A Network Approach to Define Component Modularity

by

Manuel Sosa
Steven Eppinger
and
Craig Rowles

2006/16/TOM

Working Paper Series

A Network Approach to Define Component Modularity

Manuel E. Sosa[1]
Assistant Professor of Technology and Operations Management
INSEAD
Fontainebleau, FRANCE

Steven D. Eppinger
Sloan School of Management
Massachusetts Institute of Technology
Cambridge, Massachusetts

Craig M. Rowles
Pratt & Whitney Aircraft
East Hartford, Connecticut

Abstract

Modularity has been previously defined at the product and system level, however little effort has been applied to define and quantify modularity at the component level. We consider complex products as a network of components that share technical interfaces in order to function as a whole. Building upon previous work in graph theory and social network analysis, we define three measures of component modularity based on the notion of centrality. Our measures consider how components share direct interfaces with adjacent components, how design interfaces may propagate to non-adjacent components in the product, and how components may act as bridges between other components through their interfaces. We calculate and interpret all three measures of component modularity by studying the product architecture of a large commercial aircraft engine. We illustrate how to use these measures to test the impact of modularity on component redesign. Our results show that the relationship between component modularity and component redesign depends on the type of design interfaces connecting product components. Directions for future work are discussed.

[1] Corresponding author: manuel.sosa@insead.edu. Tel: +33 (0)1 60 72 45 36. Fax: +33 (0)1 60 74 61 79.

Introduction

Previous research on product architecture has defined modularity at the product and system level [1,2,3], however little effort has been dedicated to study modularity at the component level [4]. Although complex products are typically considered as a network of components that share interfaces in order to function as a whole [5,6,7], there are no quantitative measures that allow us to distinguish components based on how they share interfaces with other components in a product. In this paper we define measures to quantify the relative degree of modularity of components in complex products based on their connectivity with other components within the product.

We formally define *component modularity* based on the patterns of a component's design interfaces with other components. Understanding architectural properties, such as component modularity, is particularly important for established firms which often fail to identify and manage novel ways in which components share interfaces [8]. Managing interfaces between components becomes even more difficult when developing complex products, hence it is critical for managers to proactively identify the components that will require particular attention during the design process [9,10]. Many important design decisions depend on how components connect with other components in the product, yet we still do not have accepted measures to capture how disconnected (i.e. how modular) a component is. *Do modular components require more (or less) attention from their design teams during their development process? Are modular components easier to redesign or outsource?* In order to answer such questions, we propose to quantitatively measure modularity at the component level.

The need for measuring modularity was implicitly highlighted by Saleh [11] in his recent invitation "to contribute to the growing field of flexibility in system design" (p. 850). Saleh [11] laments that "there isn't yet a coherent set of results that demonstrates how to embed flexibility in the design of complex engineering systems, nor *how to evaluate it* and trade it against other system attributes such as performance or cost." [p. 849, emphasis added]. Defining and measuring modularity at the component level (as opposed to the product level) is an important step in addressing this void in the engineering design literature because it can provide quantitative approaches to *evaluate* the flexibility associated with components embedded in complex products. Our proposed definitions of component modularity can therefore be the starting point of a long-needed discussion about architectural properties of product components.

Our work makes two important contributions. First, we integrate the literature of product architecture, social networks, and graph theory to define and measure modularity at the component level based on the notion of centrality. We apply our definitions to determine the modularity of the components of a large commercial aircraft engine. Second, this paper illustrates how to test the impact of component modularity on important design decisions such as component redesign. In particular, we show that the relationship between component modularity and component redesign is not trivial and depends on the type of design interface connecting product components. Our approach illustrates how to study the relation between component modularity and other important performance attributes of components. We conclude the paper with discussion of results and comments for future work.

Literature Review

This work builds upon streams of research in product architecture and social networks. We also refer to graph theory, which has provided the foundation to define properties of both products and social networks when considered as graphs and digraphs of connected nodes. We blend these research streams together by defining and measuring three types of component modularity.

Product Architecture

The literature on product decomposition and product architecture begins with Alexander [12] who describes the design process as involving decomposition of designs into minimally coupled groups. Simon [5] elaborates further by suggesting that complex systems should be designed as hierarchical structures consisting of "nearly decomposable systems" such that strong interfaces occur within systems and weak interfaces occur across systems. This is consistent with the independence axiom of axiomatic design which suggests decoupling of functional and physical elements of a product [6]. Taking a more strategic view, Baldwin and Clark [13] argue that modularity adds values by creating options that enables the evolution of both designs and industry.

Ulrich [1] defines the architecture of a product as the "scheme by which the function of a product is allocated to physical components." A key feature of product architecture is the degree to which it is modular or integral [14]. In the engineering design field, a large stream of research has focused on methods and rules to map functional models to physical components [15,16]. Digraph representations have been used to model networks of connected components comprising complex products [17] and to develop physical layouts [18]. Graphs, trees, and matrices have also been used to study decomposition of complex products [19,20,21]. As for measures of modularity, most of the previous work has been concentrated at the product level [4]. Modularity measures consider similarity and dependency links between product components [21,22,23].

As Ulrich [1] suggested, establishing the product architecture not only involves the arrangement of functional elements and their mapping to physical components but also the specification of the interfaces among interacting components. In order to capture the structure of product architectures in terms of component interactions we use the design structure matrix (DSM) tool. The DSM is a matrix-based graphical method introduced by Steward [24] and used by Eppinger et al [25] to study interdependence between product development activities. DSM representation has also been used to document product decomposition and team interdependence [26, 27, 10], and to model the risk of design change propagation in complex development efforts [9, 28, 29]. More recently, researchers have extended the use of DSM representations of complex products to analyze their architectures at the product level [30,31,32].

In an earlier paper, Sosa et al [3] use a matrix representation not only to capture the decomposition and interfaces between product components but also to extend the concepts of product modularity to the system level. In [3] we introduced a new notion of system modularity based upon the way components share design interfaces across systems. We aim to extend this work further by defining measures that allow us to categorize components based on the direct and indirect interfaces they share with other components in the product.

Social Networks

A social network is a set of actors who are connected by a set of ties. The actors or "nodes" can be people, groups, teams, or organizations. Ties connect pairs of actors and can be directed (for example, A gives advice to B) or undirected (for example, A and B are friends) and can be binary (for example, whether A gives advice to B or not) or valued (for example, frequency of interactions between A and B). Social network analysis is the study of social relations among a set of actors. Network analysts argue that how an individual behaves depends in large part on how that individual is tied into the larger web of social connections [33,34]. They also postulate that the success or failure of societies and organizations often depends on the interactions of their internal entities [35,36]. Beginning in the 1930s, a systematic approach to theory and research, based on the above notions, began to emerge. In 1934 Jacob Moreno introduced the ideas and tools of sociometry [37]. At the end of World War II, Bavelas [38] noted that the structural arrangement of ties linking members of a task oriented group may have consequences for their productivity and morale. He proposed that the relevant structural feature was *centrality*, and he defined this in formal terms. Since then, social network analysis has extended into research areas that span from analysis of people in an organization to analysis of board interlocks, joint ventures and inter-firm alliances and trade blocks - drawing upon such fields as sociology, anthropology, and mathematics [39].

We identify two streams of research in the field of social networks. First is the work focused on developing network indices to capture structural properties of social networks at the individual and group level [40, 41, 34, 36]. Second is the stream of work that focuses on how social network properties of individuals or teams impact the performance of organizational processes [42, 43, 36].

The work most relevant to our paper is the stream of research focused on developing centrality measures in social organizations. Freeman [40] discusses three different measures of centrality: degree, closeness, and, betweenness. Degree centrality refers to the simplest definition of actor centrality which indicates that central actors must be the ones that have the most ties to other actors in the network, or the ones which other actors depend upon the most. A second perspective of centrality is based on how close an actor is to all the other actors in the network, implying that an actor is more central if it can quickly reach all others. A third view on centrality is related to the role of the broker (or gatekeeper) between other actors in a social network. That is, interactions between two non-adjacent actors may depend on the other actors in the network, in particular the ones that lie on the path between the two [44]. Indices for all three centrality measures have been developed for both non-directed (symmetric) and directed (asymmetric) relations between actors and groups [40,41,34].

In addition to centrality, there are other measures of social network properties, such as power, constraint, and redundancy; however, their translation to the product domain is less apparent [34, 36]. Algorithms to compute most of these structural properties are available and implemented in network computer programs such as UCINET [45].

Graph Theory

Graph theory [46] has been widely used in social network analysis [34, 35, 36, 40, 41] and, to a lesser extent, in engineering design [19,20,47]. The most salient benefits of using graph theory to study networks include first, the usage of a common language to label and represent network properties, and second, the availability of mathematical notions and operations with which many of these properties can be quantified and measured [34, p. 93].

In the product architecture literature, previous research has used graph theory to operationalize concepts such as reachability and iterations [47]. Michelena and Papalambros [48] use hypergraph formalisms for optimal model-based decomposition of design problems. Gebala and Eppinger [49] compare DSM models with other graph based models such as program evaluation and review technique (PERT) charts and structured analysis and design technique (SADT), to study design procedures. In the social network domain, many graph theoretical notions such as graph density, node degree, geodesic distance, and bridges have provided the foundations to measure many structural properties in social networks [46,50].

One of the main notions that social network analysis derives from graph theory is to identify the most important actors in a network. Actors who are the most important (also referred as prominence or prestigious actors) are usually located in "central" locations within the network. Centrality measures aim to identify "the most important (or prominent)" actors in a social network [34,40]. In our context, this would translate to identifying the most central (or most integral) components in a complex product. Graph theory has been used to quantify the notion of centrality from different perspectives. For example, Hakimi [51] and Sabidussi [52] quantify a view of centrality by suggesting that central nodes in a graph have "minimum steps" when linking to all other nodes. Other equally sound views of centrality suggest that central nodes should have a maximum number of connections with other nodes and/or they should lie on the shortest paths between the largest possible number of other nodes [40]. We will develop further the link between these distinct views of centrality and component modularity in the next section.

Defining Component Modularity

The term 'modularity' has received widespread attention across various disciplines [1, 2, 3, 4, 13, 22, 23, 32, 53, 54] and thus far there is confusion about its definition and the ways to measure it [2]. In order to measure modularity it is important to clarify the various level of analyses on which the term can be defined. Doing so is particularly relevant when designing complex products due to their decomposition into systems and components [5]. Figure 1 shows how a product can be decomposed into several systems, and these further decomposed into components. Modularity, therefore, can be defined at the product, system, and component level.

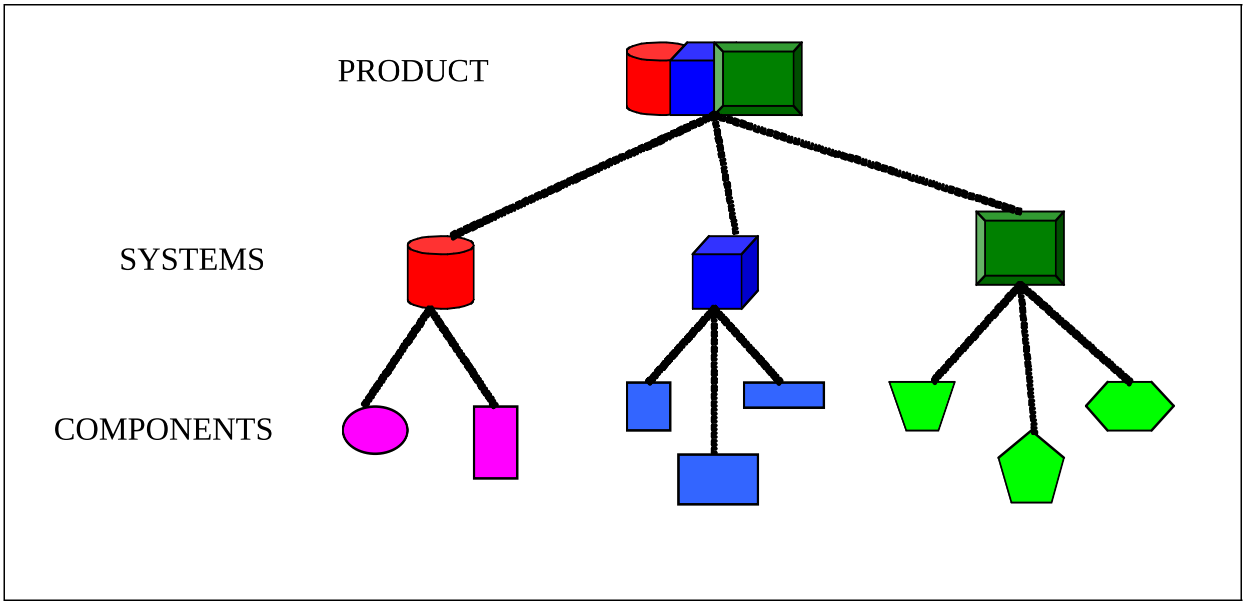

Fig. 1. Hierarchical decomposition of a product

At the product level, modularity is considered an important product characteristic that results from mapping the product's functions to its physical components [1, 14]. Moreover, Ulrich [1] defines modular product architecture as resulting from a one-to-one mapping between functional elements and physical components, and including "de-coupled component interfaces" [1, p. 423]. At the system level, Sosa et al [3] define system modularity based on how systems share interfaces with components of other systems. They define modular systems "as those whose design interfaces with other systems are clustered among a few physically adjacent systems, whereas integrative systems are those whose design interfaces span all or most of the systems that comprise the product due to their physically distributed or functionally integrative nature throughout the product". In this paper, we define and measure modularity at the component level.

In order to define component modularity, we consider products as networks of components that share design interfaces in order to function as a whole. Hence, we analyze each component's network defined by its interfaces with all other components in the product. We define component modularity as the *level of independence of a component from the other components in a product.* Hence, the more independent (or disconnected) a component is (i.e. the more "degrees of freedom" a component has), the more modular it is. We assume that components lose design independence due to their connections with other components, which we call *design dependencies*. As a result, we aim to measure component modularity by considering the patterns of design dependencies of a component. This argument is similar to the underlying proposition in social network studies, whereby various structural characteristics of nodes are defined based on their patterns of interactions. Figure 2 shows a network view of the hypothetical product decomposition, where we have added the component dependencies to the structure shown in Fig. 1. Figure 2 also shows the network of the most modular and least modular components in such a network based on their connectivity with the other components in the product. Yet, we still need to quantify the degree of connectivity of a component within a product. We do this based on the notion of centrality.

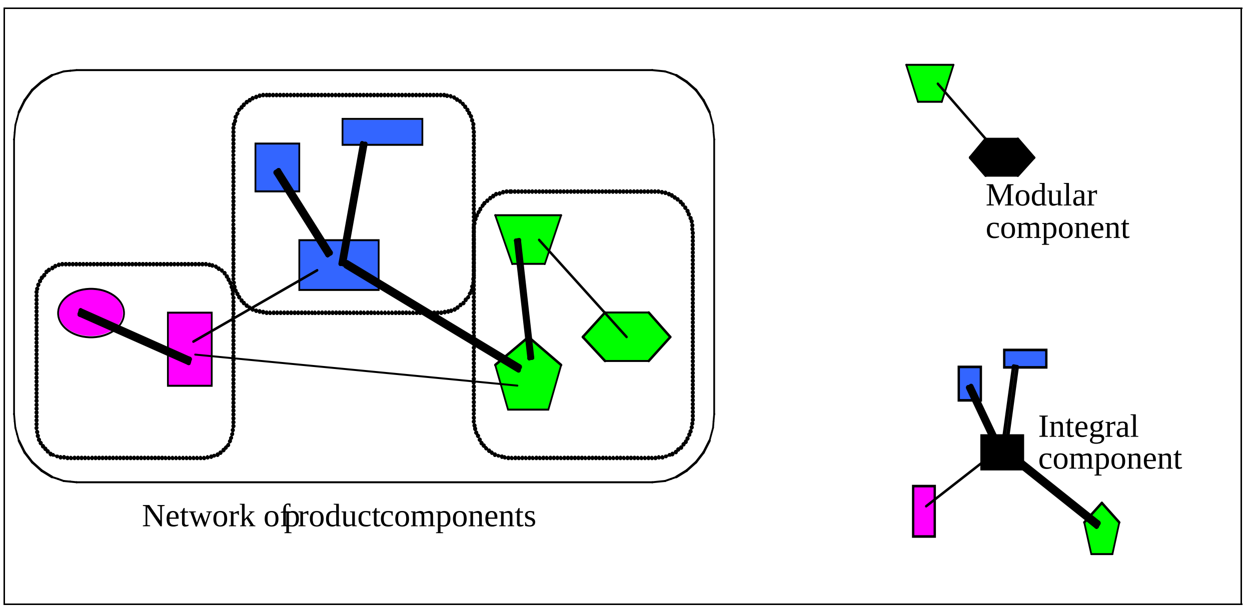

Fig. 2. Network representation of a product

Using graph theory, social network research has quantified structural properties for individuals, teams, and organizations in a social network. Centrality is one of the most important structural properties in social network analysis and also our starting point for defining modularity at the component level. The less central a component is, the fewer direct and indirect design dependencies it has upon (and from) other components.

As mentioned, we define modularity as the lack of connectivity (measured as the lack of centrality) of a component from the other components in the product. In general terms, we operationalize component modularity as the ratio of actual component "disconnectivity" to the maximum "disconnectivity" a component could have in a product of n components. Hence,

Component_modularity =(Actual component_disconnectivity)/(Max possible component_disconnectivity)

Note that the expression above is a standardized measure of component modularity that depends on how we measure the connectivity of a component within the product. We measure connectivity by considering either direct connections, indirect connections, or bridging connections. Based on this, we develop three indices for component modularity: degree, distance, and bridge modularity.

Design Dependency Matrix, X

In order to formally define modularity measures for product components, we define the design dependency matrix, **X**. **X** is a square matrix whose columns and rows are identically labeled with the components of the product. Let **X** refer to the matrix of design dependencies for any type of design dependency. Previous work in engineering design has identified various types of design dependencies between components such as spatial, structural, material, energy, and information [55, 26, 3]. Hence, **X** captures the dependency between components for any given design domain. In order to be consistent with [3], we maintain that **X** has non-zero elements, X_{ij}, if component i depends for functionality on component j. The value of X_{ij} indicates the strength of the design dependency, ranging from 0 to x_{max}. Note that the diagonal elements, X_{ii}, are defined to be zero.

Degree Modularity

Our simplest definition of component modularity is *Degree Modularity*, *M(D)*. This measure is negatively related to the number of other components with which a given component has direct design dependencies. The larger the number of components that affect, or are affected by, the design of component i, the less modular component i is.

As per graph theory conventions [46, 50], "the degree of a node is the number of lines that are incident with it" [46, p. 14]. The degree of a node therefore ranges from a minimum of 0 to a maximum of (n-1) if there are n nodes in a graph. In the product architecture domain, a node is a component and an arc (i.e. link between two nodes) is a design dependency. Since design dependencies have both direction and strength we need to extend the concept of node degree to valued directed graphs in order to define degree modularity.

The *In-Degree* of a component *i* is equal to the number of other components that *i* depends on for functionality, whereas *Out-Degree* is equal to the number of other components that depend on component *i*. Thus we define, for a product with *n* components, the *In-Degree Modularity* of component *i*, $M(ID)_i$, as

$$M(ID)_i = \frac{Actual\_degree\_disconnectivity}{Maximum\_degree\_disconnectivity} = \frac{Maximum\_degree\_disconnectivity - Actual\_degree\_connectivity}{Maximum\_degree\_disconnectivity}$$

Hence,

$$M(ID)_i = \frac{x_{\max}(n-1) - x_{i+}}{x_{\max}(n-1)} = 1 - \frac{x_{i+}}{x_{\max}(n-1)}$$ where $x_{i+} = \sum_{i \neq j}^{n} X_{ij}$ and x_{max} is the maximum value that X_{ij} can take.

Similarly, the *Out-Degree Modularity* of component *i*, $M(OD)_i$, can be defined as

$$M(OD)_i = 1 - \frac{x_{+i}}{x_{\max} \cdot (n-1)} \quad \text{where } x_{+i} = \sum_{i \neq j}^{n} X_{ji}$$

Note that the maximum degree of disconectivity is reached when a component is not connected to any other component in the product. Moreover, $M(ID)_i$ and $M(OD)_i$ range over [0,1]. The minimum value of degree modularity corresponds to a component that has <u>strong</u> design dependencies with all other (*n*-1) components of the product. Hence, such a component would be highly integral. The value of degree modularity increases linearly as the degree of a component decreases. If there are no design dependencies (either $x_{i+} = 0$ or $x_{+i} = 0$), the component is completely disconnected from others for that dependency direction and the resulting in- or out-degree modularity is equal to 1.

Distance modularity

While degree modularity captures how many other components are directly linked to component *i*, it does not consider indirect ties by which component *i* can have design dependencies with other components in the product network. Here, we argue that the modularity of component *i* also depends on how "distant" it is from all other components in the product. In social network theory, closeness centrality is the concept we build upon. Closeness centrality of an actor reflects how close an actor is to other actors in the network. As Freeman [40, p. 224] suggested, "the independence of a point is determined by its closeness to all other points in the graph." These ideas were originally discussed by Bavelas [38]. Yet, it was not until Sabidussi [52] proposed that actor closeness should be measured as a function of geodesic distance that a simple and natural measure of closeness emerged. (In graph theory, a geodesic is the shortest path between two nodes, and geodesic distance, or simply distance, between two nodes is defined as the length of their geodesic [46]). We incorporate these ideas into the product architecture domain by using the notion of "distance" between components – the more distant a component is from the other components, the further its design dependencies have to propagate, hence, the more modular the component is.

Formally, we define *Distance Modularity*, *M(T)*, to be proportional to the summation of the geodesics of component *i* with all other components in the product. Distance modularity depends on the direction, but not upon the strength, of the design dependencies.

Let *d(i,j)* denote the geodesic of design dependency between component *i* and component *j*. Thus, the *In-Distance Modularity*, $M(IT)_i$, is defined as

$$M(IT)_i = \frac{Actual\_distance\_disconnectivity}{Maximum\_distance\_disconnectivity} = \frac{\sum_{i \neq j}^{n} d(i,j)}{n(n-1)}$$

Similarly, *Out-Distance Modularity*, $M(OT)_i$, is defined as follows

$$M(OT)_i = \frac{\sum_{j \neq i}^{n} d(j,i)}{n(n-1)}$$

where *d(j,i)* denotes the shortest path of design dependency in the other direction - component *j* depends on component *i*.

A high value of $M(IT)_i$ or $M(OT)_i$ means that component *i* is far away from the others and therefore is more modular. The denominator of our index corresponds to the maximum distance of a disconnected component. For this we assume that disconnected components are *n* steps away from all other components in the product. Hence, disconnected components have a distance modularity of 1. The minimum value of distance modularity will be $(1/n)$, which is reached when component *i* is adjacent to all other components (i.e. the component is completely integral).

Bridge Modularity

A third way of measuring modularity is to focus on those components that lie in between the dependency path of two components. We can view these components as having control over design dependency flow since the design dependencies must propagate through them. In this sense, these components can be considered as bridges, or conduits that transmit design dependencies through the product network. The more a component bridges between other components, the less modular it is. We argue that components lose modularity as their bridging position increases. As a result we define *bridge modularity* of component *i* based on the number of times it is on the path of two other components.

Social network theory describes centrality in terms of the brokerage position of social actors (they call it betweenness centrality). Bavelas [38] and Shaw [56] have suggested that actors located on many geodesics are central to the network. Anthonisse [57] and Freeman [44] were the first to quantify the actor's betweenness indices.

We assume that components lying on the most geodesics will be the ones bridging the most components, and therefore the least modular. This assumption makes sense in the product domain if a design dependency between two components propagates through the minimum number of parts (i.e. the shortest path or geodesic). Hence, we calculate the ratio of all geodesics between components *a* and *b* which contain component *i* ($nd_{ab}(i)$) to the between *a* and *b* (nd_{ab}). This yields a measure of how much component *i* bridges between *a* and *b*. Note that nd_{ab} is not the geodesic distance *d* but the total number of geodesics between *a* and *b*. Summing over all pairs of components *a* and *b* in the product gives us a measure of the bridging potential of component *i*.

Our measure of Bridge Modularity, *M(B)*, then takes the form

$$M(B)_i = \frac{Actual\_bridge\_disconnectivity}{Maximum\_bridge\_disconnectivity} = 1 - \frac{\sum_{i \neq a, i \neq b, a \neq b} nd_{ab}(i) / nd_{ab}}{[(n-1)(n-2)]}$$

Note that the maximum bridge disconectivity occurs when a component does not bridge any other pair of components because it is not on any of the (*n*-2)(*n*-1) maximum possible paths between the other (*n*-1) components (not including component *i*). Note that the fewer geodesics component *i* is on, the higher the value of $M(B)_i$, and the more modular component *i* is. The minimum value of this index is 0, which is reached for a perfectly integral bridging component that is on the geodesic of all other pairs of components.

We consider the proposed measures of component modularity to be complementary of each other because they emphasize related but distinct features about the patterns of design interfaces between product components. *Degree modularity* only takes into account the effects of immediate neighbors neglecting the connections beyond adjacent components. In addition, it is the only measure we propose that captures the strength of the design dependency. Since design dependencies are not necessarily symmetric [3], we define *in-degree* and *out-degree* modularity. The lower the component degree, the more modular the component is because it is more independent from its adjacent components. *Distance modularity*, on the other hand, captures the effect of indirect design dependencies by quantifying the mean distance to all other components in the product. Hence, the further apart a component is, the more modular it is. This measure, however, does not consider the effect of the design dependency strength. Similar to degree modularity, we need to distinguish between *in-distance* and *out-distance* modularity to take into account the direction of "propagation" of design dependencies. Finally, *bridge modularity* is based on the component's role in bridging other components. The less bridging role a component has, the more modular it is. This measure assumes binary design dependencies.

All three of these measures are based on the underlying argument that as components lose degrees of freedom by sharing more design interfaces with other components they become less modular. Consequently, less modular components are components with many direct and indirect interfaces and/or occupying bridging positions in the product. Although defining these measures is important to advance our understanding of product architectures, some important questions remain to be answered: Can we assume that various design dependencies are independent of each other? What relative weight should be given to each design dependency? (Recall that component modularity measures can be defined for multiple types of design dependency, such as for spatial, structural, material, energy, and information dependencies between product components.) Are modular components less likely to fail than less modular components? Are they more or less likely to be redesigned? In the next two sections of the paper, we illustrate how to empirically address such important questions.

Measuring Component Modularity in a Complex Product

This section illustrates how to compute and use component modularity measures in a complex product such as a large commercial aircraft engine. First, we discuss how component modularity measures correlate to each other across various design dependencies. Then, we discuss the link between component modularity and component redesign.

Data

We have applied our network approach to analyze the modularity of the components of a large commercial aircraft engine, the Pratt & Whitney PW4098. The engine is decomposed into eight systems (See Fig. 3). Each of these systems is further decomposed into five to ten components each, for a total of 54 components. Six of the eight systems have been identified as modular systems, whereas the other two systems (mechanical components system and externals and controls system) are recognized as integrative systems because of the physically distributed and functionally integrative features of their components [3].

After documenting the general decomposition of the product, we identified the network of design interfaces between the 54 components of the engine. We distinguished five types of design dependencies to define the design interfaces between the physical components (Table 1). In addition, we used a five-point scale to capture the level of criticality of each dependency for the overall functionality of the component in question (Table 2). We discuss these metrics at length in [3]. Note that design dependencies only refer to interactions that impact the function of the component in question. We do not consider coincidental design dependencies, which could exist between spatially adjacent components.

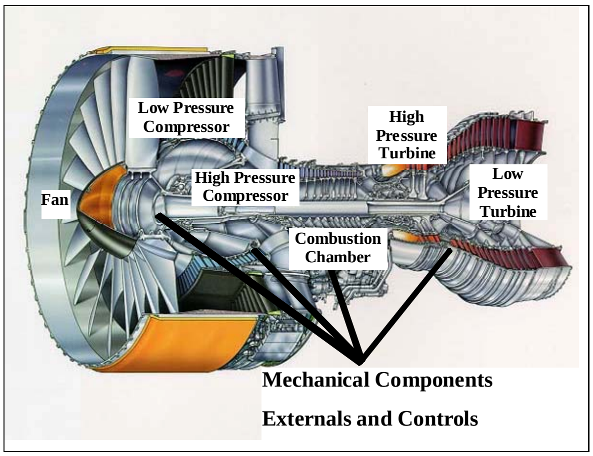

Fig. 3. PW4098 commercial aircraft engine studied

Table 1. Types of design dependency

Dependency	Description
Spatial	Functional requirement related to physical adjacency for alignment, orientation, serviceability, assembly, or weight.
Structural	Functional requirement related to transferring loads, or containment
Material	Functional requirement related to transferring airflow, oil, fuel, or water
Energy	Functional requirement related to transferring heat, vibration, electric, or noise energy
Information	Functional requirement related to transferring signals or controls

Table 2. Level of criticality of design dependencies

Criticality	Description
Required (+2)	Dependency is necessary for functionality
Desired (+1)	Dependency is beneficial, but not absolutely necessary for functionality
Indifferent (0)	Dependency does not affect functionality
Undesired (-1)	Dependency causes negative effects, but does not prevent functionality
Detrimental (-2)	Dependency must be prevented to achieve functionality

We tabulated our product architecture data in five design interface matrices corresponding to each type of design dependency. For the purpose of our analysis we consider three levels of criticality: Indifferent (0), Weak (-1, +1), and Strong (-2,+2). For illustration purposes, Fig. 4 shows the (54 x 54) design dependency matrix and the network representation for the spatial design dependencies of the engine studied. The network map shows 54 nodes, corresponding to the 54 components of the engine. The nodes are shaded differently to indicate that components belong to one of the eight major engine systems.

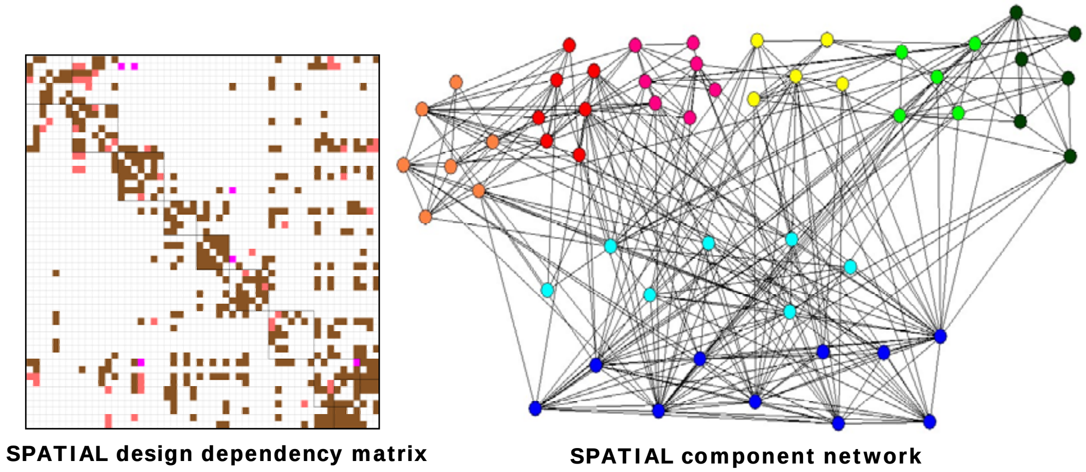

Fig. 4. Spatial design dependency matrix and spatial-type network diagram

Modularity of Engine Components

In this section we calculate and interpret modularity measures for the engine components. Our measures are calculated according to the definitions provided in the previous section. Descriptive statistics are shown in Table 3. Note that distance modularity measures are the ones that exhibit larger coefficient of variation both within and across design dependency types[2]. This is consistent with previous empirical research in social networks that has shown that distance based centrality measures are more sensitive to small changes in network configurations than degree and betweenneess centrality measures [34]. In order to visualize the variation in component network configurations associated with low and high component modularity, Fig. 5 exhibits the "ego" network of components with low and high modularity scores[3].

[2] Coefficient of variation is defined as STDEV/Mean of a random variable.
[3] The ego network of component *i* only shows the other components it directly share interfaces with as well as the interfaces among them.

Table 3. Descriptive statistics of modularity measures

	Spatial		Structural		Material		Energy		Information	
	Mean	**S.D.**	**Mean**	**S.D.**	**Mean**	**S.D.**	**Mean**	**S.D.**	**Mean**	**S.D.**
1 In-degree	0.85	0.09	0.91	0.07	0.89	0.07	0.95	0.05	0.97	0.07
2 Out-degree	0.85	0.08	0.91	0.06	0.89	0.08	0.95	0.04	0.97	0.05
3 In-Distance	0.04	0.01	0.05	0.01	0.10	0.12	0.37	0.39	0.83	0.22
4 Out-Distance	0.04	0.01	0.05	0.01	0.10	0.18	0.37	0.16	0.83	0.18
5 Bridge	0.97	0.03	0.97	0.04	0.97	0.05	0.97	0.04	0.9999	0.0005

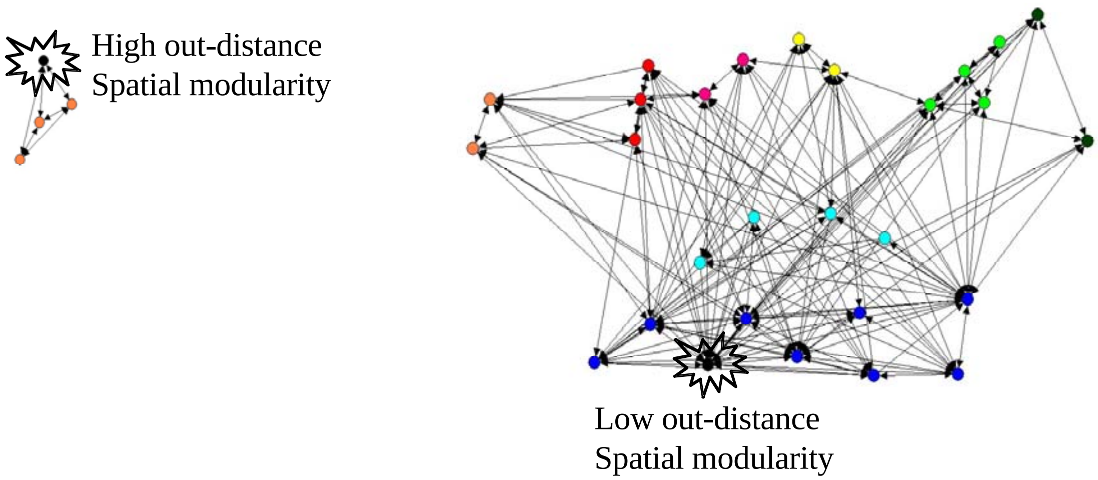

Fig. 5. Ego network graphs of components with low and high out-distance modularity measures for spatial dependencies

In order to study the relation between our component modularity measures for a given design dependency as well as to understand better the relation between the various design dependencies for each modularity measure, we perform two correlation analyses. First, we analyze the extent to which modularity measures differ from each other within each design dependency type (Table 4). This is important because if correlations are high between component modularity metrics for all dependency types, then one might be able to use only a subset of the component modularity metrics. Then, we study the extent to which modularity measures help us highlight the differences (and similarities) between design dependency types (Table 5). This is also important because this can provide empirical evidence to justify the identification and use of all five design dependency types separately.

Table 4a. Partial correlation coefficients between modularity measures

	Spatial				Structural			
	1	**2**	**3**	**4**	**1**	**2**	**3**	**4**
1 In-degree	1.0				1.0			
2 Out-degree	.770**	1.0			.536**	1.0		
3 In-Distance	.812**	.686**	1.0		.802**	.552**	1.0	
4 Out-Distance	.629**	.832**	.798**	1.0	.585**	.821**	.721**	1.0
5 Bridge	.734**	.842**	.624**	.681**	.766**	.808**	.658**	.687**

Table 4b. Partial correlation coefficients between modularity measures

	Material				Energy				Information			
	1	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
1 In-degree	1.0				1.0				1.0			
2 Out-degree	.833**	1.0			.688**	1.0			.846**	1.0		
3 In-Distance	.247	.211	1.0		.519**	.409**	1.0		.572**	.556**	1.0	
4 Out-Distance	.302*	.309*	.691**	1.0	.168	.338*	.233	1.0	.451**	.702**	.606**	1.0
5 Bridge	.791**	.833**	.131	.171	.614**	.673**	.455**	.155	.858**	.829**	.354**	.299*

*Correlation significant at the 0.05 level (2-tailed); ** Correlation is significant at the 0.01 level (2-tailed)

Table 4 shows the partial linear correlation coefficients among all the measures for each design dependency. We find significantly positive correlation coefficient among all measures of component modularity for spatial, structural, and information design dependencies. That is, within spatial, structural, and information design dependency domains, all our modularity measures greatly coincide in their assessment of component modularity. Correlation coefficients are less significant within material and energy design dependencies, particularly with respect to several of the distance modularity measures. For example, within the material domain, the variation of in-distance modularity is not strongly associated with the variation of (in- or out-) degree modularity nor of bridge modularity. Similarly, within the energy domain, the variation of out-distance modularity is not strongly associated with the variation of in-degree modularity nor of bridge modularity. Since distance modularity captures how components are connected not only with neighboring components but also with all other components in the product, this result suggests that material and energy design change propagations would follow paths that are not strongly associated with direct dependencies, which in turn are better captured by degree and bridge modularity measures. Before discussing the implications of these results to the engine we studied, let us consider the second correlation analysis.

Table 5a. Partial Correlation Coefficients Between Design Dependencies

	In-degree				Out-degree			
	1	**2**	**3**	**4**	**1**	**2**	**3**	**4**
1 Spatial	1.0				1.0			
2 Structural	.751**	1.0			.674**	1.0		
3 Material	.527**	.218	1.0		.779**	.443**	1.0	
4 Energy	.617**	.564**	.208	1.0	.565**	.392**	.315*	1.0
5 Information	.620**	.415**	.194	.711**	.570**	.080	.359**	.604**

Table 5b. Partial Correlation Coefficients Between Design Dependencies

	In-Distance				Out-Distance				Bridge			
	1	**2**	**3**	**4**	**1**	**2**	**3**	**4**	**1**	**2**	**3**	**4**
1 Spatial	1.0				1.0				1.0			
2 Structural	.836**	1.0			.844**	1.0			.741**	1.0		
3 Material	.274*	.114	1.0		-.011	-.102	1.0		.733**	.402**	1.0	
4 Energy	.188	.131	.205	1.0	.422**	.469**	-.090	1.0	.537**	.431**	.559**	1.0
5 Information	.501**	.359**	.134	.183	.669**	.487**	.021	.189	.132	-.044	.035	.249

*Correlation significant at the 0.05 level (2-tailed); ** Correlation is significant at the 0.01 level (2-tailed)

Table 5 shows the partial correlation coefficients between the five design dependencies for all measures of component modularity. In general, the results show a significantly strong correlation between spatial and structural component modularity (for all measures of modularity) whereas material, energy, and information dependencies show weaker and/or less significant correlation coefficients particularly for distance and bridge modularity measures. This provides important empirical evidence suggesting to avoid considering modularity of a component based on ONLY one type of design dependency.

Additional empirical evidence from our study is consistent with the results of the correlation analyses. In our case study, many of the materials and energy design dependencies did not necessarily correspond to other types of design dependencies. For example, the design of many mechanical components of the oil system depend on many other components for material transfer, however their design is less dependent on other components for spatial, structural and energy requirements. Additionally, material and energy dependencies are more subjective and difficult to identify than structural and spatial dependencies. For example, turbine blade design depends on temperature and pressure profile of gases flowing from the turbine vanes (material dependency), and these are less likely to be known as design dependencies than the required clearance between them (spatial dependency). In other cases, many design dependencies are unidirectional. For example, blade designs for vibration margin (an undesired energy dependency) are dependent on the number of upstream vanes but not other way around. All these empirical observations are consistent with the observed lack of significant correlation across measures for energy and material dependencies (Table 4) and across dependency types for distance modularity measures (Table 5).

The Relation Between Component Modularity and Component Redesign

In the previous section we performed a descriptive analysis of the three proposed measures of component modularity. Yet, what can these measures be used for? In addition to using these measures to rank components according to their level of "disconectivity" within the product, we can also use them to enhance our understanding of performance-related attributes of product components. This is important for managers and engineers when making decisions about product components that depend on their connectivity with other components within the product. Some of these decisions include component engineering outsourcing, mitigation of component obsolescence, and component redesign [9,11]. In this section we use our modularity measures to build new understanding of how component modularity impacts component redesign decisions. We define *component redesign* as the percentage of actual novel design content relative to the previous design of such a component included in the previous version of the product.

Previous work in engineering design has studied design changes in complex products [9, 28, 58]. Yet, the link between modularity and redesign is not well understood [59]. We formulate two important but conflicting propositions that link component modularity and component redesign based on the assumption that design changes propagate across components due to their connectivity captured as various types of design dependencies [60]. Similar to previous work in engineering design [28, 59], these propositions distinguish between initiated and emergent design changes which result in planned and unplanned redesign, respectively.

An important implication of design change propagation is on *unplanned redesign* or design rework [62]. Since components are connected through various types of design dependencies, design changes in one component are likely to propagate to other components in the product. As a result a component that depends (directly or indirectly) on many other components and/or a component that it is "in the middle" of many other components is more likely to be redesigned to accommodate unforeseen design changes (or design changes greater than planned) occurring in (or required by) other components [9, 60, 61]. That is, the more inward interfaces a component has, the higher the likelihood that unforeseen changes in other components will carry into it [59]. Hence, we formulate our proposition due to unplanned redesign as

P1: Components with low in-degree, in-distance, and bridge modularity levels are more likely to exhibit higher levels of (unplanned) redesign.

A second implication of design change propagation is on the allocation of design changes among various components in a product. In complex products, managers and engineers need to choose which components to redesign in order to fulfill the functional requirements of the new product and/or to adapt to planned changes in adjacent components. While doing so, we expect engineers to redesign components that are less likely to impact others. That is, components with fewer outward design dependencies to other components in the product are better candidates to be redesigned. This is consistent with the argument of Baldwin and Clark [13] which suggests that modularity fosters innovation because it decouples design teams to work on independent modules. Hence, we formulate our proposition due to planned redesign as follows

P2: Components with high outdegree and outdistance modularity levels are more likely to exhibit higher levels of (planned) redesign.

In order to test our propositions with our data, we needed to capture the levels of planned and unplanned redesign of each of the 54 engine components. We were able to capture only the former. We did so by asking design teams to "provide an estimate of the level of redesign required for your parts or system for the PW4098, as a percentage of the prior existing engine design." Although we did not explicitly ask for it, we believe the answer to our question mostly captured planned redesign rather than unplanned redesign (i.e. design effort of adapting the component to the new product). The reason is that engineers' estimates of % redesign were normalized by a common reference point (previous engine model) and their knowledge of what it takes, and in this case actually took, to adapt the parts into the new configuration (foreseen and planned changes). Note that in the derivative engine studied very little unplanned redesign of major significance occurred or was required. During follow up interviews to validate our data, we identified two important sources of unplanned redesign that happened during development after the initial detail designs were released to make the first development parts. Yet, the estimates of % redesign of the components involved did not change because of nature of their rework: they were redesigned already and had to be done over (i.e. the amount of work performed was much higher but not much more % of redesign of these components occurred).

Since our component redesign data only captures planned redesign, we can only formally test our second proposition (*P2)*. In order to do so, we estimate the multivariate non-linear model specified below. Since our dependent variable is a fraction, estimating an ordinary least square (OLS) linear model may be problematic because the predicted values from an OLS regression can never be guaranteed to fall within the unit interval, which can result in biased coefficient estimates. In addition, the coefficient of a linear model assumes that the effect of a predictor variable is constant across all levels of the dependent variable, which again may not be accurate. There are several ways to address these issues. A common solution is to estimate a linear model for the log-odds ratio of the dependent variable, yet this involves adjusting observations on extreme values [62, p. 402]. A better alternative is proposed by Papke and Wooldridge [63] which does not require any data adjustment. We estimate our models with such a procedure in *Stata-SE 9* using GLM with family(binomial), link(logit), and robust standard errors. Note that we estimate the model adjusting standard errors for intra-group correlation using the cluster procedure implemented in *Stata*. We do this to take into account for the fact that components were architected into eight systems which suggests that observations within a given system may not be independent. In order to test the robustness of our results we estimated linear and semi-log functional forms and obtained analogous results to the ones presented in this paper.

E(component redesign of component *i* |**x**) = G[β_0 + $\beta_{spatial}$*(*spatial modularity* of component *i*) +

+ $\beta_{structural}$*(*structural modularity* of component *i*) + $\beta_{material}$*(*material modularity* of component *i*) +

+ β_{energy}*(*energy modularity* of component *i*) + β_{info}*(*information modularity* of component *i*)]

Component redesign is the dependent variable of interest whose variation we want to explain with component modularity measures for all five types of design dependencies. *G*[.] is the logistic function. β's are the partial effects which indicate the strength of the impact of each type of component modularity on the dependent variable (See [63] for details). Since we have five proposed metrics of component modularity, each emphasizing a distinct aspect of modularity, we estimate our model for each of these measures. Note that by estimating these models we are testing whether the proposed modularity measures for each design dependency have a significant relationship to component redesign. The results of our multivariate non-linear regression analysis are shown in Table 6. Partial regression coefficients are shown for each model. We also include the *Log pseudolikelihood* for each model to indicate the goodness of fit of each model.

Table 6. Effects of component modularity on component redesign					
	Model 1 In-Degree	**Model 2 Out-Degree**	**Model 3 In-Distance**	**Model 4 Out-Distance**	**Model 5 Bridge**
Constant	-4.706 (3.913)	1.052 (5.646)	-2.185* (1.198)	-1.780 (1.892)	-356.453* (195.315)
Spatial	-3.193 (3.446)	5.743 (4.744)	47.912 (66.241)	**244.405*** (31.919)**	-6.538 (10.354)
Structural	5.281** (2.338)	-4.811 (3.740)	-3.194 (53.412)	**-156.312*** (30.137)**	6.077 (5.364)
Material	3.687 (3.091)	3.078 (2.940)	.137 (.736)	-.100 (1.045)	6.046 (4.022)
Energy	-6.076 (7.501)	-6.612 (8.178)	.030 (.492)	1.585 (1.431)	355.676* (198.342)
Information	5.170 (4.302)	1.941 (6.384)	.252 (1.245)	-1.958 (1.573)	-.494
Log pseudolikelihood	-29.247	-28.580	-29.258	-26.834	-29.561
N	54	54	54	54	54

Robust standard errors adjusted for eight clusters in the system are shown between parentheses.
Significant levels: *< 0.1; ** < 0.05; *** < 0.01

Models 1 to 5 estimate component redesign using various types of component modularity. Not surprisingly, models 1, 3, and 5 poorly fit the data which is consistent with the empirical observation that our dependent variable captures planned redesign. On the other hand, Models 2 and 4 show a better goodness of fit to our data as evident by their lower *log pseudolikelihood*. We concentrate our discussion on Model 4, as it best fits the data.

Model 4 shows significant coefficients for spatial and structural dependencies. The significantly positive spatial coefficient indicates that the more modular components (from an out-distance perspective) in the spatial domain is the more likely to exhibit higher levels of redesign. That is, components that are less likely to transmit spatial dependency to others (because they are more distant to other components) are more likely to exhibit higher levels of redesign. This result is consistent with our second proposition (*P2*). Interestingly, Model 4 also shows that structural out-distance modularity negatively impacts component redesign, which appears (at least at first) not to support *P2* because it indicates that components that are more likely to transmit forces and loads to other components (i.e. less modular from a structural out-distance viewpoint) are more likely to exhibit higher levels of redesign. Finding such opposite effects on component redesign when measuring modularity based on the same criteria (out-distance modularity) is an apparent paradox.

The results are not conflicting if we distinguish between *desired* and *undesired* design propagation. Generally, spatial dependencies are interfaces that can disrupt the design of other components if they propagate through. As a result we expect engineers to avoid redesigning components that are tightly integrated with other components. This is consistent with the Baldwin and Clark [13] view of modularity which emphasizes decoupling of components (i.e. modularization) in order to avoid disruption and to encourage innovation within modules. However, there is an alternative view of the effects of modularity and innovation that relates to performance maximizing [1, 32]. This alternative view postulates that integrality is necessary to achieve better fulfillment of functional requirements. That is, in order to meet the new functional requirements of the engine there are some dependencies that are more likely to be intentionally propagated across components. According to our results, these desired dependency propagation are more likely to correspond to structural dependencies in our case study. In order to understand these results, we need to put them in the context of the development of this engine.

The PW4098 was a derivative engine which by definition required redesigning only those systems and components necessary to achieve the new higher level of performance. The main functional requirement driving engine performance was the increase of engine thrust which entailed the intentional transmission of greater longitudinal forces through the engine. This was achieved, in short, by increasing fan and turbine capacity, thus running the high-pressure core faster and hotter. As a result, components related to bearings, fuel, oil, and air flow (with structural, material, and energy transfers) had increased performance specifications and were redesigned as required. On the other hand, redesign of some components with stronger spatial dependencies was avoided, as they tend to be more disruptive, and largely refer to "competition" for common space. These results support the result that designers are more likely to concentrate design changes on components that are more distant from a spatial viewpoint, yet structurally closer to many other components. For example, the fan (which is a system that exhibited, on average, over 70 % redesign) is structurally linked to all the cases and rotor systems of the engine but not spatially linked to all of them. On the other hand, some mechanical load components such as bearings and shafts, which are spatially close to many other components through the engine but do not impact others by structural dependencies, exhibited less than 10% redesign.

Another component that illustrates well our results is the high-pressure turbine (HPT) first blade (with 25% component redesign) which has more spatial constraints than structural ones, with those spatial constraints being very “expensive” to change. The blade airfoil length is set by the engine flowpath as it is defined going through that stage in the HPT. To change the flowpath would likely cascade into changes required in virtually every part in the HPT, as well as potentially the rest of the engine flowpath. This would be a far more complex and extensive proposal than forcing the blade airfoil length to remain unchanged and dealing with the related disadvantages of that decision. In this case, increased speed and temperature of the engine core increased loads on the blade, rotor, and case structure. Engineers in turn responded with improved cooling configurations and reinforced structures as appropriate. The axial and radial clearance changes (gapping) were also minimized for similar reasons.

The results above illustrate the importance of having various component modularity measures to capture various aspects related to the connectivity of components in a complex product. In our case study, only out-distance modularity was meaningful to study how engineers allocate redesign decisions. The final portion of this discussion relates to the definition of our modularity measures. First, note that our three measures of component modularity linearly depend on centrality measures. An important advantage of using a linear functional form to describe the relation between modularity and centrality is that non-linear functions can be then specified when regression models are estimated using component modularity as predictor variables. That is, if researchers think that certain component attribute depends in a non-linear fashion on component modularity then they can still use our measures and stipulate such non-linearity in their regression model formulation.

Conclusions and Future Work

This paper enhances our understanding of product architecture concepts by providing formal definitions and measures of modularity at the component level. We take a network approach to define three measures of component modularity based on centrality measures originally developed to study social networks [40]. Our definitions of component modularity emphasize various aspects of modularity relevant at the component level. *Degree modularity* is negatively proportional to the number and strength of design dependencies with adjacent components. *Distance modularity* is proportional to the mean distance with all other components in the product. *Bridge modularity* is negatively proportional to the number of bridging positions that a component occupies in the dependency network. We quantify and interpret these measures for all five types of design dependencies documented for the components of a large commercial aircraft engine. We also illustrate how to use component modularity measures to empirically understand component performance metrics such as component redesign.

By using our component modularity measures we were able to test whether redesign efforts are concentrated upon more modular components. In our case study analysis, we found that modular components are favored for allocating design changes that can disrupt the design of other components, yet integrally connected components are favored for design changes associated with the fulfillment of key functional requirements. While we cannot claim the generality of these results before completing similar studies in other types of products in different industries, we would expect to obtain analogous findings to explain the link between component modularity and component redesign in other complex products such as computers, automobiles, and airplanes.

Having quantitative ways to determine the architectural position of a component within the product is particularly relevant in complex products comprised of many components that share many interfaces along various design domains. Establishing the relation between component modularity and product performance metrics (beyond component redesign explored in this paper) remains an interesting challenge for future work. Are modular components less likely to fail than integral components? Which type of component modularity is better a predictor of component failure? Since component modularity is based on its connectivity within a product, the same component can have different modularity measures across products. How does component modularity affect component sourcing and quality?

In this paper we have studied component modularity for one single product. We have not explored how component modularity changes over time. Having quantitative ways to easily capture component modularity will be useful to track these measures along several product generations. Doing so can enhance our understanding of how changes in the architecture of the product affects the network properties of each component.

Although we believe our three proposed measures of component modularity have substantial meaning and are relatively simple to calculate (once the network of component design interfaces has been documented), we also believe that future efforts should be dedicated to develop alternative measures that capture other architectural properties of components based on how they share design interfaces. How can we combine these measures to have an aggregated measure of component modularity? How can we extend these concepts to the system and product level? How do architectural properties such as component modularity relate to social network properties of the organizations that develop them? Our current research efforts are focusing on answering some of these questions [64].

Finally, this work opens new opportunities for research in the area of engineering design by combining product architecture representations and social network analysis. In this paper we have benefited from previous work done to study centrality measures of social networks. Other social network concepts that merit further research by the engineering design community are structural equivalence, group cohesion, structural holes, and social influence. How can we adapt these concepts to develop better product architectures?

Acknowledgements

We appreciate the assistance of the engineers at Pratt & Whitney Aircraft in development of the example presented in this paper. We thank Anupam Agrawal for his contributions to an earlier version of this paper presented at the 2005 ASME Design Theory and Methodology Conference. We also appreciate the comments of participants in the 7th International DSM Conference. We thank Martin Gargiulo for his help on data analysis issues with *Stata*, and Christoph Loch for his comments on a previous version of this manuscript.

References

1. Ulrich, K.T., 1995, "The Role of Product Architecture in the Manufacturing Firm," *Res. Policy*, **24,** pp. 419-440.
2. Gershenson, J.K., Prasad, G.J., and Zhang, Y., 2003, "Product Modularity: Definitions and Benefits," *Journal of Engineering Design*, **14**(3), pp. 295-313.
3. Sosa, M.E., Eppinger, S.D., and Rowles, C.M., 2003, "Identifying Modular and Integrative Systems and Their Impact on Design Team Interactions," *Journal of Mech. Design,* **125**(2), pp. 240-252.
4. Gershenson, J.K., Prasad, G.J., and Zhang, Y., 2004, "Product Modularity: Measures and Methods," *Journal of Engineering Design*, **15**(1), pp. 33-51.
5. Simon, H.A. 1981. *The Science of the Artificial*. Second edition, MIT Press, Cambridge, MA.
6. Suh, N. P. 2001. *Axiomatic Design: Advances and Application*, Oxford University Press, New York.
7. Kusiak, A. 1999. *Engineering Design: Products, Processes, and Systems*. Academic Press.
8. Henderson, R., and Clark, K., 1990, "Architectural Innovation: The Reconfiguration of Existing Product Technologies and the Failure of Established Firms. *Admin. Sci. Quart*. **35**(1), pp. 9-30.
9. Clarkson, P.J., Simons, C.S., and Eckert, C.M., 2004, "Predicting Change Propagation in Complex Design," *ASME Journal of Mechanical Design*, **126**(5), pp. 765-797.
10. Sosa, M.E., Eppinger, S.D., and Rowles, C.M., 2004, "The Misalignment of Product Architecture and Organizational Structure in Complex Product Development," *Management Science*, **50**(12), pp. 1674-1689.
11. Saleh, J. H., 2005, "Perspectives in Design: The Deacon's Masterpiece and the Hundred-Year Aircraft, Spacecraft, and Other Complex Engineering Systems," *Journal of Mech. Design*, **127**(3), pp. 845-850.
12. Alexander, C. 1964. *Notes on the Synthesis of Form*. Harvard University Press, Cambridge, MA.
13. Baldwin, C. Y., and Clark, K. B., 2000, *Design rules: Volume1: The power of modularity*. Cambridge, MA: MIT Press.
14. Ulrich, K., and Eppinger, S.D., 2004, *Product Design and Development*. 3rd. Edition. McGraw Hill, New York.
15. Kirschman, C. F. and Fadel, G. M, 1998, "Classifying Functions for Mechanical Design," *Journal of Mechanical Design*. **120**, pp. 475-482.
16. Stone, R. B., Wood, K. L., and Crawford, R. H., 2000, "A Heuristic Method for Identifying Modules for Product Architectures," *Design Studies*, **21**(1), pp. 5-31.
17. Kusiak, A., and Szczerbicki, E., 1993, "Transformation from Conceptual to Embodiment Design," *IIE Transactions*, **25**(4), pp. 6-12.
18. Kusiak, A. and Wang, J., 1993, "Decomposition of the Design Process," *Journal of Mechanical Design*, **115**, pp. 687-695.
19. Michelena, N. and Papalambros, P.Y., 1995, "A Network Reliability Approach to Optimal Decomposition of Design Problems," *ASME J. Mech. Des*. **117**(3), pp. 433-440.

20. Chen, L. and Li, S., 2005, "Analysis of Decomposability and Complexity for Design Problems in the Context of Decomposition," *ASME J. Mech. Des.*, **127**, pp. 545-557.
21. Wang, B. and Antonsson, E. K., 2005, "Hierarchical Modularity: Decomposition of Function Structures with Minimal Description Length Principle," *Proceedings of IDETC/CIE 2005, 17th Design Theory and Methodology Conference.* DETC2005-85173.
22. Gershenson, J.K., Prasad, G.J., and Allamneni, S., 1999, "Modular Product Design: A Life-Cycle View," *Journal of Integrated Design and Process Science*, **3**(4), pp. 13-26.
23. Newcomb, P. J., Bras, B. and Rosen, D. W., 1998, "Implications of Modularity on Product Design for the Life Cycle," *Journal of Mechanical Design.* Vol. 120, pp. 483-490.
24. Steward, D., 1981, The Design Structure Matrix: A Method for Managing the Design of Complex Systems," *IEEE Trans. Eng'g. Manag.* **EM-28**(3), pp. 71-74.
25. Eppinger, S.D., Whitney, D.E., Smith, R.P., and Gebala, D.A., 1994, "A Model-Based Method for Organizing Tasks in Product Development," *Res. in Eng'g. Des.* **6**(1), pp. 1-13.
26. Pimmler, T.U., and Eppinger, S.D., 1994, "Integration Analysis of Product Decompositions," *ASME Conference on Design Theory and Methodology*. Minneapolis, MN. pp. 343-351.
27. Browning, T. R., 2001, "Applying the Design Structure Matrix to System Decomposition and Integration Problems: A review and New Directions," *IEEE Trans. Eng. Mgmt*. **48**(3), pp. 292-306.
28. Eckert, C.M., Clarkson, P.J., and Zanker, W., 2004, « Change and Customization in Complex Engineering Domains," *Research in Engineering Design*, **15**(1), pp. 1-21.
29. Browning, T.R. and Eppinger, S.D., 2002, "Modeling the Impacts of Process Architecture and Cost and Schedule Risk in Product Development," *IEEE Trans. Eng. Mgmt*. **49**(4), pp. 428-442.
30. Sharman, D., and Yassine, A., 2004, "Characterizing Complex Products Architectures," *Systems Engineering*, **7**(1), pp. 35-60.
31. Braha, D., and Bar-Yam, Y., 2004, "Topology of Large-Scale Engineering Problem Solving Networks," Physical Review E., **69**.
32. Hölttä, K., Suh, E.S, and de Weck, O., 2005, "Tradeoff Between Modularity and Performance for Engineering Systems and Products," *International Conference on Engineering Design*. ICED Melbourne.
33. Freeman, L. C., 2004, *The Development of Social Network Analysis : A Study in the Sociology of Science*. BookSurge Publishing.
34. Wasserman, S., Faust, K., 1994, *Social Network Analysis*. Cambridge University Press, New York.
35. Coleman, J., 1990, *Foundations of social theory*. Cambridge, MA: Belknap Press.
36. Burt, R., 1992, *Structural Holes. The Social Structure of Competition*, Harvard University Press, Cambridge, MA.
37. Moreno, J., 1934, *Who Shall Survive?: Fundations of Sociometry, Group Psychotherapy, and Sociodrama*, Washington, D.C.: Nervous and Mental Disease Publishing Co. Second edition in 1953 and Third edition in 1978.
38. Bavelas A., 1948, "A Mathematical Model For Small Group Structures," *Human Organization*, 7, pp. 16-30.
39. Borgatti, S. and P. C. Foster (2003) "The network paradigm in organizational research: A review and typology." *Journal of Management* **29**, pp.991-1013
40. Freeman, L. 1979. Centrality in social networks. Conceptual clarification. *Social Networks* **1**, pp. 215-239.
41. Carrington, P. J., Scott J. and Wasserman, W. (*Eds*.) 2005**,** *Models and Methods in Social Network Analysis,* Cambridge University Press.
42. Ibarra, H.,1993, "Network Centrality, Power, and Innovation Involvement: Determinants of Technical and Administrative Roles," *Academy of Management Journal.* **36**(3), pp. 471-501.
43. Gargiulo, M., and Benassi, M., 2000, "Trapped in Your Own Net. Network Cohesion, Structural Holes, and the Adaptation of Social Capital," *Organization Science* **11**, pp.183-196.

44. Freeman, L., 1977, 'A Set of Measures of Centrality Based on Betweenness," *Sociometry* **40**, pp.35-41.
45. Borgatti, S.P., Everett, M.G. and Freeman, L.C., 2002, *UCINET for Windows: Software for Social Network Analysis*. Harvard: Analytic Technologies.
46. Harary, F.,1969, *Graph Theory*. Reading, MA: Adison-Wesley.
47. Park, H., and Cutkosky, M.R., 1999, "Framework for Modeling Dependencies in Collaborative Engineering Design Processes," *Research in Engineering Design*, **11**, pp. 84-102.
48. Michelena, N.F. and Papalambros, P.Y., 1997, "A Hypergraph Framework for Optimal Model-Based Decomposition of Design Problems," *Computational Optimization and Applications*, **8**(2), pp. 173-196.
49. Gebala, D.A. and Eppinger, S. D., 1991, "Methods for Analyzing Design Procedures," *Proceedings of the ASME Third International Conference on Design Theory and Methodology*.
50. Hage, P. and Harary, F., 1983, *Structural Models in Anthropology*. Cambridge: Cambridge University Press.
51. Hakimi, S.L., 1965, "Optimum Locations of Switching Centers and the Absolute Centers and Medians of a Graph," *Operations Research*, **12**, pp. 450-459.
52. Sabidussi, G., 1966, "The Centrality Index of a Graph," *Psychometrik*a, **31**, pp. 581-603.
53. Sanchez, R. and Mahoney, J.T., 1996, "Modularity, Flexibility, and Knowledge Management in Product and Organization Design," *Strategic Management Journal*, **17,** pp. 63-76.
54. Schilling, M.A., 2000, "Toward a General Modular Systems Theory and Its Application to Interfirm Product Modularity," *Academy of Management Review*, **25**(2), pp. 312-334.
55. Pahl, G. and Beitz, W., (Wallace, K., ed.), 1991, *Engineering Design, A Systematic Approach*, Springer-Verlag, New York.
56. Shaw M.E, 1954., "Group Structure and the Behavior of Individuals in Small Groups," *Journal of Psychology*, **38**.
57. Anthonisse, J.M., 1971, *The Rush in a Graph*. Amsterdam: Mathematische Centrum.
58. Fricke E., Gebhard, B., Negele, H. and Igenbergs. E., 2000, "Coping with Changes: Causes, Findings, and Strategies," *Systems Engineering*, **3**(4), pp. 169-179.
59. Eckert, C., Clarkson, J., and Earl, C., 2005, "Predictability of Change in Engineering: A Complexity View," *Proceedings of IDETC/CIE 2005, 17th Design Theory and Methodology Conference*. DETC2005-85435.
60. Mihm, J., Loch, C., and Huchzermeier, A., 2003, "Problem-Solving Oscillations in Complex Engineering Projects," *Manag. Sci*. **46**(6), pp. 733-750.
61. Smith, R.P., and Eppinger, S.D., 1997, "Identifying Controlling Features of Engineering Design Iteration," *Manag. Sci.* **43,** pp. 276-293.
62. Kennedy, P., 2003, *A Guide to Econometrics*, Fifth Edition, The MIT Press, Cambridge, Massachusetts.
63. Papke, L.E., and Wooldridge, J.M., 1996, "Econometrics Methods for Fractional Response Variables with an Application to 401(K) Plan Participation Rates," *Journal of Applied Econometrics*, **11**(6), pp. 619-632.
64. Sosa, M.E., Gargiulo, M., and Rowles. C.M., 2006, "Component Modularity, Team Network Structure, and Attendance to Technical Interdependences in Complex Product Development," *INSEAD Working Paper.*