biological systems (Endy 2005, 450). The growth of living organisms is dependent upon a large number of factors unique to each system, including physiological processes, genetics, and environmental conditions. Thus, the same set of system inputs does not always result in the same set of outputs, making the design, validation, and verification of biomedical devices exceedingly difficult. Furthermore, systems designed for experimental purposes in the biomedical field are becoming progressively more technical (Csete and Doyle 2002, 1664). Researchers are now interested in biological processes at the molecular level in an effort to treat ailments at their source, while clinicians desire tools capable of faster, more accurate, and less invasive patient analysis. Due to this added complexity, the development of biomedical systems is becoming increasingly difficult and costly, since current methods for system-level design are not capable of evaluating the highly stochastic properties of biological components (Endy 2005, 450). Looking forward, new methods of designing experimental biomedical devices are needed if advances in medical science and treatment are to maintain or accelerate their current pace. The knowledge disconnect between biological and engineering domains only aids in further compounding this design problem (Endy 2005, 451). Due to the complex nature of living systems, extensive knowledge of the biological component in biomedical applications is typically limited to specialist biologists and clinicians (Oltvai and Barabasi 2002, 763).

A good case study for engineering experimental biomedical systems is in the treatment of bacterial biofilms. The growth of bacterial biofilms such as that shown in figure 1 has been linked to as much as 65 percent of all microbial infections in the human body (Potera 1999, 1837). Biofilms are complex communities composed of communicating groups of bacteria, shown in green and red in figure 1, and an extracellular matrix, shown in blue (Costerton, Stewart, and Greenberg 1999, 1318). The presence of the extracellular matrix limits molecular diffusion within the biofilm, while bacterial gene exchange in the biofilm structure promotes the development of antibiotic resistance (Costerton, Stewart, and Greenberg 1999, 1319; Donlan 2001, 277). As a result, bacterial biofilms often require 500–5000 times the concentration of antibiotics for effective treatment compared to bacterial suspensions, making them of great interest in public-health fields (Costerton et al. 1994, 2803). Such communities of microbes display naturally stochastic growth characteristics. The difficulty of predicting their development is therefore a limiting factor to design engineers who are pursuing new methods of treating or investigating these biological systems. Thus, while

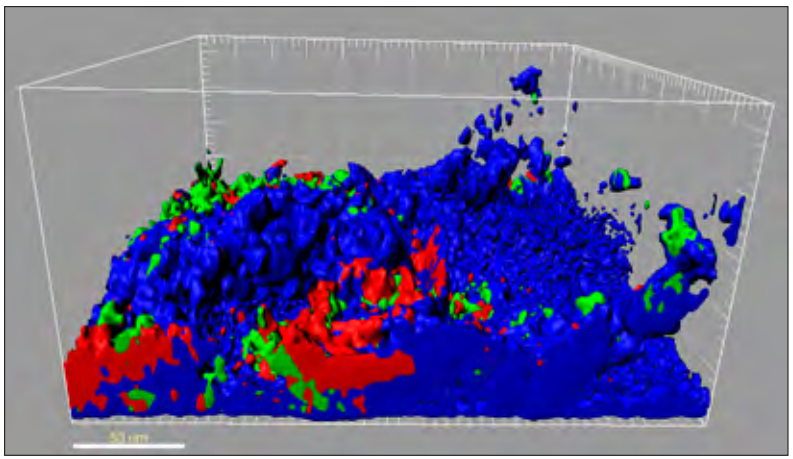


Figure 1. Surface reconstruction of a bacterial biofilm grown in a microfluidic device, showing the highly variant nature of commonly studied biological systems

biologists or clinicians may understand the intricacies of the biological system but not the technologies required to address their application, the design engineer may understand the relevant technological aspects but lack the clinical background to efficiently apply this knowledge.

To address this problem, design techniques must implement a method to enable validation and verification of system performance in the context of highly stochastic biological components, thereby assimilating the biological and engineering domains (Endy 2005, 451). Drawing upon the capabilities of systems engineering tools to model systems in the design phase, the development of platforms for engineering experimental devices is a large step towards producing more effective biomedical systems. Figure 2 presents the method by which these platforms allow for the integration of biological and engineering system domains. The application space defined by the clinician or biologist provides the necessary knowledge for the engineer to model the operation of stochastic biological components. The developed platforms then allow for the integration of biological models with potential system architectures to create an overall design space that can effectively address the system requirements. In order to capitalize upon the added capabilities of such a technique, two key tenets of this research are that (1) engineers must develop methods to succinctly model a breadth of biological systems and (2) these models must be able to integrate with system-level models capable of describing the performance of the entire engineering system. Current methods and techniques for experimental biomedical device development simply are not capable of such full-system modeling.

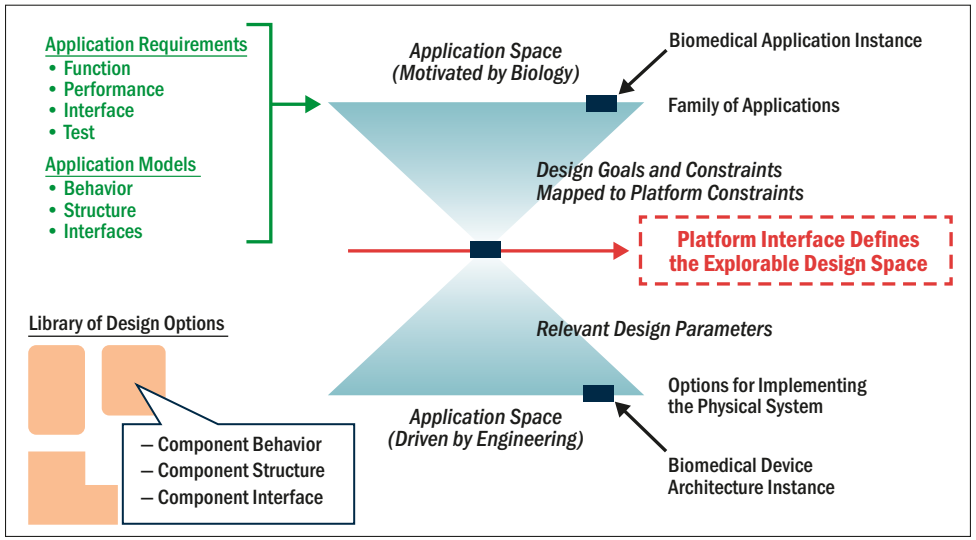


Figure 2. The design space is defined by (1) an application space driven by biology and (2) an architecture space driven by engineering.

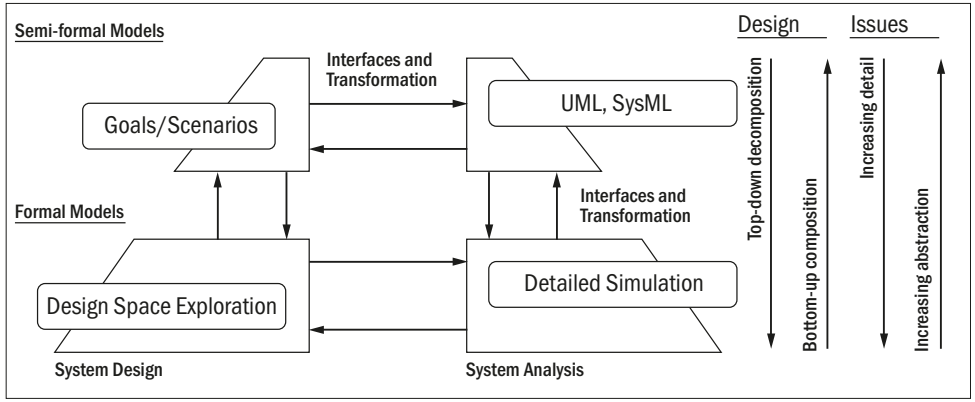


Figure 3. Abstraction as a tool for the design of biomedical systems with integrated models of stochastic biological components

In order to address the limitations of current design methods for experimental biomedical systems, this article presents platforms for the modeling, validation, and verification of device systems that contain highly stochastic biological components. By integrating models of biological systems with those of physical engineering systems, one can obtain a set of potential architectures that satisfy the requirement specifications of the application. Such models can aid in the analysis of biomedical systems intended for applications in medical science, where the stochastic elements are the biological components themselves. The models can also help with systems for patient diagnosis and prognosis, in which the stochastic elements are the physiological responses of patients to a particular assay.

By successfully implementing such platforms, device designers and engineers can ensure that results obtained from experimental tests are trustworthy representations of the biological system's development. These same techniques can also prevent unstable operation of the final system architecture by enabling the early detection of design flaws that would be otherwise unforeseeable using traditional design methods. Figure 3 shows ways to implement these concepts at various levels of abstraction.

Semi-formal models of the proposed system architecture, using such modeling languages as UML and SysML, can provide engineers with a high-level understanding of system performance, thus aiding in more efficient and cost-effective redesign, validation, and verification. These semi-formal models are supported at lower levels of abstraction through detailed simulations of the system, including components to embody the stochastic biological elements. Integrating these stochastic components with well-defined physical systems enables researchers to place more confidence in the experimental testing of biomedical devices than they could previously. The benefits of this approach are far-reaching: engineers, biologists, and clinicians can work together to develop devices that are best suited

for their applications, and thus most beneficial to clinicians, patients, and medical researchers.

Experimental Biomedical Systems

Experimental Processes. A typical experimental process utilizing a device architecture is shown in figure 4. The researcher or clinician begins with a hypothesis about the subject that is developed from prior data or patient symptoms (Tomlin and Axelrod 2007, 336–339). For a medical researcher or systems biologist, this hypothesis may involve a parameter or process that the experiment is intended to verify. Examples commonly include a metabolic process, the effects of a compound on a biological system (such as a candidate drug), or verification of the unique characteristics of a particular organism. For the clinician, a hypothesis may involve a patient diagnosis or prognosis, or may be geared toward determining an effective treatment for a patient's verified medical condition. With this hypothesis in place, the researcher or clinician begins an experiment under ideally controlled conditions. At the conclusion of the established assay, the researcher or clinician inspects the outcome to determine if the test was successful or if alterations or repetition of the experiment is required. Due to the highly stochastic nature of biological systems, such a feedback process is common in order to verify experimental results. The goal of the design engineer is to develop device systems that can aid in reducing the number of iterations needed to achieve a required level of confidence in the result. This is especially important in clinical applications, due to the patient discomfort often associated with invasive testing (for instance, prick tests to determine skin allergies). Similarly, current medical research often utilizes high-cost, low-throughput methods of testing, giving strong motivation for the development of methods to limit the number of iterations needed to verify an experiment.

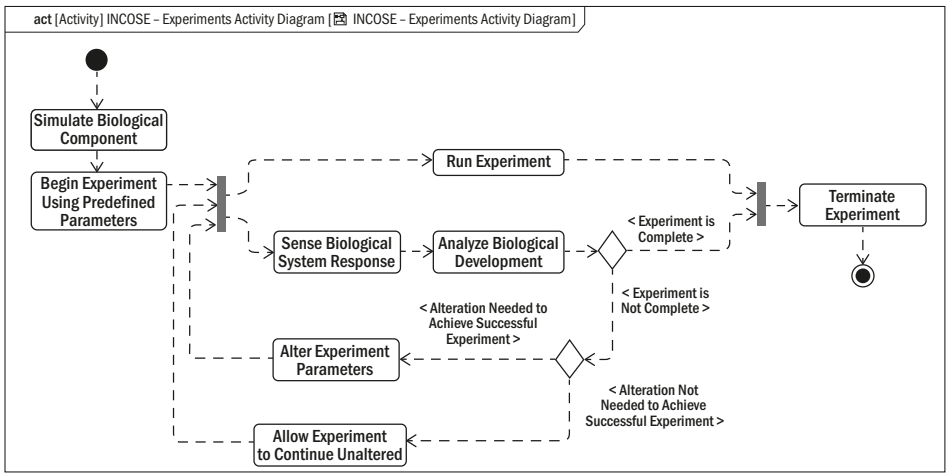


Figure 4. SysML activity diagram showing the process flow of typical biomedical experiments

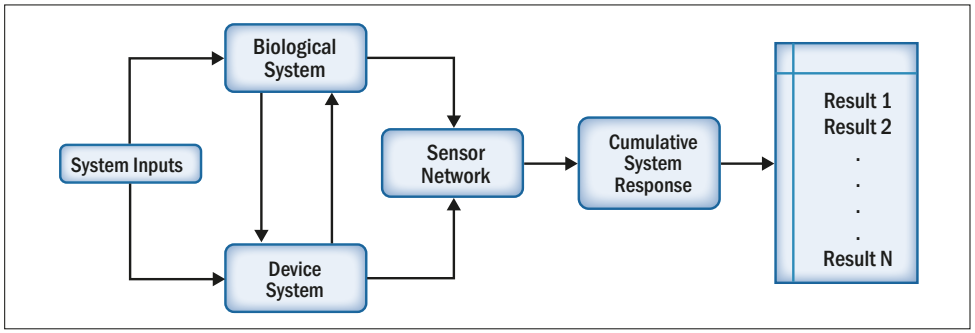


Figure 5. High-level system architecture of biomedical-device performance

In order to break from the current limiting approach to biomedical-device development, new techniques are needed to aid in the maturation of new device systems. The novel platforms presented here are a first step towards such methods, which can increase the overall efficiency of both device development and the operation of the devices themselves by optimizing the interactions of biological elements with the physical system.

Architectures for Experimental Biomedical Systems. While the physical structure of biomedical devices is diverse and typically suited to the needs of the particular application, most systems can be abstracted to the system architecture shown in figure 5. System inputs are typically comprised of a number of different domains, including environmental conditions and actuation or application conditions (that is, what is done to the biological system during the experiment). Depending upon the requirements of the assay, the physical device system can take any number of forms but will typically have three distinct structural elements: (1) a way to contain or integrate with the biological system or sample, (2) a way to control experimental conditions, and (3) a way to integrate with a sensor network for detection. The sensing mechanisms utilized for experimental devices also vary depending upon the application, though they typically aim to optimize a trade-off between minimal invasiveness and achieving the required detection limit and sensitivity of the application. The cumulative effect of the physical system's interactions with the biological element results in a set of potential experimental results, each having a unique probability of occurrence. These probabilities are dependent upon the stochastic biological system, providing at the simulation level a range of statistically relevant outcomes that can be used to confirm experimental results. Figure 6 provides a high-level implementation of the system elements and their interactions at the component and subcomponent levels.

Most biomedical devices are constructed through a similar architecture, providing strong support for the development of generalized platforms for experimental device engineering. The platforms discussed subsequently exhibit a

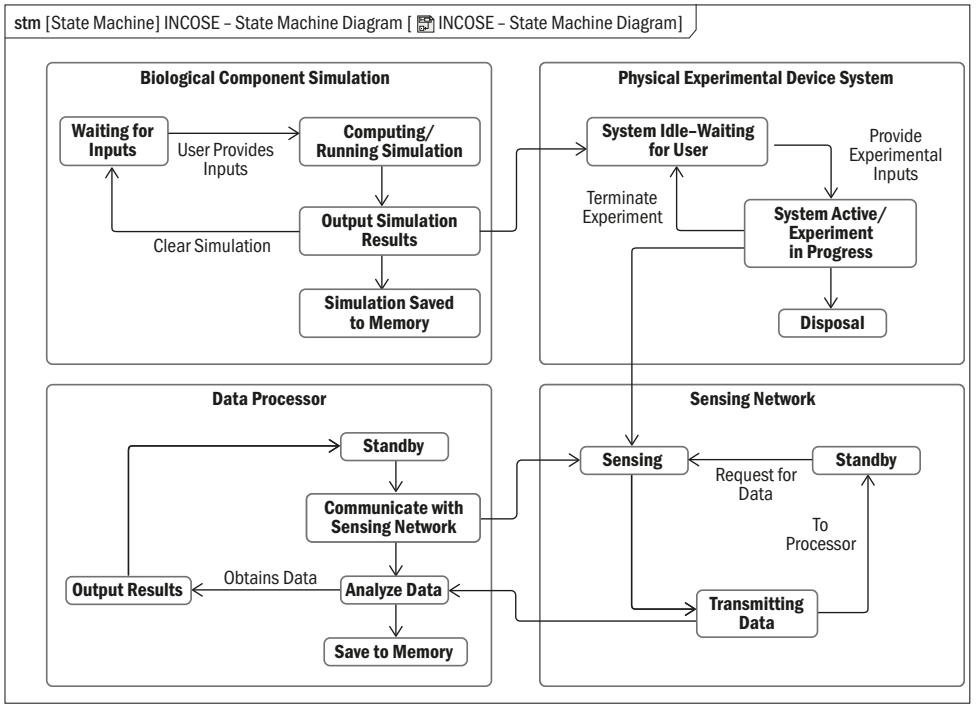


Figure 6. SysML state machines of biomedical-device components

flexible structure that can be adapted to numerous applications in the biomedical field, thus expanding the scope and influence of this work. The development of libraries of components to represent physical system and sensory network elements would aid in the efficient development of new devices and the adaptation of existing devices to new application areas. Additionally, formal platforms that can integrate such libraries with models of the stochastic biological components enable full-system modeling. This modeling can effectively aid efficient and proper design, validation, and verification of biomedical systems. Implementation of such a platform using existing systems languages like UML and SysML takes advantage of the mature properties of these tools, where implementing extensions to other modeling domains is a well-established practice. A tool for the succinct mathematical modeling of stochastic biomedical components would be such an extension of this platform.

Mathematical Modeling of Biological Systems

In order to enable the analysis of biomedical-device performance, engineers and designers require tools that can accurately model the development of biological-system components. These models must be able to simulate the development of the biological system with time, as well as predict changes of the biological system due to experimental conditions. In doing so, these models can then be integrated with higher-level models of the overall experimental device to complete the platform architecture. A number of modeling methods exist for biomedical systems, and these methods differ in their modularity, implementation, and overall accuracy. In this case we focus on one particular method, Markov chains and hidden Markov models, as particularly suitable for biomedical-device applications (Kim et al. 2002; Rabiner and Juang 1986; Tomlin and Axelrod 2007).

Markov Chains and Hidden Markov Models. Markov chains and hidden Markov models provide a method of modeling probabilistic systems with finite states (Rabiner and Juang 1986, 5). While this method has existed for over a century, only recently has it begun to see significant use in engineering applications to understand the development of systems over time. Additionally, Markov chains and hidden Markov models have been used in a number of other fields to model and predict the development of highly stochastic biological and population schema in order to emulate and predict their function (Baldi et al. 1994; Durbin et al. 2002, 46–159; Kim et al. 2002; Van Hulst 1978). A Markov-chain model can be easily visualized as a set of states, each with a probability of propagation to a future state. Figure 7 shows a simple Markov chain. Each state of the Markov-chain model represents a physical-system state, with arrows showing the probability (a_{XY}) of propagation from state X to state Y in one time step. The sum of all propagation probabilities from each state must sum to 1.0, with feedback or steady-state operation between states also being possible. Additionally, segmentation and hierarchical Markov Chain models are also possible, where the probabilities of a state's propagation may be dependent upon the current state of a separate, but related, Markov chain. Such a technique is easily scalable and enables the effective modeling of highly complex systems in a manner that is intuitive, adaptable, and quick to implement or alter (Rabiner and Juang 1986, 4–7).

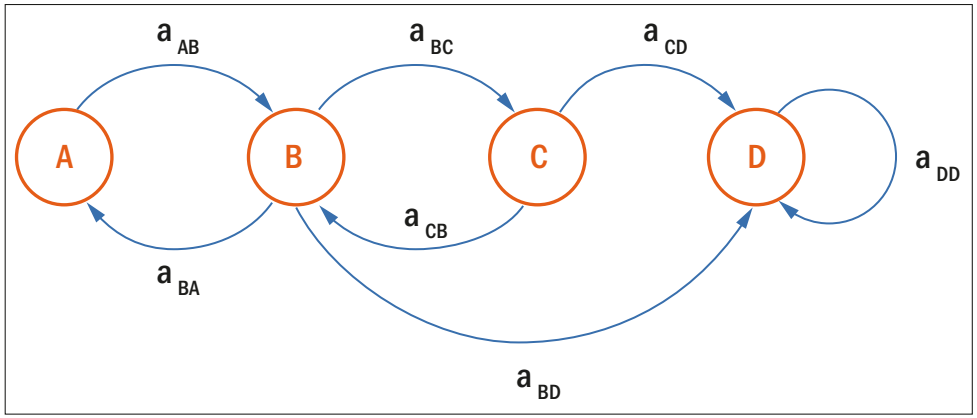


Figure 7. Markov chain showing propagation between system states

Hidden Markov models are an extension of the Markov-chain concept, where the Markov chain or network of interacting Markov chains are developed based on observed real-world performance. The behavior of a system, be it discrete in nature or a continuous spectrum, is tracked and documented, and then a Markov model is developed to fit this system performance. This model then enables further analysis or prediction of future system functionality (Rabiner and Juang 1986, 4). These models appear "hidden" to the model developer, who may not initially know how system states are related or the probabilities that govern the system fluctuation between states.

These characteristics make Markov-chain models and hidden Markov models a preferred method for the representation of biological systems. Highly complex biological phenomena have already been modeled with considerable success through the use of Markov chains. Kim and others (2002, 338) successfully developed a Markov model for the progression of melanoma in patients, where data were based upon the predictive relationships between 587 independent genes. By determining the factors of greatest importance to the development of the melanoma cells, they produced a Markov model describing ten interacting genes that very nearly matched the real-world development of the system (steady-state convergence of all states was higher than 0.05 significance level).

Since the development of a biological system such as melanoma is a continuous spectrum, in this method physical states are lumped to collective state vectors, thus enabling a succinct analysis of the biology. This same technique can be expanded to any number of other biological systems at varying levels of abstraction. A medical researcher in the field may be interested in the physiological changes of a system at the molecular level, thus encouraging the development of a Markov model to emulate these processes in the context of a larger biological system. Similarly, a practicing clinician may be more interested in overall patient response to a particular assay, thus encouraging the development of a Markov model to predict system response at a higher level of abstraction.

In each case, such a technique is extremely valuable to a system designer attempting to develop biomedical devices for these varied applications. Established techniques are generally available to provide biological system data in all but the most complex instances. Systems biologists and medical researchers can utilize this data to formulate simplified models of the highly complex biological systems that, in turn, become valuable assets to the design engineers. The intuitive nature of Markov models enables the engineer not only to design and simulate a system with stochastic biological components, but also to bridge the knowledge gap between complex biology and the engineering of complex biomedical devices (Tomlin and Axelrod 2007, 336–339). In doing so, the engineer can optimize the validation, verification, and potential redesign of a physical system for experimental biomedical applications to a point that is not currently achievable using established system modeling techniques.

Implementation of Platforms for Experimental Biomedical Systems

By combining the modeling mechanisms available for physical engineering systems with the Markov modeling techniques presented for biological systems, an engineer can realize a comprehensive platform for the full system design of experimental biomedical devices. Borrowing from the high-level system architecture in figures 5 and 6, this framework platform creates a union of the biological and engineering domains that enables the simulation of a full biomedical system. Figure 8 showcases how such a union is achieved, where the biological element is modeled as a component in the system architecture.

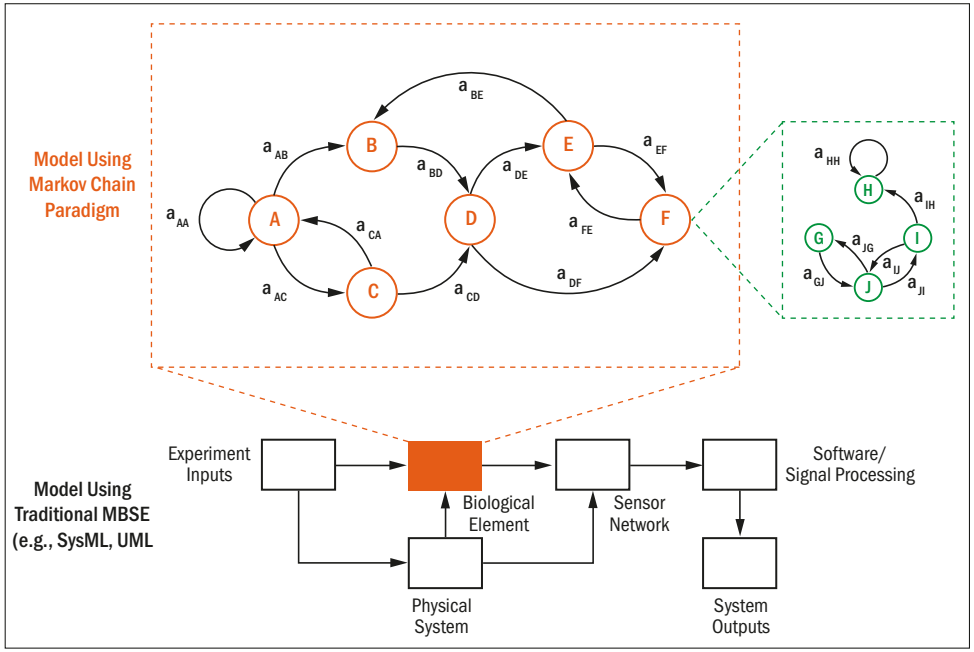


Figure 8. Implementation of the platform for engineering of experimental biomedical-device systems

It is possible to model the full system architecture by using established systems modeling platforms, such as UML or SysML, since many of these have reached a level of maturity to support extensions to other languages and tools. In order to utilize the platform for overall engineering of the biomedical system, the implementation process follows a straightforward path:

1. Gather relevant data of the biological system at a level of abstraction coincident with the application requirements. This data will be used to formulate a Markov model of the biological-system component.

2. Formulate a Markov model describing the biological component. An iterative process is often used to achieve convergence of such a model, as well as to define the appropriate segmentation of finite states for continuous systems (Kim et al. 2002; Rabiner and Juang 1986).
3. Represent the validated Markov model using a tool capable of integrating with the physical system model. This biological element will exist as an extension to the modeling platform (e.g., SysML or UML) used to define the larger device system.
4. Design the proposed physical device components and how these components relate using the modeling platform. An additional component should also be represented in the system model that will extend to the biological component.
5. Perform simulation, validation, and verification of the complete system model. The results of these analyses will provide a means of redesign and device optimization for the particular experimental application.

The outputs generated from this system analysis provide a range of potential experiment outputs based on the operation of the physical system and the development of the stochastic biological system. The value of obtaining such a resultant set is paramount to design engineers, as it allows them to directly address real-world concerns that are not otherwise visible in the design phase. In the prototyping phase of device development and beyond, this same analysis can be used to verify proper device operation, to confirm the results of experiments, and to detect and avoid undesirable system performance. Such analyses are currently difficult and exceedingly time-consuming using established methods. Therefore a platform for experimental biomedical device development has considerable value to the medical field as a whole.

Medical Drug Screening for Antibiotic Development

This section presents a prototype application of the presented platform for engineering experimental biomedical systems. Drug screening for the development of new pharmaceuticals is a major area of concentration in the biomedical field. To enable high-throughput screening of prospective antibiotics for bacterial infections, a microsystem designed in the MEMS Sensors and Actuators Laboratory at the University of Maryland utilizes a parallel architecture capable of arrayed experiments and non-invasive sensing.

The developed system contains all of the architectural components mentioned previously in this article for experimental biomedical devices. A microfluidic platform provides a physical module capable of containing the biological system, in this case an infectious bacterial biofilm. Additionally, a sensor network external to the microfluidic device enables continuous monitoring of bacterial growth or colony formation, where the cumulative outputs of the system can have a range of possibilities depending upon the stochastic biological system. Such architecture makes this application an ideal candidate for the use of the proposed design platform, since reliance on the biological component makes system performance difficult to predict.

The device itself utilizes a microfluidic channel to grow bacterial biofilms under controlled conditions. The mature biofilms are then treated with candidate drugs in order to determine their levels of efficacy in depleting the bacterial films. Bacterial suspensions, growth media, and the candidate antibiotics are supplied to the device via interface tubing, which allows an external syringe pump to control flow rates in the system. Sensing of bacterial growth is achieved through optical density detection. As biofilm grows, it becomes increasingly absorbent to incident light (optically dense), thereby enabling biofilm monitoring via the amount of light transmitted through a biofilm sample (Bakke, Kommedal, and Kalvenes 2001, 13). Sensing of this transmission is achieved by a one-dimensional array of photopixels placed underneath the microfluidic growth chamber, where the analog voltage outputs of the pixels are inversely proportional to the biofilm optical density at that point. The advantage of this sensing mechanism is that it provides a means of noninvasive and continuous detection of biofilm growth that is otherwise difficult to obtain (Meyer et al. 2011, 1). Additional study of the biofilm is achievable through end-point measurements of density and morphology using confocal microscopy. Figure 9 provides an overview of this architecture with the system components highlighted via images of the prototyped devices.

A Markov model of the bacterial biofilm component enables the formal validation and verification of this biomedical system. The current high-level model of the biofilm development process utilizes the tool presented by Yang. To investigate target characteristics of the network, an engineer uses the software package to specify a network of interacting Markov chains (referred to as Markov chain cells in this article) that simulates the interactions of these cells. Through reduction techniques that utilize symmetry in the Markov-chain network, highly complex models can be analyzed that would otherwise go beyond the computational capacity of most systems (Yang, Zhou, and Baras 2011).

The Markov-chain network used to describe this system is comprised of two distinct domains: the physical conditions of the experiment that affect the bacterial biofilm, and an array of identical Markov cells to describe the biofilm structure. Biofilm Markov cells represent discrete sections of the film within the microfluidic chamber, where the state of each cell is dependent upon the states of adjacent cells as well as the states of the experimental conditions. Figure 10 shows the abstraction of the bacterial biofilm system as it is currently implemented, and follows directly from the architecture presented in figure 3. As this Markov model continues

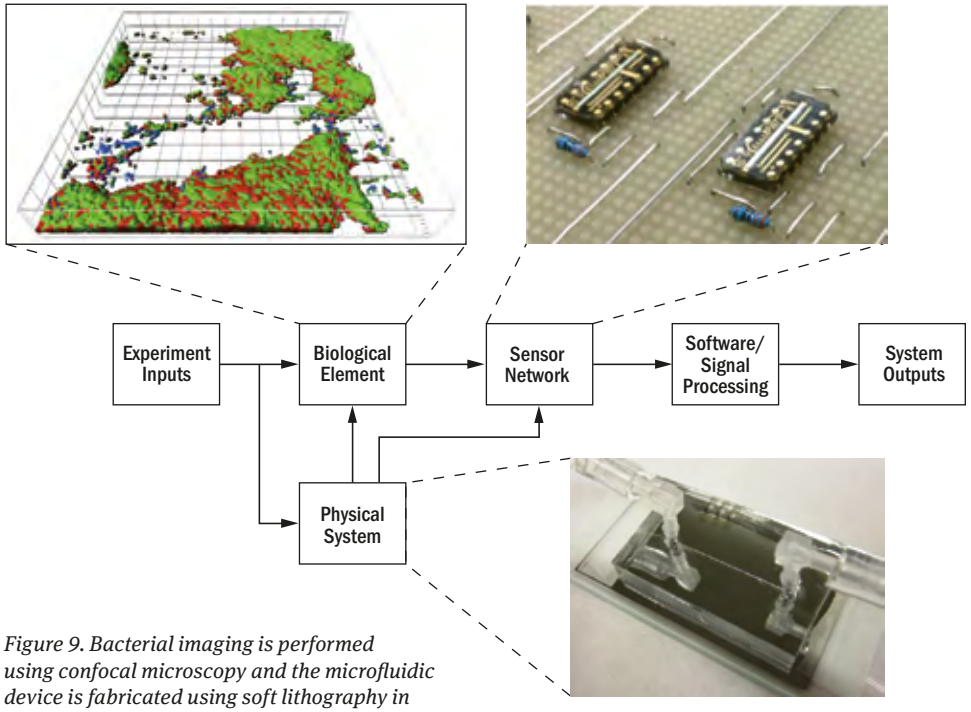


Figure 9. Bacterial imaging is performed using confocal microscopy and the microfluidic device is fabricated using soft lithography in polydimethylsiloxane. The sensors are charge-coupled devices with 128x1 pixel arrays.

to mature, a clear path is to expand the model to a generalized two-dimensional biofilm with a suite of influencing experimental factors. Such advancements will permit the model's use in any number of biomedical design processes for applications dealing with bacterial biofilms.

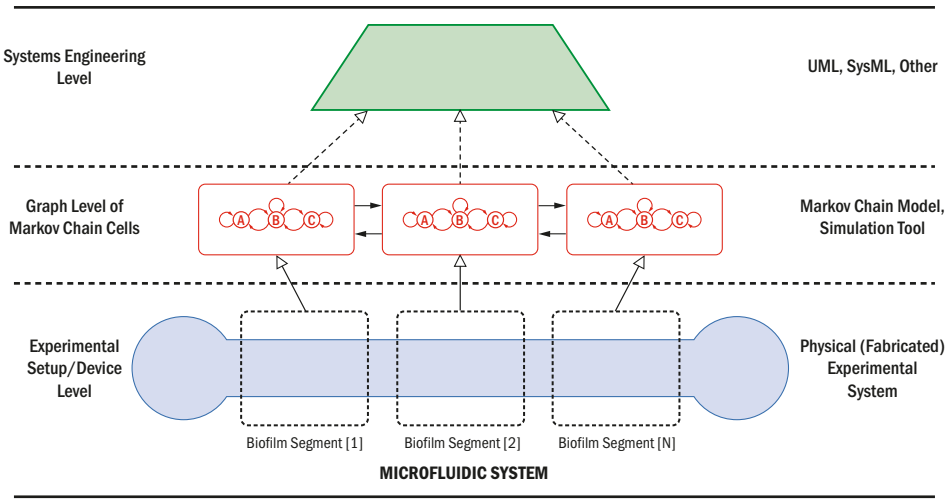


Figure 10. Visual representation of the biofilm Markov-model implementation

An N-cell biofilm model features two key experimental condition variables: nutrient concentration in the system growth media and damaging shear stress due to fluid flow around the film. Each of these variables was provided binary values (low or high), and the biofilm elements were simplified to a system of three distinct states (reduced, moderate, and mature). The next-iteration state of each biofilm element is dependent upon its own cell's current state, the current states of its adjacent cells, and the current states of the experimental conditions. Using Bayesian statistics, Yang, Zhou, and Baras (2011) found that the number of possible states for the biofilm model was $X = 2*2*3^N$. Through the tool's symmetry reduction methods, the system simulation was condensed from this set of possible states to a model with $0.75*X$ states, a 25% decrease in overall system size. By establishing an observer in the tool to track the number of biofilm cells in each of the three developmental stages, a full spectrum of theoretical biofilm growth characteristics is obtained that agrees with intuitive expectations (i.e., a near Gaussian profile). Future improvements to this model and its implementation in the simulation tool are expected to reduce this model even further, as previous implementations of its symmetric reduction principles have achieved orders-of-magnitude reductions in system size (Yang, Zhou, and Baras 2011).

Conclusions and Future Work

We have shown that the implementation of a platform for engineering experimental biomedical systems can bridge the knowledge gap between biologists and engineers and ensure more successful system development. By utilizing Markov-chain models to represent biological systems and extending these models to those of device components in established languages such as UML or SysML, one can achieve an overall model of the biomedical system. A key benefit of this method is that it enables formal system-level validation and verification of biomedical systems for experimental applications.

To bring the benefits of such a platform to fruition, further work must explore methods to integrate tools for modeling biological systems with well-established modeling languages. Such an architecture lays the foundation for a collaborative effort between biologists, clinicians, and systems engineers. Libraries of biological and device-oriented components achieved through this collaborative effort can be used in a broad number of application areas to develop new experimental systems for these disciplines.

References

Bakke, R., R. Kommedal, and S. Kalvenes. 2001. "Quantification of Biofilm Accumulation by an Optical Approach." *Journal of Microbiological Methods* (44) 1: 13–26.

Baldi, P., Y. Chauvin, T. Hunkapiller, and M. A. McClure. 1994. "Hidden Markov Models of Biological

» continues on next page

Mosteller et al. *continued*

Primary Sequence Information." *Proceedings of the National Academy of Science* 91 (3): 1059–1063.

Beck., J. R. 1983. "The Markov Process in Medical Prognosis." *Medical Decision Making* 4 (3): 419–458.

Costerton, J. W., B. Ellis, K. Lim, F. Johnson, and A. E. Khoury. 1994. "Mechanism of Electrical Enhancement of Efficacy of Antibiotics in Killing Biofilm Bacteria." *Antimicrobial Agents and Chemotherapy* 38: 2803–2809.

Costerton, J. W., P. S. Stewart, and E. P. Greenberg. 1999. "Bacterial Biofilms: A Common Cause of Persistent Infections." *Science* 284 (21): 1318–1322.

Csete, M. E., and J. C. Doyle. 2002. "Reverse Engineering of Biological Complexity." *Science* 295: 1664–1669.

Donlan, R. M. 2001. "Biofilms and Device-Associated Infections." *Emerging Infectious Diseases* 7 (2): 277–281.

Durbin, R., S. R. Eddy, A. Krogh, and G. Mitchison. 2002. *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*. Cambridge, GB: Cambridge University Press.

Endy, D. 2005. "Foundations for Engineering Biology." *Nature* 438 (24): 449–453.

Janakiraman, V., D. Englert, A. Jayaraman, and H. Baskaran. 2009. "Modeling Growth and Quorum Sensing in Biofilms Grown in Microfluidic Chambers." *Annals of Biomedical Engineering* 37 (6): 1206–1216.

Kim, S., H. Li, E. R. Dougherty, N. Cao, Y. Chen, M. Bittner, and E. B. Suh. 2002. "Can Markov Chain Models Mimic Biological Regulation?" *Journal of Biological Systems* 10 (4): 337–357.

Meyer, M. T., V. Roy, W. E. Bentley, and R. Ghodssi. 2011. "Development and Validation of a Microfluidic Reactor for Biofilm Monitoring Via Optical Methods." *Journal of Micromechanics and Microengineering* 10: 1–10.

Oltvai, Z. N., and A. L. Barabasi. 2002. "Systems Biology: Life's Complexity Pyramid." *Science* 298: 763–764.

Potera, C. 1999. "Microbiology: Forging a Link Between Biofilms and Disease." *Science* 283: 1837–1839.

Rabiner, L. R., and B. H. Juang. 1986. "An Introduction to Hidden Markov Models." *IEEE Acoustics, Speech, and Signal Processing* 3 (1): 4–16.

Richards, J. J., and C. Melander. 2009. "Controlling Bacterial Biofilms." *Chemistry and Biochemistry* 10: 2287–2294.

Tomlin, C. J., and J. D. Axelrod. 2005. "Understanding Biology by Reverse Engineering the Control." *Proceedings of the National Academy of Sciences* 102 (12): 4219–4220.

———. 2007. "Biology by Numbers: Mathematical Modelling in Developmental Biology." *Nature Reviews Genetics* 8: 331–340.

Van Hulst, R. 1978. "On the Dynamics of Vegetation: Markov Chains as Models of Succession." *Theory and Models in Vegetation Science* 40 (1): 3–14.

Yang, S., Y. Zhou, and J. Baras. 2012. "Compositional Analysis of Dynamic Bayesian Networks and Applications to CPS." *Third International Conference on Cyber-Physical Systems*, Beijing, CN, 17–19 August.

Clinical Engineering: A Systems Focus on the Point of Care

Rick Schrenker, rick.schrenker@incose.org

Clinical engineering is a subdiscipline of biomedical engineering that focuses on providing engineering services to the application of medical technology used in the delivery of health care. Clinical engineers generally hold an undergraduate engineering degree and often a graduate degree in engineering or business. They take on many engineering and technology management roles both in hospitals and the medical-device industry. From its earliest days, the profession focused on medical technology systems:

> We need first to survey user needs and define the objectives of the system to be built, based on the system in use. Once we have those objectives, then we can define requirements. [The engineer] must engage in research, market analysis, and user consultation before attempting to make old systems efficient or to interface new ideas with the user [. . .].

Until you understand the centrality of safety to clinical engineers, you can't understand clinical engineering.

Taken from Cesar Caceres's now 35-year-old book *The Practice of Clinical Engineering* (1977, 6), the above remains a valid description of some of the roles of the clinical engineer. This role maps well with the description of system engineering by INCOSE's Educational and Technical Research Committee (2004):

> Systems engineering is concerned with the overall process of defining, developing, operating, maintaining, and ultimately replacing quality systems. Where other engineering disciplines concentrate on the details of individual aspects of a system (electronics, mechanics, ergonometrics, aerodynamics, software, etc.), systems engineering is concerned with the integration of all of these aspects into a coherent and effective system. Systems engineers concentrate their efforts on the aspects of the engineering process (requirements definition, top-level functional designs, project management, life cycle cost analysis [. . .]) that serve to organize and coordinate other engineering activities.

Although clinical engineering has roots that intersect those of systems engineering, only recently have members of these professions started working with one another. Why? A retrospective taken from a 2006 article about clinical engineering may provide some clues. In the 1970s, many devices were less reliable than they are today, so maintenance efforts of the clinical engineers focused on establishing some level of device reliability and safety. Hospital engineers were responsible for maintenance and reliability of hospital facility systems, including gas, suction, and electricity that support devices. The nurses, physicians, surgeons, and

technologists were responsible for bringing patient, facility, and device together to function safely in an environment that includes many other minisystems. With these split responsibilities for components within the same or other minisystems, it was assumed that the minisystem would function properly when all the components were finally brought together (Braeutigam 2006, 360).

Did we recognize we were doing "systems engineering"? We certainly used the word *system* to describe our equipment-management programs and tools. The term *patient monitoring system* was also in common use when I entered the field in 1979, to describe the integrated collection of bedside and central-station physiological monitors used to monitor patients in intensive-care units. So while we may not have recognized that we were doing what others (who we didn't know existed) were calling systems engineering, we did know that we were working in and with systems. Moreover, we knew that we were increasingly facing problems that required developing and managing these systems. It may be fair to say that clinical engineering has successfully developed to this point as a systems-focused profession without looking outside the box of health care. What value, then, can clinical engineers gain from INCOSE? What has changed?

Asking a mixed group of clinical engineers to describe clinical engineering would probably result in answers not unlike those provided by the proverbial blind men looking at an elephant.

In an article from April 2001, Todd Cooper and I tried to provide insight into the developing problem facing clinical engineering:

> Surely by now one might assume these [patient] monitoring devices supply the information they gather directly to the systems that chart patient data. Unfortunately, not to the extent one might imagine. The standardization of communication processes that has led to the explosion of telecommunications products in the consumer area has yet to take hold in the world of clinical medicine. This lack of connectivity leaves open to question the accuracy and completeness of a patient record created by harried clinicians whose attention to data entry tasks diverts their attention from patients. And without electronic capture of data and events associated with an episode of care, trending and other sophisticated data analyses are effectively impossible.

By 2004, Wayne Hibbs called the community to find "the vision" for solving this problem:

> We will put to rest the urban legend of health care automation when we monitor every patient—from admission to discharge—with smart analysis that does not require extra staffing. We will need a flight-data recorder for the patient record that allows us to provide the best care and achieve the best clinical outcome at the least cost with the best staff. We need an EMR that is as graphically accurate, user friendly, rapidly responsive, and crash proof as my son's Game Boy SP, with the same ability to upgrade to new software, connect to other systems, and be replaced for under $100 when it is obsolete in 2 years. Now that my son has interfaced his 3-inch Game Boy SP to my 50-inch plasma display TV, I see that convergence of the 1983 Pac-Man video game at the Pizza Palace and my 1983 cable-ready TV is possible. All it takes is the vision.

Since then, a number of programs have arisen to address the connectivity and interoperability problems alluded to in the above (Schrenker 2008). As the outputs of these programs result in the delivery of products to the market, the nature of the medical-device system at the patient's bedside will change. Previously, these systems were implicitly integrated in the mind of the clinicians who used them, but now they will be explicitly integrated via communication and information technology. These integrated systems are intended to provide new functional properties to deliver on the promises alluded to by Hibbs. They will also add or change nonfunctional properties ("ilities") that will require revisiting the nature of medical technology management, such as system risk management. These are among the current technical drivers of clinical engineering.

While clinical engineering is struggling with these fundamental changes in the nature of medical technology systems, it is also having to deal with the cost-control pressures that are affecting all of health care. Learning from others who have been addressing systems engineering more directly could prove valuable going forward. Similarly, other subdisciplines could learn from clinical engineering. Certainly the INCOSE Biomedical Engineering Working Group could provide a forum to support this exchange. But before speaking more directly to the current state, some historical perspective on the earlier systems problems that drove and still drive clinical engineering can provide context for the next steps.

A Look (Not Too Far) Back...

For anyone looking to know "just what is clinical engineering, anyway?" the *Clinical Engineering Handbook* (Dyro 2004) provides a good introduction to the

field. The book includes the following sections:

Clinical Engineering
Worldwide Clinical Engineering Practice
Health Technology Management
Management
Safety
Education and Training
Medical Devices: Design, Manufacturing, Evaluation, and Control
Medical Devices: Utilization and Service
Information
Engineering the Clinical Environment
Medical Device Standards, Regulations, and the Law
Professionalism and Ethics
The Future

A sample of its 142 chapters includes the following:

Risk Management
Systems Approach to Medical Device Safety
Evaluating Investigational Devices for Institutional Review Boards
Operating Rooms
Health Care Quality and ISO 9001:2000
Tort Liability for Clinical Engineers and Device Manufacturers
American College of Clinical Engineering
Global Hospital in 2050: A Vision

The scope of clinical engineering's responsibilities and interests is broad, and no one clinical engineer can be an expert in all areas. Asking a mixed group of clinical engineers to describe clinical engineering would probably result in answers not unlike those provided by the proverbial blind men looking at an elephant. One's perspective of a large complex system is framed by one's point of encounter. But this author would be willing to bet that 100% of any group of respondents would all, each and every last one, cite one focal point as clinical engineering's *raison d'être*, one aspect of what we do that despite differences of opinion on any other topic we would all agree is fundamental and shared: safety. Until you understand the centrality of safety to clinical engineers, you can't understand clinical engineering.

Almost halfway into the handbook is a chapter by Malcolm Ridgway,"The Great Debate on Electrical Safety—In Retrospect" (2004). In about two pages Ridgway summarizes a journey from reports in 1961 of electrical safety hazards associated with the use of cardiac pacemakers, through a 1971 *Ladies' Home Journal* article by Ralph Nader which states that 1200 Americans were being electrocuted annually during routine medical procedures. He traces the subsequent reactions and responses from standards and regulatory agencies, through the inability of anyone thereafter to substantiate any credible evidence for the reported phenomenon of "microshock." The United States Congress attempted to address this and other problems by passing The Medical Device Amendments of 1976 to the Federal Food, Drug, and Cosmetic Act. In testimony to a Senate subcommittee while the act was being debated, Dr. Joel Nobel of the Emergency Care Research Institute pointed to a serendipitous result of the process, which was a new awareness of the need for engineering services in hospitals (quoted in Ridgway 2004, 282):

> Phony statistics have been used to promote the sales of safety equipment and manipulate the National Electric Code to require the use of specific products. We are not suggesting that the microshock electrocution issue was fabricated by the industrial and code making camps and consumer advocates. Each, however, capitalizing on the issue, has distorted both the technical problems and the priorities rather badly. The result is that many millions of dollars have been diverted from more critical areas of health care. This electrical safety issue has however, performed a useful catalytic function in drawing attention to other problems associated with the use of technology for health care. It has helped hospitals to understand the broader need for engineering support of patient care, including the judicious purchase, inspection, and preventive maintenance of medical equipment.

Thirty-one years later, the *Clinical Engineering Handbook* devoted two pages to this story and over 600 hundred more to 141 additional chapters that might never have been written, or at least not from the same perspective, had it not been for what could arguably be termed "clinical engineering's creation story."

Many, this author included, believe the field is at a similarly pivotal point today and has been for a while. In yet another chapter from the handbook, Steve Grimes described the field as being at a "strategic inflection point" and recommended the field focus on a number of areas going forward, a few of which follow (Grimes 2004, 624–625):

- Adopt a systems and process approach
- Add basic information technology and telecommunications skills
- Monitor technological, regulatory, economic, and other developments
- Become conversant with the "business" of technology
- Plan for the integration of existing and new medical technologies
- Develop systems and infrastructure to support technology in nontraditional venues

All It Takes Is the Vision

Clearly clinical engineers recognized in the first five years of the 21st century that the field had entered another pivotal period. This time, however, the critical technological interface driving the need for an engineering response was not between the patient and device but between the device and information technology. Both this interface and the one that caused a stir 40 years ago exist for the same reason: to improve the delivery of care to a patient by decreasing the uncertainty of professional caregivers about the status of the patient and to enable more effective therapeutic interventions. But this time, there is also a fundamental difference in the system model at the point of care.

No matter how incrementally evolutionary a contemplated change might seem, it can still introduce a new way for the changed system to fail. The overarching lesson is that making any change in a design can alter the entire context in which the detail is embedded and thereby introduce a failure mode that would have been impossible in the original design. A single change can change everything (Petroski 2012, 19).

So who will be doing the engineering required to address this? Back in 2004, one school of thought, represented by Hibbs (in Schrenker 2004), was that this was not a clinical engineering role:

> The future success of connectivity is not in our vision, it is in making our vision financially rewarding for the vendors. As much as I would like to believe that good clinical engineers are going to evolve into good software engineers, the US Food and Drug Administration requirements for good manufacturing practice standards in software development for health care products removes that option from the very best clinical engineer or biomedical equipment technician. The legal liabilities of clinical-based engineers adapting vendor's software are no longer acceptable from a risk-management viewpoint.

But over the last few years it has been at least tacitly understood by the clinical engineering and regulatory communities that even if the liabilities of hospital-based developers (not necessarily clinical engineers) doing software engineering were unacceptable, it was happening anyway. In 2011 the US Food and Drug Administration issued a regulation covering what they have termed "medical device data systems" (MDDS).They provide this definition:

> An "MDDS manufacturer" may be a health care facility or manufacturer that is engaged in the following activities:
>
> - Modifying a general purpose IT equipment/software or infrastructure for purposes of interfacing with medical devices and performing functionality described in the MDDS rule (transfer, store, display, or convert data).
> - Labeling a general purpose IT equipment/software as a MDDS for purposes of interfacing to medical devices and performing functionality described in the MDDS rule (transfer, store, display, or convert data).
> - Designing and implementing custom software or hardware for purposes of interfacing with medical devices and performing functionality described in the MDDS rule (transfer, store, display, or convert data).

Three health-care facilities (including mine) have registered medical-device data systems with FDA. It remains to be seen whether more delivery systems will get into the MDDS business, or if the preponderance of the work will end up with medical-device manufacturers. Perhaps some joint effort may emerge. Regardless, an MDDS by itself is not all that valuable; the important thing is what it enables: communication of patient data from a medical device to another device for storage or display (for example).No one medical-device company makes all of the medical devices that will supply data to the chart of a patient in an intensive-care unit, for example. The MDDS serves as a conduit from a medical device to enable the transfer, storage, display, or conversion of data. The information-technology networks and devices to which an MDDS provides data also provides this kind of conduit. Developing an MDDS presents a plethora of systems property-management problems, not the least of which is risk management. As Todd Cooper (2008) puts it, there is no one international standard that fully addresses the problem:

> An emerging standard seeks to address this problem area: IEC 80001[,] *Application of Risk Management for IT-Networks Incorporating Medical Devices* [. . .]. IEC 80001 picks up where ISO 14971 leaves off and addresses how accumulated and residual risks from medical and nonmedical equipment and applications should be managed in a heterogeneous networked environment [. . .]. An interesting aspect to this problem is that it lies firmly in that demilitarized zone between clinical engineering and IT [. . .]. Thus, any solution must pull together provider, medical device manufacturer, IT equipment manufacturer, regulator stakeholders, and also cross traditional organizational boundaries, to include clinical/biomedical engineering, IT, facilities, purchasing, etc.

Opportunities for Clinical Engineers and Systems Engineers

The system components described above may seem somewhat removed from the ultimate nexus for clinical engineering, which remains where it has always been—the point of care. Nevertheless, the point of care is still the focus of these system components. In her book *Misadventures in Health Care: Inside Stories*, Marilyn Sue Bogner likened the systems surrounding the patient–caregiver interface to an artichoke, working progressively outward from the interface through the

subsystems that provide the context for what goes on there: ambient conditions, physical environment, social environment, organizational factors, and the broader social factors of laws, regulations, reimbursement channels, and national cultures (Bogner 2004, 5).

Two examples illustrate how problem-rich the subsystems at the point of care can be. In the first case, I had the opportunity to work with clinicians developing extra corporeal membrane oxygenation (ECMO) programs at the Johns Hopkins Hospital (Baltimore, Maryland, US) in the 1980s and Massachusetts General Hospital (Boston, Massachusetts, US) in the 1990s. Back then ECMO was primarily used to treat newborn babies in acute respiratory distress; before ECMO the majority of affected babies did not survive. In an ECMO procedure, physicians take the system that they normally used to bypass the heart and lungs during cardiac surgery for just a few hours and bring it into the intensive care unit to treat life-threatening cardiorespiratory failure for days. The procedure is highly complex, requires a great deal of money, staff, and technology, and is fraught with risk. So it was a pleasant surprise for me to see that a systems engineering team from Georgia Tech Research Institute and Children's Healthcare of Atlanta was applying model-based systems engineering to address the system issues presented by the intervention. This team has proposed a roadmap for creating a more robust system (Pihera et al. 2012).

A more abstract systems-focused approach that intersects with a few of the subsystems identified by Bogner comes out of the Medical Device "Plug and Play" Interoperability Program. This program focuses on integrating the point of care by creating interoperability between standards-based "plug and play" medical devices. This approach promises solutions to clinical problems that cannot currently be addressed with available technology. Examples of these solutions may be found on the program's website (http://www.mdpnp.org/).

Interoperability is certainly a missing and sorely needed component of the solution to the current health-care engineering problem. All the same, interoperability alone is not the whole solution. It will take an even more comprehensive systems approach, with full collaboration between systems engineers and clinical engineers, to realize a seamlessly integrated health-care technology system. These examples point to opportunities for the INCOSE and its Biomedical Engineering Working Group to bring together systems engineering professionals to work with clinical engineers and others who are focused on improving the system that is the point of care.

References

Bogner, M. S. 2004. *Misadventures in Health Care: Inside Stories*. New Jersey, US-NY: Laurence Erlbaum Associates.

Braeutigam, D., D. Dreps, K. Ferguson, B. Hyndman, and M. Shepherd. 2006. "The Legacy of the CE and BMET: Safer Healthcare Technologies." *Biomedical Instrumentation & Technology* 40 (5): 357–362.

Caceres, C., ed. 1977. *The Practice of Clinical Engineering*. New York, US-NY: Academic Press.

Cooper, T. 2008. "IEC 80001: Proactively Managing Risks." *24x7*, July. See http://www.24x7mag.com.

Dyro, J., ed. 2004. *Clinical Engineering Handbook*. Burlington, US-MA: Elsevier.

US Food and Drug Administration (FDA). 2011. "Who is an MDDS Manufacturer?" Website of the FDA, 19 April. http://www.fda.gov/MedicalDevices/ProductsandMedicalProcedures/GeneralHospitalDevicesandSupplies/MedicalDeviceDataSystems/ucm251927.htm.

Grimes, S. 2004. "The Future of Clinical Engineering: The Challenge of Change." In *Clinical Engineering Handbook*, edited by J. Dyro, 623–627. Burlington, US-MA: Elsevier.

Hibbs, C. W. 2004. "The Future of Connectivity." *24x7*, March.

INCOSE Educational and Technical Research Committee. 2004. "Careers in Systems Engineering." INCOSE website, 10 Nov. http://www.incose.org/educationcareers/careersinsystemseng.aspx.

Petroski, H. 2012. *To Forgive Design: Understanding Failure*. Cambridge, US-MA: Belknap Press (Harvard University Press).

Pihera, L., T. Ender, B. Taylor, N. Bollweg, M. Paden, A. Lopez, and S. King. 2012. "Application of Systems Engineering to Improve Extracorporeal Membrane Oxygenation (ECMO) Therapy." Paper presented at the 22nd Annual International Symposium of INCOSE, Rome, IT, 9–12 July.

Ridgway, M. 2004. "The Great Debate on Electrical Safety—In Retrospect." In *Clinical Engineering Handbook*, edited by J. Dyro, 281–283. Burlington, US-MA: Elsevier.

Schrenker, R., and T. Cooper. 2001. "Building the Foundation for Medical Device Plug-and-Play Interoperability." *Medical Electronics Design*, April. http://www.medicalelectronicsdesign.com/article/building-foundation-medical-device-plug-and-play-interoperability.

Schrenker, R. 2004. "The Future of Connectivity." Letter to the editor in response to Hibbs (2004), followed by a reply from Hibbs. *24x7*, May.

———. 2008. "Interoperability is Just Around the Corner—Right?" *24x7*, Jan.

Applying Systems Engineering to Improve Extracorporeal Membrane Oxygenation Therapy

L. Drew Pihera, drew.pihera@incose.org**; Tommer Ender, Brian Taylor, Nick R. Bollweg, Matthew L. Paden, Andrew Lopez, and Scott King**

The tools of model-based systems engineering, specifically SysML, were shown to bridge the gap in communication between engineers and medical professionals.

This article describes an effort to develop a structured process leveraging systems engineering best practices for development of a complex medical system, one which was originally developed and assembled in an ad hoc and unstructured manner. This article is based on the findings of a 2011 capstone project for students in the Professional Master's of Applied Systems Engineering program at the Georgia Institute of Technology (Atlanta, Georgia, US), in collaboration with medical professionals representing the extracorporeal membrane oxygenation (ECMO) unit of Children's Healthcare of Atlanta. This collaboration was the first in a long-term strategy to apply systems engineering to improve operation of ECMO, locally at Children's Healthcare and at a national level in conjunction with the Extracorporeal Life Support Organization. The capstone projects use model-based systems engineering and other novel visualization techniques.

This article will make the case for applying systems engineering to extracorporeal membrane oxygenation therapy, given the inherent complexity of its architecture and often improvised operational use. Given that the term "improvement" is vague, the project team first characterized and formally documented the systems making up the ECMO therapy. The team then continued to find potential short and long-term ways that future projects could incrementally add value to ECMO from the perspectives of the minute-to-minute operators (ECMO specialists), physicians directing care at a more intermittent time scale, the hospitals as a whole, and the nationwide ECMO community, which would benefit from broader standardization and approval by the United States Food and Drug Administration. A "roadmap" to possible implementation of the future projects was a major artifact of this effort.

The purpose of this article is threefold. First, the article describes the application of relevant systems engineering processes and practical tools to improve ECMO, which has not been attempted previously. Second, the article will detail why and how the chosen process and tools facilitated understanding and discussion between engineering students and the medical professionals sponsoring them. Finally, the article offers several proposed research topics and challenges to implementation as areas of future study, applicable to applied research or master's-level capstone projects.

The context of ECMO is very broad, and as such some areas of work performed during this effort are outside the scope of this article. One such area was the application of human-systems-interaction strategies, given the heavy workload required by the various medical professionals. Another was classification of the ECMO therapy as a system of systems, using Jamshidi's definition (2008). These findings may be presented in future publications.

Description of Extracorporeal Membrane Oxygenation Therapy

Failure of a patient's cardiac or respiratory system is a common problem in the intensive-care unit. Extracorporeal membrane oxygenation therapy provides temporary life-saving heart and lung replacement when traditional methods are failing and where the chance of survival is estimated at less than 25%. ECMO has been used to provide support to over 50,000 critically ill patients with life-threatening diseases, and has achieved an average survival rate of 73% (ELSO 2012).

In brief, the procedure removes blood from the body via large-bore *cannulas* (tubes inserted into the body); an external pump delivers this blood first to a gas-exchanging membrane to provide the function of the lung (oxygenation and carbon-dioxide clearance), before returning the blood to the patient. The site of the returning blood, either arterial (oxygenated) or venous (deoxygenated), determines whether the system is providing solely respiratory support or combined cardiorespiratory support. The volume of extracorporeal blood is often twice the amount that is inside the patient's body. Due to the blood's reaction with the extracorporeal circuit, anticoagulation methods must be used to prevent the blood from clotting while outside the body.

Traditionally, ECMO has been used primarily to treat life-threatening, acute neonatal respiratory failure. However, over the past decade this technique has been increasingly used for a diverse group of indications including cardiac failure (in all ages), pediatric and adult respiratory failure, and septic shock (Swaniker et al. 2000; Marasco et al. 2008; Bartlett 2007). Additionally, the use of ECMO has been increasing in the transplant community, where patients have successfully been bridged to both cardiac and lung transplantation. The procedure is also used in the post-transplantation period (Broome et al. 2008). Thus, the ECMO therapy has been shown to be effective in supporting patients of all ages with life-threatening cardiorespiratory failure of varying causes.

Although ECMO has been shown to result in improved survival rates for patients with life-threatening disease, it is also associated with other complications. Complications include patient-centered problems such as bleeding, kidney failure, and stroke, as well as technological problems—mechanical complications of the ECMO circuit, such as tubing ruptures, pump failure, and oxygenator failures. These complications are common and negatively impact survival rates (ELSO 2010).

Monitoring for these problems and treating the resulting complications requires a separate ECMO specialist to be at the bedside 24 hours a day. The complexity of the technology requires the ECMO specialist to continually monitor more than 50 different values (including pressure, flow, and temperature), displayed on more than 20 different displays (as shown in figure 1). This complex technical setup and the demand for intensive supervision by highly trained staff restricts the application of ECMO to a broader population of patients.

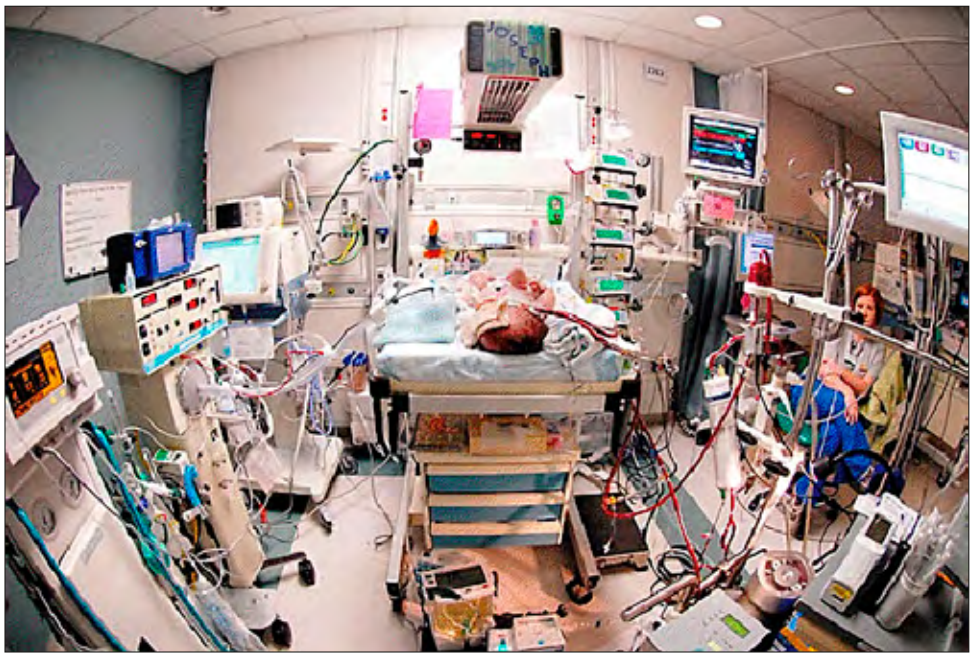


Figure 1. An operational ECMO system (Photo courtesy of Children's Healthcare of Atlanta)

Charter of the First Georgia Tech ECMO Team

This effort was to be the first collaboration between Children's Healthcare of Atlanta (CHOA) and the Professional Master's in Systems Engineering program at the Georgia Institute of Technology. The project was intended to describe the earliest phases of systems engineering in relation to ECMO. As such, the charter of the first team was to characterize and document the ECMO system as it currently stands, and use sound systems engineering methodologies to propose a roadmap for the future Georgia Tech teams and Children's Healthcare, charting a course to ever-increasing levels of improvement to ECMO. In addition to providing the documentation and planning a way forward, the team also developed a prototype of an improved data-visualization display.

In regards to the greater systems engineering process, specifically requirements elicitation, this effort would barely scratch the surface. The project was, however, an important and necessary first step toward such an end. It is hoped that each future capstone team will be able to further apply the systems engineering process to achieve success in the proposed future projects.

Mission Statement. The relationship between Children's Healthcare and Georgia Tech's professional master's program is expected to extend beyond the initial project in 2011. Therefore a mission statement was required to frame the long-term interaction between the two entities. The groups wrote the following mission statement at their initial meeting:

> CHOA and the Georgia Tech Professional Masters of Applied Systems Engineering (PMASE) program will partner, in the spirit of continuous improvement, to facilitate a state transformation of the ECMO system of systems. Each PMASE project team will refine the understanding of the current state, concepts for the ideal future state, and the architecture for state transformation while providing tangible value and incremental progression toward the desired future state.

Need Statement. In cooperation with Children's Healthcare, the Georgia Tech team wrote the following statement to guide the team's actions during the initial 2011 iteration of the partnership. The goal is to provide the highest possible benefit to all stakeholders including health-care professionals, student participants, and academic faculty:

> A need exists to improve the timeliness and synthesis of the information used by the physician to plan, the specialist to operate, and the biomedical engineer to maintain safe and effective patient therapy with extracorporeal membrane oxygenation (ECMO).

Systems Engineering Process

The following sections describe the team's systems engineering process and show which tools they used to further their knowledge of the problem domain and to convey the value of the systems engineering process to the medical practitioners. Given the differences in backgrounds and vocabularies between the engineers and most of those in the medical field, employing models that could bridge those gaps was essential.

Domain Characterization. The first meeting between the Georgia Tech team and their collaborators from Children's Healthcare allowed the students a chance to describe the systems engineering process to the CHOA representatives. They conveyed how the process might be applied to ECMO, asked questions to gain further insight into the system, and presented potential current and future projects in the problem space. Given the students' nascent understanding of ECMO and the desired end states, they presented these projects in the order that seemed most logical. The meeting also allowed the CHOA representatives the opportunity to talk through their problems with the current implementation, explain the end states they wish to achieve and discuss how they related to the proposed projects and road map. This also provided CHOA the opportunity to describe "perfect-world" scenarios that may be difficult or impossible to reach given the current state of medicine and regulation. The brainstorming exercise led to invaluable data, which allowed the Georgia Tech team to refine the roadmap from the current state of ECMO to the desired, yet feasible, end state.

In addition to the process of simply discussing the problem space with the sponsor to determine what was needed and how to proceed, the engineering students also used several systems engineering tools and methodologies. They employed a mix of Systems Modeling Language (SysML) diagrams to begin the conversation of initial stakeholder identification and system characterization. Possible project approaches were presented to elicit and prioritize potential solution paths. These initial elicitation presentations included graphical concepts, a short description, entry and exit criteria, and future work recommendations. The entry and exit criteria in these presentations were used for justification of the project ordering.

Stakeholder Identification. One of the team's first tasks was identifying the relevant stakeholders in the project domain. They captured these data in a SysML use-case diagram. They chose this format primarily because it is easily understood even by those with no formal training in SysML. The simplicity of the diagram clearly conveys who the stakeholders are and why they were included.

The team's early iterations of this diagram were naïve, containing little more than the project team, the lead physician, the ECMO specialist, and the patient. Iteration with the sponsoring CHOA representatives and other members of the ECMO staff greatly improved these diagrams. There was strong evidence that using SysML use-case diagrams in collaboration with subject-matter experts can provide extremely valuable domain information that is easily used in the system engineering process. The team realized they needed to expand their initial list of stakeholders to include surgeons, biomedical engineers, and the Extracorporeal Life Support Organization. When collaborating with the domain experts, this inconsistency between the model and the system was easily uncovered by both parties and corrected. The final use-case diagram of stakeholders for the initial effort is shown in figure 2. It is clear from the diagram that there are two concurrent activities, one being the continued patient treatment performed by the CHOA staff, and the second being the system improvement performed by the Georgia Tech team. This diagram clarifies the identity and functions of all the stakeholders.

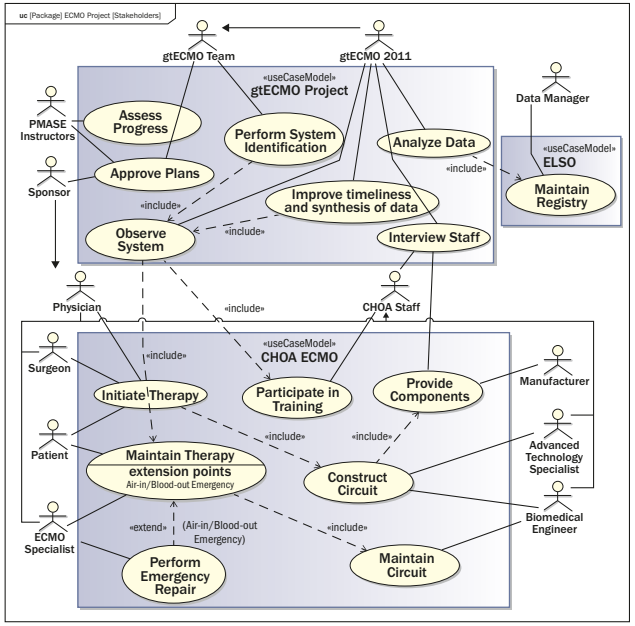


Figure 2. Stakeholder identification using a SysML use-case diagram

Problem Decomposition. At the start of this effort, the team developed a preliminary roadmap of the most logical path forward given the current understandings for the long-term partnership. They chose three primary systems engineering tools for the initial problem decomposition: work breakdown structures, N2 diagrams, and Operational View Level 1 (OV-1) diagrams. These tools enabled a better understanding of the ECMO functions and relationships, and influenced the roadmap for the future. In addition, the tools helped the Children's Healthcare representatives easily understand the proposed way forward and what would be gained by each future iteration of the Georgia Tech project.

Work Breakdown Structure. Proper planning is the first and most important step in the improvement and optimization of the existing ECMO system at Children's Healthcare. To this end, the team created two work breakdown structures: one used to schedule the actual work in the 2011 project, and another as a longer-term plan for the proposed future work. This second work breakdown structure is intended to become a living document and be updated by each future Georgia Tech team as

resolution and fidelity of the proposed projects are increased due to new findings or redirection from CHOA.

An important aspect of the work breakdown structure is the ability to show dependencies between tasks. In the work breakdown structure for proposed projects this ability was vital, because it justified why the proposed actions should be completed in the order specified. An example of this sort of dependency occurred between the redundancy-characterization project and the hardware-instrumentation project. Redundancy characterization requires equipment failure data; however, very little equipment failure data was readily available. The hardware instrumentation project, on the other hand, provides failure-rate data and should therefore be completed first. This was easily observed with the work breakdown structure.

N2 Diagram. This tool is used to identify major functional and physical interfaces for the system of interest. Though typically used for describing software interfaces, it is also a valuable way to show the flow of information throughout ECMO. In conjunction with activities identified in the work breakdown structure, the Georgia Tech team developed an N2 diagram for the current and future states of ECMO. Arrows between components in the diagram represent feed-forward loops (arrows to the right and down) and feed-back loops (arrows to the left and up) of data transfer or required physical interface. These interrelationships may change as the long-term effort evolves. Figure 3 and figure 4 present the N2 diagrams for the current state and proposed future state. The blue dashed outlines show organizational boundaries (between Children's Healthcare of Atlanta and the Extracorporeal Life Support Organization), while the red dashed outline shows the area targeted for improvement.

When comparing the current-state and future-state N2 diagrams side by side, it becomes apparent that the area of improvement (in red) of the future state exhibits more automation in the system than the current design does, specifically in data entry. This change would allow the supervising physician to spend more time with the patient rather than at the computer. The team used the N2 diagram as a tool to quickly identify an area of improvement that would be easily understood by the decision-makers at CHOA. The N2 diagram helped the health-care specialists realize that information could be transferred into the ECMO system in a much more timely and accurate way using a real-time input/output and control component for data entry both in CHOA's data stores as well as in the ELSO registry.

Operational View One (OV-1). The OV-1, taken from the United States Department of Defense Architecture Framework, is used to present a high-level view of the system of interest along with its operational elements and major data flow. The diagram was used in this study to provide a snapshot of ECMO in its future state. The OV-1 diagram in figure 5 represents the overall ECMO system of systems, including

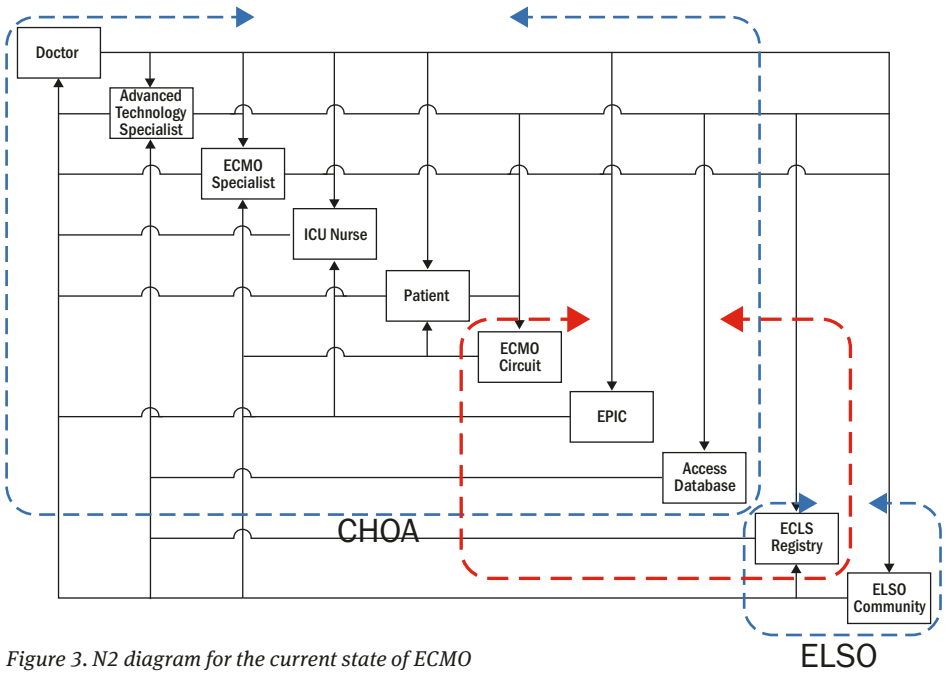


Figure 3. N2 diagram for the current state of ECMO

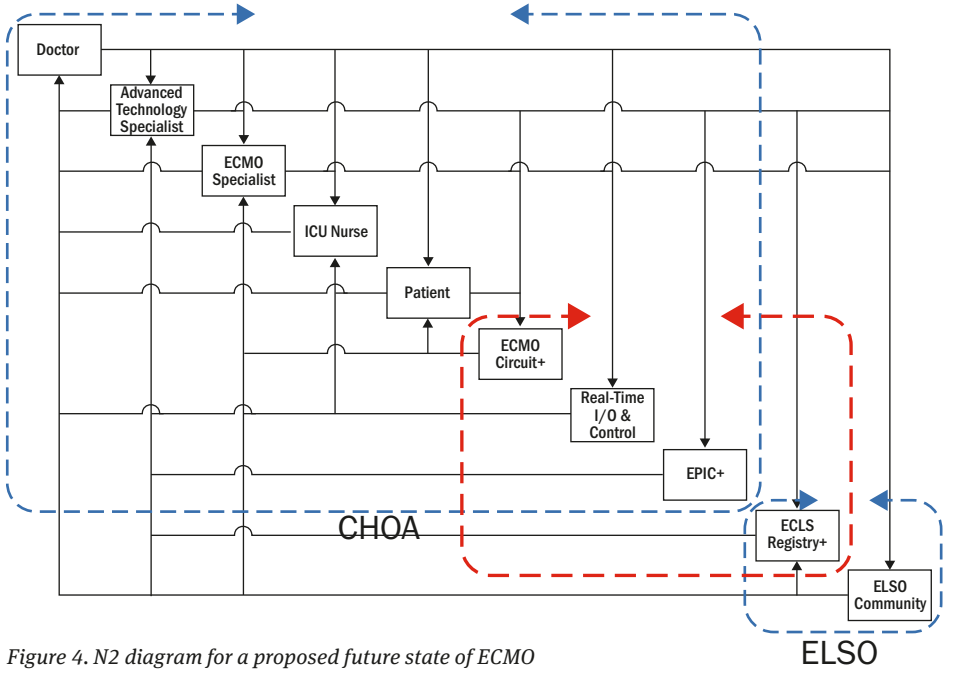


Figure 4. N2 diagram for a proposed future state of ECMO

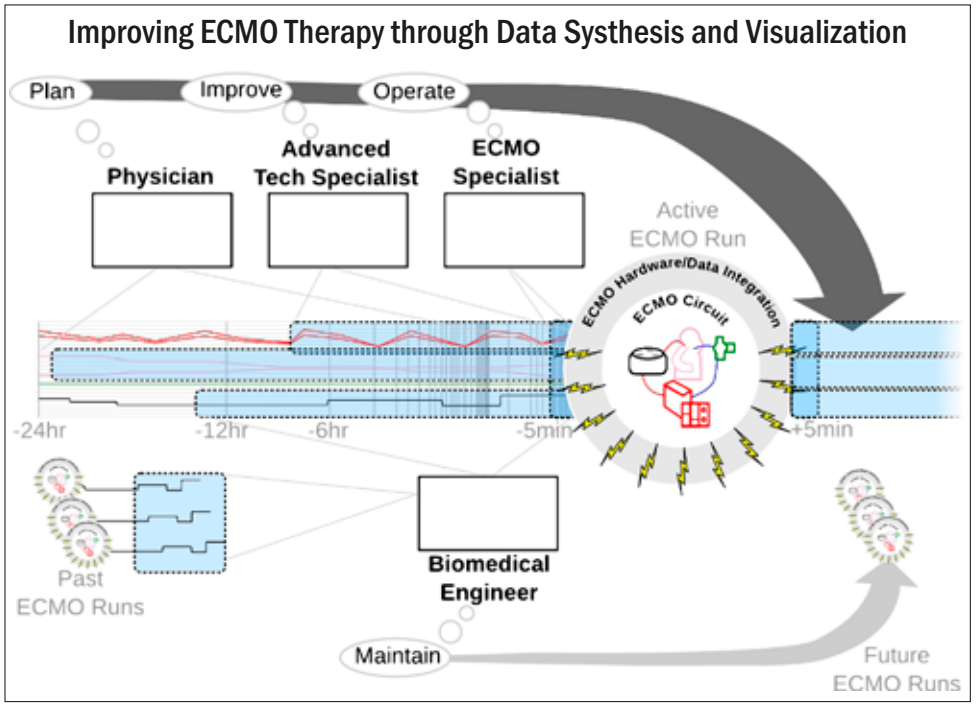


Figure 5. OV-1 depicting proposed ECMO end state

the key components, operators, and environment after the Georgia Tech project's improvements to data synthesis and visualization. The diagram shows that information visualization (shown by the graphs to the left of the ECMO circuit and the past ECMO runs graph) and data synthesis (represented by ECMO Hardware/Data Integration) allow each ECMO circuit operator to focus on patient information that is tailored to specific needs, and at the most relevant time intervals.

Stakeholder Interviews. With some knowledge of the system of interest, the Georgia Tech team met with relevant stakeholders to gain a clear understanding of the expectations for the project. While the team was able to gather a large amount of information, there were no clearly defined system requirements. It was clear, however, that the scope of the challenges presented by the ECMO system were much broader than any one academic team could complete in twelve weeks.

Rather than attempt to develop requirements for the system as a whole with stakeholders who were not sure where to begin, the team performed a gap analysis to determine what kind of improvements may be possible in twelve weeks' time. As a result, they proposed a series of current and future projects, using outputs of early projects as inputs to later ones. By proposing these much smaller projects and their potential inputs and outputs, the team hoped that the future student teams and the stakeholders would be able to focus on manageable slices of improvement. Some projects may entail a formal requirements-development phase, while some may be better suited for an agile or iterative approach, without formal requirements. The team feels our approach does not hinder either method and leaves the decision to future teams and stakeholders.

The team conducted interviews with eight users, and collected the information informally as notes and more formally as context maps. The team interviewed seven CHOA personnel including a pediatric cardiologist, a pediatric pulmonologist, four advanced technology specialists, and one ECMO Specialist, as well as a representative from the pump-apparatus manufacturer used by CHOA. One missing class of stakeholder that would have been most valuable was that of the biomedical engineer who is responsible for maintenance of the system, but meeting with the biomedical engineer was not possible in the time available. Future teams are strongly encouraged to seek out this input. The interviews led to a series of invaluable observations that the team has integrated into its approach. These are described in more detail below.

Preservation of Best Practices. Among the key observations was a need for maintaining therapy patterns and strategies over time. This applies to the application of ECMO within an organization, as well as in the community as a whole. Currently, one of the best ways to arrive at an informed plan for each unique patient situation is to personally contact a member of the ELSO community, or a site (such as CHOA) that performs many therapies a year. Very little automation exists, and the long turnaround for requests for data from the ELSO registry (on the order of days to weeks) is not sufficient to meet therapeutic goals (on the order of hours). On learning of this, the Georgia Tech team added a project of working with the ELSO community to create a cross-site ontology and access mechanism. This effort would help the registry become a resource for daily operations instead of merely for research.

Signal-to-Noise Ratio. Upon visiting the intensive-care unit at Children's Healthcare, and after many of the conversations with stakeholders, it became clear that there is a potentially overwhelming amount of information presented to the operators at any given time from many different locations around the bedside. Additionally, the operator must constantly assess the relevance of each piece of information to the therapy at hand. Based on this observation, the team proposed ways to improve the presentation of information to the operator in a more unobtrusive manner, in line with the time-critical nature of the many tasks required.

Physical Analogs. In the medical world, the presentation of information and the design of control surfaces are sometimes misleading. In some cases, interfaces have printed information in non-English languages without English equivalents. All the information presented should be as close as possible to the patient as possible. In

response to this problem, the team proposed to create design flows that mimic the layout of the circuit and placement of the patient in the displays as closely as possible.

Design Perspective of Equipment. Aside from presentation format, the content of presented information on medical equipment is variable. A significant dichotomy exists between systems with a machine-focused concept of data and those that are patient-focused. An example is two different brands of dialysis machines, both employed in various ECMO environments within CHOA. One brand presents fluid flow in terms of the volume of flow brought into the machine, while another represents the volume returned to the patient. This led to confusion in some cases of documentation, specifically if the fluid volume should be shown as additive or subtractive. On the whole, the patient is best served by taking measurements relative to the patient.

People Don't Use Technologies that Don't Work Every Time. Several conversations revealed a major limitation to technology insertion in the intensive-care unit. A prime example is the proximity-access mechanism that was integrated with the CHOA data carts. In theory, this mechanism would allow a user to sit in front of a cart, and without touching the keyboard, be identified and have current data of interest presented to that user. However, in practice, the system would authenticate someone walking by the system, logging out the current user. Further, the system required users to carry a separate, bulky card on their lanyard in addition to the employee-identification card they already carried. As a result, users opted to not carry the cards and instead perform authentication manually. This shows that unless the improvements perform as designed and benefit the user, they may fall by the wayside and cost more time and effort than they save. When inserting technology into this setting, the future state of the ECMO system must abide by the above design observations. These observations can be used to judge to value of introducing additional hardware devices, cables, biometrics, and other elements.

Domain and System Documentation and Modeling. Before the initial meeting with stakeholders, the Georgia Tech team formulated basic structural and functional models in SysML. These formal SysML models represented the team's imprecise initial understanding of the ECMO system, which allowed them to find misunderstandings quickly and easily. Through iteration with the stakeholders, the team refined these models into a state where the stakeholders could agree on the content and fidelity of the model. While the SysML characterization of the domain and system were initially foreign to the medical staff, the final models were easily understood once the engineers explained some basic concepts of SysML. The end result was well received by the CHOA representatives since they had not previously seen a formal visual representation of the ECMO system.

This effort involved information the team acquired from users, documentation, and observation. Figure 6 shows the block definition diagram used to document

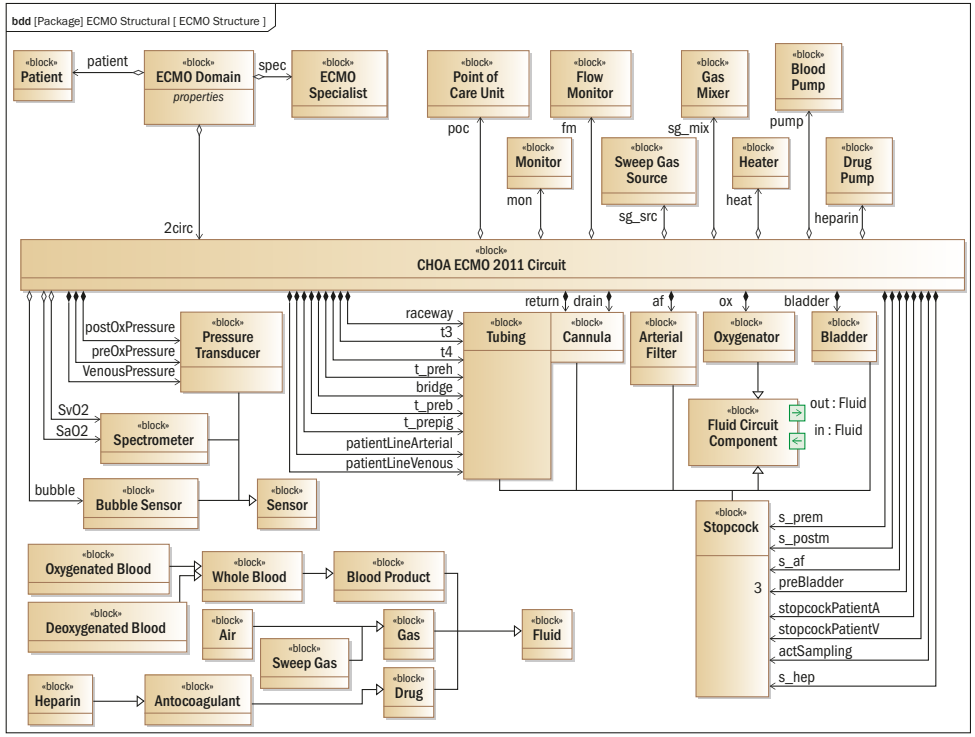


Figure 6. Current state of ECMO in a block definition diagram

the current structure of the ECMO system. This diagram shows the ECMO circuit, its SySML part properties (such as tubing), its SysML reference properties (such as the bubble sensor), and the hierarchy of fluids that pass through the system. The team also created an internal block diagram presenting interrelation information and flow of blood through the circuit.

Mock-ups and Prototypes. In preparation for the system presentation, the team built mock-up models of the data-visualization system that would be delivered as well as other possible future systems. These were assessed internally by the team for usability and value. Normally, the next step would have been user feedback and iterative design techniques, but the academic time frame for the project was limited.

Based on an initial mock-up, the team created an interactive prototype of the proposed data-visualization system. The purpose was to show a possible bedside-dashboard view, combining data in one display that is currently presented on many different machines in many different locations at the bedside. Each individual measurand is shown as a "widget" in the dashboard. The data displayed by default represents one hour of runtime. The widgets are organized both inside and outside a central box (which represents the patient). The widgets outside of the box repre-

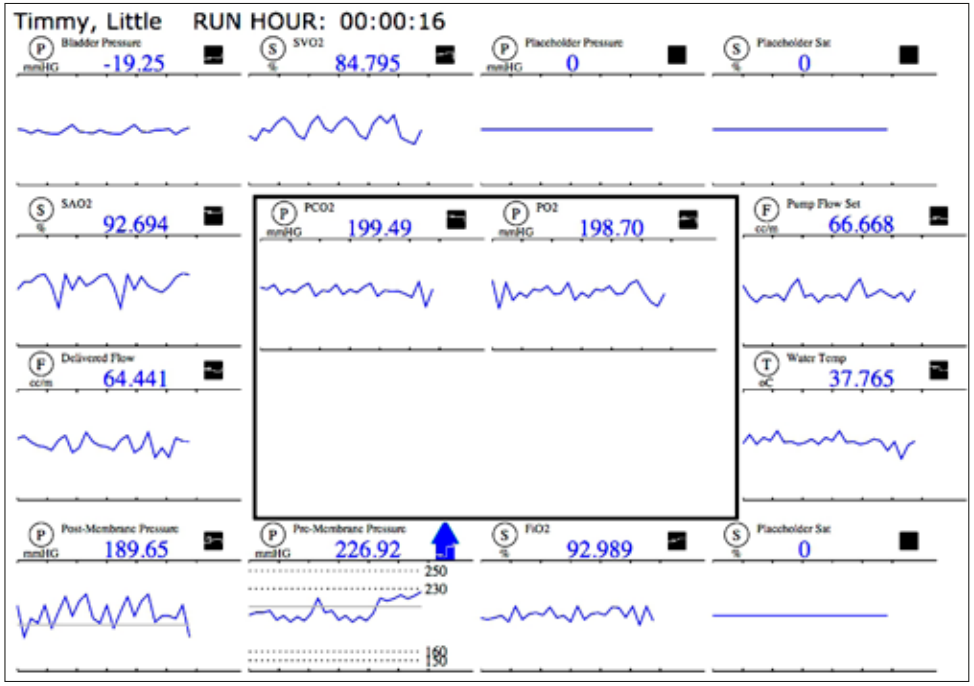


Figure 7. Screenshot of data-visualization prototype

sent measurands of the circuit itself, and widgets inside the box are specific to the patient vitals. The layout of the circuit widgets (clockwise starting from the top-right) follows the order of events which occur in the ECMO circuit.

Each widget displays the name of the measurand, and has a symbol representing the type of data being measured (as in P for pressure). The maximum, minimum, and average values for the last time increment displayed are also shown. In addition, the minimum and maximum setpoints for each measurand are also shown with a graph of the measurand over the last time increment. Finally, a small trend symbol (a red or blue arrow, or a black square) shows the general trend of the measurand over a much smaller time increment (such as five minutes by default). This allows the minute-to-minute operators to see the immediate trends, while a physician can have a more generalized overview of the last time increment.

In its current state, this display would serve primarily as a point of departure for further discussion with the target users, and help determine the requirements for the next design iteration. Figure 7 presents a screenshot of the running prototype. This is a web-based application with a customizable scenario generator. The graphs, numeric values, and trend symbols all update during a run and demonstrate the ability to have a single display for all relevant measurands. The dashboard would be customizable, so ideally a specialist could remove widgets deemed unnecessary while a physician could select a completely different set for their needs.

Proposed Future Projects

As stated previously, one of the artifacts of the Georgia Tech project was a roadmap for future proposed projects. The following describes a selection of the proposed projects and challenges as areas of future research. It is important to note that the scope of each project needed to be small enough that a team of master's students could accomplish it in twelve weeks (the duration of a capstone project), or to accomplish a significant portion while developing a path forward for a future student team.

Information Integration. This project would seek to refine the data collected by the ECMO therapy to the point that it can be more directly integrated into the electronic records system used at Children's Healthcare. This would reduce the workload of ECMO therapists, as well gather more information for use in making strategic decisions. In addition to allowing the specialist more time for patient treatment while reducing the manual input of data, this would also reduce the likelihood of incorrect data entry due to human error. Inputs to this project would be formalized key performance indicators, measures of performance, measures of effectiveness, mean time between failure, and mean time to repair.

Therapeutic Sensor Integration. This project would seek to identify therapeutic elements of the system that would benefit from measurement and control (such as time to clot or O_2/CO_2 levels) and how these elements could be measured. The long-term goal would be to automate manual measuring procedures performed periodically (on the hour, twice a day, or at other intervals). The team would research what federally approved sensors exist in the market and what control loops are associated with them. Required information for this project would be formalized therapeutic measurands of interest and the means to measure them, and could be gathered during the previous project. The output would be a market-research report of sensors approved by the Food and Drug Administration, a list of associated manufacturers, and the beginning of a plan to integrate them into ECMO.

Hardware Instrumentation. This project would seek to identify hardware elements of the system that would benefit from measurement and control (such as vibration or duty cycle) and a means to measure each. The team would perform a market survey of industrial and medical sensors. The long-term goal would be to collect data that could be used to predict the need for preemptive maintenance or replacement of hardware in the system. This project would require formalized measurands of interest and the market survey of sensors. Results would be a framework for analysis and an approach to inventory-management that would provide the basis for detection and eventual prognostication; all this would support a condition-based maintenance program.

Redundancy Characterization. This project would seek to characterize how redundancy of system components could improve failure modes to something other

than catastrophic, which is currently the case. By creating a systems dynamic model of ECMO, or using practices such as functional hazard assessment to assess risk, the team could accomplish the long-term goal of an analysis of alternatives for configurations including redundant components. The input for such a project would be formal documentation and models of the existing configurations, all available failure data, trade-space analysis and technical specifications for components of the existing system. The output of the project would be a systems dynamics model for redundancy characterization, among other analyses. To truly perform this project, meaningful failure data would be required, as would a survey of the current configurations of sites other than CHOA. The team feels this would require the hardware instrumentation to happen first, and would need the data gathered from these sensors for some time before attempting this project.

Portability Analysis. This project would analyze the potential for creating a more portable configuration of the ECMO system. Using the information gained from the systems dynamics model and the development of requirements for portability, the team would analyze the shortcomings of portability in the current configuration. Inputs to the project would be the systems-dynamics-model output of the redundancy-characterization project and formal requirements for portability, with an emphasis on therapeutic requirements for sustainability and reliability. The output would be a comprehensive analysis of shortcomings of current components in terms of transportability and in-transit usage, as well as possible ways to alleviate these shortcomings.

Future Challenges. The efforts of the 2011 Georgia Tech project were based mostly on collaboration with the sponsor, Children's Healthcare of Atlanta. However, it was the intent of the group to reach out to a broader community. In this case, the Extracorporeal Life Support Organization provided such a community. The means attempted were direct analysis of a dataset curated by the University of Michigan (the ELSO Registry) and an informal questionnaire of ECMO sites about some of the characteristics of their sites and population demographics.

The results of this effort were minimal: no data was forthcoming, preventing any direct analysis. Future teams should redouble the efforts to gain access to this data, even in an anonymous, site-restricted form, in order to arrive at new conclusions. Additionally, ELSO did not publish the questionnaire in time for inclusion in our analysis. If and when this is published, this data will become available to future teams.

There is also no currently FDA-approved ECMO circuit. As such, each site can construct the circuit in any method approved by the management and legal team of a given site. Rigorous analysis would be necessary to determine the best overall circuit design as well as the one most likely to gain FDA approval. In addition, if an approved, standardized circuit were to be developed, work would be required to standardize the training for that circuit as well.

Conclusions

This article has shown how a team of engineering graduate students at Georgia Institute of Technology successfully used systems engineering to model the current state of extracorporeal membrane oxygenation therapy at Children's Healthcare of Atlanta. The tools of model-based systems engineering, specifically SysML, were shown to bridge the gap in communication between engineers and medical professionals. This suggests that the same benefit may be seen when working with professionals in other non-engineering domains. The ability to formally model aspects of the system, and display them simply enough for all stakeholders to understand, proved invaluable.

It was also shown that a systems engineering approach to ECMO could be used not only to characterize the system, but also to identify gaps that could be addressed in future projects. The prioritization of these gaps can be defined by examining the relationships between the projects and logically determining what order tasks need to be completed in. Even at this early phase of the relationship between Children's Healthcare and Georgia Tech, these processes and tools have placed both groups on the same page and provided a clear path forward to improving the state of ECMO. The group's work will benefit the specialists who operate the circuit, the physicians leading the therapy, the ECMO community as a whole, and most importantly, the patients.

References

Bartlett, R. H. 2007. "Extracorporeal Support for Septic Shock." *Pediatric Critical Care Medicine* 8 (5): 498–499.

Broome, M., K. Palmer, A. Schersten, B. Frenckner, and F. Nilsson. 2008. "Prolonged Extracorporeal Membrane Oxygenation and Circulatory Support as Bridge to Lung Transplant." *The Annals of Thoracic Surgery* 86 (4): 1357–1360.

ELSO (Extracorporeal Life Support Organization). 2012. "ECLS Registry Report: International Summary." Website of ELSO, http://www.elsonet.org/index.php/registry/statistics/limited.html.

Jamshidi, M. 2008. "System of Systems Engineering: New Challenges for the 21st Century." *IEEE Aerospace and Electronic Systems Magazine* 23–(5): 4–19.

Marasco, S. F., G. Lukas, M. McDonald, J. McMillan, and B. Ihle. 2008. "Review of ECMO Support in Critically Ill Adult Patients." *Heart, Lung and Circulation* 17 (Supplement 4): S41–47.

Swaniker, F., S. Kolla, F. Moler, J. Custer, R. Grams, R. H. Bartlett, and R. B. Hirschl. 2000. "Extracorporeal Life Support Outcome for 128 Pediatric Patients with Respiratory Failure." *Journal of Pediatric Surgery* 35 (2): 197–202.

The Facility Location for Emergency Response: A Multi-Objective Approach

Ivan Hernandez, ihernand@stevens.edu**; and Jose Emmanuel Ramirez-Marquez,** j.ramirezmarquez@incose.org

The objective of this article is to show systems engineers how to assist decision-makers in creating a system for dispatching the medicines as efficiently as possible.

Even though global resources to counter disease emergence are poorly allocated (Jones et al. 2008), some governments have devised plans for countermeasuring anthropogenic hazards. For example, in New York (New York, US), the Department of Health and Mental Hygiene has established a dispensing system for medication in the case of a disease outbreak. As the department explains on its website (2012), "If a bioterrorist attack or widespread disease outbreak ever occurs, New York City officials will open temporary emergency 'Points of Dispensing' (POD) in every affected neighborhood to provide emergency medication to protect against the threat." Dispatching these medications quickly and effectively to millions of people requires a well-organized plan involving the interplay of different organizations and different decisions. For example, various state agencies might assist in the process of selecting the best locations for stocking and dispensing emergency medications throughout the city.

The objective of this article is to show systems engineers how to assist decision-makers in creating a system for dispatching the medicines as efficiently as possible. As illustrated in figure 1, the methods presented in this paper are part of a project that intends to develop a quantitative framework to help guide emergency planners in their decision-making process. This framework concentrates on three interrelated issues that have not been concurrently analyzed with respect to emergency units:

1. Location: Determine which are the best places for dispatching medicines.
2. Staffing: Determine how many staff members and of which type to assign to emergency units.
3. Layout: Determine how to physically arrange the emergency units.

The first problem is related to the facility-location problem. In its simplest form, this problem considers a set of locations at which one may build a facility (in this case an emergency unit), where the cost of building at location i is given by f_i. Furthermore,

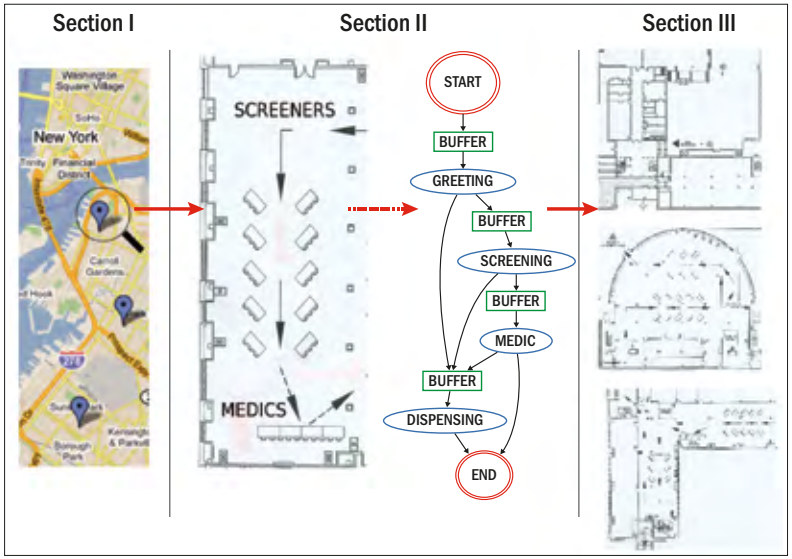


Figure 1. Emergency-plan framework

there is a set of client locations (such as neighborhoods) that need to be served by a facility, and if a client at location j is assigned to a facility at location i, a cost c_{ij} is incurred. Such a cost may be proportional to the distance between i and j, for example. The objective in the facility-location problem is to determine an optimal set of locations at which to open facilities so as to minimize the cost of the facilities and the assignment (Shmoys, Tardos, and Aardal 1997). To clarify, figures 2 and 3 illustrate a region with corresponding facilities and demand centers. Figure 2 describes a data set taken from the literature, which contains 100 demand centers (Nogueira Lorena 2010), while figure 3 describes a data set with 55 demand centers (Swain 1971). Both of these figures illustrate coordinates of potential facilities' sites in the Cartesian plane. Two important considerations need to be mentioned with respect to both figures: (1) generally, in the facility-location problem, every demand center is a potential site for a facility, and (2) for both figures, the size of the points is proportional to the demand.

The facility-location problem can be analyzed based on two

perspectives: (1) the uncapacitated case, where each facility can service an unlimited number of clients, and (2) the capacitated case, where each facility can serve a certain number of clients (for example at most u clients). In the uncapacitated case, for a given set of open facilities, finding the optimal assignment of demand centers to facilities is trivial. Each demand center is assigned to the location for which its distance is minimum (Shmoys, Tardos, and Aardal 1997). Conversely, in the capacitated case, the assignment is nontrivial. Once the capacity of a particular facility has been saturated, no more demand centers can be assigned to it (even though the facility might be their closest one). There are two variants of the capacitated facility-location problems, one variant with a single source and the other, with multiple sources. In the single-source case, it is assumed that the entire demand of a demand center needs to be served by the same facility. In the multiple-sources case, it is assumed that the demand of a particular location can be served by multiple facilities (Shmoys, Tardos, and Aardal 1997).

The uncapacitated facility-location problem has been implemented for emergency-response planning. For instance, in 2001, the United States Federal Emergency Management Agency required every county in the state of Florida to identify potential sites for disaster-recovery centers. The Emergency Management Division of the Alachua County sponsored a team project in order to identity potential sites. The project team used a mathematical-analysis tool called the covering-location model in a two-stage approach to find, recommend, and accept the locations (Dekle et al. 2005).

For the capacitated facility-location problem, Nogueira Lorena and Franca Senne (2003) have developed a set of heuristics that perform well compared to other meta-heuristic approaches, but with less computational time. For implementation examples, Church, Scaparra, and Middleton (2004) focused on supply chains and on identifying the set of facilities that would affect service delivery the most if the facilities were lost. The models presented by those researchers can be used to identify the most critical facility assets in a service/supply system.

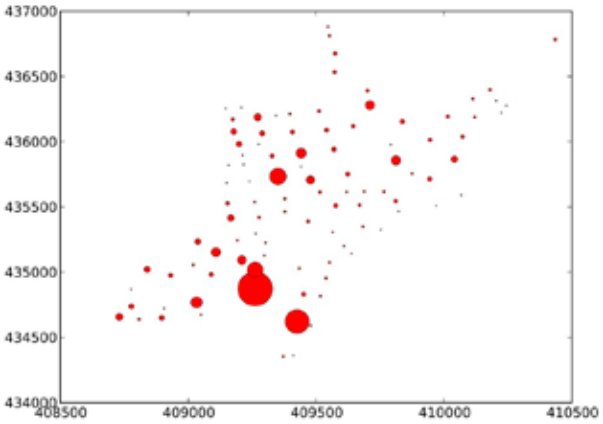


Figure 2. Coordinates of the demand centers of the SJC1 data set (Nogueira Lorena 2010). The size of the points is proportional to the demand.

Nevertheless, the current approaches found in the literature for the problem of locating emergency units may not be immediately applicable to some emergency planners. We would raise two main concerns:

1. For a given population demanding medication, how are facilities chosen from a pool of facilities with fixed processing capacity, in order to satisfy the demand in a timely manner?
2. For a given population and assuming that the facilities (emergency units) can satisfy all demand, how would the facilities' failure effect the effort to satisfy all population demand?

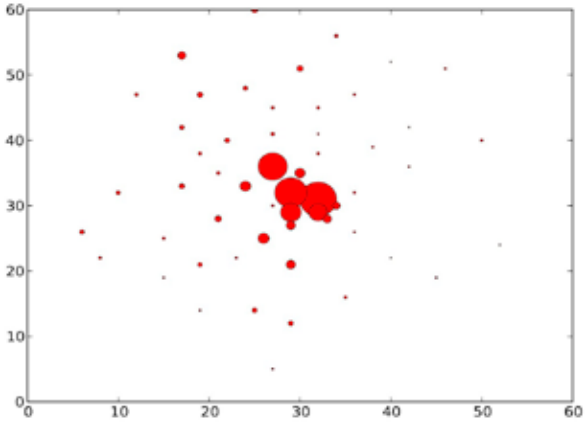


Figure 3. Coordinates of the demand centers of the Swain data set (Swain 1971)

Finding a solution to these two problems will allow response and emergency-planning agencies to identify the most critical facilities in a region. Additionally, for the first problem, solutions will illustrate trade-offs between unsatisfied demand and active emergency units as a function of cost. Additionally, solutions to the second problem will allow response agencies to evaluate the impact on time efficiency for dispensing medication as facilities fail. We believe that some emergency planners do not require a single solution but rather, an understanding of the trade-offs among different objectives of interest (for example: minimize the travel distance and minimize the number of facilities to open). Thus, we have transformed the two concerns into separate multi-objective optimization models, which we solve based on the implementation of Pareto analysis of solutions and via evolutionary algorithms. We developed these algorithms because (1) they provide insights for trade-off analysis, (2) the solution space for these problems is very large (thus, obtaining exact solutions is time-consuming and in some cases infeasible), (3) these algorithms can find multiple Pareto-optimal solutions in one simulation run.

The remainder of the article is organized as follows. In the Technical Background section, the facility-location problem and multi-objective optimization problems are introduced mathematically. The next sections, Problem Statement and Solution Approach for Our Two Models, provide the mathematical description of both optimization problems and our approach. In the Experimentation section, two data sets from the literature are applied to our approach, and highlight how the approach works. The Conclusions highlight the inferences that can be made with

our approach, the contributions made and directions for future work. We will use the following notation:

F = set of possible facilities to open.
D = set of demand centers.
K = subset of the most critical facilities of F that will fail, $K \subseteq F$.
Q = subset of facilities that remain open after a failure occurs ($F \backslash K$).
r = number of most critical facilities that will be considered to fail, $|K|$.
f_i = cost of opening facility i.
c_{ij} = cost of traveling from demand center j to facility i.
x_{ij} = binary decision variable that equals 1 if demand facility i serves demand center j.
y_i = binary decision variable that equals 1 if facility i is open.
d_j = demand of the demand center j.
u_i = capacity of facility i.
v_i = binary decision variable that equals 1 if facility i fails.

Technical Background: Uncapacitated Facility-Location Problem

For a given set of locations $N = \{1, \ldots, n\}$ and distances between them c_{ij}, $i, j = 1, \ldots, n$; there is a subset $K \subseteq N$ of locations at which facilities can be opened, and a subset $D \subseteq N$ of locations that must be assigned to an open facility; for each location $j \in D$, there is a positive integral demand d_j that must be processed by its assigned location. For each location $i \in F$, the non-negative cost of opening a facility at j is c_{ij}, per unit of demand processed. These costs also satisfy the following: $c_{ij} = c_{ji}$ for all i, $j \in D$ and $c_{ij} + c_{jk} \geq c_{ik}$ for all i, $j, k \in D$. The objective is to find a feasible assignment of each location in D to an open facility so as to minimize the total cost incurred. The problem can be stated as the following binary programming problem, where the 0-1 variable y_i, $i \in F$ indicates if a facility is opened at location i, and the 0-1 variable x_{ij}, $i \in F$, $j \in D$ indicates if location j is assigned to a facility at i (Shmoys, Tardos, and Aardal 1997):

$$\text{Minimize: } \sum_{i \in F} f_i y_i + \sum_{i \in F} \sum_{j \in D} d_j c_{ij} x_{ij} \quad (1)$$

Subject to:

$$\sum_{i \in F} x_{ij} = 1, \text{ for each } j \in D \quad (2)$$

$$x_{ij} \leq y_i \quad (3)$$

$$x_{ij} \in \{1,0\}, \text{ for each } i \in F, j \in D \quad (4)$$

$$y_i \in \{1,0\}, \text{ for each } i \in F \quad (5)$$

The objective function represents the total cost understood as the addition of the setup cost, $f_i y_i$, and the assignment cost, $d_j c_{ij} x_{ij}$. The constraints described by (2) ensure that each demand center is assigned to a facility, while the constraints in (3) ensure that demand centers are assigned only to facilities that are open. Constraints (4) and (5) describe the binary nature of the decision variables.

The Capacitated Facility-Location Problem

As previously mentioned, the capacitated facility-location problem can be analyzed assuming indivisible demand (such that the entire demand of a particular demand center needs to be served by the same facility), and assuming a particular demand center can be served by different facilities. In this article we focus only on the version of the problem with these assumptions, known as the single-source capacitated facility-location problem. The mathematical formulation of the capacitated problem is similar to the uncapacitated case, but requires additional constraints. Consider u to be the capacity of the facilities and d_j, $j \in D$ the demands of the demand centers:

$$\sum_{j \in D} d_j x_{ij} \leq u_i y_i \text{ for each } i \in F \quad (6)$$

Evolutionary Computing

Evolutionary computing is the collective name for a number of problem-solving techniques based on principles of biological evolution, such as natural selection and genetic inheritance. These techniques are being increasingly applied to a variety of problems, ranging from practical applications in industry and commerce to leading-edge scientific research (Eiben and Smith 2003). Evolutionary-computing techniques include evolutionary strategies and genetic algorithms.

Genetic algorithms were formally introduced in the United States in the 1970s by John Holland at University of Michigan. The continuing performance improvements of computational systems have made these algorithms attractive for solving some optimization problems. In particular, genetic algorithms work very well on mixed (continuous and discrete), combinatorial problems. They are less susceptible to getting stuck at local optima than are gradient search methods, but they tend to be computationally expensive. To use a genetic algorithm, you must represent a solution to your problem as a genome (or chromosome). The genetic algorithm then creates a population of solutions and applies genetic operators such as mutation and crossover to evolve the solutions in order to find the best one. The three most important aspects of using genetic algorithms are (1) definition of the objective function, (2) definition and implementation of the genetic representation, and (3) definition and implementation of the genetic operators (Wall 2011).

Multi-Objective Optimization

The rationale behind multi-objective optimization is that a significant portion of research is devoted to single-objective optimization, but most real-world problems involve more than one objective (Deb 2001, 15). In the single-objective case, optimization is a procedure of finding and comparing feasible solutions until no better

solution can be found. Solutions are termed good or bad with respect to an objective, which can be cost, for example.

Unfortunately, the traditional perspective of optimizing a single function does not allow decision-makers to concurrently contemplate different considerations (for example, to minimize the number of facilities to construct alongside the minimization of the distance between demand and facilities) and on the impact these considerations have among each other (i.e., understand the trade-off space).

When considering the selection of the facilities' locations, single-objective approaches cannot address diverse stakeholder needs. For example, using only a single objective does not allow one to analyze multiple competing objectives and multiple prospective solutions that may change based on the preference of the decision-maker. To address issues of competing optimization needs, multi-objective optimization provides a way to find solutions for mathematical models that have multiple objective functions.

Unlike optimization models with a single objective function, where a solution may satisfy the optimization criteria (that is, become the optimal solution), multi-objective models are concerned with obtaining solutions that best represent the conflicting nature of different optimization criteria. A multi-objective model allows one to find a set of solutions that describe the interaction of the different criteria. These solutions show how the improvement of a single-objective function value impacts the value of other objectives (Rocco, Ramirez-Marquez, and Salazar 2010). The solutions are usually referred to as the Pareto-optimal set, and each of the elements in the set as a Pareto optimal solution.

Thus, in the multi-objective case, the motivation is to simultaneously optimize conflicting objectives (Deb 2001), such as minimizing the number of facilities to open and minimizing the distance to travel. Given that an optimization problem that involves maximization of a function $g(x)$ can be turned into an equivalent minimization problem ($G(x) = -g(x)$), the general multi-objective optimization problem is posed as follows:

$$\text{Minimize: } G(x) = [g_1(x), \ldots, g_k(x)]^T \quad (7)$$

Subject to:

$$h_l \le 0,\ l = 1, \ldots, e \quad (8)$$

$$r_z = 0,\ z = 1, \ldots, s \quad (9)$$

where k is the number of objective functions, e is the number of inequalities and s is the number of equality constraints (Marler and Arora 2004).

The characterization of the Pareto set is as follows. Suppose each of the solutions of the Pareto set lies in an n dimensional space R^n, where n is the number of objective functions. If the coordinates of a vector Z that belongs to R^n measure positive attributes, like a firm's profit, utility of a decision maker, or the quantity of a certain good to a non-satiable consumer, it Pareto-dominates a vector V that belongs to R^n if $Z_i \ge V_i$ for all coordinates i, with strict inequality for at least one coordinate. Conversely, if coordinates measure negative attributes (loss, disutility, quantities of "bads"), Z Pareto-dominates V if $Z_i \le V_i$ for all coordinates i, with strict inequality for at least one coordinate. If an alternative X is not Pareto-dominated in a given set of alternatives, it is Pareto optimal (Voorneveld 2003).

In this article, we use evolutionary algorithms as a heuristic approach to uncover Pareto-optimal solutions to the facility-location problem. The advantage of using evolutionary algorithms is their ability to deal with noncontinuous, nonconvex and nonlinear objectives and constraints, as well as problems whose objective function is not explicitly known (for example, the output of Monte Carlo simulation) (Rocco, Ramirez-Marquez, and Salazar 2010).

It should be noted that there are other approaches to solve multi-objective optimization problems. One approach is to combine all objective functions into one single objective by assigning different weights (w_i) to each objective function g_i, a method known as the weighted sum (Marler and Arora 2004):

$$\text{Minimize: } \sum_{i=1}^{k} w_i\, g_i(x) \quad (10)$$

One then needs to vary the weights in order to find different optimal points.

A second approach optimizes a single objective function and uses the other objective functions as constraints (Marler and Arora 2004):

$$\text{Minimize: } g_j(x), \text{ for some } j \quad (11)$$

Subject to:

$$g_i(x) \le k, \text{ for all } i \ne j \quad (12)$$

One then needs to vary the values of the constraints to find the different trade-off solutions. This approach is named the bounded-objective method.

Another approach is to use goal programming, in which one sets target levels for each objective instead of maximizing or minimizing the objective functions. These target goals are treated as soft constraints (Lee 2002). For a review of the different methods used in multi-objective optimization, the reader should refer to Marler and Arora (2004).

Multi-Objective Evolutionary Algorithms

For the problems analyzed in this article, we have implemented two evolutionary algorithms: MO-PSDA (Multi-Objective Probabilistic Solution Discovery Algorithm) and NSGA-II (Non-Dominated Sorting Genetic Algorithm II).

MO-PSDA generates potential solutions (a selection of facilities to open) based on an initial specified probability distribution. For each of these solutions, this algorithm also provides their associated objective function values (the number of

facilities opened and maximum travel distance). We then analyze these solutions for Pareto optimality based on a comparison of their objective function values. After performing this step, we update the initial probability distribution (to generate potential facilities locations) as a function of the current Pareto-optimal solutions. We restart the cycle until this distribution converges to a constant set of optimal solutions or until we enforce a stopping criterion. For a more detailed description of the algorithm, the reader should refer to Ramirez-Marquez (2008).

NSGA-II is specifically tailored for dealing with multi-objective optimization problems. NSGA-II is an elitist genetic algorithm that implements a fast, nondominated sorting approach. Simulation results on difficult test problems show that in most cases the algorithm can find solutions that are better than Pareto-Archived Evolution Strategies and Strength-Pareto Evolutionary Algorithm. Details about the algorithm can be found in the work of Deb and others (2002).

Problem Statement and Solution Approach for Our Two Models

Model 1

The first model presented in this article considers facilities that do not fail and that can serve a demand that is less than or equal to u. The objectives to optimize are these: (1) the minimization of the number of facilities to open, (2) the minimization of the maximum distance, (3) the minimization of the unsatisfied demand (that is, demand not processed by any facility), and (4) the minimization of the excess of load received by the facilities (the difference between the capacity of the facility and the demand assigned to it). The corresponding mathematical representation is as follows:

$$\text{Minimize } \sum_{i \in F} y_i \quad (13)$$

$$\text{Minimize: Max: } d_j x_{ij} c_{ij}, \text{ for all } i \in F, j \in D \quad (14)$$

$$\text{Minimize: } \sum_{i \in F} \sum_{j \in D} d_j (1 - x_{ij}) \quad (15)$$

$$\text{Minimize: } \sum_{i \in F} \sum_{j \in D} u_i - d_j x_{ij} \quad (16)$$

To the best of our knowledge, the capacitated facility-location problem has not been previously analysed considering multiple objectives. The rationale behind this model is as follows. The objective function in (13) is minimized because opening a facility implies the use of resources (primarily money). However, as per (14) the interest is also in minimizing the maximum distance, a conflicting interest with (13). Note that this is a worst-case-scenario approach for the traveling distance. Ideally, demand should always be satisfied, and thus (15) requires the minimization of unsatisfied demand. However, such a requirement can translate into having overflow by a facility. Thus (16) requires the minimization of excess load.

In order to solve this problem with the evolutionary algorithms previously described, we have developed a vector (or chromosome in evolutionary-algorithm jargon) P, where each p_i is an integer number with $i = 0, \ldots, |F|$. The length of the chromosome is $|D|$. The rationale of having an integer-based chromosome is to let the evolutionary algorithms not only choose which facilities are going to be open, but also to let the algorithm select the assignment of demand centers to facilities. The element p_i indicates which facility the demand center i is assigned to. If it is zero, it means the demand center is not assigned to any facility. The Pareto set obtained by the algorithms provides a trade-off space among the solutions. For example, emergency planners would be able to understand how the allocation of resources for opening more facilities translated into the reduction of travel costs and reduction of unsatisfied demand.

Model 2

For this model we assume that (a) facilities can experience failures (for example, due to catastrophic events) and (b) facilities have infinite capacity. The objectives to optimize are these: (1) the minimization of the number of facilities to open; (2) the minimization of the maximum distance; (3) the minimization of the maximum distance, given that the most critical facilities will fail; and (4) the maximization of the number of facilities that will fail. Its mathematical representation is as follows:

$$\text{Minimize: } \sum_{i \in F} y_i \quad (17)$$

$$\text{Mininimize: Max: } d_j x_{ij} c_{ij}, \text{ for all } i \in F, j \in D \quad (18)$$

$$\text{Minimize: Max: } d_j x_{ij} c_{ij}, \text{ for all } i \in Q, j \in D \quad (19)$$

$$\text{Maximize: } r = \sum_{i \in K} v_i \quad (20)$$

This second model is considered for cases where the response agency tries to optimize the worst-case scenario (that is, minimize the maximum distance). It simultaneously considers the cases when the facilities work perfectly—objective function (18)—and when there are r failures, as in objective function (19). The solutions to this multi-objective-optimization model would be robust: if two solutions were to have the same maximum distance when properly working, the one that has the smallest maximum distance after the failure of r facilities would be considered better.

In order to solve this problem with evolutionary algorithms for multi-objective optimization, we designed a chromosome P, where each p_i is a binary value that equals 1 if facility i is open and zero otherwise. The length of the chromosome is $|F|$.

It is worth explaining that the subset K of the r most critical facilities is determined as follows: First, the demand center whose distance to an open facility is the greatest is determined (d_{max}). Then, the distances of all open facilities to d_{max} are computed and stored in a list L, where $l_i \in L$. Finally, the list L is sorted in ascending order. The maximum distance from all demand centers to an open facility, in case the r most critical facilities fail, is given by the element of L at position r,

namely L_r. For the experiments conducted using model 2, all possible values of r are considered. A more detailed explanation of how to determine the subset of most critical facilities using the maximum distance is presented by Medal, Pohl, and Rossetti (2011).

Experimentation

For the experiments presented in this article, based on the chromosome P, we used the framework called Evolutionary Computations in Python (Garrett 2011). This framework is an open-source library that contains a generic implementation of NSGA-II and also provides a set of data structures with which we implemented the algorithm MO-PSDA.

In order to obtain a better approximation to the true Pareto set, and due to the probabilistic nature of the algorithms, we analyzed four independent runs of the algorithms NSGA-II and MO-PSDA. The final Pareto set is a combination of the eight independent runs, pruned via the Pareto rationale. This approach allowed us to improve the results obtained by each algorithm alone. It also allowed us to understand the behavior of each evolutionary algorithm by comparing the results of the two different algorithms. The experimentation framework is as follows:

General Algorithm

- For $i = 0$ to *maxRuns* do
  1. Create a population P_n, $|P_n| = p$.
  2. Evolve the population P_n with NSGA-II for g generations and obtain the nondominated solutions archive A_n.
  3. Create a population $P_m = |p|$.
  4. Evolve the population P_m with MO-PSDA for g generations and obtain the nondominated solutions archiveA_m.
  5. Append A_m and A_n to the *FinalArchive*.
- Remove non-dominated solutions from *FinalArchive*.
- Return *FinalArchive*.

Experimentation Test 1

This case considers the Nogueira Lorena data (2010) set, as previously described in figure 2, for the multi-objective-optimization problem described in model 2. For both algorithms we used a population of size 50. The number of generations was set also to 50. The total time (for the four runs) for MO-PSDA was 447.39 minutes. The total time for NSGA-II was 78.15 minutes. MO-PSDA took a relatively long time to compute because the algorithm keeps a very large set of nondominated solutions through each cycle. Every time MO-PSDA performs a check to determine if a candidate solution should be included in the nondominated set, it performs several comparisons. MO-PSDA found 36.79% of the final solutions, out of a total of 280. We can see that for this example NSGA-II performed better than MO-PSDA.

Figure 4 shows the results of four different runs of the NSGA-II algorithm. Each Pareto set shows the different trade-off solutions that decision-makers can choose among facilities to open and the distance needed to travel. Also, one can trade off the unsatisfied demand with the two aforementioned objectives. The figure shows all solutions from the Pareto set for which the overload of the facilities—objective function (16)—was positive.

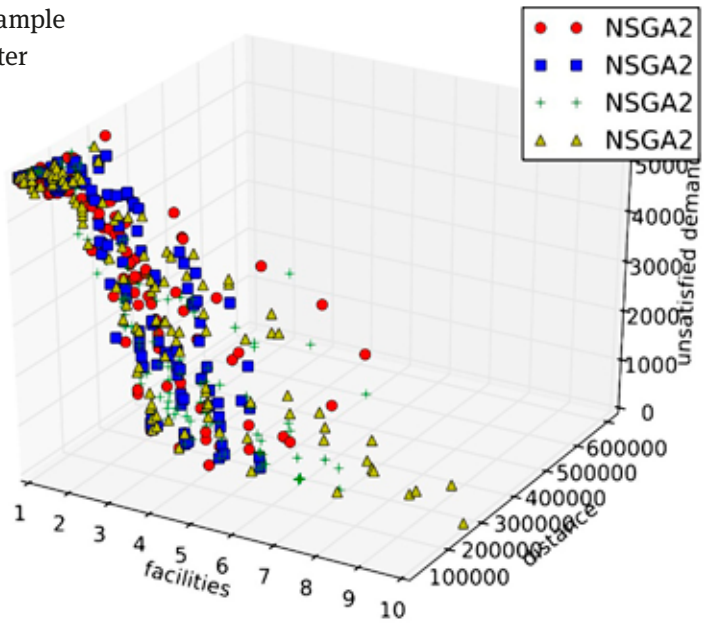


Figure 4. Four Pareto sets obtained through NSGA-II

Figure 5 shows the Pareto set that is obtained by combining the four Pareto sets of NSGA-II and the four Pareto sets of MO-PSDA. It represents the final set of solutions from which decision makers choose the solution that best meets their criteria. It can be seen from the picture that certain regions of the solution space are not explored by either algorithm.

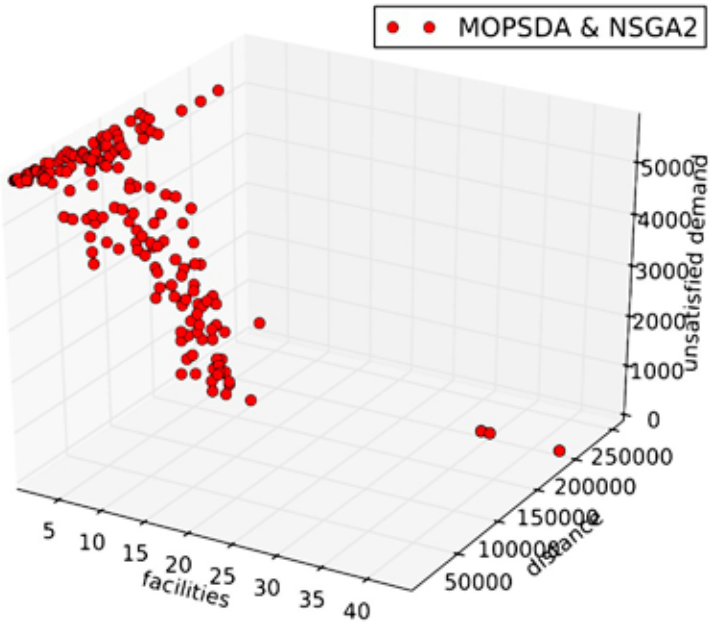


Figure 5. Combined Pareto set

From the Pareto set shown on figure 5, a sample of eight solutions is shown in table 1.

Table 1. Sample of the Pareto set for model 1

Open Facilities	Maximum Distance	Unsatisfied Demand	Load Excess
0	INF	5807.0	0
1	359.22	5803.0	-716.0
1	58544.08	4123.0	964.0
2	4031.77	5802.0	-716.0
3	13829.79	5779.0	-716.0
4	52080.76	3803.0	-700.0
36	190161.11	152.0	-719.0
44	188744.44	152.0	-719.0

Experimentation Test 2

This case considers the Swain data set (Swain 1971), as previously described in figure 2, for the multi-objective problem described in model 2. For NSGA-II and MO-PSDA, a population of size 50 was used. The number of generations was set also to 50. The total time recorded for MO-PSDA equalled 2.61 minutes while for NSGA-II was 2.80 minutes. PSDA found 29.55% of the final solutions, out of a total of 44.

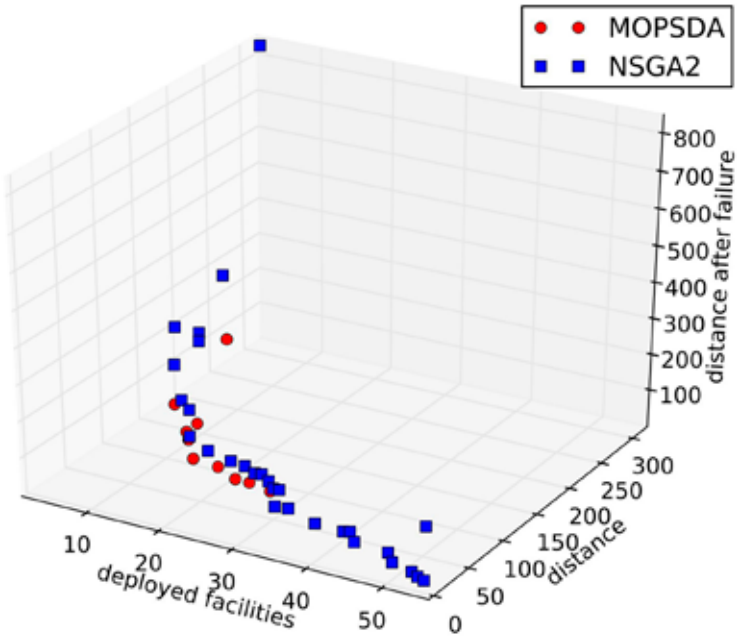


Figure 6. Comparison of MO-PSDA and NSGA-II Pareto sets

Figure 6 shows the comparison of the algorithms MO-PSDA and NSGA-II. Each Pareto set depicted corresponds to the four runs that each algorithm performs. It can be seen that both algorithms find similar solutions.

Figure 7 shows the combined Pareto set of the MO-PSDA and NSGA-II Pareto sets. The Pareto set shows the different trade-off solutions that decision-makers can choose among facilities to open and the distance needed to travel. Also, for each solution one can determine which would be the distance after the most critical facilities fail. The figure shows all solutions from the Pareto set for which $r = 1$.

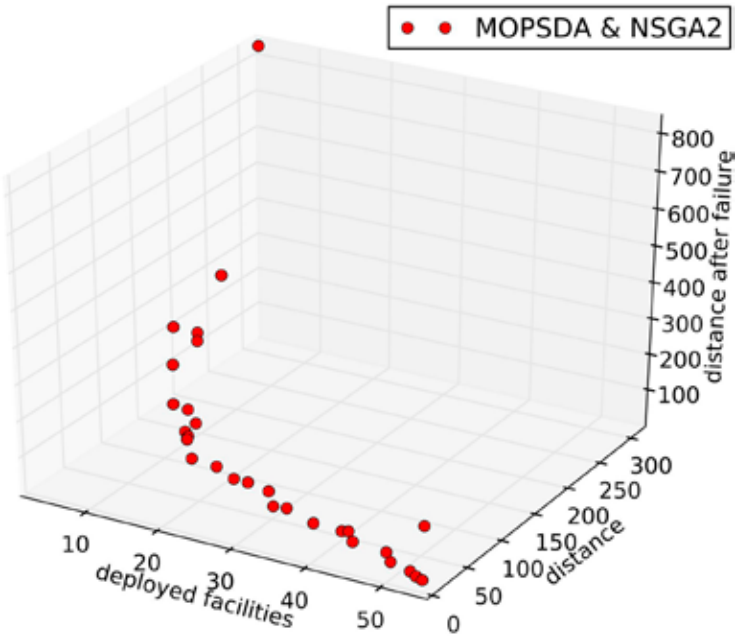


Figure 7. Combined Pareto set of NSGA-II and MO-PSDA

The Pareto sets shown on figure 7 are finally combined into a single Pareto set. From this Pareto set, eight sample solutions are shown in table 2.

Table 2. Sample of the Pareto set for model 2

Open Facilities	Maximum Distance	Maximum Distance after *r* Failures	*r* (no. of facilities that failed)
9	196.061	196.061	1
9	196.061	277.272	2
9	196.061	472.177	3
9	196.061	682.000	4
9	196.061	815.482	5
9	196.061	921.696	6
9	196.061	1117.720	7
9	196.061	1178.000	8

Conclusion

Our first model shows that opening more facilities, where in the 1 to 20 range,

significantly reduces the unsatisfied demand. However, after more than 20 facilities have been opened, there is no substantial gain (with respect to unsatisfied demand) in opening more. A similar conclusion can be reached by analyzing the total demand and the total capacity of the facilities. However, the Pareto set allows response agencies to understand solution trade-offs with respect to maximum distance for each possible solution. The second model allows response agencies to evaluate "what if" scenarios: that is, for a given set of open facilities, it allows decision-makers to understand the change in the maximum distance if the most critical facilities were to fail. The second model also lets decision-makers trade off open facilities and maximum distance to travel.

To the best of our knowledge, this is the first article to treat the capacitated facility-location problem from a multi-objective perspective. Directions for future research include the resolution of the other problems faced by response agencies, like staff allocation, and how the staff-allocation problem can be integrated with the facility location in order to model the complete problem. Another direction of future research is the use of different distance functions (such as the total distance instead of maximum distance).

References

Church, R. L., M. P. Scaparra, and R. S. Middleton. 2004. "Identifying Critical Infrastructure: The Median and Covering Facility Interdiction Problems." *Annals of the Association of American Geographers* 94 (3): 491–502.

Deb, K. 2001. *Multi-Objective Optimization using Evolutionary Algorithms*. Chichester, GB-WSX: Wiley.

Deb, K., A. Pratap, S. Agarwal, and T. Meyarivan. 2002. "A Fast and Elitist Multi-Objective Genetic Algorithm: NSGA-II." *IEE Transactions on Evolutionary Computation* 6 (2): 182–197.

Dekle, J., M. S. Lavieri, E. Martin, H. Emir-Farinas, and R. L. Francis. 2005. "A Florida County Locates Disaster Recovery Centers." *Interfaces* 35 (2): 133–139.

Eiben, A. E., and J. E. Smith. 2003. *Introduction to Evolutionary Computing*. New York, US-NY: Springer.

Garrett, A. 2010. "ecspy 1.1: A Framework for Creating Evolutionary Computations in Python." Website of Python Software Foundation. http://pypi.python.org/pypi/ecspy.

Jones, K. E., Patel, N. G., Levy, M. A., Storeygard, A., Balk, D., Gittleman, J. L., Daszak, P. 2008. "Global Trends in Emerging Infectious Diseases." *Nature* 451: 990–994.

Lee, E. K. 2002. "Optimization with Multiple-Objectives". Paper presented at the NCI-NSF (US National Cancer Institute and National Science Foundation) Workshop on Operations Research in Radiation Therapy, Washington, US-DC, 7–9 Feb.

Marler, R. T., and J. S. Arora. 2004. "Survey of Multi-Objective Optimization Methods for Engineering." *Structural and Multidisciplinary Optimization* 26: 369–395.

Medal, H. R., A. E. Pohl, and M. D. Rossetti. 2011. "An Integrated Model for Facility Location and Hardening." Unpublished paper.

Nogueira Lorena, L. A. 2010. "Problem Instances." Data file from personal website of Luiz Antonio Nogueira Lorena, Instituto Nacional de Pesquisas Espaciais, Brazil. http://www.lac.inpe.br/~lorena/instances/pmedian/SJC1.dat.

Nogueira Lorena, L. A., and E. L. Franca Senne. 2003. "Local Search Heuristics for Capacitated p-Median Problems." *Networks and Spatial Economics* 3 (4): 407–419.

New York (US-NY) Department of Health and Mental Hygiene. 2012. "Be Prepared! 10 Quick Facts on Emergency Medication 'Points of Dispensing.'" Website of the Department of Health and Mental Hygiene. http://www.nyc.gov/html/doh/html/em/mrc-pod-fact.shtml.

Ramirez-Marquez, J. 2008. "Port-of-Entry Safety via the Reliability Optimization of Container Inspection Strategy through an Evolutionary Approach." *Reliability Engineering and System Safety* 93: 1698–1709.

Rocco, C. M., J. E. Ramirez-Marquez, and D. E. Salazar. 2010. "Bi and Tri-Objective Optimization in the Deterministic Network Interdiction Problem." *Reliability Engineering and System Safety* 95 (8): 887–896.

Shmoys, D. B., É. Tardos, and K. Aardal. 1997. "Approximation Algorithms for Facility Location Problems." Paper presented at the 29th Annual ACM (Association for Computing Machinery) Symposium on Theory of Computing, El Paso, US-TX.

Swain, R. W. 1971. "A Decomposition Algorithm for a Class of Facility Location problems." PhD diss., Cornell University, Ithaca, US-NY.

Voorneveld, M. 2003. "Characterization of Pareto Dominance." *Operations Research Letters* 31 (1): 7–11.

Wall, M. 2011. "Introduction to Genetic Algorithms." Personal website of Matthew Wall. http://lancet.mit.edu/~mbwall/presentations/IntroToGAs/.

A Systems Approach to Medical-Device Compliance with IEC 60601-1:2005

Chad Gibson, chad.gibson@incose.org**; Fritz Eubanks,** fritz.eubanks@incose.org**; and Felicia Hobson,** felicia.hobson@incose.org

The systems engineer is the ideal candidate to manage the interfaces between end users, engineering, regulatory, human factors, and project management.

In order to place medical devices on the market, many countries or regions require or recommend compliance to certain international standards. One such standard is IEC 60601-1, the International Electrotechnical Commission's standard for medical electrical equipment. This standard has now been released in a third edition, identified as IEC 60601-1:2005 (IEC 2005)†. Even in countries where compliance is not mandated by regulations (such as those of the United States Food and Drug Administration), compliance with this and other standards helps facilitate regulatory clearance or approval. The international standard provides regulators with a well-known framework that helps demonstrate the safety and effectiveness of the device.

IEC 60601-1:2005, hereafter referred to as "the third edition," is titled *Medical Electrical Equipment—General Requirements for Basic Safety and Essential Performance*. Mechanical-only medical devices do not fall within this standard. The standard focuses on safety and performance of the device, and presents a multitude of requirements, including safety risk management processes, usability-engineering processes, electrical and mechanical safety testing, and labeling. The third edition requires the use of other "collateral" standards that pertain to certain aspects of safety and performance, such as electromagnetic compatibility. Certain medical devices must follow "particular" standards, which amend sections in the base IEC 60601-1 standard. One such example is IEC 60601-2-4, which defines particular requirements for cardiac defibrillators.

Changes from the Second Edition

The third edition was a significant departure from the second edition of the standard, which it is intended to replace. The second edition was published in 1988 and is now being withdrawn in many countries. The third edition makes these notable changes:

- Expands upon the concept of "Essential Performance"
- Requires heavy reliance on the results of a safety-risk management process
- Requires a usability engineering process
- Incorporates previously separate standards and requires compliance to additional standards

Essential Performance. Essential Performance is defined in the standard as "performance necessary to achieve freedom from unacceptable risk" (IEC 2005, § 3.27). The Essential Performance must be defined by the manufacturer as part of their safety-risk management process. Many of the clauses and tests of the third edition refer to Essential Performance. For example, the manufacturer must mitigate the effects of radio-frequency interference, which may cause degradation of Essential Performance. This definition of Essential Performance is a significant departure from the second edition, which relied heavily on a standard set of tests and inspections of the device (such as inspection of electrical labeling symbols or leakage current testing), and these tests were not necessarily tied to the device's Essential Performance.

Safety Risk Management. Clause 4.2 of the third edition states: Compliance is checked by inspection of the risk management file. The requirements of this clause and all requirements of this standard referring to inspection of the risk management file are considered to be satisfied if the manufacturer has:

- Established a risk management process
- Established acceptable levels of risk
- Demonstrated that residual risk(s) is acceptable (in accordance with the policy for determining acceptable risk).

The third edition requires conformance to the risk-management standard ISO 14971:2007, *Medical Devices—Application of Risk Management to Medical Devices* (ISO 2007). The first two bullets above can be satisfied by explicitly stating the manufacturer's risk-

† **Abbreviations of Standards, Correlated with Reference-List Citations**

IEC 60601-1:2005	(IEC 2005), referred to as "the third edition"
IEC 60601-1-6:2010	(IEC 2010), a "collateral standard" to IEC 60601-1:2005 (IEC 2005)
IEC 62366:2007	(IEC 2007)
ISO 14971:2007	(ISO 2007)

management policy via standard operating procedures or other forms of corporate policy that address the requirements of ISO 14971:2007, clause 3. The third bullet is satisfied through the hazard-identification, risk-evaluation and risk-control process defined in ISO 14971:2007, clauses 4, 5, and 6.

Note that the identification of essential performance is also a risk-based process, except that it assumes that a sequence of events has taken place, such that the feature or function in question has been lost or degraded, resulting in a hazardous situation. The mechanics of this process will be covered in a later section.

Usability Engineering. Consideration of usability is now required as part of third-edition compliance. Usability is the characteristic that establishes effectiveness, efficiency and operator learnability and satisfaction; it is defined by the standard IEC 60601-1-6:2010 (IEC 2010). Many principles outlined in the third edition agree with those outlined in the "safety" and "survivability" human-systems-integration domains outlined in the INCOSE *Systems Engineering Handbook*, section 9.12 (Haskins 2011, 335–336).

The third edition outlines discrete activities to address usability. Though only several pages in length, the usability collateral standard (IEC 60601-1-6:2010) requires tight integration with the entire device-development process. Usability should be considered early in the lifecycle to understand and "design out" potential usability issues before the products are realized in the physical domain. The application of the device, its primary operating functions, and safety labeling serve as inputs to the usability-engineering process (IEC 2010, §6.2.2). Usability must be verified and validated in an actual or simulated end-use environment.

Compliance with Other Standards. The third edition references two types of additional standards—"collateral" and "particular" standards. Particular standards, denoted by 60601-2-x, are standards that apply to specific medical devices. For example, IEC 60601-2-52 applies specifically to medical beds. The particular standards often define specific tests and override clauses in the base standard. Collateral standards are denoted by 60601-1-x, and are required to be evaluated along with the base standard (IEC 60601-1:2005). The second edition generally did not require collateral standards to be evaluated. The collateral standards for alarms and usability engineering (-1-8 and -1-6, respectively) have not been previously evaluated by many medical device manufacturers, and will require consideration throughout the product's entire lifecycle.

Key Compliance Activities. Most medical-device manufacturers contract with a certification body to assess a device to the third edition. The certification body's role has changed significantly from its role in assessing products to the second edition. Most notably, the certification body will review the manufacturer's usability-engineering and risk-management files in addition to product inspections and tests. The documentation provided to the certification body is much more extensive than for the second edition. Typically, second-edition documentation consisted of labeling (such as operator's manuals) and a few critical drawings or specifications in addition to a product sample for physical testing and inspections.

Approaches to Complying with the Third Edition

Medical-device development that includes compliance with the third edition provides ample opportunities for the systems engineer. Many of the more difficult problems device manufacturers face with the third edition can be addressed by leveraging existing systems engineering practices. Compliance must be addressed by multiple stakeholders from the inception of device development. In particular, safety-risk management and usability engineering span the entire system lifecycle and impact many disciplines.

This section is split into two subsections: New Product Development and Addressing Gaps in "Second Edition" Products. Many manufacturers were faced with the third-edition adoption date of 1 June 2012 in Europe. This means that all medical devices placed on the market after this date—even existing medical devices—must meet the third edition of the standard. The United States, Canada, and many other countries and regions will follow shortly thereafter (though in some regions and countries, no second-edition withdrawal date has been given).

New Product Development

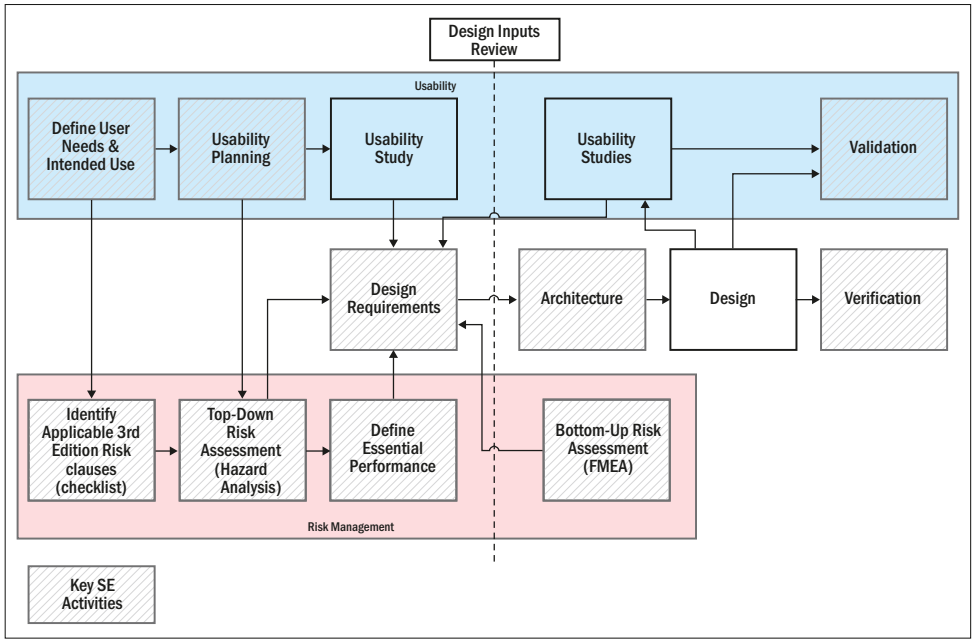


Figure 1. Typical medical-device development outputs and relationships

The third edition requires a more process-oriented approach than the second edition did. Compliance to the third edition can be approached in a similar manner to the way risk is approached in ISO 14971:2007 (details of this process are described in the risk-management section below). The systems engineer is uniquely positioned to facilitate the third-edition compliance process as the owner of the trace matrix and other key development outputs submitted and reviewed for compliance by the certification body (such as the safety-risk management file). Figure 1 shows the key activities that support third-edition compliance for new product development and highlights where the systems engineer plays a central role. Arrows mean "provides input to."

The following subsections discuss the key elements in the compliance process to be facilitated by the systems engineer, with an emphasis on the new elements required in the third edition. The activities described below are often iterative in nature, requiring active monitoring and updating throughout the product lifecycle.

Definition or Acquisition of Stakeholder Requirements. Stakeholder requirements lay the foundation for subsequent design input planning. The systems engineer may or may not be the owner of this document—often these requirements may come from other divisions of the company. The systems engineer should ensure that all relevant information is included, particularly the intended use, indications for use, user population, and intended use environment. These requirements are often fleshed out in more detail as part of the initial usability assessment and use-case definitions (outlined in subsequent sections).

Safety Risk Management Planning. By one count, the words "risk management" appear in over 100 separate clauses of IEC 60601-1:2005. This is a sampling of frequently used phrases that include those words:

- "verified by review of risk management file"
- "as indicated in risk management file"
- "risk associated with [. . .] addressed in risk management process as indicated in risk management file"
- "as determined by application of risk management process"
- "addressed in risk management process as indicated in risk management file"

The risk-management file for the purposes of determining IEC 60601-1:2005 compliance will include not only the risk-management plan, assessments, and summary reports, but also product requirements, hardware and software specifications, and verification test reports. Traceability from the outputs of the safety risk-management process to design and verification documentation will be an important element in the compliance-evaluation process.

The systems engineer might prepare for the risk-management process by determining which clauses of IEC 60601-1:2005 apply to the new product. A well-defined intended-use statement, often in the stakeholder requirements, is an important input to this activity. The clauses that apply would then be ported into the hazard analysis and addressed as part of the risk-management process. A justification should be documented for the clauses that are deemed not to apply to the new product. It will be necessary to provide this information to the certification body for their third edition compliance assessment.

The first two elements of IEC 60601-1:2005, clause 4.2 can be satisfied by explicitly stating the manufacturer's risk-management policy via standard operating procedures or other forms of corporate policy. The third element is satisfied through the hazard-identification, risk-evaluation and risk-control process defined in ISO 14971:2007, clauses 4, 5, and 6. The manufacturer must answer these questions for the evaluator:

- Have all known and foreseeable hazards been identified?
- Have all known and foreseeable causes resulting in hazardous situations been identified?
- Have all unacceptable risks been either (1) controlled or (2) shown to be as low as reasonably practicable by appropriate risk-benefit analysis?

Top-Down Risk Analysis. A typical top-down risk assessment conducted in accordance with ISO 14971:2007, clause 4, starts by identifying potential hazards (sources of harm), then proceeds to identify events or causes resulting in hazardous situations that in turn have the potential to cause harm. Checklists and guide questions like those appearing in ISO 14971:2007, annexes C and E, provide good tools for identifying general hazards associated with medical devices. Hazards that are not applicable to a specific device are usually not addressed in a hazard analysis or a failure-modes-and-effects analysis.

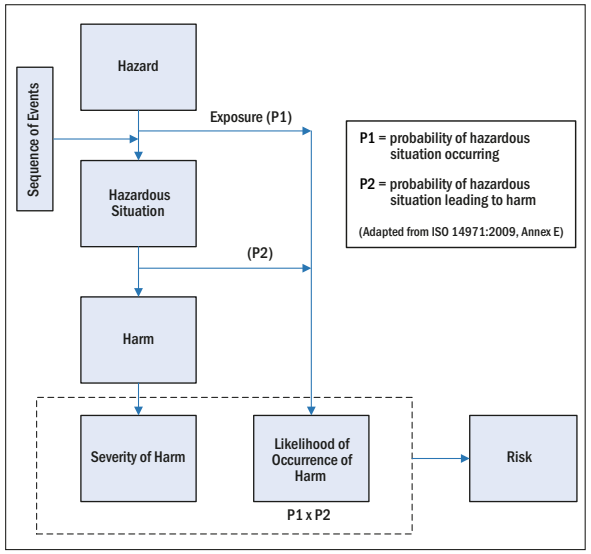


Figure 2. Relationship of risk-assessment elements (Adapted from ISO 14971:2007, figure E.1)

An additional checklist will be required. This checklist must explicitly demonstrate that the safety-risk management process has addressed all clauses that require inspection of the safety-risk management file for compliance. Clauses that are not applicable would be

designated as such, where clauses that are applicable would contain pointers to the document containing the estimation and evaluation of the safety risk.

The risk is estimated as the combination of the severity of the harm and the composite likelihood that (1) the sequence of events results in a hazardous situation, and (2) the hazardous situation results in harm. The process is shown graphically in ISO 14971:2007, annex E, figure E.1, and is adapted here in figure 2.

The estimated risk is compared to the risk-acceptability criteria established by the manufacturer. Initial safety risks rated as acceptable do not require the implementation of risk-control measures. If the initial safety risk is not acceptable, the next stage is to progress to the risk reduction via the implementation of risk-control measures in accordance with ISO 14971:2007, clause 6.

Definition of Essential Performance. The recommended method of identifying essential performance is stated in IEC 60601-1:2005, annex A, subclause 3.27, as follows:

> Assessment of this risk is made on the assumption that the performance aspect in question has been lost or degraded, and takes account of the probability that harm would then occur (which in some instances could be 100%) and the severity of that harm. Application of the risk management process then ensures that the probability of loss of the performance aspect is low enough to make the residual risk acceptable.

The identification of essential performance described above assumes that a sequence of events has taken place, such that the feature or function in question has been lost or degraded, resulting in a hazardous situation. Therefore, the level of risk used to determine whether to evaluate essential performance is equal to the combination of the severity of the harm and the likelihood that the hazardous situation results in harm, as shown in figure 3.

Note that the likelihood of occurrence for essential performance risk (P2) is not the same as the composite likelihood of occurrence for safety risk (P1 x P2). If the manufacturer defines safety-risk acceptability only in terms of the composite likelihood, a separate likelihood index will need to be developed for use in determining essential performance. Risk evaluation proceeds in accordance with ISO 14971:2007, clause 5.

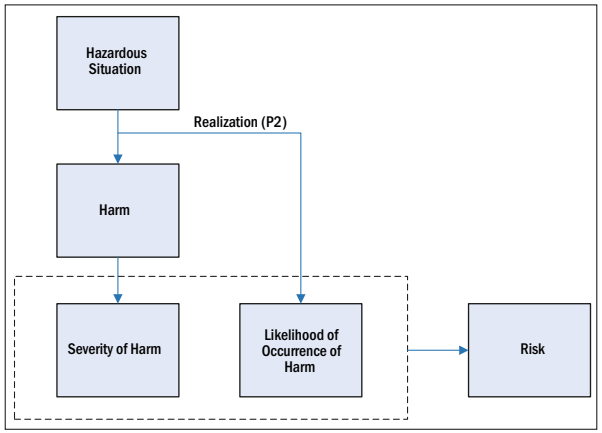


Figure 3. Essential-performance identification diagram

The process and criteria for determining essential performance (analogous to the criteria for safety-risk acceptability) are defined by the manufacturer and can be written into a safety-risk management plan or into a separate document and maintained in a safety-risk management file.

As in the case of hazard identification for safety risk, the analysis for essential performance must examine all known and foreseeable functional failures that could result in hazardous situations with unacceptable risk. Sources for compiling a list of functional failures include the statement of intended use, product-performance requirements, and applicable regulations and compliance standards.

Usability Engineering. A usability-engineering program is required for compliance to the third edition. IEC 60601-1-6, a collateral standard to the third edition, describes how to develop and follow a usability engineering process. It should be noted that IEC 60601-1-6 will eventually be replaced by IEC 62366; the standards are nearly equivalent, though IEC 62366 expands the usability assessment to include nonelectrical medical equipment. Compliance with one standard can easily be applied to the other. The primary evidence used to demonstrate compliance to the standard is a Usability Engineering File. This file provides evidence that the usability process was followed, and typically consists of "pointers" to other documentation.

Understanding the use environment early in design will help design out potential usability issues. Understanding this environment late in design (or not at all) may result in the device being "patched" or mitigated through less effective means such as labeling or training. The systems engineer should incorporate usability early in the development cycle and as part of existing engineering lifecycle practices, and continue to track and manage the effort through verification and validation.

The usability engineering process may be incorporated with many existing medical device-development activities. Since usability engineering is tightly integrated and concurrent with safety-risk management activities, many required elements can be incorporated in existing safety-risk management processes. For example, use error can be considered as part of a device hazard analysis. Once the device's critical functions are defined (that is, the "intended use" of the device), use cases, sequence diagrams, and other architecture tools and practices (including those built into SysML) can be used to understand and convey these user-device interactions. In most cases, usability engineering is best addressed through multiple validation efforts because, as IEC 62366:2007 specifies, "no validated techniques are known to exist to predict, in advance, the likelihood of a person committing a use error" (annex A.2, clause 5). An example may include an early validation of a graphical-user-interface prototype to understand potential sources of error or confusion. This may help guide selection of a particular display

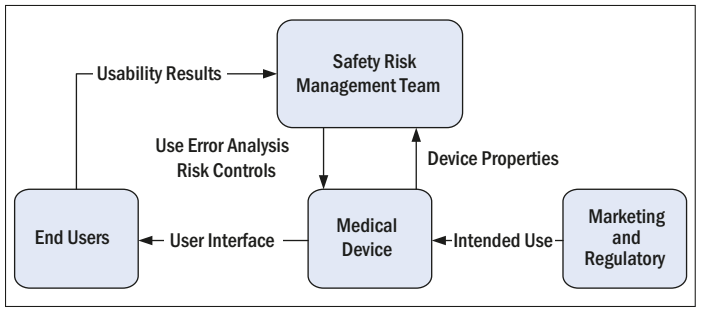


Figure 4. Usability-engineering stakeholders

technology, which would be more difficult to redesign if the usability issue was discovered later in the process.

The systems engineer may be ideally suited to bridge the gap between the users and the design, with usability experts (like cognitive psychologists or behavioral scientists) constructing the formative and summative validation efforts. Figure 4 shows the major relationships between the medical device and different usability stakeholders. Note that these flows may be iterative in nature.

Requirements and Architecture. Management of compliance to the third edition can be a significant undertaking in a project to develop a medical device. The sections noted above are only part of the development effort. In many cases, particularly with complex electrical medical devices, a well-defined requirements and architecture package may help control an otherwise difficult to manage project. While design requirements are mandatory for most medical devices sold in the United States, the European Union, Canada, Japan, and other major markets, architecture is often not mandatory (exceptions include documentation for higher-risk devices with software in the United States).

Model-based systems engineering, in particular, offers several advantages when managing compliance to the third edition. Usability engineering can be well documented through use-case and sequence diagrams. These diagrams can be used (either directly or as input) for early validation activities. Physical or functional blocks considered to be user interface elements can be identified and tracked through the use of attributes. Design elements related to essential performance can be flagged as such to aid in design and prevent downstream manufacturing issues (e.g., swapping critical parts for cost savings). Figure 5 shows examples of model-based systems engineering used to model user-centric behavior.

In the authors' work at the Battelle Memorial Institute in Columbus, Ohio (US), we have found model-based systems engineering to be quite effective when communicating with certification bodies, users, and other nontechnical stakeholders. A thorough, functional system walk-through with use-case and activity diagrams helps set the stage without pouring through detailed requirements. This has helped end users and clinical experts provide early formative validation, and it has facilitated compliance assessments with

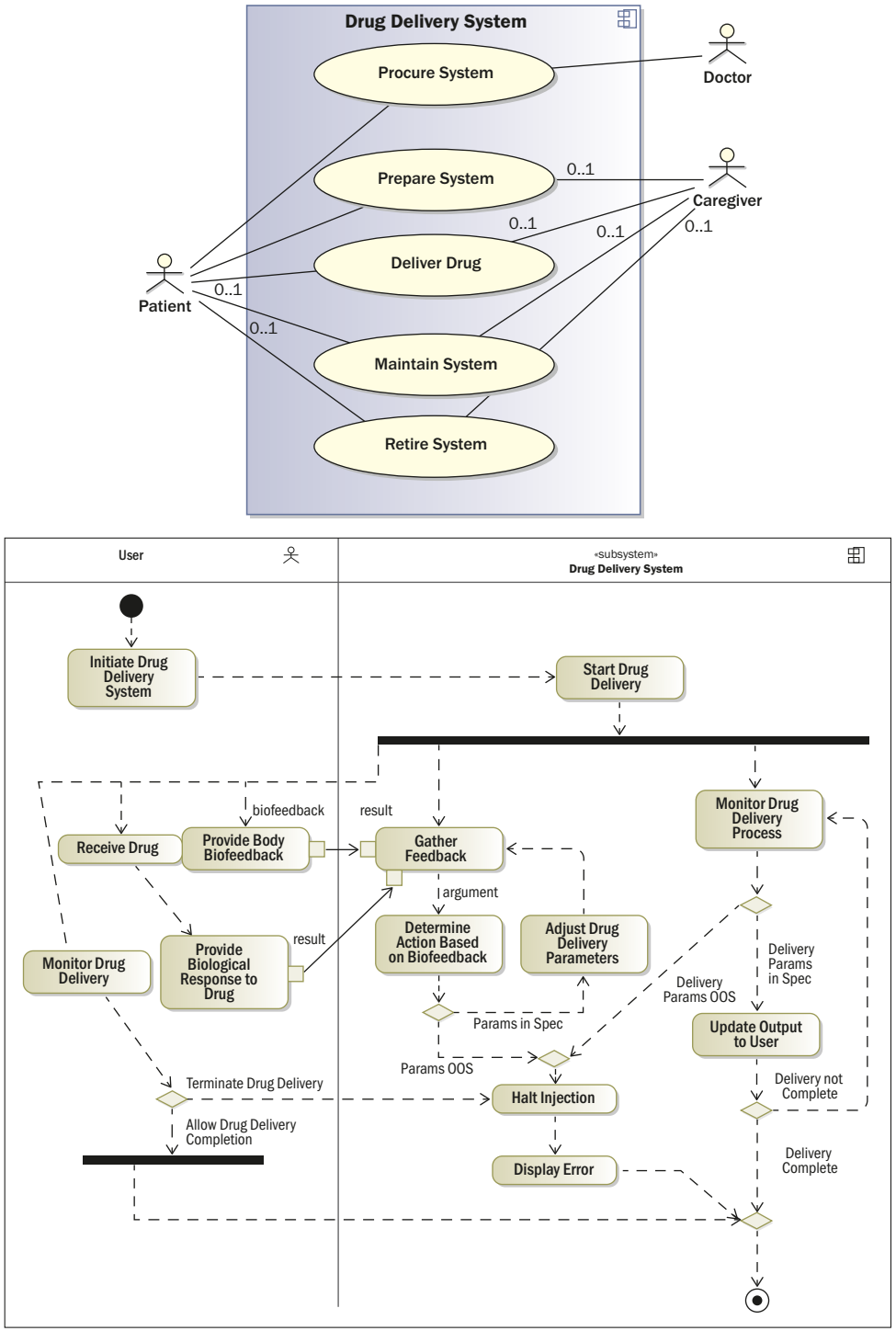


Figure 5. Usability definition using model-based systems engineering

certification bodies by allowing them to focus on meaningful activities as opposed to "getting up to speed" on product functionality.

Verification and Validation. Verification and validation provide results that are used as the evidence that the product meets its requirements—including those identified through risk management and usability engineering. The evidence completes the trace matrix (managed by the systems engineer), which is the roadmap central to a third-edition compliance review.

Addressing Gaps in "Second Edition" Products

In nearly all cases, the design and documentation of a medical device designed to the second edition must change to comply with the third edition. Many manufacturers will have to perform this activity when countries and regions begin to require compliance to the third edition, particularly those regions like the European Union that do not allow existing products to be "grandfathered" in. As shown in figure 1, the systems engineer is often the owner of design inputs and outputs that demonstrate third-edition compliance. The following subsections highlight the systems engineer's role in defining the scope and managing the implementation of changes.

Project Planning and Scope Assessment. The scale of the design and documentation deficiencies can vary widely between products; therefore, the recommended first step is to perform an initial assessment of the current design to identify both procedural gaps (in risk management, for example) and design gaps (as in new labeling). Our experience in the evaluation of several "second edition" products has shown that the primary drivers of cost and schedule will likely be these:

- Essential performance and assessment of risks outlined in the third edition will drive new risk controls, requiring product changes.
- Usability engineering was limited to design validation, and will have to be assessed more fully as part of the risk-management process.
- Miscellaneous new design-level requirements—such as those related to alarm harmonics, fire enclosures, or mains supplies—may require significant redesigns.

Unfortunately, the scope may not be fully known after the initial gap assessment, since the risk-management process itself is intended to identify required risk controls. In general, the more thorough the existing risk-management and usability processes are, the less potential there is for a significant redesign motivated by risk control.

Risk-Management Process. The second edition did not specifically require a risk-management process compliant with ISO 14971:2007, leaving the possibility that a manufacturer's risk-management process may not strictly comply with all of the clauses of that standard. In this case, the manufacturer should perform a gap analysis of its current risk-management process against the specific clauses of ISO 14971 to assure that the process can meet the requirements of clause 4 of the third edition. In most cases, risk assessments and traceability matrices will be in place to meet the filing requirements of the US Food and Drug Administration. Typical gaps are the lack of safety-risk management plans and summary reports.

In addition, a gap analysis should also be performed to determine where the existing risk-management file does not explicitly demonstrate that all clauses where inspection of the safety-risk management file is required have been addressed as part of the safety-risk management process. As stated in the previous section, an additional checklist will be required to show which clauses are not applicable, and which clauses that are applicable have not been specifically addressed in the existing risk-assessment documents. If a clause is applicable, even though the associated risk was so low as to warrant no specific consideration in the past, it must be evaluated in the risk-assessment process as evidence that the associated risk was evaluated.

Essential Performance. Because it was not required, the manufacturer may not have determined essential performance for their existing products, so the manufacturer will have to analyze this element as described above in the "New Product Development" section. Even if essential performance is known, it may not be documented in the risk-management file, so that the documentation of the risk-management file will have to be updated with the process and findings of the essential-performance evaluation.

Usability Engineering. Since a usability assessment was not required in the second edition, most manufacturers will have gaps in their design-history file for usability. If usability was considered as part of the initial development process, addressing the gaps may be as simple as creating a "pointer" file referencing existing documentation. If usability was not considered (aside from design validation), activities may be more significant and include both formative and summative user validations. Often, the systems engineer will lead the safety-risk management portion of usability engineering, while more specialized individuals (e.g., human-factors engineers or cognitive psychologists) may lead formative usability studies.

Conclusion

The third edition represents a significant change in the way medical devices have been developed. Compliance has shifted from a test-based approach to a process-based approach. Compliance now requires a systems perspective to ensure that safety-risk management and usability engineering is considered and integrated throughout the development lifecycle. The systems engineer is the ideal candidate to manage the interfaces between end users, engineering, regulatory, human

» continues on next page

Gibson et al. *continued*

factors, and project management. The systems engineer can implement a methodical approach to comply with the standard in a cost-effective manner, while at the same time ensuring the safety and effectiveness of the device.

References

Haskins, C., ed. 2011. *Systems Engineering Handbook: A Guide for System Life Cycle Processes and Activities*. Version 3.2.2 Revised by M. Krueger, D. Walden, and R. D. Hamelin. San Diego, US-CA: INCOSE.

IEC (International Electrotechnical Commission). 2005. IEC 60601-1:2005. *Medical Electrical Equipment—Part 1: General Requirements for Basic Safety and Essential Performance*. Geneva, CH: IEC.

———. 2007. IEC 62366:2007. *Medical Devices—Application of Usability Engineering to Medical Devices*. Geneva, CH: IEC.

———. 2010. IEC 60601-1-6:2010. *Medical Electrical Equipment—Part 1-6: General Requirements for Basic Safety and Essential Performance—Collateral standard: Usability*. Geneva, CH: IEC.

ISO (International Organization for Standardization). 2007. ISO 14971:2007. *Medical Devices—Application of Risk Management to Medical Devices*. Geneva, CH: ISO.

Technical Operations

Lean Systems Engineering Working Group Links Program Management to Systems Engineering

Josef Oehmen, josef.oehmen@incose.org**; Bohdan W. Oppenheim,** bohden.oppenheim@incose.org**; and Deborah Secor,** deborah.secor@incose.org

INCOSE's "LEfSE" Project

The successful application of the wisdom of lean thinking to systems engineering is not new to the INCOSE community. The work of the INCOSE Lean Systems Engineering Working Group between 2006 and 2009 culminated with the publication of a product entitled *Lean Enablers for Systems Engineering* (LEfSE). The enablers represent 147 best practices of systems engineering and relevant aspects of enterprise management based on lean principles. These principles are not covered in traditional systems engineering handbooks. They strengthen the delivery of value to program stakeholders, eliminate waste, shorten program schedule, lower cost, and vastly improve stakeholder satisfaction—a decidedly needed approach given that the largest 96 US Department of Defense engineering development programs accumulated a cost overrun of nearly USD 300 billion over a ten-year period, a staggering amount. The average schedule overrun was close to two years. Clearly, both cost overrun and schedule underperformance are not sustainable.

At its simplest, lean thinking promotes "doing the right work right the first time," "working smarter, not harder," and "doing more with less," elegantly captured into core lean principles. The LEfSE product has been reported in ***INSIGHT*** (Oppenheim 2009), in *Systems Engineering* (Oppenheim, Murman, and Secor 2011), and in the book *Lean for Systems Engineering with Lean Enablers for Systems Engineering* (Oppenheim 2011). The book contains detailed explanations of the enablers, implementation suggestions, lagging factors, case studies, as well as basics of lean thinking in the engineering productdevelopment environment. The book was reviewed in ***INSIGHT*** (Pohlmann 2012).

Numerous online artifacts exist on the working-group website (http://www.lean-systems-engineering.org/), including presentations, a video lecture, a brochure, a reference guide, published papers, and numerous other documents. The original LEfSE work was recognized with two prestigious awards: 2009 INCOSE Product of the Year Award, and 2010 Shingo Award.

A new section 3.4.3 on lean was inserted into the INCOSE *Systems Engineering Handbook*, version 3.2.2. LEfSE project leaders were invited to offer over 50 lectures and workshops in 12 countries on 3 continents, many INCOSE venues, without exception to extremely positive reception. The Lean Systems Engineering Working Group has become one of the largest of all INCOSE working groups. LEfSE has been incorporated into academic programs at Air Force Institute of Technology, Loyola Marymount University, the Massachusetts Institute of Technology, and Stanford University. Most importantly, the awareness of LEfSE among systems engineers is now more widespread than ever before. Several companies reported successes implementing selected LEfSE, including Rockwell Collins, Thales, Rafael, Honeywell, and others (see http://www.lean-systems-engineering.org/).

The Joint INCOSE-PMI-MIT LAI "LEfMEP" Project

The significant success of the LEfSE project led to the subsequent project, jointly undertaken in 2011 and 2012 by the Project Management Institute (PMI), a 600,000 member nonprofit professional society, together with INCOSE and the Lean Advancement Initiative (LAI) at the Massachusetts Institute of Technology (MIT). Dr. Josef Oehmen of LAI led and managed the project. The effort started in January 2011, when a group of LAI consortium members met at Boeing in Seal Beach, California (US), to discuss future research activities and collaborations. The meeting included lean product-development and program-

management experts from Boeing, Rockwell Collins, Raytheon, United Launch Alliance, United Technologies, and MIT. The group decided to form the core of a wider effort to bring the power of lean thinking to bear on program management. Inspired by the success of INCOSE's LEfSE project, the group set the target of developing a set of lean enablers for program management by January 2012. Collaboration with INCOSE, PMI, and MIT LAI drove the team to operate as a collaborative working group. Key success factors for this project included the support from both INCOSE and PMI. These organizations supported this project from team conception and continues today, including support from the highest levels of both organizations.

This project benefitted from several lessons learned in the LEfSE project. Just as the latter, it started by organizing a group of subject-matter experts from industry and government organizations, as well as academic institutions with a strong experience base in program management. Fourteen experts were involved in the project on a regular basis, conducting weekly teleconferences. Several experts involved in the earlier LEfSE project joined the current SMEs, including two co-chairs of the INCOSE Lean Systems Engineering Working Group. This turned out to be a profound milestone, assuring subsequent seamless integration of lean systems engineering and lean program management.

The project included the subject-matter experts (SMEs) and a larger community of 140 practitioners from 80 organizations, called the Joint INCOSE-PMI-MIT Lean Program Management Community of Practice. This community was invaluable in several phases of the project responding to surveys, taking part in several large professional meetings reviewing the work, and providing valuable feedback. As was the case with LEfSE, we decided to develop and release the results to the public free of charge.

The phase of brainstorming LEfSE was based on a rather informal tacit knowledge approach (Webb 2008), relying on the wisdom and experience of participating experts. The present project, by contrast, adopted a more formal two-step approach:

1. Identify the most relevant challenges in program management from the program management community
2. Collect a set of proven lean enablers that overcome these challenges.

Using the experience of SMEs, the LAI research database, literature search (including the publications of the US Government Accountability Office, and a comprehensive survey of the Joint Community of Practice, we identified 160 challenges, prioritized them, and organized them into 10 challenge themes, listed in table 1.

Table 1. Major challenge themes in engineering programs that lean enablers help to address

1. Firefighting—reactive program execution
2. Unstable, unclear, and incomplete requirements
3. Insufficient alignment and coordination of the extended enterprise
4. Processes that are locally optimized and not integrated for the entire enterprise
5. Unclear roles, responsibilities, and accountability
6. Mismanagement of program culture, team competency, and knowledge
7. Insufficient program planning
8. Improper metrics, metric systems, and key performance indicators
9. Lack of proactive program risk management
10 Poor program acquisition and contracting practices

The initial project intent was to develop lean enablers for program management, to be used as a complement to the *Lean Enablers for Systems Engineering*. It soon became obvious to the subject-matter experts that the two fields are strongly interrelated, and that a separate, "silo" approach to the two domains was incorrect, leading to well-known frictions and competition for resources. These findings led to a team decision to fully integrate lean systems engineering and lean program management. Next steps included incorporating all of LEfSE into the new set of enablers, and releasing the final product under the name *Lean Enablers for Managing Engineering Programs* (LEfMEP). This study was the product of collaboration across three domains of management wisdom: lean thinking, systems engineering, and program management. Using the operations-management theory of lean thinking, program management and systems engineering have been integrated to develop a set of unique, relevant, and actionable recommendations for program managers—the LEfMEP. The result represents an exemplary case of multidisciplinary cooperation, praised at the highest levels of both PMI and INCOSE, including INCOSE's Corporate Advisory Board.

The final result was released as *The Guide to Lean Enablers for Managing Engineering Programs* (Oehmen 2012), published jointly by PMI, INCOSE, and MIT LAI. The Lean Systems Engineering Working Group became the home for the current project: its website features the guide as well as subsequent works (http://www.lean-systems-engineering.org/). The core of the guide contains the 10 challenges mentioned above, as well as LEfMEP consisting of 43 lean enablers with 286 sub-enablers (total of 329 practices) to overcome these challenges, better integrate program management and systems engineering, and lead engineering programs to excellence.

Four workshops were organized by the team leaders during 2011: at MIT, at the INCOSE International Workshop and International Symposium, and at the PMI Global Congress. These events gave working-group members and interested workshop participants the opportunity to engage in stakeholder dialogue and elicit feedback from more than 180 participants. Two surveys of industry and government practitioners validated the findings of the group's work. One prioritized the program-management challenges, and the other validated the suggested LEfMEP. The LEfMEP were validated further by comparing the enablers with the management practices of published highly successful programs. The enablers correlated strongly with the successful programs, and also the unsuccessful programs analyzed demonstrated the much weaker use of the enablers. The survey conducted to validate the LEfMEP clearly showed that programs that use enablers demonstrate a significantly stronger performance in all dimensions—cost, schedule, quality, as well as stakeholder satisfaction (Oehmen 2012).

The 329 enablers and subenablers are presented in the guide under six lean principles: respect for people, capturing value as defined by the customer, mapping the value stream, maintaining flow through value-adding processes, letting customers' needs pull value, and pursuing perfection in all processes.

The guide contains extensive sections on fundamentals of lean thinking, with excerpts from Oppenheim (2011). These sections include key concepts for better integration of program management and system engineering; the major engineering program management challenges; a comprehensive list of the enablers and subenablers; concrete advice on how to implement the LEfMEP in both new and existing programs; and industrial, governmental, and academic barriers to the use of the Lean Enablers in the current program environment.

An appendix to the guide presents seven different mappings of the enablers and subenablers onto the following domains, to assist in identifying the enablers that are most relevant for a particular program:

- Detailed mapping of LEfMEP against engineering program challenges
- Detailed mapping of LEfMEP against program-management performance domains
- Detailed mapping of LEfMEP against the INCOSE systems engineering processes
- Detailed mapping of the LEfMEP against the LEfSE

The guide provides valuable insights for key stakeholders in an engineering program. Stakeholders include program managers, functional managers, continuous improvement and auditing functions, risk managers, commercial and government customers, corporate leadership, and all professionals in an engineering program.

The guide contains extensive advice for implementation of the enablers. It is not necessary (or advisable) to implement all the LEfMEP at once. Clearly the improvement needs for any given program should be prioritized based on the 10 major challenges discussed in the guide. Those lean enablers which promise the highest level of improvement should be selected for implementation effort.

Lean thinking, aiming to create the best value for the program stakeholders, with minimum waste and in a minimum of time, is common to all types of programs: commercial and government, engineering and social transformation, large and small. The lean enablers presented in the guide were developed from the challenges observed in recent large-scale engineering programs, requiring millions to several billions of dollars, which included aerospace and defense programs, systems or missions, large-scale infrastructure developments, development and integration of complex information-technology systems, and development of new commercial product lines. However, the experts who developed the enablers made a significant effort to ensure that the enablers were applicable to other types of programs, for example, organizational-change programs (as in cost reduction, restructuring, or post-merger integrations), and social-transformation programs (such as reducing childhood obesity or preventing and treating post-traumatic stress disorder). While the subject-matter experts are mostly based in the United States, strong attempts were made to incorporate a global perspective through the extended Joint Community of Practice and the international workshops where the results were discussed.

Conclusions

We have come to accept that big programs mean big problems, big cost, big risks and big delays. In addition, we accept that there is friction and misaligned goals between functional silos. There are conflicts among customers, contractors, and suppliers that lead to frequent irritations, animosity, and open hostility. Critical skilled resources compose reports rather than engineering or managing programs. Conveniently, the excuses for doing so are endless: no time for managing the program better because everyone is busy fixing problems, requirements change all the time, regulations and compliance replace efficiency, new technologies fail, suppliers do not stick to their promises, and qualified people are impossible to find.

The guide has been written for managers and engineers who are willing to take on the challenge to lead their program to excellence. This guide can help them execute a program where the key program stakeholders understand how they make a difference for their customers, their internal organization, and society at large; where professionals collaborate seamlessly over functional and organizational boundaries; where processes run like clockwork, delivering what is needed and

» continues on next page

Oehmen et al. *continued*

when it is expected. In short: a lean program! You can run this world-class program, and the guide has been written to help you do that. The best practices for managing engineering programs, which have been condensed into the lean enablers, are basically "good sense." We hope the guide will contribute to making them "common sense" as well.

References

Oehmen, J., ed. 2012. *The Guide to Lean Enablers for Managing Engineering Programs*. Version 1.0. Cambridge, US-MA: Joint MIT–PMI–INCOSE Community of Practice on Lean in Program Management. http://hdl.handle.net/1721.1/70495.

Oppenheim, B. W. 2009. "Lean Enablers for Systems Engineering." *INSIGHT* 12 (1): 35–36.

Oppenheim, B.W. 2011. *Lean for Systems Engineering with Lean Enablers for Systems Engineering*. Hoboken, US-NJ: Wiley.

Oppenheim, B.W., E. M. Murman, and D. A. Secor. 2011. "Lean Enablers for Systems Engineering" *Systems Engineering* 14 (1): 29–55.

Pohlmann, L. D. 2012. "*Lean for Systems Engineering With Lean Enablers for Systems Engineering*." *INSIGHT* 15 (1): 55–56.

Webb, L. 2008. "Knowledge Management for Through Life Support." Manuscript of unfinished PhD diss., RMIT University, Melbourne, AU.

Agile Systems and Systems Engineering Working Group Chartered — Kickoff Planned for International Workshop

Rick Dove, rick.dove@incose.org

Rick Dove of Stevens Institute of Technology and cochairs Ron Lyells of Honeywell and Michael Coughenour of Lockheed Martin, have submitted a charter for an Agile Systems and Systems Engineering Working Group, which has been approved.

Purpose. The purpose of this working group is to identify and develop a body of knowledge that will inform systems engineering and related processes that require agile system capability. Agile systems of interest to this working group include both systems engineering processes and systems produced through these processes.

This working group views agility as a sustainable system capability, enabled and constrained fundamentally by system architecture. This architecture delivers agile capability as reconfiguration, augmentation, and evolution of system and process functionality, during operation throughout the lifecycle. The architecture enables the system or process to respond to new and immediate situational requirements effectively. Effectiveness of response is measured in response time, response cost, response quality, and response scope sufficient to sustain functional intent.

Need. The need to understand sustainably agile system design and project management exists on multiple fronts:

- Agile-systems-engineering development processes have become of interest to the companies on INCOSE's Corporate Advisory Board, who are asking that INCOSE develop appropriate guidance.
- Defense organizations have an interest in how agile system concepts might inform agile acquisition processes.
- Quick-reaction capability has been a defense-acquisition need for some time and would benefit from an agile response capability by suppliers. But generally quick-reaction capability is achieved today by the employment of costly and error-prone overtime work and the increased risk of relaxing formal systems engineering processes.
- Both commercial and governmental organizations are finding that the pace of technology and growing user expectations are reducing the effective life time of deployed systems.

Confusion exists in the relevance of agile software development processes to more general systems development processes, and in the relationship of lean concepts to agile concepts. This confusion needs clarifying perspective.

A large body of experience and a variety of beneficial process approaches now exists in the area of agile software development. In the growing interest for more general agile system-project-management processes, these agile software-development processes appear to many to be a model for more general systems engineering development; but they are tailored to the specifics of the software development environment, and exist in a variety of different approaches more akin to brand-specific practice (such as Scrum and XP).

In a very general interpretation, the lean approach values efficiency of operation and achieves this mainly through process principles; the agile approach values effective response ability and achieves this mainly through architectural principles. To be sure, both are concerned with operational effectiveness. Since the two have a different means for achieving different ends they are not necessarily in direct conflict—but they often are. When efficiency dominates the requirements, a lean concept of operations should dominate, taking additional value from agile if and only if lean requirements (as required by stakeholders) are not adversely compromised, and stakeholder requirements recognize some value from agility. Vice versa, when an agile concept of operations is called for by stakeholder requirements, the design focus goes to architecture, streamlining the process with lean principles if and only if dominating agile requirements are not adversely compromised. A useful set of requirements will make the nature of lean vs. agile design tradeoff clear, when tradeoff is unavoidable. In general, an agile design should be as efficient as possible, and a lean design should be as agile as possible; but focus and values are found in the requirements.

Scope. The primary focus of this working group is on fundamentally necessary and sufficient architectural concepts and

concept-employment principles that enable any system or process to be agile, and to show how these architectural concepts and principles are or might be applied advantageously to a variety of INCOSE-relevant systems and processes of interest. These examples will be directed at the application of necessary and sufficient agility-enabling concepts and principles, avoiding prescriptive interpretation and disclosure of organization-specific competitive-advantage differentiation. Application examples will include, for instance, systems engineering and management processes, quick-reaction capability, and acquisition processes, to name only a few.

Goals

- Fundamental system engineering concepts and principles supported with application examples that enable an agile systems engineering development processes.
- Fundamental systems engineering concepts and principles supported with application examples that can inform agile acquisition processes.
- Fundamental systems engineering concepts and principles supported with application examples that can inform supplier design of quick-reaction capability.
- Fundamental systems engineering concepts and principles that can inform the design of agile systems which must respond effectively to the pace of technology and growing user expectations.

Intended Outcomes and Products

- Identification and justification (short-term) of necessary and sufficient fundamental concepts for any system or process to be agile.
- Identification (mid-term) and development (long-term) of a relevant body of knowledge appropriate for the Systems Engineering Body of Knowledge.
- Development of appropriate contributions to the INCOSE *Systems Engineering Handbook* (ongoing) that provide fundamental enabling concepts and considerations for engineering agile systems and processes and for employing agile development processes.
- Development (short-term) of an understanding of how lean concepts and agile concepts can be complimentary, and how trade-offs between the two concepts can be reconciled.
- Identification and development (mid-term) of informative examples of fundamental agile architectural concepts employed in a variety of relevant system or process applications, including acquisition.
- Socialization of work efforts (ongoing) with papers for INCOSE's journal *Systems Engineering*, papers and tutorials at the International Symposium, ***INSIGHT*** theme issues, and educational and tutorial Webinars.

Next Steps

The group will be organizing through the end of 2012, in preparation for the kick-off event at the International Workshop. This will include the creation and population of a SharePoint site, recruitment of working-group members from both INCOSE membership and external sources, development of a mailing list for opt-in announcements, and an agenda for the 2013 International Workshop in January. To get on the announcements mailing list, write to Rick Dove at the address above and include any thoughts you may have.

SEBoK Goes Live

Jack Riley, jack.riley@incose.org

On Friday 14 September 2012, INCOSE and the Institute of Electrical and Electronics Engineers (IEEE) released the *Guide to the Systems Engineering Body of Knowledge* (SEBoK), version 1.0. It is now available worldwide at http://www.sebokwiki.org/. This is one of the key deliverables of the BKCASE Project, initiated three years ago by the United States Department of Defense and the Systems Engineering Research Center. BKCASE (pronounced "Bookcase") is the acronym for the Body of Knowledge and Curriculum to Advance Systems Engineering. The BKCASE project is led by a university partnership between the Stevens Institute of Technology and the US Naval Postgraduate School. The project scope is to define a Systems Engineering Body of Knowledge (SEBoK) and use the SEBoK to develop an Advanced Graduate Reference Curriculum for Systems Engineering (GRCSE, pronounced "Gracie").

INCOSE and IEEE are the two key sponsors of BKCASE and will jointly ensure the stewardship of SEBoK and GRCSE when they are released in the fourth quarter of 2012.This version 1.0 of SEBoK, intended for broad worldwide use, adds to the rapidly maturing discipline of systems engineering. It consists of seven parts broken into 26 knowledge areas, with 112 topics. There are five use cases, seven case studies, and six vignettes to illustrate the contents. The glossary has 363 entries, and there are 224 primary references plus hundreds more additional references. Seventy contributors from around the world authored the SEBoK. Several hundred reviewers provided comments. Each author and reviewer made an important contribution to the final product.

SEBoK will make a strong impact on INCOSE Technical Operations and products. One of the first consequences is the next revision of the *Systems Engineering Handbook*, version 4.0, planned for Sep 2013.

INCOSE Operations

Feedback Now Provided on the INCOSE Certification Exam

Dave Walden, ESEP, david.walden@incose.org

Based on inputs from the community of systems engineering stakeholders, INCOSE has updated the core CSEP/ASEP exam to provide additional feedback for candidates who do not pass. Previously, only a Pass/Fail indication was given to the candidates. INCOSE still does not provide specific scores or the percentage of questions answered correctly. However, the following diagnostic aids are now provided to candidates who do not pass the exam should they choose to retake it in the future:

- **Proficient.** The score you obtained in this area is at or above the expected level. A review of this area may be helpful prior to retaking the exam.
- **Marginal.** The score you obtained in this area is slightly below the expected level. Additional study of this area is suggested prior to retaking the exam.
- **Deficient.** The score you obtained in this area is significantly below the expected level. Substantial study of this area is recommended prior to retaking the exam.

The diagnostics are provided to help identify strengths and weaknesses in each of the examination's top-level learning objectives categories. These top-level learning objectives categories loosely map with the INCOSE *Systems Engineering Handbook* as follows:

INCOSE CSEP/ASEP Exam Top-Level Learning Objective Categories	Primary Mapping to the INCOSE *Systems Engineering Handbook*
General systems engineering knowledge	Chapters 1–3
Systems engineering technical processes	Chapter 4
Systems engineering technical management processes	Chapter 5
Systems engineering organizational/ enterprise and agreement processes	Chapters 6–8
Specialty engineering activities	Chapter 9

Please note that there are different numbers of exam items in each category. Also note that a passing score is based on the overall exam results and not the results of each individual section. Therefore, improvement in any section may improve the overall chances for success on future examination attempts.

This additional feedback is currently available on the core CSEP/ASEP exam. Feedback will be added to the Acq extension exam at a later date. For more information on INCOSE's Certification Program, please use the following link: http://www.incose.org/educationcareers/certification/.

INCOSE and SEP Logo Items Now Available for Purchase

Dave Walden, ESEP, david.walden@incose.org

INCOSE has partnered with Promotion Select to offer high-quality apparel and other merchandise. The INCOSE Storefront online features both Port Authority and Cutter & Buck short-sleeve shirts, long-sleeve shirts, and wind jackets. Portfolios, travel wallets, business-card holders, and writing instruments can be ordered from the site. In addition, miscellaneous items such as travel mugs, picture frames, and acrylic awards are available. All items can be ordered with an INCOSE, ESEP, CSEP, or ASEP logo.

Group orders may take advantage of additional savings. The **INCOSE Storefront**[1] can be reached via links from the **"INCOSE Store"**[2] and the new **"Resources for SEPs"**[3] web pages on http://www.incose.org/.

1. http://www.promoplace.com/ws/ws.dll/Pres?DistID=36218&AC=459907&Login=Login
2. http://www.incose.org/ProductsPubs/incosestore.aspx
3. http://www.incose.org/educationcareers/certification/sepresources.aspx

INCOSE Spotlight

INCOSE Spotlight on. . . Brad Peck

Sandy Young, info@incose.org

Name: Brad Peck
Title: Senior Director, Systems and Human Factors Engineering, Neuromodulation
Organization: Medtronic (Member of INCOSE's Corporate Advisory Board)
Place of birth: Dubuque, Iowa (US)
Current Residence: Minneapolis, Minnesota
Domain: Biomedical
Studied: Electrical Engineering, Analog Integrated Circuit Design
Year joined INCOSE: 2010
Years in systems engineering: 12 years

How would you describe systems engineering to an eight-year-old?

Lots of things today are made up of many parts. A modern car, for example, has thousands of parts in it. Systems engineers help make sure everything works well when all the parts come together.

What did you want to do for a job when you were a little kid?

I wanted to be some form of super hero. Super strength was the preferred super ability, but the ability to fly was always a plus.

When did you first learn about systems engineering?

I first learned about systems engineering as an analog integrated circuit designer at Medtronic. At Medtronic Cardiac Rhythm Disease Management, systems engineers are the individuals who understand best how our pacemakers work therapeutically and what features physicians look for.

How did you become a systems engineer?

I was working on sensor technology for a new pacemaker feature that could automatically detect if a heart contracted from a stimulation pulse and adjust outputs accordingly. During the course of development, I migrated from working in sensor design to systems engineering, becoming responsible for the overall feature design. I basically followed the sensor technology into the therapeutic application effort.

What inspired you to become a systems engineer?

For me, systems engineering seemed to be closer to the center of the action. I was very interested in how our technology was being applied clinically and this drew me into the systems engineering role.

What does Medtronic do? How does it utilize systems engineers?

Medtronic specializes in implantable medical-device systems, primarily used in chronic-disease management. Our systems can be complex, ultra-low-power and ultra-small implantable devices with sophisticated closed-loop biological-feedback mechanisms. Typically, they communicate wirelessly to external programming that can interoperate with the larger health-care information-technology environment.

In simple terms, we use systems engineering in three main ways: (1) to collect and understand design input requirements, (2) to design and architect the system, and (3) to validate final design.

How do you explain your job to others?

"I'm responsible for systems, human factors, and technical communications within Medtronic Neuromodulation Research and Development" would be the stock answer. However, I've learned that for folks increasingly distanced from the engineering and medical professions, I often need to simplify the message by saying "I'm in research and development" or "I'm an engineer" or the ever-popular "I make pacemakers."

What current projects are you working on?

My group supports development of all new products at Medtronic Neuromodulation. We are developing implantable stimulators and

» continues on next page

infusion systems for the management of several chronic diseases, from deep-brain stimulation for Parkinson's disease to intrathecal baclofen for spasticity.

What work accomplishment are you most proud of?

The project I worked on when I first became a systems engineer is the one I am most proud of: I was fortunate to invent some ventricular-evoked response-detection circuitry that solved a non-linear challenge involving the tissue-electrode interface. I had the opportunity to take circuitry from first-principles research through to commercial release, while working within a team of engineers. It was a great experience.

What trends do you see in health care?

Personally, I see two key drivers of change in health care right now: digital transformation and economic reform. Health-care information technology is starting to catch up with the global IT trend (for instance, electronic records, connectivity and interoperability, mobile computing, cloud computing, social networking, low-power local wireless networks), which is changing both the definition and regulatory boundaries of our systems. Also, economic reform is raising the design and analysis of the economic benefits for our systems to be on par with the clinical benefits. We need to address both of these topics without compromise toward the therapeutic efficacy, safety, and reliability of our systems.

Describe your role in INCOSE. How is the organization beneficial to you?

I represent Medtronic on the Corporate Advisory Board and am cochair with Meaghan O'Neil of the Biomedical Working Group. I particularly appreciate learning how systems engineering is applied in other industries. I think it's beneficial to compare and contrast our business approach with individuals from a variety of company and industry backgrounds.

What do you like to do outside of work?

Since my wife and I are first-time empty nesters this fall, we are asking ourselves the exact same question! We're working on what to do with ourselves beyond season tickets to the theater and bothering our three kids for updates.

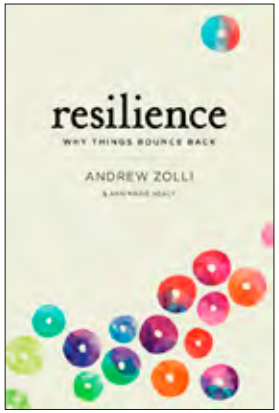
Book Reviews

The Structure of Rebounding

RESILIENCE: *Why Things Bounce Back*

By Andrew Zolli and Ann Marie Healy
New York, US-NY: Free Press, 2012 (ISBN-13: 978-1-4516-8380-6)
323 pp., including endnotes and index

Reviewed by Denise Howard, s.denise.howard@incose.org



Engineers who have been watching the nascent field of resilience through binoculars, waiting for the dust to settle and clarity to emerge, might want to put down the binoculars for a while and take up this book. Reservations about a popular-press book from relatively unknown authors aimed at a wide audience can be set aside too. *Resilience* has much to offer. It is worth returning to with pencil and paper.

Clear, sensible definitions and distinctions, at least as good as many found in the current research literature, appear in the introduction and set a foundation that is reasonably observed throughout the text. This in itself is something of an accomplishment, especially considering the breadth of the illustrative examples. They include the Mexican tortilla riots of 2007, fisheries, the fall of Lehman Brothers, terrorist networks, tuberculosis, the North American power grid, cities, World War II orphans, warfare, arsenic in Bangladeshi wells, and gun violence. The scope may seem beyond comprehension, but this diversity may actually improve the potential for discovering universalities.

The examples, along with the systems thinking integrated into their presentation, will likely draw the interest of the intended wide audience. ***INSIGHT*** readers may find value in two less prominent aspects of the book: first, the authors' search for commonalities and deeper patterns, and second, a brief description of an emerging area of systems architecting, the architecting of ecosystems.

Common structural features and solutions are sketched in the introduction and are of sufficient quality to enable an extraction of some working heuristics for systems architecting (see table 1). Having heuristics on hand as reminders and guides may make the everyday task of building for resilience easier.

Table 1. Twelve provisional resilience heuristics

Name	Heuristic	Page
Small adaptations	Since a thousand small adaptations may be much easier than a single large one, architect systems to be "alert and responsive": take advantage of sensors; micro-monitor surroundings; respond quickly and appropriately.	9
Contingent system	Whenever at least some of the conditions of disruption can be anticipated, to improve resilience consider architecting an alternative system, which will remain dormant until it is triggered at the right threshold.	10
Dynamic restructuring	In unanticipated circumstances, to improve resilience, architect the system to promote rapid, effective restructuring and reconfiguration.	10
Swarming	Especially when an enhanced response is needed, to improve resilience, consider architecting the system so that multiple units may band together for an advantage, then disband when the advantage no longer obtains.	12
Resilience in diversity	Seek sufficient diversity, since diversity may improve resilience by increasing options for rebounding.	
Diverse clustering	Consider clustering diverse assets together to improve resilience. Then guard that cluster.	12
Modularization for resilience	To improve resilience, notice how modularization decisions might impact a reconstituting of the system after disruption. When possible, make the decision in support of resilience	11
Decoupling	Consider architecting in options for selectively decoupling from the environment when disruption threatens. (Options might include reducing, reassigning, recharacterizing, buffering, temporarily suspending, or eliminating incoming resource flows.)	10
Simple core	Resilient systems are perhaps best structured with simplicity and uniformity at the core and complexity and diversity at the periphery.	11
"Holistic" intervention	Be prepared to "work in more than one mode, one domain, and one scale at a time" to improve resilience.	17
Time signatures	Be aware of time signature mismatches (e.g. milliseconds vs. months), which may "shear" and trigger cascades during disruptions.	17

Book Review *continued*

Name	Heuristic	Page
Adhocracy	Encourage adhocracy for greater resilience. ("[Adhocracy is] characterized by informal team roles, limited focus on standard operating procedures, deep improvisation, rapid cycles, selective decentralization, the empowerment of specialist teams, and a general intolerance of bureaucracy. … If it were a musical genre, adhocracy would be jazz." Skunkworks organizations may be similar.)	264

One of the most significant surprises in *Resilience* is the description of an attempt to architect an ecosystem. Certainly engineering and ecosystems have crossed paths before. Teams of ecologists, geologists, and engineers have occasionally been called on in hopes of restoring ecosystems after the removal of engineering projects, to the point of raising the idea that a restoration phase should sometimes be placed at the end of the project lifecycle. Marks (2007) and Pringle (2012) provide an introduction to this concept. This project is different, however. Microbiologist and forester Willie Smits is attempting to address and harmonize human economic needs with environmental preservation in Borneo at Samboja Lestari, a difficult, vital, and urgent emerging form of systems architecting.

Given the immediacy and impact of systems problems, having a few extra insights and some working tools taken from reading notes may indeed make a positive difference, even while the resilience field is sorting itself out. And that is the reason for taking up *Resilience*.

References

Marks, J. C. 2007. "Down Go the Dams." *Scientific American* 296 (3): 66–71.

Pringle, R. M. 2012. "How to Be Manipulative." *American Scientist* 100 (1): 30–37.

Final Thoughts

From the Chief Editor

Bob Kenley, insight@incose.org

Contributors to the upcoming issue on certification have responded to Certification Advisory Group chair Jerry Fisher with a set of articles that celebrate 10 years of the INCOSE Systems Engineering Professional certification program. This issue will demonstrate the importance and value of certification to the individual contributors and their organizations.

In addition to guiding the System Security Working Group to produce its third theme issue of ***INSIGHT***, Rick Dove will be rallying a new working group to produce a theme issue for July 2014. The inaugural meeting of the Agile Systems and Systems Engineering Working Group will occur on 28 January 2013 at the International Workshop. Those who wish to join the working group in achieving its first major milestone should attend the meeting in January or contact Rick with a proposal for an article.

As the project team leader for the 2013 International Symposium, Mike Gehringer is looking for innovative ideas for the symposium issue of ***INSIGHT***. If you have any ideas for coverage of the symposium, please contact him, or drop by and visit him during planning meetings he will be leading during the International Workshop in January.

The Forum Académie-Industrie was sponsored by AFIS (the French chapter of INCOSE) and was held at École Nationale Supérieure de Techniques Avancées ParisTech, one of the foremost schools of engineering in France, and concluded on 30 November. Hervé Panetto will be bringing a collection of essays from French doctoral students who presented their research at this forum in Paris to the global INCOSE audience.

The Standards Group, under the leadership of Ken Zemrowski, is on task to publish an update of its 2007 theme issue of ***INSIGHT*** that will provide news and information about the advances that INCOSE and its partners have made in international standards for systems engineering.

Finally, INCOSE ***INSIGHT*** is conducting its 1st annual survey of its readers to determine awards for the best issue and the best article. The survey will remain open until 17 January 2013 to allow these awards to be given at the International Workshop in 2013. Please visit the survey site to cast your vote at: ***INSIGHT*** Readers' Choice Survey (https://connect.incose.org/admincomm/comm2/insight/Lists/INSIGHT Readers Choice Survey/overview.aspx).

Upcoming submission deadlines and themes for ***INSIGHT***

Issue	Submission Date for General Articles	Theme	Theme Editors
1st Qtr 2013	15 February 2013	Certification	Jerry Fisher
2nd Qtr 2013	15 May 2013	The Buck Stops Here: Systems Engineering's Responsibility for System Security	Rick Dove
3rd Qtr 2013	8 July 2013	2013 International Symposium Coverage: Philadelphia, PA (US)	Mike Gehringer
4th Qtr 2013	15 October 2013	AFIS Doctoral Symposium: Systems Engineering Research Challenges in French Universities	Hervé Panetto
1st Qtr 2014	15 February 2014	Standards	Ken Zemrowski
2nd Qtr 2014	15 May 2014	Agile Systems and Systems Engineering	Rick Dove

INSIGHT
International Council on Systems Engineering
7670 Opportunity Road, Suite 220
San Diego, CA 92111-2222