A General Framework for Measuring System Complexity

MAHMOUD EFATMANESHNIK AND MICHAEL J. RYAN

University of New South Wales - Canberra, Australian Defence Force Academy, Campbell, ACT, Australia

Received 21 July 2015; revised 23 January 2016; accepted 26 January 2016

In this work, we are motivated by the observation that previous considerations of appropriate complexity measures have not directly addressed the fundamental issue that the complexity of any particular matter or thing has a significant subjective component in which the degree of complexity depends on available frames of reference. Any attempt to remove subjectivity from a suitable measure therefore fails to address a very significant aspect of complexity. Conversely, there has been justifiable apprehension toward purely subjective complexity measures, simply because they are not verifiable if the frame of reference being applied is in itself both complex and subjective. We address this issue by introducing the concept of subjective simplicity—although a justifiable and verifiable value of subjective complexity may be difficult to assign directly, it is possible to identify in a given context what is "simple" and, from that reference, determine subjective complexity as distance from simple. We then propose a generalized complexity measure that is applicable to any domain, and provide some examples of how the framework can be applied to engineered systems. © 2016 Wiley Periodicals, Inc. Complexity 000: 00–00, 2016

Key Words: complexity measure; subjective simplicity; objective complexity; graph complexity

1. INTRODUCTION

The lessons to be drawn from complexity theory are of interest to systems engineers to assist in managing the growth of complexity in modern systems [1–3]. A predominant preoccupation in systems engineering, as well as in product engineering, has been the development of justifiable complexity measures that can be of assistance in all aspects of the system lifecycle but particularly in system design. In product engineering limited efforts have been focused on developing complexity measures that are helpful in estimating the amount of resources required to complete a design task, and to estimate design difficulty [4]. A recent research project (META) sponsored by DARPA (Defence Advanced Research Project Agency) focused on identifying complexity metrics for engineered systems that correlate with as well as predict project cost, schedule or reliability, which can also be used to compare designs/system concepts alternatives [5–7]. However, there is no consensus about how to measure complexity in the context of systems engineering, which is further exacerbated by the fact that the applicability of such complexity measures to predict project success remains fundamentally dubious [8]. The search for statistically significant

Correspondence to: Mahmoud Efatmaneshnik,
E-mail: m.efatmaneshnik@adfa.edu.au

DOI 10.1002/cplx.21767
Published online 00 Month 2016 in Wiley Online Library
(wileyonlinelibrary.com)

correlation between different complexity measures (e.g., the size of project or engineering problem) and project outcomes has shown no promise at all [3].

In this article, we briefly review the concept of complexity and the implications associated with measuring it. We propose a generalized complexity measure that explicitly accounts for both subjective and objective complexity. We identify a mechanism for determining subjective complexity based on the use of subjective simplicity as a reference. We show how some well-known measures of complexity implicitly follow this framework. We conclude with examples of how to apply the framework for measuring complexity in engineered systems.

1.1. What Is Complexity?

From an etymological perspective "complexity" is an English noun, which comes from the Latin noun complexus meaning: (1) an aggregate of parts, (2) embraced, and surrounded (3) an embrace. Thus, complexity is the quality of a thing, which consists of an aggregate of parts, and/or of a situation in which a part is embraced or surrounded by other parts. Complexsus itself comes from "complecto" a compound of "com- "(together) and "plecto" (I weave, I twist).

However, "The term complexity has been used without qualification by so many authors, both scientific and non-scientific, that the term has been almost stripped of its meaning" [9]. Heylighen [10] maintains that, "Complexity has turned out to be a very difficult notion even to define. Dozens of definitions that have been offered tend to have a structure specific to the problem being addressed, classifying something as complex, which we intuitively would see as simple, or denying an obviously complex phenomenon the label of complexity. Moreover, these definitions are either only applicable to a very restricted domain, such as computer algorithms or genomes, or so vague as to be almost meaningless."

The first time the term "complexity" appeared in a scientific paper was in 1948, in an article titled "Science and Complexity" [11] in which Weaver [11] stated that "physical science before 1900 ... was largely concerned with two-variable problems of simplicity; whereas the life sciences, in which these problems of simplicity are not so often significant, had not yet become highly quantitative or analytical in character."

Edmonds [12] and Sussman [13] presented lists of different accounts of complexity. Edmonds [12] noted that complexity has unjustifiably been equated with ignorance, variety, midpoint between order and chaos, improbability, ability to surprise, irreducibility, amongst other more common descriptions such as size and minimum size. Both authors however made close connections between complexity and unpredictability. According to Edmonds [12], "Complexity is that property of a model which makes it difficult to formulate its overall behavior in a given language, even when given reasonably complete information about its atomic components and their inter-relations." A similar view was presented by Sussman [13,14] who stated that "a system is complex when it is composed of a group of related units (subsystems), for which the degree and nature of the relationships is imperfectly known. Its overall emergent behavior is difficult to predict, even when subsystem behavior is readily predictable."

The descriptions of both Edmonds [12] and Sussman [13] imply that complexity is a subjective notion as much as it is objective, although that observation is not acknowledged directly. The characteristics of an agent observing a system, or the agent's internal model of an external system, have much to do with the perception of complexity. The behaviour of a complex system may appear to be unpredictable, because even a complete description of an object does not lead to an adequate understanding of its behaviour due to a lack of comprehension or experience of the observing agent, or due to incompleteness in the agent's internal model relative to the external observations. Explicitly accounting for the subjective dimension of complexity is therefore important in a useful measure of complexity in a given context.

1.2. Context Dependency and Viewer Dependency of Complexity

Gell-Mann [15] states that: "As measures of something like complexity for an entity in the real world, all such quantities are to some extent context-dependent or even subjective. They depend on the coarse graining (level of detail) of the description of the entity, on the previous knowledge and understanding of the world that is assumed on the language employed, on the coding method used for conversion from that language into a string of bits, and on the particular idealized computer chosen as a standard."

In the case of an engineered system, the context can be defined as a sufficiently defined domain of knowledge to which a system has relevance. An engineered system has life cycle stages that are also system contexts. For example, a computer system can be viewed, *inter alia*, in its design context, manufacturing context, operational use context, and repair context. So a system can be meaningfully viewed from the perspective of a given context and may consequently be assessed as having different complexity in each of those contexts. Further, each context may have different stakeholders who may well have different perspectives [16–19].

For example consider Kolmogorov complexity and Effective complexity [15]. Kolmogorov complexity roughly speaking is the size of the shortest program that can regenerate an object or a set of strings. Kolmogorov complexity is clearly language dependent, in other words it

DOI 10.1002/cplx

has different values in the context of different languages. Effective complexity conversely is the length of a concise description of an object's regularities, or the amount of nonrandom information [15]. Regularity might have different meanings from the perspective of different agents. Both Kolmogorov complexity, and effective complexity, have been criticised for being respectively context dependent and viewer dependent [12]. We believe that the context dependency and subjectivity criticisms for complexity measures is primarily due to the fact that the literature does not contain any formal method to capture the relativity of a measure of complexity to the context and subjective viewpoints. Here, we present a general framework that explicitly captures both context dependency, and subjectivity respectively in an objective complexity, and subjective components of complexity.

2. TO KNOW THE COMPLEX KNOW THE SIMPLE

Although theories of complexity are relatively recent, principles of parsimony date back two millennia as the core of rational thinking. "We may assume the superiority *ceteris paribus* of the demonstration which derives from fewer postulates or hypotheses" is attributed to Aristotle and "truth is ever to be found in simplicity, and not in the multiplicity and confusion of things" is attributed to Newton. Yet the principle of parsimony, also known as Ockham's razor—"entities should not be multiplied unnecessarily"—does not mean to imply that "whatever is simple is true." Rather, science (to know) is focused on developing explanations for that which can be "knowable." Here, we demonstrate that a rather important part of complexity, subjective complexity, can be made knowable by measuring the departure from simplicity when the observer can be confident of what is "simple" in a given circumstance.

Yet, although "complexity is necessarily in the eye of beholder" [20] and therefore has subjective components, it clearly has objective components as well. The need for a dual perspective to characterize the complexity of systems is therefore deeper than highlighted by previous researchers: here, we propose that a useful complexity measure should have an objective and a subjective component.

2.1. A Framework for Measuring Complexity

We propose a general framework for a measure of system complexity, which is a function of objective and subjective complexity, which we define as follows:

- *Objective complexity* is a measure of system size, or the size of the minimum description of the system. Objective complexity is independent of any observer viewpoint; however, it may be domain/context, and object/goal dependent.
- *Subjective complexity* is a measure of the departure from a reference simplicity. Subjective complexity therefore depends on the observer's selection of a suitable reference model.

Figure 1 shows the proposed framework to measure system complexity based on the above principles. Subjective complexity is reference-dependent, which implies relativity to a context, subject, or viewpoint. For example, system engineers may deem system requirements that resemble a perfect tree structure as simple [21]. If there is link between leaves, or between requirements of different branches of the tree-like requirements breakdown structure then the design might be very complex, because the allocation of functional requirements to physical components is not able to be conducted as a one-to-one mapping. Therefore, a set of requirements resembling a perfect tree is simple to allocate—that is, the allocation problem could not be any simpler. As shown in Figure 1, every subjective viewpoint has an idea of what is simple in a given context, which we call *subjective simplicity.* The subjective simplicity in the context of a system forms a pattern that is reference simplicity. In the above example a one-to-one allocation is subjective simplicity from a systems engineering point of view and the reference simplicity is a tree-like pattern in systems requirements. The reference simplicity drives the subjective complexity, and in the remainder of the article we look at ways that subjective complexity can be measured.

With the above introduction we are in a position to formulate the framework for a general complexity measure that has both subjective and objective components for the observed complexity of an object:

$$\mathcal{K}(S)=\mathcal{F}(\mu(S),\mathcal{D}(SR)) \tag{1}$$

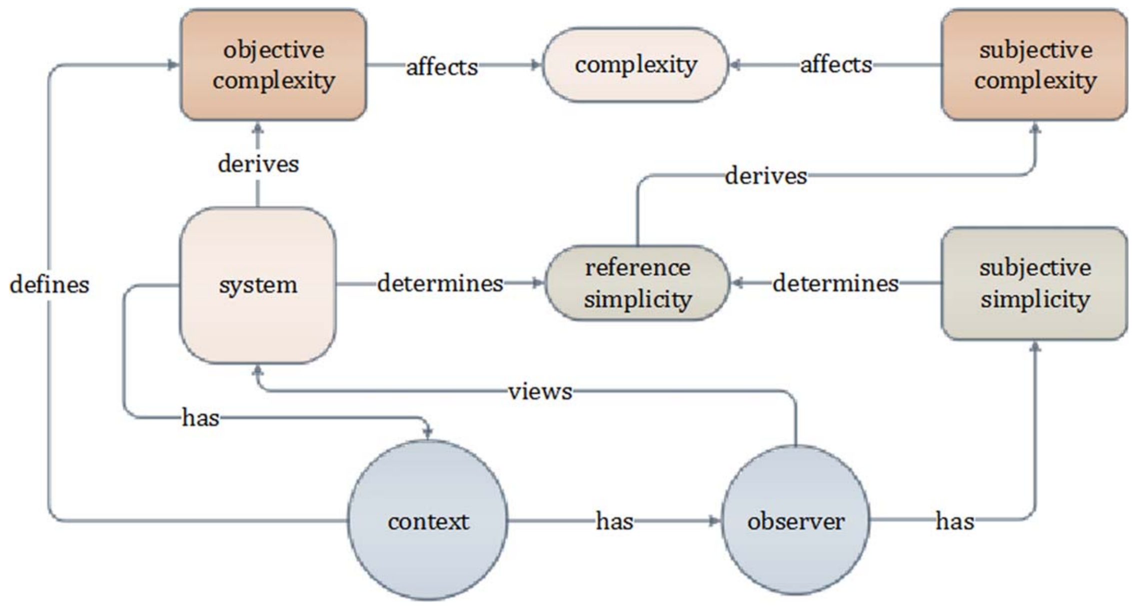
S is the object under study (such as a binary string, a random variable, a set of elements, or a system). K is *system* complexity,[1] and $\mathcal{F}:\mathbb{Z}^2\rightarrow\mathbb{Z}$ is a monotonically increasing function with respect to both of its inputs. $\mu(\bullet)$ denotes the size of the minimal description in a given context. $\mathcal{D}(\bullet)$ is a distance function and $\mathcal{D}(S\|\mathrm{R})$ is the distance of S from R, which is the reference model of simplicity from the viewpoint of observer and for a given context of object S. All three functions $\mathcal{F}(\bullet)$, $\mathcal{D}(\bullet)$, and $\mu(\bullet)$ are context- and/or goal-dependent.

Obviously, function $\mathcal{F}$in the form of multiplication of two inputs provides a very simple instantiation of (1). In our demonstrative examples we use multiplication as the

[1] *Here, we can use other terms such as general complexity, effective complexity, observed complexity, total complexity, combination complexity, and mixture complexity to refer to the intended concept.*

DOI 10.1002/cplx

FIGURE 1



A framework for measuring system complexity. Subjective complexity is relative to the observer; objective complexity is independent of the observer. All elements in the figure are context-dependent.

complexity function, nonetheless, with different size measures and distance metrics. For completeness, simple algebraic definitions of size and distance functions are provided next.

Consider X, as a set of subsets of another set M. We define a size function as a monotonic function μ on X such that:

1. $\mu(x) \geq 0, \forall x \in X$ (positivity)
2. $\mu(\varnothing)=0$ (null for empty set)
3. $\mu(x_i) \leq \mu\left(x_j\right) \Longleftrightarrow x_i \subseteq x_j x_i, x_j \in X$ (monotonicity)

The main properties of distance functions are discussed in set theory and general topology theory [22,23]. The distance is defined for two sets, as a positive-definite real number that satisfies the following conditions 22]:

1. $\mathcal{D}(x,y)=0 \Longleftrightarrow x=y$
2. $\mathcal{D}(x,y)=\mathcal{D}(y,x)$ (symmetric)
3. $\mathcal{D}(x,z) \leq \mathcal{D}(x,y)+\mathcal{D}(y,z))x,z \neq y$ (triangle inequality).

A distance function that meets all three criteria is known to be a metric [22]. The distance function is known as pseudo-metric, quasi-metric, and semi-metric, if it respectively does not meet the first, second, and third criterion [22].

In summary, complexity is a monotonic function of size and distance from the reference simplicity. An interesting complexity measure is the self-dissimilarity complexity measure in Ref. 24, which assumes the pattern of data at one scale as a reference for another scale (higher or lower) of the same system [24]. The self-dissimilarity measure is low when there is uniform pattern in all scales. This complexity measure simply has a variable reference simplicity that is derived from views of the system at different chosen scales. The next section shows that existing statistical measures of complexity fit into our general framework of measuring complexity.

2.2. Relation to Statistical Complexity Measures

According to Feldman and Crutchfield [9], statistical complexity captures structure, organization, patterns, regularities and symmetries in systems. Consider a random variable X that can take on N values each with a probability $\Pr(X=x_i)=p_i$, $1 \leq i \leq N$. $p_i$is the probability of each of the N system states described by X. The Shannon entropy of X is defined as:

$$H[X]=-\sum_{1}^{N} p_i \log p_i \tag{2}$$

Shannon Entropy is a valid measure of size according to the conditions for a measure as stated in previous section. The amount of internal structure of a system can be captured by measuring its distance from, or difference to, randomness [25]. López-Ruiz, Mancini, and Calbet [25] and Feldman and Crutchfield [9] proposed two statistical complexity measures by multiplying disorder and a distance

DOI 10.1002/cplx

function of the current state of the system from complete disorder. These measures followed the general template of:

$$C[X]=H[X]D[X] \tag{3}$$

where $D[X]$ can be described as distance from complete disorder. Given that Shannon entropy is a measure of size of a random variable in bits, the measure template in (4) is clearly consistent with our definition in (1). López-Ruiz, Mancini and Calbet [25] used disequilibrium as the distance term, which for any random variable X of size N (X can take on N values x) with probability p_i is defined as:

$$D[X]=\sum_{i=1}^{N}\left(p_i-\frac{1}{N}\right)^2 \tag{4}$$

Feldman and Crutchfield [9] proposed that the departure from uniformity captured by the Kullback–Leibler distance from a uniform distribution was more representative of the distance component, which is also known as departure from uniformity. Note that the uniform distribution has maximum entropy and can create a maximally disordered series of numbers. In general, the Kullback–Leibler distance of any two probability distributions $\mathrm{Pr}_1(X)$ and $\mathrm{Pr}_2(X)$ is 9]:

$$D(\mathrm{Pr}_1(X)\parallel \mathrm{Pr}_2(X))=\sum_x \mathrm{Pr}_1(X)\log_2\left(\frac{\mathrm{Pr}_1(X)}{\mathrm{Pr}_2(X)}\right) \tag{5}$$

For measuring distance from uniformity we have $\mathrm{Pr}(X=x_i)=1/N$. Disequilibrium (4) is a metric, but Kullback–Leibler distance (5) is a quasi-metric (not symmetric).

In both cases the authors assumed that the reference for simplicity is always a uniform distribution, which is not an invalid assumption for natural physical systems, where uniformity has no useful information for any observer for the purpose of hypothesis building. Shiner et al. [26] also introduced a statistical measure of complexity as:

$$\Gamma_{\alpha,\beta}=\Delta^{\alpha}(1-\Delta)^{\beta} \tag{6}$$

where Δ is disorder:

$$\Delta=\frac{H[X]}{H_{\max}[X]} \tag{7}$$

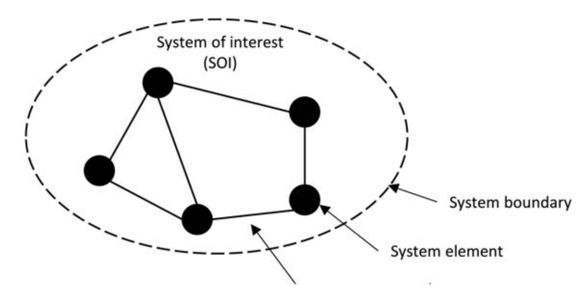
$$H_{\max}[\mathrm{X}]=\log N \tag{8}$$

$H[X]$ is Boltzman–Gibbs–Shannon entropy and $H_{\max}$ is its maximum possible entropy. Δ is clearly a measure of size, and we can show that $1-\Delta$ is a very simple distance measure from uniformity. After some simple manipulations we can see that:

$$1-\Delta=\frac{\sum_{i=1}^{N}\left(p_i\log(p_i)-\frac{1}{N}\log\left(\frac{1}{N}\right)\right)}{\log(N)} \tag{9}$$

The term in the numerator is a nonsymmetric distance function of X from uniformity (quasi-metric), thus Shiner et al. [26] assume uniformity as a reference simplicity. Therefore (6), apart from using a quasi-metric as a distance measure, is a valid complexity measure so long as α, $\beta>0$. The latter condition is necessary for complexity to be a monotonic function of size and distance from simplicity.

FIGURE 2



A system as a set of interconnected elements. [30].

We now turn our attention to graph complexity that is widely used to characterize engineered systems. Several graph complexity measures and distance functions for graphs are introduced. We show that cyclomatic complexity measure (which is extensively used in software engineering) is an implicit instantiation of the general framework proposed in (1).

3. COMPLEXITY IN ENGINEERED SYSTEMS

The objects of study in Refs. 9,25–28 were string of symbols, or a series of numbers reducible to binary strings. The object of interest in this article, however, is a human-made or engineered system. In the next section, we show that, because of the very definition of an engineered system, graph theory is the most appropriate tool for modeling engineered systems in general and for providing a rigorous mathematical basis for measuring engineered system complexity.

"A system is a combination of interacting elements organized to achieve one or more stated purposes**"** [29]. A system necessarily has a boundary through which its elements interact with elements or systems outside the boundary [30]. The notion of boundary is important, since by virtue of a finite boundary, the objective complexity of any engineered system is finite. Further, within the boundary, systems have a limited number of elements at any chosen level of description or abstraction. Since the definition of a system considers systems as networks of constituent elements (Figure 2), a complexity measure for engineered systems can therefore be obtained from the product of a graph complexity measure as an objective

DOI 10.1002/cplx

measure of complexity and the distance of the graph from a baseline graph that represents subjective simplicity.

The general complexity measure in Eq. (1), can provide a general complexity measure for engineered systems with two components:

- *An objective component*: For engineered systems this is an increasing function of number of system elements/objects and their relations. This component can be thought of as objective complexity, context-independent complexity, or absolute complexity. The absolute complexity may be thought of as being synonymous with the system size.
- *A subjective component*: Which is the distance of the system from a selected reference model. It might also be thought of as context-dependent complexity or relative complexity.

As noted earlier, an engineered system exists in a number of life cycle stages [31] such as design, manufacturing/implementation, utilization/operation, replacement/maintenance, and retirement. Each of these stages is a context for an engineered system, but it is not necessary for us to pursue that complication here. In the next two subsections, we look at some graph theory and graph theoretic complexity measures than can have objective, subjective, and combined components.

3.1. Aspects of Graphs

An undirected graph $G(n,m)$ with n nodes and m links can be formally defined with its adjacency matrix $A(G)=[a_{i,j}]_{n\times n}$. For simplicity of explanation, we only consider undirected graphs here for which $a_{i,j}=1$ if there is link between nodes i and j, otherwise $a_{i,j}=0$. A subgrpah of G, is any graph that is contained in G, or in other words its nodes and edges are subsets of those of G.

Previously we briefly described the Kullback–Leibler distance as a measure of the distance between two probability distributions. Now we briefly explain what distance for graphs could mean. Consider a graph G, with $A(G)$ as its adjacency matrix. The graph distance is usually referred to as graph edit distance, or topological difference [32–36]. The topological distance of this graph from any reference graph (R with $A(R)$ as its adjacency matrix) is related to the size of the Maximum Common Subgraph (MCS) between G and R [33,34] and for example can be calculated as:

$$D(G,R)=1-\frac{|MCS(G,R)|}{|G|+|R|-|MCS(G,R)|} \tag{10}$$

where |.| denotes cardinality or the number of nodes, or any other objective measure of graph complexity.

For illustration of the MCS concept, Figure 3 shows a simple example of the MCS of two graphs. The nodes and links shown in red to highlight the MCS, which is the intersection of the two graphs. It is not always straightforward to extract the MCS of two graphs. Finding MCS is computationally an NP hard problem [34]. A systemic way of extracting the MCS of two graphs is to determine the maximum clique (maximum complete subgraph) of the modular product of the two [37,38].

FIGURE 3

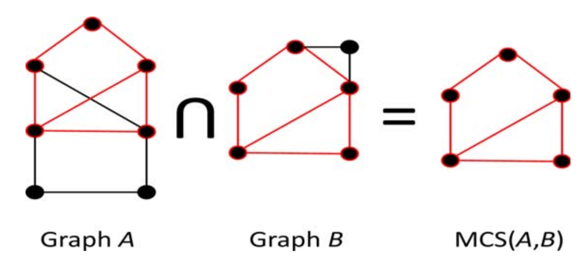


Illustration of the MCS concept.

There are numerous graph distance measures, however all measures are based on the concept of MCS. Two examples from Burke [39] and Wallis et al. [32] are:

$$D(G,R)=1-\frac{|\mathrm{MCS}(G,R)|}{\max(|G|,|R|)} \tag{11}$$

$$D(G,R)=1-\frac{|\mathrm{MCS}(G,R)|}{|MinCS(G,R)|}=1-\frac{|G\cap R|}{|G\cup R|} \tag{12}$$

where *MinCS* is the *Minimum Common Supergraph* which represents the union of the two graphs. Interestingly, when MCS(G,R) = R or MCS(G,R) = G, all three distance measures are equal. It should also be noted that all three distance measures are metrics (satisfy the triangle inequality) [32,39].

3.2. Complexity Measures for Graphs

Comprehensive reviews for graph complexity measures are presented elsewhere [40,41]. Here, we provide a brief overview for the purpose of identifying the utility of such measures as part of a system measure of complexity for systems that can be modeled as a graph. Broadly, graph complexity measures can be classified into entropy-based [42–44], and non-entropic classes [42,45–48].

In general the term entropy describes the diversity and variety. Graph entropy measures state the diversity or spread of a specific graph property (or a property of its nodes or links) [40,41]. For example, one of the properties of a node in a graph is its degree, which is the number of links related to that node:

$$d_i=\sum_{j=1}^{n}a_{i,j} \tag{13}$$

DOI 10.1002/cplx

d_i is the degree of node i. Assume that graph G has n_d nodes with distinct degrees. Then the degree distribution $p(k_i)$ is the number of nodes with their degree as k and $i = \{1 \ldots n_d\}$ divided by total number of nodes (n). An entropy-based graph complexity measure is the entropy of node degree distribution:

$$C_{ge}(G) = -\sum_{i=1}^{n_d} p(\mathrm{k}_i) \log (p(\mathrm{k}_i)) \tag{14}$$

C_{ge} provides information about diversity of degree distribution across all nodes of G. As shown in an experiment in the next section, C_{ge} does not necessarily increase with an increase in the number of nodes and the number of links. With regards to the entropy of degree distribution, graphs such as a star, circular, and complete graph are not complex because they have peaked degree distributions respectively around $k = 1$, $k=2$, *and* $k = n - 1$ ($n =$ number of nodes). This means very low entropy for star graph, and zero entropy for circular and complete graphs. Other entropy measures can be proposed based on the in-between-ness of nodes, in-between-ness of links or any other graph property [42–44].

Simple nonentropic measures include measures of the numbers of graph elements (nodes and links), or the sum or product of those elements. Spectral examples of nonentropic graph measure are maximum eigenvalue of adjacency matrix [49], sum of eigenvalues of adjacency matrix (also known as graph energy [6]), and determinant of adjacency matrix (which is the product of eigenvalues). Kolmogorov graph complexity measures such as linear graph complexity [50] are based on size of the minimal code that reconstructs the graph. For example, the size of a binary representation of adjacency matrix (which is a minimal representation of graph because it excludes the edges weights) is deemed as Kolmogorov graph complexity by Mowshowitz and Dehmer [40].

Other nonentropic graph complexity measures include [40, 41, 48]:

- The *number of spanning trees* [41] which is the number of distinct subgraphs with tree structures that also have same number of nodes as the graph they span.
- *Off-diagonal complexity* measures the diversity of node-node correlation matrix, is maximum for graphs with power law degree distribution (small world property). The underlying assumed reference is a regular graph (all nodes with constant degree) [41].
- *Graph efficiency* is the arithmetic mean of all inverse shortest path lengths. The underlying assumed reference is scale-free (power-law degree distribution) [41].
- *Cyclomatic complexity* [51] (the number of independent loops in a graph) is based on a preference for a tree-like structure as the reference simplicity. The measure was shown to be statistically correlated to the number of programing errors, and has been used to reduce the number of loops in structured software because cyclic operational dependence in system software (such as feedback loops) is also described as the primary source of emergent response or unintended consequences in a system [52].

3.3. Objectivity and Subjectivity of Graph Complexity

For use in the system complexity proposed in Eq. (1), we seek an objective measure for graph complexity. For a graph, the objective complexity measure must be a strictly increasing function of the number of nodes and the number of links. To be more precise, the measure must satisfy the following criterion of a connected graph (a graph that has no isolated components):

- The objective complexity of a connected graph must be strictly greater than the objective complexity of any its subgraphs [53,54].

An equivalent form of expressing this criterion is:

- The objective complexity must strictly decrease by removing a link or a node of a connected graph.

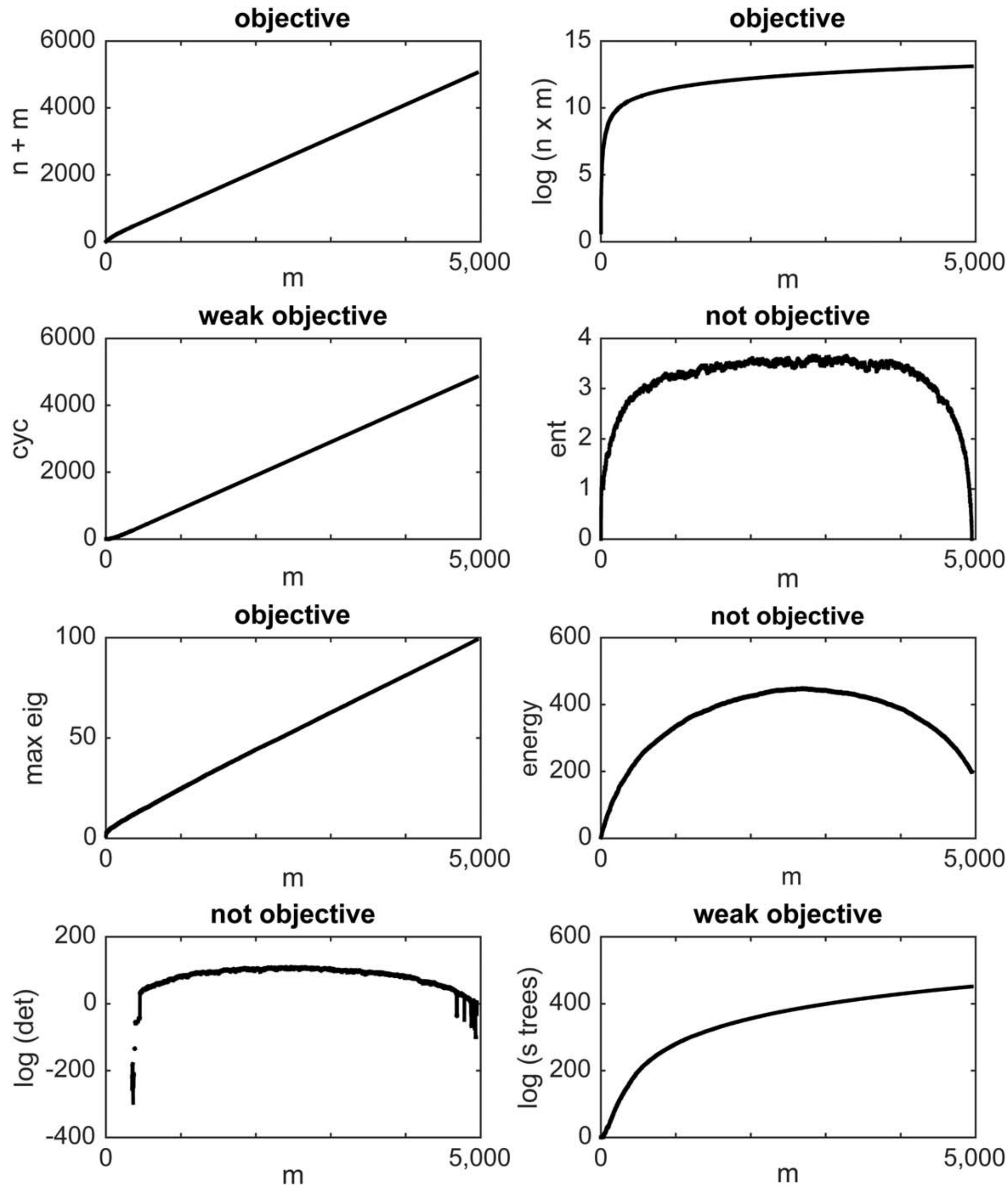
If a complexity measure meets the above criterion it is called *objective*. If it meets the criterion, except for the strictness part (i.e., e.g., the complexity measure remains unchanged by removing a link, or a node) then we can term the measure as *weakly objective*. Figure 4 shows the behavior of different complexity measures in a simple experiment, in which a node and a link were randomly added to a graph in steps up to a maximum of 100 nodes. After that point, only a link was randomly added at each step until a clique/fully connected graph was reached (with a maximum of 4950 links). As shown in Figure 4 from eight common graph measures (number of links plus number of nodes, number of links times number of nodes, cyclomatic number, entropy of degree distribution, maximum eigenvalue, graph energy, determinant, and number of spanning trees), only three measures are objective (sum and product of n and m and maximum eigenvalue), two (cyclomatic complexity and number of spanning trees) are weakly objective and three (degree distribution entropy, graph energy and determinant) are not objective.

3.4 Objective and Subjective Assumptions of Cyclomatic Complexity

Some measures described in the previous section have explicit assumptions about a reference model. Complexity

DOI 10.1002/cplx

FIGURE 4



Eight complexity measures and their relationship with the number of links. These measures from top to bottom are number of links plus number of nodes, number of links times number of nodes, cyclomatic number (cyc), entropy of degree distribution (ent), maximum eigenvalue (max eig), graph energy, determinant (det), and number of spanning trees (s trees).

measures such as off-diagonal complexity and graph efficiency are designed to have a preference for a particular form (i.e., they have low/high values for scale-free graphs). However, most measures do not make explicit their assumptions about a subjective simplicity or reference such as the determinant of the adjacency matrix. The problem with such measures is that they are focused on a specific use, or might prove useful in a specific context

DOI 10.1002/cplx

without a clear understanding of how the measure might be applied in a different context. Without a reference for simplicity, the measurement of subjective complexity is itself subjective and therefore unreliable which has led to justifiable criticism of such measures in the past. When the assumptions of an underlying reference model are surfaced, the utility of combination measures can be made more explicit.

For example, although it does not expressly separately identify subjective complexity, we can show that cyclomatic complexity (the number of loops as a measure of graph complexity) conforms to the general measure of Eq. (1) when it is recognized that the measure is based on a preference for tree-like structure as reference simplicity. For a connected graph, $G(n,m)$, cyclomatic complexity is calculated as [51]:

$$\mathrm{Cyc}(G) = m - n + 1 \tag{15}$$

We can reproduce the cyclomatic complexity measure using Eq. (1), if we assume any spanning tree of graph $G(n,m)$ as the reference model for objective simplicity, with an objective complexity calculated as the sum of n nodes and m links. The proof is simple. Assume an objective complexity measure as follows:

$$|G| = n + m \tag{16}$$

Since every spanning tree has $n-1$ links:

$$|\mathrm{ST}| = 2n - 1 \tag{17}$$

where ST is a spanning tree of G. Since the intersection of G and ST is ST itself:

$$\mathrm{MCS}(G,\mathrm{ST}) = \mathrm{ST} \Rightarrow D(G||\mathrm{ST}) = 1 - \frac{|\mathrm{ST}|}{|G|} = \frac{m-n+1}{m+n} \tag{18}$$

and finally, using Eq. (1):

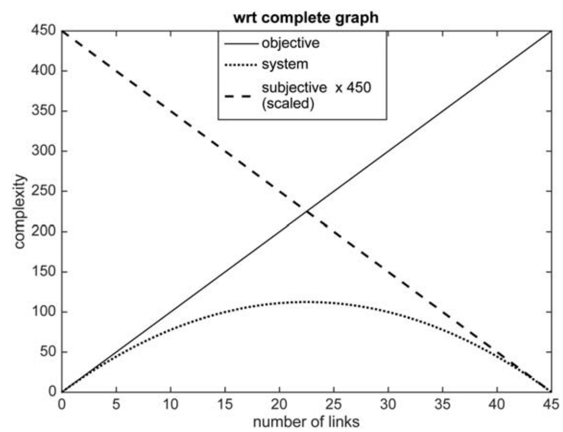
$$\mathcal{K}(G) = |G| \times D(G||\mathrm{ST}) = m - n + 1 \tag{19}$$

which is the same as the cyclomatic complexity in Eq. (15). So, without being explicit within the measure, cyclomatic complexity is generated by a subjective desire for tree-like graphs (i.e., graphs with no loops), and also by regarding the addition of links and nodes as objective complexity.

4. SOME ILLUSTRATIVE EXAMPLES

This section provides several examples to illustrate the utility of the general measure of system complexity. Unless otherwise stated, by way of example and without loss of generality, we use the following simple measure of the objective complexity a graph:

FIGURE 5



Objective, subjective and system complexity for a graph is created by regarding a complete 10-node graph as reference simplicity.

$$|G(n,m)| = n \times m \tag{21}$$

where n is the number of nodes and m is the number of links. It should be noted that, since we have MCS(G,R) = R or MCS(G,R) = G for all examples, all distance measures (10), (11), and (12) yield the same results.

4.1. Complete Graph as Reference

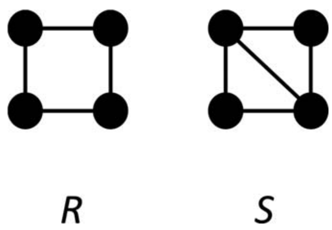
A complete graph can be considered to be as simple as a graph with no edges, because its description is very short and only knowledge of the number nodes is needed to reconstruct the complete graph in both cases. In graph theory, there has also been interest in creating complexity measures that vanish at the extremes [26,41] by multiplying an objective normalized graph complexity measure by one minus itself. Here, we show that by assuming a complete graph as reference simplicity, we arrive at the same result.

To illustrate this we conducted a small experiment to determine the behavior of the complexity of a graph as the number of links increases objectively and where the reference simplicity is a complete graph. In the experiment a random link was added to graph of maximum 10 nodes and at each step objective and subjective complexity was measured. Figure 5 shows that objective complexity increases monotonically as more links and nodes are added, because the amount of information needed to describe the new graph increases.

Conversely, subjective complexity decreases as the subject graph builds in size to be closer to the reference graph (to the reference simplicity). When there is a small number of nodes and links there is a maximum distance from the complete graph with 10 nodes. The distance decreases as the subject graph becomes more similar to

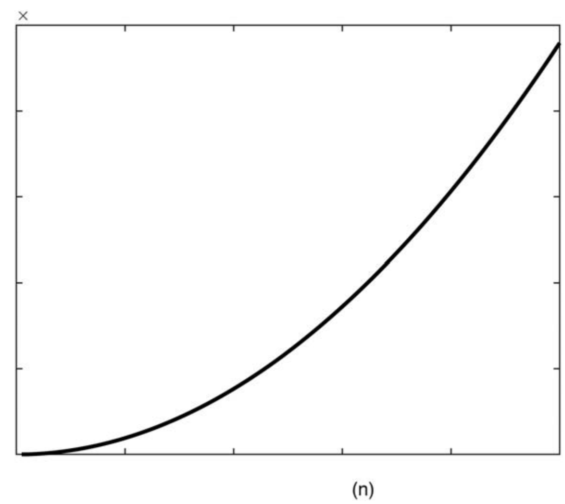
DOI 10.1002/cplx

FIGURE 6



The structures of a selected reference system, *R*, and a system under study, *S*.

FIGURE 8



Objective complexity of a modular system, S_n, where $n = 1100$.

the complete reference graph. By the time all possible links are made, the distance is zero. The subjective complexity in Figure 5 is scaled by a factor of 450 to display with the other values.

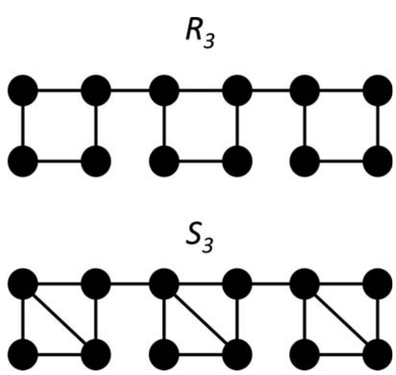
4.2. Complexity of Repeating Patterns

Suppose that we have perfect knowledge about a reference system R (see Figure 6) from a single perspective such as design, manufacturing/assembly, integration, maintenance, or operation, or from the perspective of any individuals or groups involved in those processes. We want to know complexity of the system, S, from the same perspective as the reference system. Since in this case the structural intersection (MCS) of the reference and the system is also equivalent to the entire reference system, we have:

$$|S|=20, |R|=16,\ \ D(S||R)=0.2,\quad \mathcal{K}=4 \qquad (22)$$

Now consider a system, S_n, that is composed of n modules, and each module is identical to the system S above, and integrated by a single link between two adjacent blocks, as shown in Figure 7 for $n = 3$. Figure 8 shows the objective complexity of S_n as a function of the number of modules (n). The objective complexity increases monotonically with the size of S_n.

FIGURE 7



A reference system, R_n, with $n = 3$ and a system, S_n, under consideration with $n = 3$.

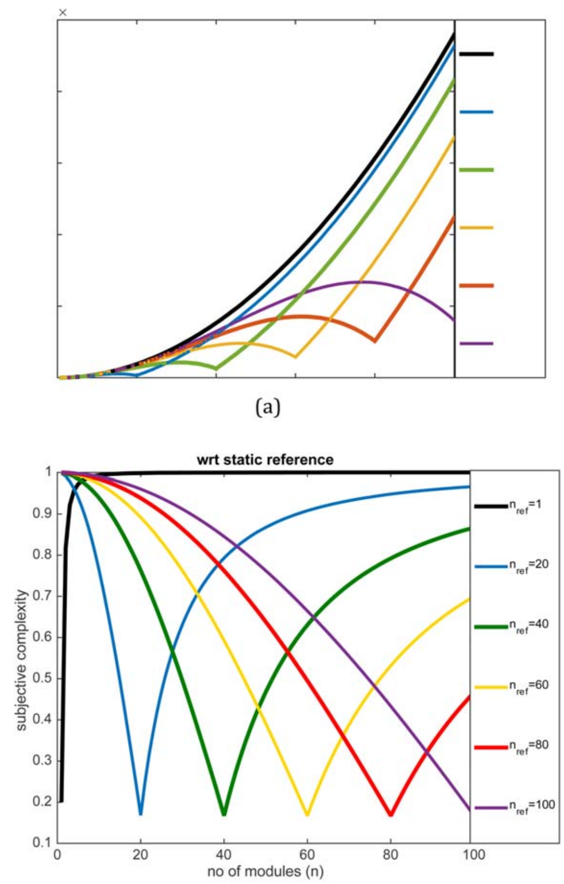
Now, we can consider two different reference types for simplicity: static and dynamic. Assume n_{ref} as the size of the reference model. If n_{ref} is independent of system size (n), the reference model is static. Figure 9 shows the subjective complexity and system complexity of a modular repeating system with reference to a range of static reference models (characterized by n_{ref}).

Figure 9(a) shows the variation in subjective complexity with the number of modules. Consider, for example, the case of a static reference simplicity $n_{\text{ref}} = 60$. The subjective complexity for a small number of modules is high because the difference from the reference system is large. As the number of modules increases the distance, and the consequent subjective complexity, decreases to a minimum when the number of modules equals n_{ref}. At that point ($n = 60$, in this case) the subjective complexity is at minimum—it would be zero if the two graphs were identical, but a small difference between G and R mean that there is a small subjective complexity. After that point, the subjective complexity again begins to rise because, as n increases, the static reference model becomes less representative and the distance between G and R grows.

For any given n_{ref}, under the influence of objective complexity, the system complexity in Figure 9(b) increases up to a maximum as the system size (n) increases, and then, because of the influence of subjective complexity reduces toward a local minimum at $n = n_{\text{ref}}$, before both subjective and objective components combine to create a steadily increasing complexity rate for $n > n_{\text{ref}}$. An

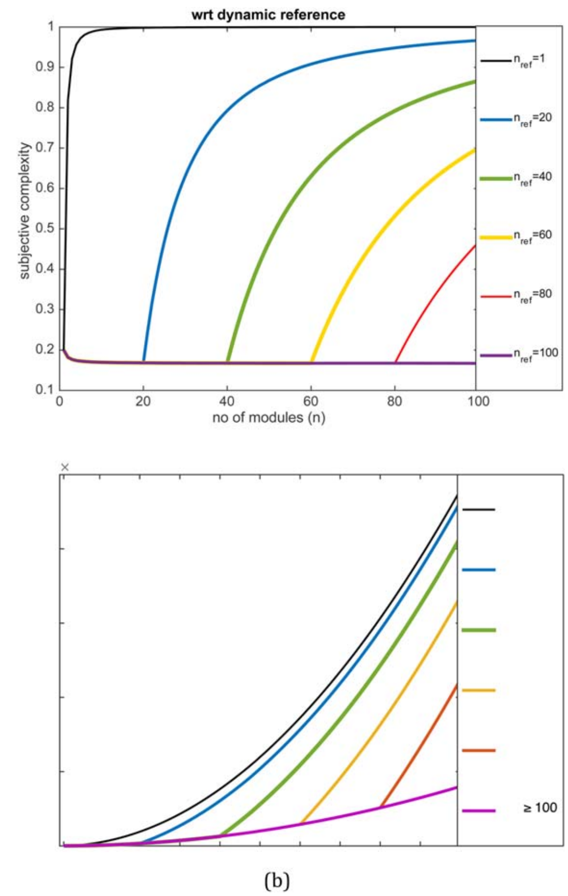
DOI 10.1002/cplx

FIGURE 9



Complexity as a function of number of modules, n, with reference to different static references with n_{ref} modules (a) for subjective complexity and (b) for the system complexity.

FIGURE 10



The effect on observed complexity of the use of dynamic reference models (a) for subjective complexity and (b) for system complexity.

interesting observation here is that the local minimum and maximum complexity values increase with n_{ref}, which is a result of a larger distance between the reference model and the system at these points as n_{ref} increases. Since this distance is relatively small for ($n_{\mathrm{ref}} = 1$), the minimum complexity is very close to zero, however the minimum complexity increases slowly as n_{ref} increases.

With a dynamic reference, the size of the reference model is allowed to change so that it always remains commensurate in size with that of the system (or $n_{\mathrm{ref}} = n$), as is the case for cyclomatic complexity. We can also cap this changeability of n_{ref} up to a maximum number of modules after which the reference remains static. Figure 10 shows the complexity profiles for subjective complexity: Figure 10(a), and system complexity, Figure 10(b). It is evident that observed complexity can only increase with respect to dynamic references. Consequently, for $n > n_{\mathrm{ref}}$ the reference model stays constant at n_{ref} in size, as occurred in the static case. Figure 10(a) shows the subjective complexity remains low since the difference between the n-module subject and the n_{ref} -module reference remains small until $n > n_{\mathrm{ref}}$ after which n_{ref} remains constant and the subjective complexity rises sharply. An interesting observation here is that for $n_{\mathrm{ref}} = 100$ the subjective complexity is lower than in the static case, and if we could plot the subjective complexity for a very large number of modules with a dynamic reference with no cap, the relative distance would be very small. Figure 10(b) shows the resultant observed complexity which for any size subject remains low until $n > n_{\mathrm{ref}}$ at which time the system complexity rises exponentially.

5. CONCLUSION

Although there are numerous complexity measures, each tends to be context-dependent and there have been

DOI 10.1002/cplx

few attempts to provide a generic structure that might provide a useful framework for such measures. Here, we show that a useful complexity measure for engineered systems has two components. The first is an objective component, which is an increasing function of the number of system elements/objects and their relations. This component can be thought of as objective complexity, context-independent complexity, or absolute complexity. The second component of complexity measure is the subjective component, which is the distance of the system from a reference model for simplicity. The subject component can be thought of as subjective complexity, context-dependent complexity, or relative complexity. For systems design, the objective component must be graph theoretic and conform to a number of other criteria set out in the article. The subjective component of the measure is a distance function expressing the topological distance from a reference model (structure) which is subjectively simple for the purpose for which the complexity measure is being applied.

Previous considerations of appropriate measures have not directly addressed the fundamental issue that the complexity of any particular matter or thing has a significant subjective component in which the degree of difficulty depends on available frames of reference. Any attempt to remove subjectivity from a measure of complexity therefore fails to address a very significant aspect of complexity. Conversely, there has been justifiable apprehension toward purely subjective complexity measures, simply because they are not verifiable if the frame of reference being applied is in itself both complex and subjective. We address this issue by introducing the concept of subjective simplicity—although a justifiable and verifiable value of subjective complexity may be difficult to assign directly, it is possible to identify in a given context what is "simple" to the observer and, from that reference, determine subjective complexity.

The system complexity measure proposed here allows the study of system complexity from the perspectives of multiple stakeholders who naturally have a diverse set of reference simplicities. In each case, a candidate solution has a constant objective complexity and a number of subjective ones depending on the points of view of stakeholders. Solutions can then be compared based on:

- That with the lowest objective complexity.
- That with the lowest subjective complexity for certain stakeholders.
- That with the smallest deviation in subjective complexity over all stakeholders.

The proposed system complexity measure has the additional advantage of being adaptable in terms of subjective complexity with reference to dynamic models that evolve with the learning of observers.

ACKNOWLEDGMENT

We would like to thank two anonymous reviewers of the complexity journal for their contribution to this work by thorough, insightful and invaluable remarks.

BIOGRAPHY

Dr Mahmoud Efatmaneshnik is a researcher at School of Information Technology and Electrical Engineering, University of New South Wales at Australian Defence Force Academy, in Canberra. He has a PhD in Complexity management of Design Process (2009), and a ME in manufacturing engineering (2005) both from UNSW. He has a BE in aerospace (1999) from Tehran Polytechnic university. He is author of more than 35 book chapters, journal papers, and refereed conference papers.

Dr Mike Ryan is a Senior Lecturer with the School of Information Technology and Electrical Engineering, University of New South Wales, Canberra, at the Australian Defence Force Academy. He holds Bachelor, Masters, and Doctor of Philosophy degrees in electrical engineering as well as a Graduate Diploma in Management Studies. He lectures and regularly consults in a range of subjects including communications and information systems, systems engineering, requirements engineering, and project management. He is the conference chair of two international conferences each year, he is the editor-in-chief of the Journal of Battlefield Technology, and is chair of the Requirements Working Group in the International Council on Systems Engineering (INCOSE). He is the author or co-author of eleven books, three book chapters, and over one hundred technical papers and reports.

REFERENCES

1. Beckerman, L.P. Application of complex systems science to systems engineering. Syst Eng 2000, 3, 96–102.
2. Calvano, C.N.; John, P. Systems engineering in an age of complexity. Syst Eng 2004, 7, 25–34.
3. Sheard, S. Assessing the Impact of Complexity Attributes on System Development Project Outcomes; PhD Thesis, Stevens Institute of Technology, 2012.
4. Summers, J.D.; Shah, J.J. Mechanical engineering design complexity metrics: Size, coupling, and solvability. J Mech Des 2010, 132, 021004
5. Stuart, D.; Mattikalli, R. META II complexity and adaptability. In: DARPA META Project Final Report; Boeing Company, 2011.

DOI 10.1002/cplx

6. Willcox, K.; Allaire, D.; Deyst, J.; He, C.; Sondecker, G. META II stochastic process decision methods for complex cyber-physical systems. In: DARPA META Project Final Report; MIT, 2011.
7. Brian, T.; Murray, A.P.; Skelding, R.; de Weck, O.; Zhu, H.; Nair, S.; Shougarian, N.; Sinha, K.; Bopardikar, S.; Zeidner, L. META II complex systems design and analysis (CODA). In: DARAP META Project Final Report; United Technologies Corporation, 2011.
8. Sheard, S.A.; Mostashari, A. Complexity measures to predict system development project outcomes. INCOSE Int Symp 2013, 23, 170–183.
9. Feldman, D.P.; Crutchfield, J.P. Measures of statistical complexity: Why? Phys Lett A 1998, 238, 244–252.
10. Heylighen, F. The growth of structural and functional complexity during evolution. In: The Evolution of Complexity; F. Heylighen, J. B. A. R., Ed.; Kluwer Academic, Dordrecht, 1999, p 17–44.
11. Weaver, W. Science and complexity. Am Sci 1948, 36, 536–544.
12. Edmonds, B. Syntactic Measures of Complexity; University of Manchester, 1999.
13. Sussman, J.M. Ideas on Complexity in Systems–Twenty Views; MIT, Ed., 2000. Available at: http://web.mit.edu/esd.83/www/notebook/20ViewsComplexity.PDF
14. Sussman, J.M. The new transportation faculty: The evolution to engineering systems. In: Perspectives on Intelligent Transportation Systems (ITS); Springer US, 2005; pp 101–116.
15. Gell-Mann, M. What is complexity? Remarks on simplicity and complexity by the Nobel Prize-winning author of The Quark and the Jaguar. Complexity 1995, 1, 16–19.
16. Antunes, L.; Souto, A.; Vitanyi, P.M. On logical depth and the running time of shortest programs, in press. arXiv:1310.6976 2013.
17. Bennett, C.H. Logical Depth and Physical Complexity; Springer, 1995.
18. Gell-Mann, M.; Lloyd, S. Information measures, effective complexity, and total information. Complexity 1996, 2, 44–52.
19. Wackerbauer, R.; Witt, A.; Atmanspacher, H.; Kurths, J.; Scheingraber, H. A comparative classification of complexity measures. Chaos Solitons Fractals 1994, 4, 133–173.
20. Standish, R.K. On complexity and emergence, in press. arXiv nlin/0101006 2001.
21. Dick, J.; Jones, B. On the Complexity of Requirements Flow-down Structures; INCOSE: Rome, Italy, 2012.
22. Kelley, J.L. General Topology; Springer-Verlag, New York, 1975.
23. Arkhangel'skiĭ, A.; Fedorchuk, V.V. The basic concepts and constructions of general topology. In: General Topology I; Springer, 1990; pp 1–90.
24. Wolpert, D.H.; Macready, W. Using self-dissimilarity to quantify complexity. Complexity 2007, 12, 77–85.
25. López-Ruiz, R.; Mancini, H.L.; Calbet, X. A statistical measure of complexity. Phys Lett A 1995, 209, 321–326.
26. Shiner, J.S.; Davison, M.; Landsberg, P.T. Simple measure for complexity. Phys Rev E 1999, 59, 1459–1464.
27. Crutchfield, J.P.; Young, K. Inferring statistical complexity. Phys Rev Lett 1989, 63, 105
28. Crutchfield, J.P.; Feldman, D.P.; Shalizi, C.R. Comment I on "Simple measure for complexity". Phys Rev.E 2000, 62, 2996
29. ISO/IEC-15288 Standard. System and Software Engineering - System Life Cycle Processes, 2008.
30. Faulconbridge, I.; Ryan, M.J. Systems Engineering Practice; Argos Press: Canberra, Australia, 2014.
31. INCOSE. Systems engineering handbook—A guide for system life cycle processes and activities, Version 4, 2015. In: Version 4. Walden, D.; Roedler, G.; Forsberg, K.; Hamelin, R.; Shortell, T., Eds.; 2015.
32. Wallis, W.D.; Shoubridge, P.; Kraetz, M.; Ray, D. Graph distances using graph union. Pattern Recognit Lett 2001, 22, 701–704.
33. Fernández, M.L.; Valiente, G. A graph distance metric combining maximum common subgraph and minimum common supergraph. Pattern Recognit Lett 2001, 22, 753–758.
34. Bunke, H.; Shearer, K. A graph distance metric based on the maximal common subgraph. Pattern Recognit Lett 1998, 19, 255–259.
35. Bunke, H.; Jiang, X.; Kandel, A. On the minimum common supergraph of two graphs. Computing 2000, 65, 13–25.
36. Xiao, Y.; Dong, H.; Wu, W.; Xiong, M.; Wang, W.; Shi, B. Structure-based graph distance measures of high degree of precision. Pattern Recognit 2008, 41, 3547–3561.
37. Levi, G. A note on the derivation of maximal common subgraphs of two directed or undirected graphs. Calcolo 1973, 9, 341–352.
38. Barrow, H.G.; Burstall, R.M. Subgraph isomorphism, matching relational structures and maximal cliques. Inf Process Lett 1976, 4, 83–84.
39. Bunke, H. On a relation between graph edit distance and maximum common subgraph. Pattern Recognit Lett 1997, 18, 689–694.
40. Mowshowitz, A.; Dehmer, M. Entropy and the complexity of graphs revisited. Entropy 2012, 14, 559–570.
41. Kim, J.; Wilhelm, T. What is a complex graph? Phys A Stat Mech Appl 2008, 387, 2637–2652.
42. Dehmer, M.; Mowshowitz, A. A history of graph entropy measures. Inf Sci 2011, 181, 57–78.
43. Krivovichev, S. Topological complexity of crystal structures: Quantitative approach. Acta Crystallogr A 2012, 68, 393–398.
44. Simpson, J.J.; Simpson, M.J. Entropy metrics for system identification and analysis. Syst Eng 2014, 17, 140–156.
45. Grone, R.; Merris, R. A bound for the complexity of a simple graph. Discrete Math 1988, 69, 97–99.
46. Jukna, S. On graph complexity. Comb Probab Comput 2006, 15, 855–876.
47. Berwanger, D.; Grädel, E. Entanglement—A measure for the complexity of directed graphs with applications to logic and games. In: Logic for Programming, Artificial Intelligence, and Reasoning; Springer, 2005; pp 209–223.
48. Standish, R.K. Complexity of networks (Reprise). Complexity 2012, 17, 50–61.

DOI 10.1002/cplx

49. Efatmaneshnik, M. Towards immunization of complex engineered systems: Products, processes and organizations. In: Mechanical and Manufacturing Engineering; University of New South Wales, 2009.
50. Neel, D.L.; Orrison, M.E. The linear complexity of a graph. In: Advances in Network Complexity; Wiley-VCH Verlag GmbH & Co. KGaA, 2013; pp 155–175.
51. McCabe, T. A complexity measure. IEEE Trans Software Eng 1976, 2, 308–320.
52. Erdi, P. Complexity Explained; Springer: Berlin, 2008.
53. Hornby, G.S. Modularity, reuse, and hierarchy: Measuring complexity by measuring structure and organization. Complexity 2007, 13, 50–61.
54. Efatmaneshnik, M.; Reidsema, C.; Marczyk, J.; Balaei, A. Immune decomposition and decomposability analysis of complex design problems with a graph theoretic complexity In: Measure Smart Information and Knowledge Management; Szczerbicki, E.; Nguyen, N., Eds.; Springer Berlin/Heidelberg, 2010; pp 27–52.

DOI 10.1002/cplx