

## ON A MATURITY MODEL FOR COMPLEXITY, COMPLEX SYSTEMS, AND COMPLEX SYSTEMS ENGINEERING

B.E. WHITE

Complexity Are Us←Systems Engineering Strategies, USA.

### ABSTRACT

Complexity, complex systems, complex systems engineering, and other related terms are defined. As yet not enough about these topics is broadly known, understood, and applied to claim the existence of a maturity model. Nonetheless, one can contemplate what might constitute such a model and to begin its development. Ways of stimulating a more effective maturation process are suggested. Criteria are outlined for recognizing when maturity is approached. An initial draft maturity model is offered that includes examples of complexity, a collection of complex system characteristics, behaviors, and problems; a set of complex systems engineering principles for attacking and making progress in alleviating the world's more difficult problems, and an assessment of systems engineering practices. The main objective is to stimulate further thinking on these topics with the goal of constructing an improved maturity model through the collective interactions of interested parties.

*Keywords: complex systems, complex systems engineering, complexity, maturity models, system of systems, systems engineering.*

### 1 INTRODUCTION

“The term *systems engineering* (SE) can be traced back to at least the 1940s” [1, 2]. The practice of SE is now roughly 65 years old. Currently there are three recognized forms of SE, viz., (1) classical, conventional, or traditional SE; (2) System of Systems (SoS) Engineering (SoSE); and (3) Complex Systems (CSs) Engineering (CSE) [3] representing regimens for addressing problems of increasing engineering difficulty.

#### 1.1 Definitions

To appreciate the SE context, it is useful to consider some definitions.

##### 1.1.1 System and system of systems

A system is an interacting mix of elements forming an intended whole greater than the sum of its parts. These elements may include people, cultures, organizations, policies, services, techniques, technologies, information/data, facilities, products, procedures, processes, and other human-made (or natural) entities. The whole is sufficiently cohesive to have an identity distinct from its environment.

A SoS is a collection of systems that functions to achieve a purpose not generally achievable by any subset of the individual systems. Each system can operate independently (in the same environment as the SoS) and is generally managed primarily to accomplish its own separate purpose. There are four different forms of SoSs, directed, acknowledged, collaborative, and virtual [4].



This paper is part of the Proceedings of the New Forest Conference on Complex Systems 2016  
(Complex Systems 2016)  
[www.witconferences.com](http://www.witconferences.com)

### 1.1.2 Complexity and complex systems

Complex means more than complicated, the simplest form of complexity. Complicated problems are typically attacked by first trying to subdivide or partition the entity into separate pieces in a hierarchical structure until the pieces are sufficiently small that they can all be understood in a rather thorough fashion.

If a portion of a CS is separated from the CS, that component is often not able to operate on its own within the environment of the CS.

A complex entity may be characterized by a number of attributes such as

- Uneven, varied, diverse, and changing compositions of multi-sized elements
- Both tightly-coupled and loosely-coupled regions of network connectivity
- Elements and connections among them that continually evolve through self-interactions and exposure to the entity's environment
- Rich textures and multiple colors, sounds, and patterns
- Perceived as having different approximations of its underlying reality.

The more such attributes apply, the more complex the entity.

A CS is an adapting and evolving open (unconstrained by artificial means) system with continually cooperating and competing elements and unlimited interactions among its components (often including autonomous or semi-autonomous agents) and its environment. Relationships among CS elements and the effect of their interactions on the whole CS or its external environment are difficult to describe, manage, design, and/or change, let alone understand or predict.

### 1.1.3 Engineering, systems engineering, and complex systems engineering

Engineering consists of methodically conceiving and implementing viable solutions to existing problems, usually of the human-made variety.

SE is an iterative and interdisciplinary management, development, and implementation process that defines the needs and/or desires of interested stakeholders and transforms them into an operational system. Activities include conceiving, researching, architecting, utilizing, designing, developing, fabricating, producing, integrating, testing, deploying, operating, sustaining, and retiring system elements. There is emphasis on selecting, analyzing, synthesizing, and applying the appropriate scientific, mathematical, and other technological knowledge.

CSE is the deliberate leadership and accelerated management of natural and human processes that shape useful CS developments. CSE involves more environmental, economic, political, philosophical, social, organizational, ethical, moral, and other non-technological aspects. Accordingly, it is recommended that all key stakeholders be included in the CS intentionally from the start of any CS project. Because people are willful, unpredictable, and difficult to lead and manage, one is not in control of a CS and cannot prespecify or predict CS outcomes with any high degree of confidence.

## 1.2 Examples

Here are some examples of SoSs, complicated entities, and CSs.

### 1.2.1 Systems of Systems

- The Space Shuttle (not counting the crew) (a directed SoS).
- A satellite communications system (an acknowledged SoS).

- The internet (a collaborative SoS).
- A terrorist organization (a virtual SoS).

#### 1.2.2 Complicated entities

- Wiring diagram for a large electrical installation.
- Architectural plans for a building complex.
- Technical specifications of a military electronic system.
- Instructions for conducting elementary school classes.
- Rules for driving roadway vehicles.
- Engineering standards handbook.

#### 1.2.3 Complex systems

- An anthill, beehive, bird flock, or fish school.
- A seashore.
- The weather.
- The human body.
- The auto industry.
- Any group of people called to work together for a common purpose.
- A concert orchestra.
- A professional society.

## 2 BACKGROUND

In this section, we claim only an immaturity model for complexity, CSs and CSE. Then we explain by reviewing the current state-of-the-art. Next we suggest some challenges and what might be done to develop such a maturity model. Finally, we characterize how we might recognize when maturity has been attained.

### 2.1 Immaturity Model

CSs are well understood by those that study complexity science.

However, most engineers often say complex when they mean complicated. Surprisingly, relatively few systems engineers grasp the richer meanings complexity; most apply SE as if they live in a merely complicated world.

If we hope to deal effectively with our most difficult problems, CSE must deal with the multifarious spectrum of complexity. We have formidable challenges and have a lot more work to do before we can honestly state that we have a mature model for complexity, CSs, and CSE.

What we have presently can only be labeled an immaturity model!

### 2.2 Where we are now

Significant progress is being made in the past decade in learning about complexity, offering definitions of CSs, characterizing CS behaviors, and offering principles for CSE. More professionals are being exposed to these ideas and some are becoming sufficiently interested and motivated to join early adopters of this burgeoning field.

CSE practitioners and CS theorizers enjoy interacting with each other and preaching to the choir, as it were. Unfortunately, much of this effort enters the ether and has little influence in furthering progress in modernizing the practice of SE. As an example, many senior members of the International

Council On Systems Engineering (INCOSE), the world's largest SE society (currently about 9312 members as of December 2015, <http://www.incose.org/about>, are still wedded to classical, traditional, or conventional SE, which also tends to hamper the education and development of more junior systems engineers concerning complementary techniques that might have greater impact in solving our most difficult problems.

INCOSE has published a wonderful forward-looking vision of SE [5] but there is precious little offered on just how to get there. People are not considered part of the system, and processes and technology are relied upon to solve the problems. However, the importance and relevance of the social sciences is recognized.

INCOSE's latest SE handbook version [6] is a voluminous and detailed exposition of what is meant by SE. Below we highlight a few areas that still need fixing despite many recent handbook revisions.

The handbook advocates a hierarchical approach of partitioning subsystems indefinitely until they are small and simple enough to be understood. Best is used indicating a mindsight of optimization, rather than balance and "satisficing" more appropriate in CSE. Although recognizing that reductionism may not work well for a CS, no more cogent CSE approach is offered.

There is an overemphasis on the life cycle and sequential methods, although incremental and iterative techniques are also discussed. Too much is devoted to defining requirements and mitigating risks which are, respectively, quite hard to know and treated thoroughly at the expense of pursuing opportunities.

Also, there is little advocacy for the early construction of true guiding architectures. In practice, reference architectures tend to change with system development, and this area typically degenerates into presenting so-called architectural views before an underlying architecture is firmly established.

But many good things are included, e.g., systems thinking, Checkland's soft systems methodology [7] (which emphasizes understanding the problem as opposed to assuming requirements are knowable), Forrester's system dynamics [8, 9], Boardman's and Sausser's systemigrams [10], SE leadership vs. management, the importance of case studies, and several case study examples are rightly highlighted.

### 2.3 The challenge

Many purveyors of SE methodologies believe they are in control, know how to solve complex problems, and advocate a straightforward step-by-step process they think is sufficient to make practitioners successful. Unfortunately, this is inadequate for CSs, as many failed SE attempts documented in case study examples amply illustrate.

So how can these systems engineers be dissuaded from such fruitless pursuits and convinced to learn about or explore and adopt more promising methods? Even their experience with failed approaches does not help when they continue to repeat their standard methods expecting different results!

What about the rest of the world? CSE is something that can be applied in many domains, not just engineering fields. Those wrestling with formidable problems in many diverse non-engineering fields likely already understand complexity and bring a healthy dose of humility when trying to make progress. Why can't engineers embrace the same mindsight when encountering and dealing with their challenges?

### 2.4 What we might do

Reaching out to and establishing collaborative interactions with practitioners in other complexity domains about what works and what does not should help improve CSE.

So how might one ignite this endeavor? Systems engineers should be encouraged to participate in other forums to learn similar concerns, make personal contacts, and begin dialogs for mutual benefits. Protagonists from other fields should be invited to participate in SE forums for the same reasons. These interactions can stimulate greater understanding and fresh ideas for accelerating progress in SE through diversity, taking advantage of the Medici effect [11], where innovative ideas derive from seemingly unlikely sources. Another suggestion is to get systems engineers involved with the Club of Rome and their offshoots, to motivate and orient them and their colleagues to better address big world problems within the SE profession as part of their job.

## 2.5 How to recognize maturity

It would be nice to know when we might be attaining enough maturity to finalize and accept a viable maturity model. Here are a number of areas (with brief explanations) to watch in ascertaining the extent to which certain CS behaviors and CSE principles are being acknowledged and applied in a sufficiently widespread manner; and, clearly, there may be several other meaningful indicators.

- Intentionally including stakeholders in the system to be improved: this makes the system unpredictable because people are willful and difficult to control.
- Investing significant time and effort up-front to understand the problem: charging ahead with perceived solutions impatiently is often counterproductive.
- Embracing transdisciplinary fields to garner additional valuable insights: take advantage of the Medici effect which often suggests fruitful innovations
- Sharing information freely to illuminate the underlying reality collectively: this fundamental means of establishing trust and understanding should be rewarded.
- Modifying the system boundary in establishing a proper context: this offers the potential of establishing a more attractive solution space leading to a simpler objective while recognizing the original complex problem may remain unsolved.
- Focusing on system capabilities instead of requirements: initial requirements are largely unknown by the stakeholders or users and are continually changing; as desirable and useful capabilities are developed they should be introduced incrementally while watching how well they work over time, and then adapting.
- Developing and adopting a guiding (and layered) architecture early on; a true underlying architecture that will not change much compared to subsequent system configurations must be established before promulgating popular descriptive architectural views that otherwise may have little behind them.
- Recognizing and pursuing opportunities as well as identifying and mitigating risks: opportunities typically abound in CSs, and the biggest risk may be in not pursuing an opportunity, thereby missing a great solution, while trying to get back on a rigid, preconceived development track or path.
- Planning for emergent properties comprised of the unexpected overall system outcomes that result from the confluence of interactions within the system or with its environment; surprises that cannot be explained can be illuminating.
- Concentrating on a holistic approach instead of resorting to reductionism and collectivism: subdividing the problem and optimizing subsystem solutions does not work well in CSs; by the time one performs this cycle the CS has moved on.
- Striving for balance among subsystems rather than optimization: treat competing objectives together to fairly represent the interests of each effect.

Table 1: Complex system characteristics.

1	Unique: irreproducible; no other exact copies exist
2	Ill-Defined: ambiguous shifting boundary and content
3	Unpredictable: cannot foresee or prespecify outcomes
4	Emergent: unexpected results; multiple scales involved
5	Evolutionary: concurrent/continual development and operation
6	Un-Stable: transitory; largely non-hierarchical structure
7	Self-Organized: highly interactive and re-integratable
8	Collaborative: thrives on cooperation and competition
9	Experimental: explores and tests new possibilities
10	Adaptable: learns from and responds to current conditions
11	Open: influenced by and influences external environment
12	Inter-Relational: long-range effects but short-ranges dominate
13	Robust/Inefficient: relies on diversity and effective holistically
14	Anti-Fragile: strengthened by small random perturbations

Until most of the above aspects are explicitly, broadly, and routinely addressed, how can our understanding of complexity and approach to CSE be considered mature?!

### 3 COMPLEX SYSTEMS

In this section, we elaborate on CS characteristics, behaviors, and problems.

#### 3.1 Characteristics

Table 1 lists 14 CS characteristics along with a brief explanation of each.

Highly CSs are unique in that no exact replicas exist or can be reproduced. CSs are difficult to define and have no fixed boundary; one needs to discuss and agree on where the “fuzzy” boundary may lie. They are unpredictable and have (unexpected) emergent properties. CSs evolve at several different scales which need to be viewed concurrently to be understood. Although a CS may become relatively stable at times, it will soon move on, fluidly reaching and passing through other states. CSs are self-organized, collaborative, adaptive, and open. They exchange energy and information with their environments, and are heavily interdependent on the multifarious interactions of their component portions. CSs tend to be inefficient in the usual sense because they possess extra capabilities which ensure overall effectiveness. Healthy CSs are robust and can become even stronger from small random perturbations that induce some stress but create a greater ability to face future threats.

#### 3.2 Behaviors

Table 2 lists 14 CS behaviors along with a brief implication of each.

A common behavior is the stimulation of different observer perspectives of the multiple scales of a CS’s evolution. A CS thrives on diversity, self-organization, and freewheeling interactions. All this helps to integrate the many factors at play to ensure adaptable and survivable, i.e., robust operation; if possible stimulate anti-fragility. The CS is ever changing, evolving on its own as a whole, and

Table 2: Complex system behaviors and recommended responses.

1	Stimulates Different Perspectives: seek understanding
2	Thrives on Diversity: study inter- and external relationships
3	Self-Organized: note collaboration, competition, and cooperation
4	Internal and External Relationships are Key: learn from experiments
5	Acts Robustly: recognize adaptability, survivability, and anti-fragility
6	Many Factors are at Play: seek to build upon areas of common ground
7	Evolves on its Own as a Whole: appreciate holistic un-predictability
8	Ever Changing: endeavor to pre-prepare in managing uncertainties
9	Unexpected Properties Emerge: welcome them and wrestle with surprises
10	Sensitive to Small Effects: carefully record and mind initial conditions
11	Informs the Observer: rein in emotions and practice objectivity
12	Exhibits Tight and Loose Couplings: map and categorize interactions
13	Performs Openly: attempt to interpret causal relationships with humility
14	Operates at Edge of Chaos: marvel at reflected health where best happens

exhibiting emergent properties that are largely unexpected and even surprising. A slight change of initial conditions can yield quite different results (the “butterfly effect”). CSs generally perform openly but it is challenging to observe objectively, especially when one is inside the CS. A healthy (chaordic) CS operates at the edge of chaos.

### 3.3 Examples of complex systems problems

In Table 3, we have offered specific CS examples, suggested to what degree they are complex, and have indicated what some solution approaches might entail.

Since people are complex, systems that include them are also complex. Challenging systems, even those that “boggle the mind,” but that do not include people as part of the system, are only complicated, or at least not as complex, compared to those that do. Fundamentally people are semi-autonomous, semi-independent CSs [12]; their interactions are what make situations especially complex.

## 4 COMPLEX SYSTEMS ENGINEERING

In this section, we (1) propose CSE principles that may be more effective than those of classical, conventional, or traditional SE; and (2) discuss SE practice vs. theory.

### 4.1 Complex systems engineering principles

Table 4 lists this author’s latest CSE principles and short explanations. The first 13 principles have existed for years [13, 14] but there is a new 14th principle.

First, we must suppress hubris in trying to quickly solve the problem; instead devote considerable effort to understand the problem by gathering facts and speaking with experienced practitioners within the domain of interest.

Table 3: Examples of complex system problems and potential approaches.

Complexity	CS Challenges	Remedy?
Very High	Combat International Terrorism	Address Why They Hate Us
Very High	Marginalize Iran and N. Korea	Create Economic Incentives
High	Accept Global Climate Change	Reduce Harmful Emissions
High	Bolster Renewable Energy	Transfer Fossil Fuel Efforts
Medium	Update Government Acquisition	Move to Paying for Results
Medium	Develop New Military Systems	Focus Only on What is Needed
Medium	Improve Air Traffic Control	Modernize Existing Technology
Medium	Rebuild National Infrastructure	Stimulate United States to Invest
Low	Fix Privacy and Cyber Security	Give Devices Unique Passwords
Low	Innovate in Social Networks	Concentrate on Educating Public
Complicated	Complete Income Tax Forms	Maintain Accurate Records
Complicated	Keep Expense Budget	Create Thorough Spreadsheet
Complicated	Use Technical Paper Format	Pay Attention to Detail

Table 4: Complex systems engineering principles.

1	Bring Humility: acknowledge shortcomings and seek understanding
2	Follow Holism: focus on the whole system rather than just the pieces
3	Achieve Balance: shun optimality but improve overall capabilities
4	Utilize Trans-Disciplines: apply good techniques of related domains
5	Embrace POET*: define the CS boundary and keep the big picture in mind
6	Nurture Discussions: listen to others and reward information sharing
7	Pursue Opportunities: take informed risks by pursuing alternative solutions
8	Formulate Heuristics: formulate rules of thumb and improve decision taking
9	Foster Trust: brave personal vulnerability in gaining help from others
10	Create Interactive Environment: open and facilitate meaningful interchanges
11	Stimulate Self-Organization: nurture, reward, and celebrate collaborations
12	Seek Simple Elements: deal with the most pervasive common concerns
13	Enforce Layered Architecture: establish early a powerful guide to progress
14	Elevate Future Goals: endeavor with others to help solve world problems

\*politics, operations, economics, and technology

Concentrate on the whole CS and the interactions among its components and with the environment. Unless the problem is just complicated, avoid the usual SE approach, reductionism and constructionism; these techniques are fine but only when they apply!

Rather than trying to optimize the parts or even the whole, instead be mindful of achieving balance among competing system aspects to improve overall capabilities.



Utilize trans-disciplines to learn from other, perhaps widely separated, domains.

Nurture discussions and share relevant information (leaders should reward but not punish this!) to garner different points of view so collectively there is a better understanding of the problem's underlying reality and the CS boundary definition.

CSs opportunities often abound and they should be pursued while being informed of potential associated risks. If one only identifies and tries to mitigate risks, attractive solutions that might be found by different paths could be missed.

One way of helping decision takers make better decisions is to recall, develop, or propose good heuristics. Sometimes postponing decisions works well because more information is available on what is happening with previous interventions.

One engenders trust by increasingly sharing information that is reliable and safe in both directions. This takes some courage to show vulnerability in sharing but the prospect of gaining greater understanding from others' knowledge should be worth it.

An interactive environment is critical in facilitating meaningful interchanges. Those in charge should stimulate self-organization by creating attractive conditions for collaboration, competition, and cooperation.

Concentrate on those things that everyone cares about rather than relatively minor "stovepipe" concerns. This can create compelling simplicity within complexity which can achieve results that have widespread benefits such as horizontal interoperability.

The early building of a guiding architecture that does not change much compared to the system being developed is recommended. To the extent possible the architecture should be layered to be more assured of simple and stable interfaces between layers.

Elevate future goals, hopefully with appropriate incentives, so as to reach for more ambitious achievements and even greater success in solving world problems that could benefit all humankind. What higher goals might one's organization aspire to!

#### 4.2 Systems engineering practice vs. theory

In science, which is about understanding the universe, often theory drives practice. For example, based upon their objective observations physicists first develop a theory which gains some degree of acceptance, at least among fellow scientists. This holds until the theory is proven wrong by the practice of conducting experiments.

In engineering, which is about solving human made problems, the reverse tends to be true, i.e., practice drives theory.

Suppose 1950 is when classical, traditional, or conventional SE began. In this era M.I.T.'s Jay Forrester invented system dynamics [9], for example. Assume that SE was practiced from the late 50s through about 1990 (and beyond) when the National Council on Systems Engineering (NCOSE) (now INCOSE) was formed in the US. So 1990 might be the demarcation where SE theory began.

The practice of SoSE began circa 1998 with Mark Maier's definition of SoSs [15]. Efforts toward a theory of SoSE started about 2008 with a seminal paper by Judith Dahmann, et al., on the four different forms of SoSs, mentioned earlier [4].

The scientific field of complexity also extends back into at least the 1940s. The practice of CSE began circa 1993 with Yaneer Bar-Yam's efforts [16] in making the study of complexity more practical and actionable. Theoretical methodologies for CSE began in earnest prior to 2005 with, for example, regimen proposals by Doug Norman, Mike Kuras, and Brian White [17, 18].

Table 5: Essential elements of a complexity, CSs, and CSE maturity model.

Benefits	Sample indicators
Development/Provision	Informed strategies to better influence difficult outcomes
Consistency/Relatedness	Synergistic characteristics, behaviors, and principles
Effectiveness/Usefulness	Greater CSE impacts than traditional methodology
Alignment/Progress	Education, understanding, and acceptance of approach
Action/Awareness	Widely proselytized, promulgated, and publicized ideas
Impact/Value	Measurements of successful applications and practices

Just recently, two interesting works included in the conference, Complex Systems 2016, have appeared: [19] is a literature review assembling thoughts on launching a science of (CS) emergence; and in [20], although the author seems to limit complexity to natural, not human or human-made CSs, he advocates doing more to help solve critical problems the world faces.

## 5 CONCLUSION

We summarize this paper by suggesting what we might expect from working toward a maturity model. Six essential milestones, each of which provides some perceived benefits, are envisioned and described briefly in Table 5.

The first step is to develop an understandable collection of ideas that will provide the impetus for realizing a signature maturity model. The focus should be on strategies for facilitating desired outcomes in helping to solve our most difficult problems.

The descriptions of the characteristics and behaviors of CSs, and the espoused principles of CSE must be synergistic, internally consistent, and interrelated to propel a cohesive way forward.

Of course, whatever is advocated should be significantly more useful and effective than the traditional SE approach.

As people are exposed and educated to these ideas their understanding of complexity CSs and CSE will likely improve, and they should become more aligned and receptive to such a modernization of SE.

As the elements of the model are explained and distributed in ever widening arenas, in time more practitioners will become acutely aware of how they can apply CSE in their normal work, and motivated toward acting in a fashion consistent with trying to solve more challenging problems.

Naturally, the ultimate assessment of the viability of CSE will depend on its positive impact and value based on measurements of its application in practice, supported in parallel by case studies.

## REFERENCES

- [1] Systems Engineering, Wikipedia, the free encyclopedia, [https://en.wikipedia.org/wiki/Systems\\_engineering](https://en.wikipedia.org/wiki/Systems_engineering) (accessed 14 January 2016).
- [2] Schlager, J., Systems engineering: key to modern development. *IRE Transactions*, **3**(3), pp. 64–66, 1956.  
<http://dx.doi.org/10.1109/IRET-EM.1956.5007383>
- [3] White, B.E., A complex adaptive systems engineering (case) methodology — the ten-year update. *IEEE Systems Conference*, Orlando, FL, 2016.
- [4] Dahmann, J.S., Rebovich, G. Jr. & Lane, J.A., Systems engineering for capabilities. *CrossTalk, The Journal of Defense Software Engineering*, pp. 4–9, 2008, <http://www.stsc.hill.af.mil/crosstalk/2008/11/index.html>

- [5] A world in motion — systems engineering vision 2015. International Council On Systems Engineering (INCOSE), 2015.
- [6] Systems engineering handbook — a guide for system life cycle processes and activities. International Council On Systems Engineering (INCOSE), 4th edn, TP – 2003-002-004, Wiley: Hoboken, NJ, 2015.
- [7] Checkland, P., *Systems Thinking, Systems Practice — Soft-Systems Methodology : a 30-year Retrospective*, Wiley: New York, 1999.
- [8] Forrester, J.W., The beginning of system dynamics, banquet talk. *International Meeting of the System Dynamics Society*, Stuttgart, Germany, 1989, <http://web.mit.edu/sysdyn/sd-intro/D-4165-1.pdf>
- [9] Forrester, J.W., System Dynamics, circa 1958, [http://en.wikipedia.org/wiki/Jay\\_Wright\\_Forrester](http://en.wikipedia.org/wiki/Jay_Wright_Forrester)
- [10] Boardman, J. & Sauser, B., *Systems Thinking – Coping With 21st Century Problems*, CRC Press: Boca Raton, FL, 2008.
- [11] The Medici Effect: Breakthrough Insights at the Intersection of Ideas, Concepts, and Cultures, Book by Frans Johansson.
- [12] Gorod, A., White, B.E., Ireland, V., Gandhi, S.J. & Sauser, B.J., *Case Studies in System of Systems, Enterprise Systems, and Complex Systems Engineering*, CRC Press, Taylor & Francis Group: Boca Raton, FL, 2015.
- [13] White, B.E., A personal history in system of systems, special session on system of systems (SoS). *International Congress on Ultra Modern Telecommunications and Control Systems (ICUMT-2010)*, Moscow, Russia, 2010.  
<http://dx.doi.org/10.1109/ICUMT.2010.5676561>
- [14] White, B.E., Managing uncertainty in dating and other complex systems. *Conference on Systems Engineering Research (CSER) 2011*, Redondo Beach, CA, 2011.
- [15] Maier, M.W., Architecting principles for systems-of-systems. *Systems Engineering*, **1**(4), pp. 267–284, 1998.  
[http://dx.doi.org/10.1002/\(SICI\)1520-6858\(1998\)1:4<267::AID-SYS3>3.0.CO;2-D](http://dx.doi.org/10.1002/(SICI)1520-6858(1998)1:4<267::AID-SYS3>3.0.CO;2-D)
- [16] Bar-Yam, Y., *Making Things Work – Solving Complex Problems in a Complex World*, 1st edn, Knowledge Press: New England Complex Systems Institute (NECSI), 2005.
- [17] Norman, D.O. & Kuras, M.L., Engineering complex systems. Chapter 10. *Complex Engineered Systems—Science Meets Technology*, eds. D. Braha, A. Minai & Y. Bar-Yam, New England Complex Systems Institute (NECSI), Springer: Cambridge, MA, 2006.  
[http://dx.doi.org/10.1007/3-540-32834-3\\_10](http://dx.doi.org/10.1007/3-540-32834-3_10)
- [18] White, B.E., Complex adaptive systems engineering. *8th Understanding Complex Systems Symposium*, University of Illinois at Urbana-Champaign, IL, 2008, available at <http://www.howhy.com/ucs2008/schedule.html>
- [19] Lichtenstein, B., Complexity at a crossroads—toward a science of emergence. *Oral Presentation, Complex Systems 2016*, New Forrest, UK, Subsequent version is being submitted to a journal for publication, 2016.
- [20] Motloch, J.L., Unlocking complexity: big science project and research agenda. Complex systems 2016, new forrest, uk, 1-3 june 2016, to be published by the Wessex Institute of Technology (WIT) press in a special issue of the. *International Journal of Design & Nature and Ecodynamics*, ISSN:1755-7437; Digital ISSN:1755-7445.