



An investigation on the network topology of an evolving product family structure and its robustness and complexity

Kijung Park¹ · Gül E. Okudan Kremer²

Received: 15 May 2018 / Revised: 2 February 2019 / Accepted: 5 February 2019 / Published online: 28 February 2019
© Springer-Verlag London Ltd., part of Springer Nature 2019

Abstract

Products continuously change over time through product innovations satisfying new customer needs and technologies. The successive emergence of new products within a product family can be considered as product family evolution that necessarily involves changes in a product family design structure. Although product family design has been widely discussed in the extant literature, inherent evolving properties in a product family design structure have not been sufficiently explored in an analytical manner. To tackle this issue, this research aims to characterize underlying properties in an evolving product family structure based on a network science approach. First, a product family structure is represented as a network to describe relationships among the components of a product family structure. Then, topological properties and patterns in a product family structure network at each time period are investigated through a case study using the smartphone models of a major company. The results show that each product family structure network follows the topological properties observed in other real networks; the product family structure network evolves with both scale-free and small-world properties and with common and specific motifs during each specific time duration. Also, findings from this study suggest that a design structure with a scale-free network topology with commonality can have topological robustness due to a decrease in structural complexity.

Keywords Product family evolution · Product family structure · Network science · Network topology · Topological robustness · Structural complexity

1 Introduction

Companies continuously develop new products to achieve market dominance and high profits. Indeed, new products upgraded from previous products or completely new models substituted for older generation models steadily appear in competitive marketplaces. This continuous emergence of new products is regarded as the evolution of a product that is affected by inherent product structures (Shibata et al. 2005) as well as new product environments such as new customer needs, technologies, and markets (Otto and Wood 1998; Rinne 2004). The evolution of individual products can be considered within their associated product family.

Herein, the successive emergence of new products within a product family, resulting from inherent structural properties and adaptation to new product environments, is referred to as product family evolution.

The past company-oriented manufacturing environment, primarily focusing on maximizing profits through mass production, does not ensure business success under today's accelerating global market competition and rapidly changing customer needs (Tu et al. 2001). Adjusting to the challenging business environment, companies have realized the importance of product variety and customer satisfaction to gain competitiveness in customer-driven markets (ElMaraghy et al. 2013; Sanderson and Uzumeri 1995); this change in the business paradigm has led to a transition to mass-customization (Duray et al. 2000; Huang et al. 2008). Although the idea of providing a wide variety of products for customers is regarded as essential to generate new profit paths, introducing a single product at a time imposes inefficiency in the use of materials and components due to the lack of commonality and modularity among products (Anderson et al. 1997; Meyer and

✉ Kijung Park
kjpark@inu.ac.kr

¹ Department of Industrial and Management Engineering,
Incheon National University, Incheon 22012,
Republic of Korea

² Department of Industrial and Manufacturing Systems
Engineering, Iowa State University, Ames, IA 50011, USA

Lehnerd 1997). Thus, a focus on product variety itself causes additional costs and complexity in product lines to handle distinct raw materials, parts, and processes occurring from product diversification (Tseng et al. 1996).

Both academia and industry have tried to mitigate the negative impacts of product variety on manufacturing performance, and the concept of product family design based on a common platform has been widely accepted as an effective method to fulfill the need for an increasing product variety as well as the economy of scale (Halman et al. 2003; Jiao and Tseng 1999; Simpson et al. 2006). Platform-based product family design provides operational advantages such as complexity reduction in manufacturing multiple products, design quality improvement of new products, easy expansion and upgrade of existing products, and flexibility of manufacturing processes (Halman et al. 2003; Sawhney 1998).

The structure of a product family is an important aspect in product family design since it can involve different types of modularity (Salvador et al. 2002) and different levels of design and manufacturing complexity (Park and Okudan Kremer 2015). Existing studies analyze and model a product family structure to derive better design solutions from relevant structural characteristics (Jiao et al. 2007; Pirmoradi and Wang 2011). However, they tend to view product family design as creating an alternative product platform or a product family against an existing one through improvement in specific structural characteristics; the one-time design decision making approaches often overlook the underlying structural behavior of a product family that cannot be captured through the narrow view of product family changes in specific design parameters without understanding changes in the entire topology of the product family structure.

One of the critical factors in product family design is how much potential the product platform can realize to generate a continuous stream of new products during its lifetime (Meyer 1997; Meyer and Utterback 1993; Sköld and Karlsson 2013). From the perspective of product family evolution, earlier research primarily conceptualizes changes in a product family at a broad level in strategic marketing and management contexts to provide managerial directions (Meyer 1997; Meyer and Utterback 1993; Ohvanainen and Hietikko 2012; Ulrich 1995; Wheelwright and Sasser Jr. 1989). Recent studies more specifically address the behavior of a product family structure based on theoretical models and frameworks (Hou et al. 2013; Liu et al. 2008; Liu and Özer 2009; Shao et al. 2012; Xu et al. 2008). However, continuous changes of an actual product family over time have not been explored through a data-driven analysis in the extant literature. The lack of in-depth observation for continuous changes in a product family structure often results in approaches that are too general or ad hoc modeling without considering inherent evolving characteristics in a product family structure.

The research herein aims to characterize the inherent evolving nature of a product family design structure based on a data-driven network analysis for an actual time-dependent product family. The study focuses on the structural changes of products within a product family over time to reveal the underlying structural properties of a product family architecture. Topological variations in a product family structure network, representing the design structure of a product family, during successive time periods are viewed as consequences from the evolution process of a product family architecture. This view facilitates effective investigation to identify the evolving properties of a product family by keeping track of the changes of a product family structure network.

This study employs a network science approach (Barabási 2016; Newman 2010), that has been used to investigate the evolving properties and topological changes of complex networks in many different domains, to quantitatively analyze a continuous stream of new product family structures. The universality of topological properties observed in many different types of empirical and theoretical networks has reshaped the understanding of systems in that systems believed to be complex are not based on randomness but universal mechanisms governing their evolution (Barabási 2009). Adopting this point of view, this research considers a product family structure as a complex network with various components and their structural interactions that can be characterized by certain network topologies and evolving properties. In this paper, the evolution of a product family structure is addressed through a case study of historical smartphone product families revealed by a major company to effectively observe the evolution process due to the short life cycles of smartphones. The topological characteristics of the smartphone product family structure network in each time period are extracted through network measures and motifs, then their time-dependent changes are examined to see if the product family structure network evolves by following the universality of network characteristics. Furthermore, structural complexity measures are proposed to explain a relationship between structural complexity and robustness from the context of evolutionary opportunities in a product family design structure. Findings from this research serve as a basis to describe underlying dynamics shaping a product family structure beyond prevailing approaches to analyze a product family structure from a limited development scope.

2 Literature review

This section reviews studies that are associated with the concept and analysis of product family evolution. Three major approaches in the extant literature are categorized in Table 1.

Table 1 Research approaches related to product family evolution

Approach	Literature	Description
Bottom-up product family design	Thevenot et al. (2005), Ong et al. (2006), Shooter et al. (2007), Salhieh (2007), Thevenot and Simpson (2009), Liu et al. (2014) and Giovannini et al. (2016)	Redesign and reuse to efficiently improve current product family design
Product family evolution strategies	Tucker and Kim (2008), Ferguson et al. (2009) and Bryan et al. (2013)	Modification and rearrangement of structural configurations in a product family
	Wheelwright and Sasser Jr. (1989) and Meyer and Utterback (1993)	Representation of product family evolution in a map
	Meyer (1997), Seepersad et al. (2002) and Ohvanainen and Hietikko (2012)	Platform leveraging strategies in a market segmentation grid
	Ulrich (1995) and Shibata et al. (2005)	Categorization of evolution of a product (or a product family) architecture based on evolutionary properties
Product family evolution modeling	Xu et al. (2008), Liu and Özer (2009), Shao et al. (2012) and Hou et al. (2013)	Product family evolution modeled with dynamic factors and mechanisms to guide the evolution process
	Liu et al. (2006, 2008) and Sui and Zhong (2008)	Modeling an evolving product family by employing complex network theory

2.1 Bottom-up product family design

Redesign, design reuse strategy, and reconfiguration of product families are bottom-up approaches that deal with efficient design improvements by modifying existing product variants (Alizon et al. 2007; Pirmoradi and Wang 2011). Although most studies in this category do not imply the concept of product family evolution, their methods to revitalize existing product families may be useful to reveal the evolutionary process of product families. Product family redesign is a typical bottom-up approach to find improved product family designs by revising original product portfolios or product families such as redesigning existing product families based on commonality (Thevenot et al. 2005; Thevenot and Simpson 2009), deriving an optimal product family architecture through component redesign to satisfy customer and budget requirements (Liu et al. 2014), and transforming individual product portfolios into product families (Salhieh 2007). Similarly, design reuse aims to decrease design risks, occurring from uncertainty and insufficient information in the early design stage, by increasing design and production efficiency for building new product families through reusing existing design elements. Ong et al. (2006) developed a design reuse methodology for product family design that facilitates the synthesis and evaluation of design information. Giovannini et al. (2016) proposed an approach for the representation of the retrieval and the reuse of design knowledge for product family design.

The reconfiguration of a product family structure is a process to rearrange available modules and structural configurations. Tucker and Kim (2008) incorporated a predictive data mining technique and a multilevel optimization method to reflect changing customer performance preferences on product family reconfiguration. Ferguson et al. (2009) presented a multilevel multidisciplinary design optimization approach to derive a core product family architecture that can generate products reconfigured by adaptable design parameters. Bryan et al. (2013) proposed a reconfiguration planning method to generate cost effective assembly systems to produce a product family that evolves over time.

These bottom-up approaches enable a new product family to meet various structural and functional requirements under limited time and cost conditions (Pirmoradi and Wang 2011). They can be regarded as possible options by which a current product family generation may evolve into its next generation. The bottom-up approaches provide important managerial implications for certain changes in a product family design or architecture. However, they are often limited to generating specific structural changes within a product family at a given time period. To understand the time-dependent evolutionary characteristics of

a product family design structure, it is necessary to have a more comprehensive approach that can define generic properties and patterns given a continuous stream of structural changes in a product family.

2.2 Product family evolution strategies

Mapping product family evolution is a representative early work that directly conceptualizes product family evolution, based on visualized maps to conceptually depict the evolution of product families from the perspectives of core capability and product development strategy (Meyer and Utterback 1993; Wheelwright and Sasser Jr. 1989). Wheelwright and Sasser Jr. (1989) proposed a generic product development map to illustrate relationships among product offerings in each generation through a series of maps. On these maps, product offerings are categorized into core and leveraged products, and leveraged products are further separated into enhanced, customized, cost reducing, and hybrid products. Meyer and Utterback (1993) provided a product family map to illustrate product family evolution over time and to create brand-new product families through new technology adaptation, cost reduction, and feature reconfiguration. They stated that the successive new generations of existing product families renew older platforms through improved designs and technologies to satisfy new customer streams in a targeted market. At the same time, new product families based on new product platforms can be simultaneously developed to pioneer new markets by expanding core technologies, market knowledge, and manufacturing capabilities from existing platforms.

Although the above mapping approaches to product family evolution can be effectively used to plan product positioning, they do not clearly explain how the underlying product platform of a product family can influence the successive generations and overall product development. To tackle this issue, product family evolution was discussed at the product platform level to understand the evolutionary properties of product families in more depth. Meyer (1997) defined strategies (i.e., horizontal leverage, vertical leverage, and beach-head leverage) for leveraging platforms across different market niches and/or price and performance levels in a market segmentation grid. Seepersad et al. (2002) developed a multi-objective decision model based on goal programming to create multiple product platforms for an evolving product portfolio, considering possible options for product platform extension from current product platforms to expand the initial product portfolio. Ohvanainen and Hietikko (2012) concluded that product platforms should evolve with incremental innovations to gain competitive advantages through their empirical findings on product platform changes.

Types and characteristics of a product architecture were also investigated to guide the evolution of a product

architecture with associated configurational changes (Shibata et al. 2005; Ulrich 1995). Ulrich (1995) addressed evolution motives in terms of product changes as: (1) product change through technological upgrades and component additions occurring over the lifecycle of a product, and (2) product change in a product line over successive generations. Shibata et al. (2005) addressed that the evolution of a product architecture follows the order of the integral architecture, the modular architecture, and the open architecture; however, the evolution direction may be reversed from the modular architecture to the integral architecture when a new technology is incorporated into an existing product system.

Earlier studies based on the above approaches conceptually describe product family evolution in strategic marketing and management to give managerial directions for product family and product platform design. The discussion regarding product family evolution deals with changes in product platforms and architectures along with technology and market changes. In line with the prior discussion on the conceptualization of product family evolution, more recent studies on product family evolution involve specific methodologies and frameworks to model evolving properties in product families and product platforms beyond comprehensive directions; a summary of these studies is provided in Sect. 2.3.

2.3 Product family evolution modeling

One of the critical factors that can directly affect the generation and degeneration of products in a product family is technological change. With technology improvement, new functions and components are added to previous platforms and architectures to upgrade existing product families. In this regard, product family evolution was modeled with technology changes to guide the evolution process of a product family. Xu et al. (2008) developed a framework to estimate the evolution of a product family architecture through technological innovations guided by the directed evolution of theory of inventive problem solving (TRIZ). Liu and Özer (2009) developed a decision framework to manage a generational product family under stochastic technological changes to identify the relationship among product replacement cost, product profitability, and technology evolution. Dynamic modeling for product family evolution was conducted by considering components in a product family as basic entities, which are inevitably added, removed, or updated in an evolution process. From this perspective, Shao et al. (2012) proposed the concept of a structure semantic unit to assign inherent structural and assembly information to each part component to model the evolution of components in a product family architecture.

Recent studies consider a product family as a complex network and deal with the evolutionary properties of a product family through the network topology. Based on the

scale-free network model (Barabási and Albert 1999), Liu et al. (2006, 2008) proposed a complex network model for a steam turbine product family to establish the underlying rules of structural relationships between parts in the product family. Similarly, Sui and Zhong (2008) theoretically examined the evolving rules of a part relation network for a product family based on the scale-free network model.

Given the summary on the extant works, the following research opportunities are identified. First of all, change characteristics in an actual product family structure over time have not been scrutinized in the previous studies. The recent research that attempts to model the process of product family evolution relies on theoretical modeling rather than in-depth discussion through an actual case study to observe a continuous stream of product family structure changes. Secondly, most studies in modeling product family evolution are based on network representation due to its simplicity and usefulness in analyzing the complex structure of a product family. In this regard, an inspiring aspect for product family evolution can be found in network science. Network science has been widely employed for various types of complex systems including social, biological, and technological networks to effectively identify underlying network properties and to build models representing their complex networks (Newman 2010). As the nature of real networks has been widely explored in various domains, investigating underlying dynamics in product design and development using network science theories and methods has been also initiated. Braha and Bar-Yam (2004a, 2007) investigated product development networks where design tasks and their relationships are represented as directed networks. They found that product development networks follow topological characteristics and network perturbation patterns commonly observed in other real networks. Braha (2016) investigated real-world large-scale product design and development networks and found that they have both small-world and scale-free network patterns. Based on these characteristics, Braha (2016) demonstrated that complex design networks are highly robust to uncontrolled errors at randomly selected design components but weak for failures at hub design components; direct changes in hub design components can also significantly improve the performance of the entire design system. Similarly, a network science approach can be useful to represent a product family and to analyze its evolutionary behavior in a product family structure since a product family can be also regarded as a complex network consisting of components and their interactions. Especially, evolving network properties (Albert and Barabási 2002), which have been studied in complex network areas, would serve as a foundation to identify structural properties in product family evolution. The fact that the static topology of a network cannot fully capture the dynamics of a complex system (Braha and Bar-Yam 2006, 2009) supports that a product family

should be analyzed as a time-series network corresponding to its historical changes to understand underlying properties in product family evolution.

3 Methodology

This section illustrates principal steps to analyze the evolving properties of a product family structure across generations and time periods through a network science approach. The proposed methodology can be an effective tool to analyze the characteristics of a product family structure since any product family having a complex structure can be represented in a simplified form of a network. Based on the network representation of a product family structure, underlying patterns and properties of the design structure can be easily extracted through network measures and network motifs.

There are four main analysis steps performed in this study as illustrated in Fig. 1: (1) representation of product family networks, (2) derivation of network properties, (3) identification of evolution patterns, and (4) analysis of topological robustness and structural complexity. A product family structure in each time period is represented as a network at the first step. In the next step, all networks constructed for different time periods are analyzed to identify the characteristics of topological changes in the product family structure network over time periods. Then, significant network motifs (sub-graphs) in the product family networks are extracted to find evolutionary patterns during the whole time span. Finally, topological robustness and structural complexity of the product family structure network in each time period are investigated to understand structural dynamics in product family design.

3.1 Representation of product family structure networks

The underlying structure forming a product family architecture needs to be properly converted into a network. Since a product family is a set of products rather than a single product, interrelations among components in individual products within a product family should be represented in a network. For the network representation of a product family structure, product platform and architecture information embedded in a product family is considered to reflect structural types of components as well as relationships among components.

Nodes in a product family structure network are defined as components which are used in the product variants of a product family. Components in a product family can be categorized into two types: common base and differentiation

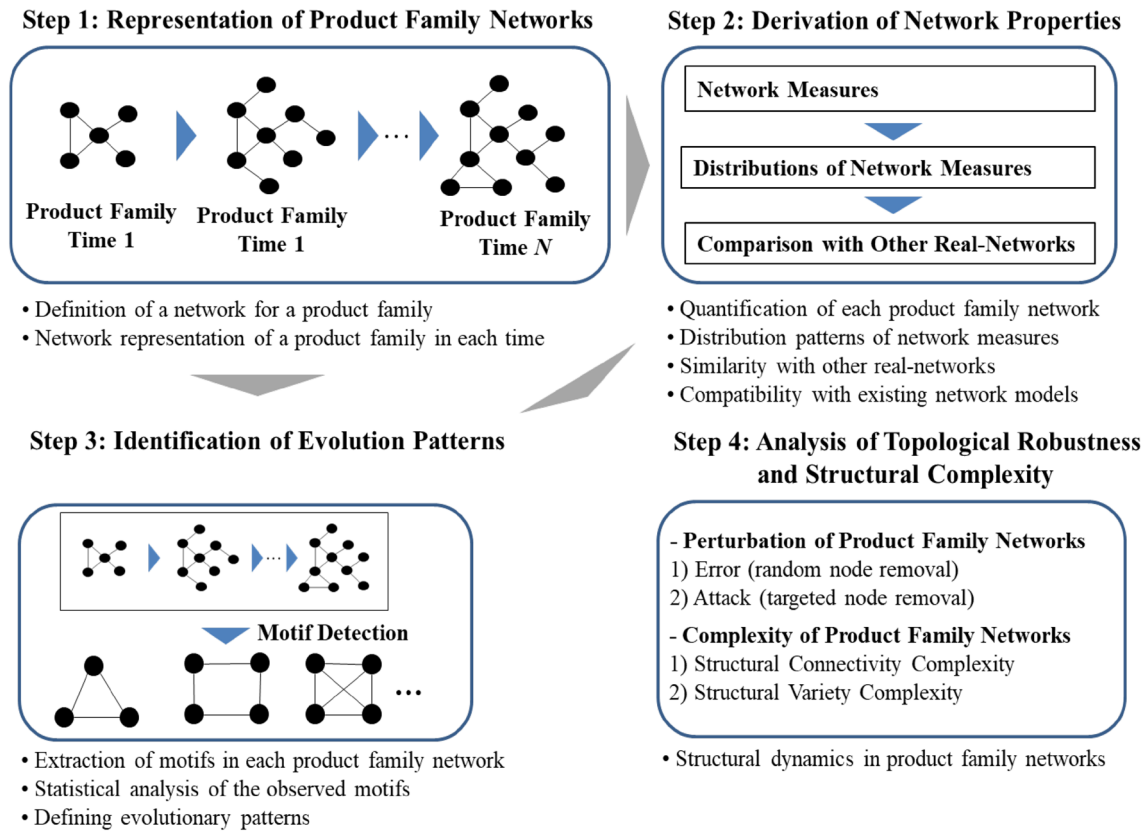


Fig. 1 Main steps and tasks for structural characterization of product family evolution

enabler (Du et al. 2001). The common base components indicate that the components are shared and provide the same functional features within a product family. On the other hand, the differential enabler components refer to the specific and unique components that provide product variants with different features. In accordance with the roles of components in a product family, nodes are categorized into common components and variant components to reflect the structural types of components in a network. Variant components are further divided into partly shared variant components and non-shared variant components.

Regular edges in a product family structure network indicate structural interactions to describe physical interdependencies between components in a product variant. This type of edges is defined as a structural connection

edge. Components connected with these edges indicate that they are physically connected/related in the same product variant. Additionally, inhibitory edges to describe negative structural relationships between the components of different product variants are considered when one component provides the same but a different level of functionality from another component. This type of edges is called a functional scale-leveraging edge. These edges indicate the scale variation and non-coexistence of components connected with the edges. Since functional scale-leveraging edges connect components having the same functionality but different performance, this type of edges typically describes the situation where a component in one product variant is upgraded or downgraded in another product variant; thus, components connected with this type of edges are not compatible to exist

Table 2 Edge types in a product family structure network

Type	Form	Description
Structural connection edge		A and B provide different functions, and they are connected to share the same interface.
Functional scale-leveraging inhibitory edge		A and B provide the same function but different performance. They cannot co-exist in the same product.

together in the same product. Functional scale-leveraging edges are additionally considered for the detection of network motifs to include the functional relationships in finding network patterns. Otherwise, structural connection edges are mainly taken into account for analysis. The edge types used in a product family structure network are illustrated in Table 2. Based on the above definitions of nodes and edges, a product family structure network is built by merging the individual networks of product variants.

A design structure matrix (DSM) (Eppinger et al. 1994) can be useful to build a product family structure network. A DSM is an effective tool to represent relationships between design elements such as components in a product and tasks in a design process (Eppinger et al. 1994; Steward 1981). A basic DSM has the same form and principle as an adjacency matrix in network theory for the matrix representation of a network; element (i, j) in an n by n matrix can have either 0 (i.e., no connection) or 1 (i.e., connection from node i to node j). The DSM method is useful to represent a network for a single product, but it cannot be directly used for a product family. As the way to construct a product family network, the following process is proposed to assist the creation of a product family structure network from individual product structure networks.

As the first step, DSMs for individual product family members and component types are identified. Let $PF = \{p_1, \dots, p_n\}$ be a set of products representing a product family and $C = \{c_1, \dots, c_m\}$ be a set of components appearing in PF. Let C_p ($C_p \in C$) denote a set of components used in a product $p \in PF$. Then, a DSM (\mathbf{D}) of a product p is:

$$D_{ij}(p) = \begin{cases} 1 & \text{if there is a structural connection edge from } i \text{ to } j \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where $i, j \in C_p$.

Let $C_{CC} \in C$ be a set of common components, $C_{VC} \in C$ be a set of variant components, $C_{VC(\text{shared})} \in C_{VC}$ be a set of variant components that are partly shared in a product family, and $C_{VC(\text{non-shared})} \in C_{VC}$ be a set of variant components that are completely not shared. Then, the type of components within a product family is determined by:

$$\begin{aligned} C_{CC} &= C_{p_1} \cap C_{p_2} \cap \dots \cap C_{p_n} \\ C_{VC} &= C - C_{CC} \\ C_{VC(\text{shared})} &= \{(C_{p_1} \cap C_{p_2}) \cup \dots \cup (C_{p_1} \cap C_{p_n}) \cup \dots \\ &\quad \cup (C_{p_{n-1}} \cap C_{p_n})\} - C_{CC} \\ C_{VC(\text{non-shared})} &= C_{VC} - C_{VC(\text{shared})} \end{aligned} \tag{2}$$

where $C_{p_n} \neq \emptyset, C_{p_1} \neq \dots \neq C_{p_n}$.

Next, a new DSM considering all variant components in a product family is generated to check relationships between variant components in all product family members. Elements

with respect to physical connections in the DSM have the same values ($= 1$) as those in the individual product DSMs. For negative structural relationships between variant components, if any pair of two variant components in different products is in a functional scale-leveraging relationship, then -1 is assigned to the element in the DSM. This step is described as follows:

$$D_{ij}(VC) = \begin{cases} 1 & \text{if there is a structural connection edge from } i \text{ to } j \\ -1 & \text{if there is a scale leveraging inhibitory edge from } i \text{ to } j \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where $i, j \in C_{VC}$.

At the final stage, a DSM to represent a product family structure network is created by merging the DSMs generated in the previous steps. The construction of the DSM for a product family follows:

- An empty matrix (\mathbf{D}) for all components appearing in a product family is generated to make a DSM for a product family $D_{ij}(PF)$.
- The same elements in individual $\mathbf{D}(p_n)$ are incorporated by the OR logic to fill in the same elements of the product family $D_{ij}(PF)$. $\mathbf{D}(VC)$ is also merged into $\mathbf{D}(PF)$ through the same way with the OR logic.

The above procedure can be also formulated as follows. Let $DSM = \{\mathbf{D}(p_1), \mathbf{D}(p_2), \dots, \mathbf{D}(p_n)\}$ be the DSMs of product family members and $\mathbf{D}(VC)$ denote a DSM between variant components. Then, the DSM of the product family $\mathbf{D}(PF)$ can be constructed by

$$D_{ij}(PF) = D_{i_{p_1}j_{p_1}}(p_1) \cup D_{i_{p_2}j_{p_2}}(p_2) \cup \dots \cup D_{i_{p_n}j_{p_n}}(p_n) \cup D_{i_{VC}j_{VC}}(VC) \tag{4}$$

where, $i_{p_n}, j_{p_n} \in C_{p_n}$ and $i_{VC}, j_{VC} \in C_{VC}$.

Figure 2 illustrates an example to summarize the DSM process to construct a product family structure network.

3.2 Derivation of network properties

After building a product family network in each time period, the properties of the network are analyzed by network measures. Objective information about a network structure can be obtained by network measures since they quantify various topological characteristics of a network. The main network measures that are commonly used to characterize networks are listed in Table 3 (Albert and Barabási 2002; Cui et al. 2010; Newman 2010). In the context of relationships among product development tasks, these measures were introduced and employed for complex design task networks to effectively analyze their topological properties (Braha and Bar-Yam 2004a, b, 2007).

In real networks, there are certain properties shared by different types of networks regardless of their origins and

Fig. 2 Representation of a product family structure network based on DSM

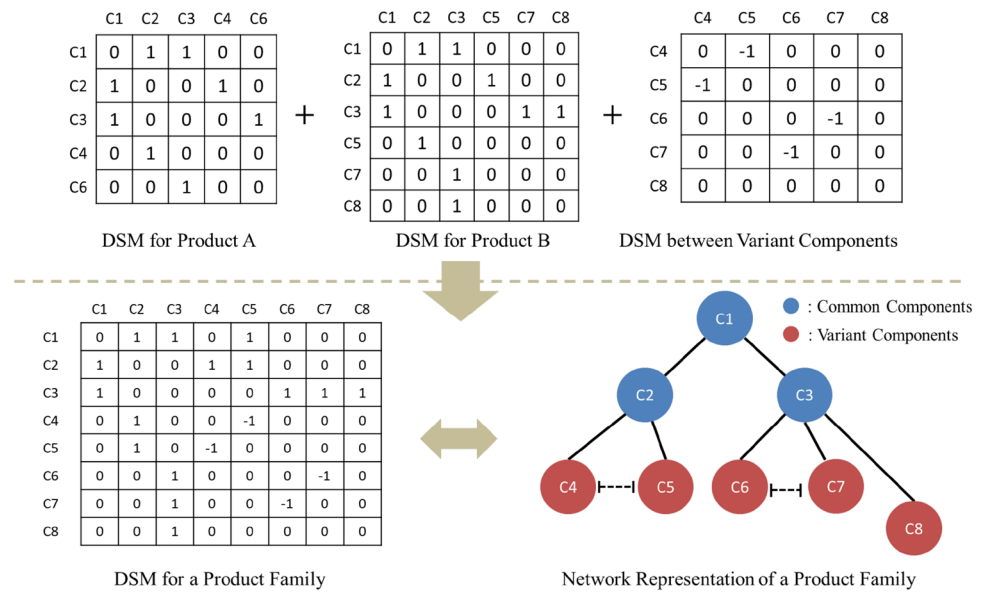


Table 3 Major network measures (Albert and Barabási 2002; Cui et al. 2010; Newman 2010)

Parameter	Description	Equation (for undirected networks)
Node degree	The number of links connecting the node to its neighbors	$k_i = \sum_{j=1}^n A_{ij}, A_{ij} = \begin{cases} 1, & \text{node } i \text{ and } j \text{ are connected} \\ 0, & \text{otherwise} \end{cases}$ (Average node degree: $\langle k \rangle = \frac{1}{n} \sum_{i=1}^n k_i$, $n = \text{the number of nodes}$)
Degree distribution	The probability distribution of the degree of each node in the network	$P(k) = \frac{n_k}{n}, n_k = \text{the number of nodes with } k \text{ degree}$
Average path length	The average value of shortest distance between any pair of nodes in the network	$\langle l \rangle = \frac{1}{n(n-1)} \sum_{i \neq j} d_{ij}$ $d_{ij} = \text{the shortest distance between node } i \text{ and } j$
Clustering coefficient	The local transitivity in the network	$C_i = \frac{2n_i}{k_i(k_i-1)}$ $n_i = \text{the number of edges among the nodes adjacent to } i$ (Average clustering coefficient: $C = \frac{1}{n} \sum_{i=1}^n C_i$ (for $i; k_i > 1$))

functions (Albert and Barabási 2000, 2002). The first common property is that the degree distribution of numerous large-scale networks follows a power-law function, which is shown in Eq. 5. This kind of networks is described as scale-free networks (Barabási and Albert 1999). The power-law degree distribution of a scale-free network indicates that a few nodes are likely to be linked with many other nodes, playing the role of hubs; and many nodes have a few links to other nodes. The second common property is that the average distance between any pair of nodes logarithmically scales with the number of nodes, following the average path length of a random network in Eq. (6) (Albert and Barabási 2002). The last common property is that the clustering coefficient of real networks is larger than that of equivalent random networks and does not depend on the network size, deviated from the clustering coefficient of a random network as shown in see Eq. (7) (Albert and Barabási 2002).

The last two common properties are typically observed in small-world networks (Albert and Barabási 2002; Barabási 2016), where they have many nodes with a specific node degree value.

$$P(k) \sim k^{-r} \tag{5}$$

where $P(k)$ = the distribution of node degree, r = power-law exponent.

$$\langle l \rangle \approx \langle l_{\text{random}} \rangle \tag{6}$$

where $\langle l_{\text{random}} \rangle = \ln(n)/\ln(\langle k \rangle)$.

$$C/\langle k \rangle \neq 1/n; \quad C \gg C_{\text{random}} \tag{7}$$

where $C_{\text{random}} = \langle k \rangle/n$.

Based on the network measures in Table 3 and the above common properties in real networks, a product family structure network in each time period is investigated to

characterize structural properties in the evolution process of a product family.

3.3 Identification of evolution patterns

Significant network motifs extracted in a product family structure network for each time period are identified in this step. Network motifs are defined as certain patterns in a network significantly recurring more frequently than those in equivalent random networks (Shen-Orr et al. 2002). Network motifs indicate specific rules governing their original network (Milo et al. 2002). Common network motifs are found in various real networks (Yeang et al. 2012) (see Fig. 3).

Evolution patterns are demonstrated by statistically analyzing frequencies of network motifs in a product family structure network at each time period. For this purpose, the frequency of each network motif obtained in each product family structure network is compared to that in its equivalent randomized network. Three-node and four-node motifs are extracted in each product family structure network through FANMOD (Wernicke and Rasche 2006), a network motif detection tool using the RAND-ESU algorithm (Wernicke 2005). All the node and edge types illustrated in Sect. 3.1 are considered for motif detection. Then, 1000 equivalent random networks are generated to identify the significance of detected motifs; statistically significant motifs that are more frequently observed in each network than in its equivalent random networks are extracted. The following measures are considered to check the statistical significance of motif occurrence (Wernicke and Rasche 2006): (1) Z score is the original motif frequency minus the average random motif frequency divided by the standard deviation of random motif frequency and (2) p value is the number of random networks, where a motif occurs more often than in its original network, divided by the total number of random networks. Z score greater than 3 and p value less than 0.05 in an original network are considered for screening detected motifs.

3.4 Analysis of topological robustness and structural complexity

3.4.1 Topological robustness

Many empirical results show that various complex systems have inherent tolerance against errors, typically represented by random node and/or edge removal in a network (Albert et al. 2000; Barabási 2016). For example, the overall

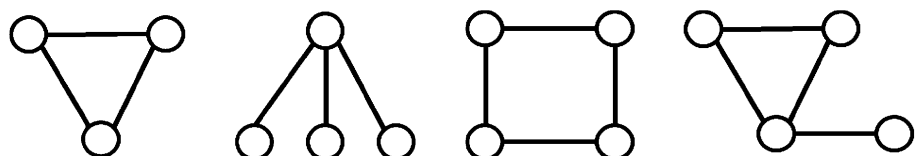
topology of the yeast protein interaction network is not significantly affected by the removal of randomly selected yeast proteins (Jeong et al. 2001). The connectivity loss of the North America power grid network is tolerant for random transmission substation removal (Albert et al. 2004). The modeled military supply chain network shows robustness in the average path length and the diameter for random node removal (Thadakamaila et al. 2004). Communication networks such as the World-Wide Web and the Internet (Albert et al. 2000) and social networks (Centola 2008; Perc 2009) also show a great degree of error tolerance. However, complex networks commonly show weakness in the targeted attacks of critical nodes in connectivity (e.g., hubs), typically represented by the selective removal of highly connected nodes (Albert et al. 2000, 2004; Centola 2008; Jeong et al. 2001; Perc 2009; Thadakamaila et al. 2004). The error robustness and the attack vulnerability of complex networks result from their scale-free network topology that is characterized with an inhomogeneous connectivity distribution (Albert et al. 2000). Many nodes with a few links in a scale-free network can have much higher probability to be selected under random node removal without significantly affecting the overall network structure. On the other hand, the removal of hub nodes in a scale-free network can significantly damage the overall network structure.

The topological resilience of a product family structure network is examined in this step. Random errors and targeted attacks in the product family network of each time period are, respectively, addressed by node removal through random selection and targeted selection in decreasing order of node degree. Then, the relative size of the largest cluster (S), expressed as a fraction of the total network size, is tracked as a function of the fraction of removed nodes (f).

3.4.2 Structural complexity

Complexity is a common concept that has been discussed in many scientific and engineering domains such as computer science, biology, and physics (Mitchell 2009). However, there is no one specific definition and framework to describe complexity in systems since each domain deals with diverse systems where complexity can be distinctively interpreted from various perspectives (Barabási 2009; Park and Okudan Kremer 2015). Despite the lack of widely agreed conceptualization of complexity, complexity is commonly defined to measure the density of interactions and the amount of information a system

Fig. 3 Significant network motifs in real networks (Yeang et al. 2012)



has (Chu 2011). Indeed, general methods to quantify complexity have been developed in a wide variety of areas through benefits of network representation for systems to understand their dynamics (Bonchev and Buck 2005).

The efforts to conceptualize and quantify complexity have also been expended in the field of product and engineering design. Many researchers have applied the concept of complexity to product design to understand the embedded uncertainty in product design and the impact of complexity on operational performance (Ameri et al. 2008; Park and Okudan Kremer 2015). From the structural perspective, complexity in product-related systems is associated with the amount of uncertainty in structural configuration affected by the number, diversity, and interrelationships of elements (Closs et al. 2008; Jacobs 2013; Orfi et al. 2011; Park and Okudan Kremer 2015). Thus, an information-theoretic approach (Shannon 2001) has been widely used to conceptualize complexity in product and operational systems due to its effective interpretation for linkage between uncertainty and complexity, such that a system increasing in uncertainty becomes more complex and thereby results in more information required to describe the system's state (Braha and Maimon 1998; Park and Okudan Kremer 2015; Sivadasan et al. 2006).

Herein, two complexity measures for a product family structure network are introduced based on information theory: (1) complexity of structural connectivity, and (2) complexity of structural variety. Connectivity is a basic characteristic to describe the structural state of a network and represents relationships among nodes. The dynamics between connectivity and uncertainty in a network can be explained by the expected value of the information content of node degree (Bonchev 2003). Based on the entropy expression of node connectivity, the complexity of structural connectivity is expressed as a logarithmic function of a structural connectivity distribution from the product and product family design context (see Eq. 8):

$$C_{sc} = - \sum_{i=1}^n \frac{k_i}{k} \log_2 \frac{k_i}{k} \quad (8)$$

where k_i is the number of connections of component i in a product or a product family, and k is the sum of k_i .

The complexity of structural connectivity is defined as a measure of the expected amount of uncertainty in realizing the structural connectivity of a product or product family structure. Given no structural knowledge and data of a new component, the probability for a new component to be attached to an existing component can be estimated depending on the number of structural connections of the existing component. Then, it is most uncertain for a new component to be connected to an existing set of components if each existing component has equal connectivity. This results in the maximum uncertainty (or information) of the design structure to describe its connectivity state since it is

the most unpredictable. Consequently, this situation makes the connectivity of the design structure the most complex, where the complexity increases with the number of components. Networks shaping with the form of an n -clique ($k_i/k = (n-1)/n(n-1) = 1/n$) or an n -cycle ($k_i/k = 2/2n = 1/n$) lead to the most complex structure in connectivity.

Under the condition that there is no isolated component in the design structure, on the other hand, the expected amount of uncertainty in structural connectivity becomes minimum if one specific component dominates connectivity in the design structure. The design structure with one centralized component presents the least uncertainty to estimate an existing component to which a new component is likely to be connected. Thus, the least information is required to describe the connectivity of the design structure, and this leads to the least complexity. Networks forming an n -star structure ($k_{i(\text{centralized})}/k = (n-1)/2(n-1) = 1/2$ and $k_{i(\text{non-centralized})}/k = 1/2(n-1)$) involve the least complex situation in connectivity.

The variety of a design structure can be represented by: (1) intra-variety associated with part variety (i.e., how many various components are used in a design structure), and (2) inter-variety associated with product variety (i.e., how much each component is shared in different product variants) (Park and Okudan Kremer 2015). The variety sources bring uncertainty to a design structure in that a design structure can be difficult to be realized if it should handle many unique elements that make the structure complex. The complexity of structural variety is defined as a function of uncertainty to describe the total amount of information embedded in the variety sources of a design structure (Park and Okudan Kremer 2015). It is expressed as the total information content of commonality for each component i as seen in Eq. (9):

$$C_{sv} = - \sum_{i=1}^n \log_2 \frac{p_i}{p} \quad (9)$$

where p_i is the number of product variants that share component i , and p is the number of product variants.

The measure for the complexity of structural variety in Eq. (9) is based on the commonality of product variants (p_i/p), representing the degree of sharing a component within a product family. Since a product family is designed to effectively handle product variety based on a common platform, commonality plays a vital role in determining the design structure of a product family (Johnson and Kirchain 2010; Roy et al. 2011; Simpson 2004). A product family designed with too low commonality results in very high component variety, causing an increase in the uncertainty of realizing the design structure in manufacturing. Thus, each product variant or product family with low commonality needs more information required to describe the state of the design structure than that with high commonality. Equation (9) captures this uncertainty caused from commonality

within a product family by quantifying the information content of commonality for each component. The complexity of structural variety in Eq. (9) also reflects structural variety caused by the size of a design structure (i.e., the number of product variants and the number of components). With the same p , C_{sv} increases if p increases. Also, the more components a product variant or a product family has, the higher complexity it has in structural variety. Thus, C_{sv} reflects both intra- and inter-structural variety. Based on the above two complexity measures, structural complexity in a product family structure network in each time period is analyzed.

4 Illustration of a smartphone case study

In this research, a smartphone product family was analyzed to characterize the evolutionary properties of a product family structure. Due to the short life cycle and highly competitive markets of smartphones, their historical information can be effective to observe the evolution process. All the smartphone generation models of a major provider in the smartphone industry, including the first generation released from June in 2007 to the latest generation produced until September in 2016, were investigated for the case study. First, main models and their variants in the smartphone product line (i.e., a total of 14 main models and 81 product variants) were searched on the web, and their released and discontinued dates were collected to compile the company's product timeline from its first generation to the latest. Then, each time period of product family evolution for the company's product line was determined as each time period when there is no change in the product line. In total, 14 time periods were identified in the timeline of the smartphone product family; this shows that the smartphone product family evolved 14 times.

Then, the product structures of each main model and its variants were examined to construct a product family structure for each time period. The main components including major parts, modules, and chipsets were considered for each product structure. Information about these components was collected from teardown information in <http://www.electronicproducts.com/WhatsInside.aspx>, <http://www.ifixit.com>, <http://www.techinsights.com>, and other available web sources. Each model is structured with similar types of key assemblies as a product platform: (1) display, (2) home button, (3) battery, (4) rear case, (5) camera, (6) loudspeaker, (7) dock/lightning connector, (8) logic board, (9) power button, (10) silent button, (11) volume button, (12) antenna, and (13) vibrator (see Fig. 4). The associated modules and parts of the assemblies and their structural connections are varied depending on the smartphone models. The product architecture of each model was considered with the least unit of

identifiable components (i.e., modules, parts, and chipsets) in the above assemblies during their disassembly processes.

The logic board of each model has various chipsets and these chipsets play significant roles in realizing smartphone functions. Major chipsets relevant to the following functions were also considered to illustrate a more detailed product architecture for each smartphone model: (1) application processor with DRAM, (2) wireless memory (SRAM), (3) flash memory, (4) cellular modem (radio frequency baseband), (5) radio frequency transceiver, (6) bluetooth/Wi-Fi, (7) GPS (global positioning system), (8) NFC (near field communication), (9) accelerometer, (10) gyroscope, (11) audio codec, (12) electronic compass, (13) barometer, (14) processor power management, (15) radio frequency power management, and (16) touch screen controller.

Structural relations among the components in each smartphone model were determined from its disassembly steps, provided in iFixit (2016). Structurally connected components within a smartphone product are defined as components that are structurally connected with each other to realize direct mechanical and/or functional relations for the case study. The following types were considered to determine whether components are structurally connected: (1) directly connected by screws and adhesives, (2) directly connected by sockets, connectors, and holes, and (3) direct structural design relation (shape and form/fit) to a specific component. Chipsets were considered as individually placed on a bare logic board, and internal relationships between them were disregarded for the simplification of the product structure. Components associated with the above three types were linked with structural connection edges in each product family structure network. In addition, functional scale-leveraging edges in each network were drawn for components that provide main functional features with different performance within each product family. Individual product



Fig. 4 Example of smartphone assemblies (iFixit 2015)

family networks were then analyzed through the proposed methodology.

5 Results and discussion

The product family structure network of the smartphone models in each time period of the product family evolution is shown in Fig. 15 in Appendix. The size of each node is scaled with the degree of each node. Blue, purple, and red nodes, respectively, represent completely shared common, partly shared variant components, and completely non-shared variant components. Grey solid edges and green dotted edges, respectively, describe structural connection edges and functional scale-leveraging edges. The topological properties of the constructed network in each time period are summarized in Table 4.

5.1 Topological properties

5.1.1 Network size

The size of the product family structure network significantly grows over time (see Table 4; Fig. 5). From the first period to the latest period, the number of nodes and the number of edges increase 5.3 times and 6.6 times, respectively. N in the smartphone product family structure network seems to increase in proportion to E . The Pearson's correlation between N and E ($=0.997$) supports a strong positive linear relationship between them. This can be interpreted that the increase in the number of components in the smartphone

product family does not cause an overgrowth in the number of structural connections.

The linear growth in structural connections may result from the property of commonality of a product family. Since the smartphone product family in each time period shares components all or partly available in product variants, structural connections among those components are also shared with different product variants. Shared components and their structural connections within a smartphone product family form the structural commonality of the product family and lead to decreasing uncertainty in realizing the product family.

The number of shared (completely and partly) components in the smartphone product family structure network more exponentially increases over time than the total number of components (see Figs. 5, 6). The fitted exponential function, with 95% of confidence, of the number of shared components for $t=0, \dots, 111$ th (month) is $N_{\text{shared}} = 18.59e^{0.02353t}$ ($R^2: 90.75\%$) and that of the number of total components for the same interval is $N_{\text{total}} = 50.42e^{0.01504t}$ ($R^2: 91.82\%$).

Since the number of shared components cannot exceed the number of total components, the upper bound of the fitted exponential function of the number of shared components over time becomes an intersection point between these plots in Fig. 6. This intersection point can be named as the efficient product platform design point at which all components in a product family are shared (partly or completely) within the product family when the current product platform strategy is maintained. The number of components at the efficient product platform design point is considered as the expected number of components in a product family where the platform of the product family does not generate

Table 4 Properties of smartphone product family structure networks

#	Time-period		# of nodes (N) [CC: # of CCs, VC: # of total VCs (# of partly shared VCs)]	# of edges (E)	Average node degree ($\langle k \rangle$)	Average path length ($\langle l \rangle$)/% of reachable pairs	Average clustering coefficient (C)
	Starting time	Ending time					
1	June 2007	September 2007	46 [CC: 44, VC: 2 (0)]	88	3.83	2.62/100%	0.55
2	September 2007	February 2008	45 [CC: 45, VC: 0 (0)]	87	3.87	2.61/100%	0.55
3	February 2008	July 2008	46 [CC: 44, VC: 2 (0)]	88	3.83	2.62/100%	0.55
4	July 2008	June 2009	45 [CC: 43, VC: 2(0)]	81	3.60	2.31/100%	0.64
5	June 2009	June 2010	60 [CC: 29, VC: 31 (14)]	120	4.00	2.43/100%	0.50
6	June 2010	February 2011	93 [CC: 1, VC: 92(47)]	176	3.78	3.45/100%	0.64
7	February 2011	October 2011	117 [CC: 1, VC: 116 (73)]	267	4.56	3.42/100%	0.52
8	October 2011	September 2012	153 [CC: 1, VC: 152(63)]	360	4.71	3.75/100%	0.50
9	September 2012	September 2013	160 [CC: 2, VC: 158 (77)]	378	4.70	3.56/100%	0.51
10	September 2013	March 2014	151 [CC: 0, VC: 151(99)]	315	4.17	2.85/54.97%	0.59
11	March 2014	September 2014	152 [CC: 0, VC: 152 (99)]	316	4.16	2.86/55.12%	0.59
12	September 2014	September 2015	199 [CC: 3, VC: 196(152)]	454	4.56	3.93/100%	0.55
13	September 2015	March 2016	251 [CC: 0, VC: 251 (249)]	580	4.62	4.43/100%	0.56
14	March 2016	September 2016	242 [CC: 1, VC: 241 (239)]	581	4.80	4.00/100%	0.56

Fig. 5 Structural changes in smartphone product family structure network

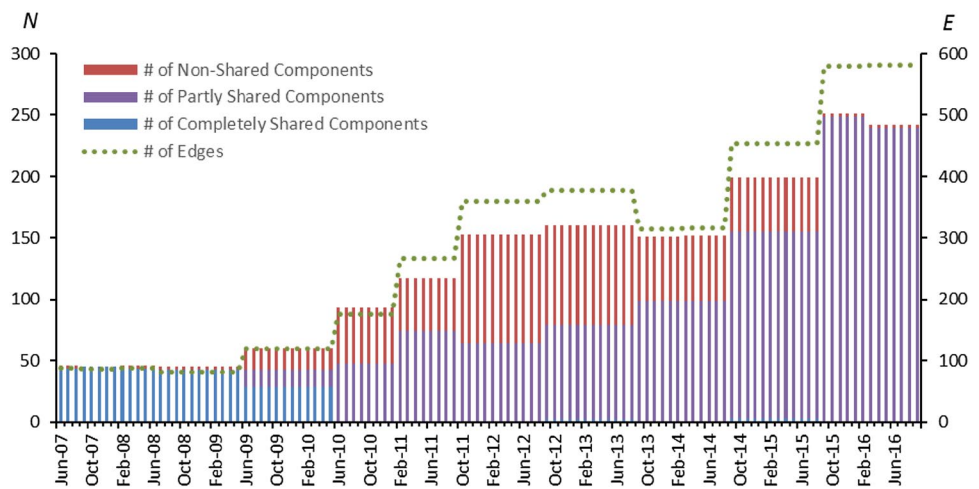
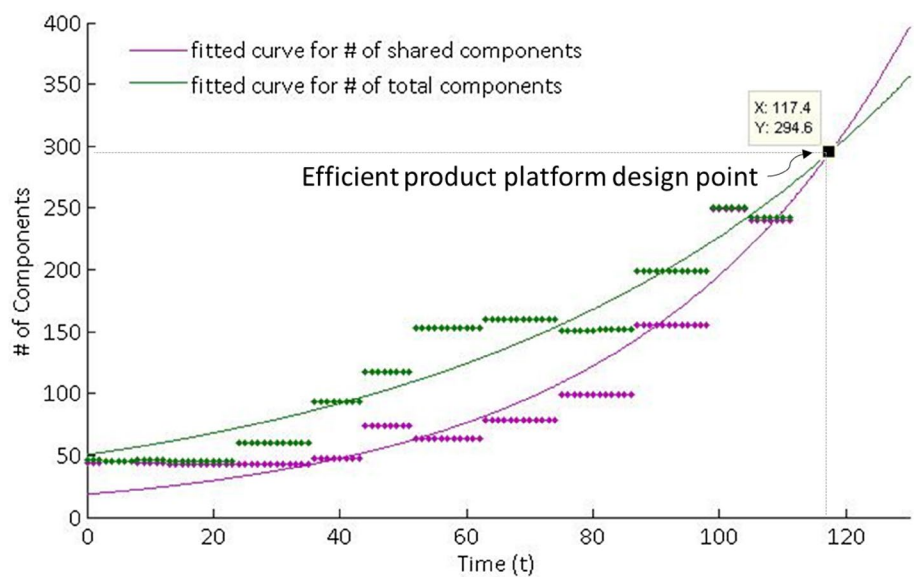


Fig. 6 Fitted exponential growth of smartphone product family structure network



non-shared variant components. The projected time at the efficient product platform design point shows the expected timing of efficient product family occurrence. In this smartphone product family case, the efficient product platform design point (t, N_{shared}) is $(117.4, 294.6)$, which indicates that the product family has approximately 294 shared components at 117th month (≈ 10 years) after the appearance of the first product family in the market.

The efficient product platform design point can be regarded as a signal of maturity for the current product platform design. After the efficient product platform design point, all components in a product family may be still shared following the plot for the number of total components, or the number of shared components may decrease if disruptive changes through strategical, technological, product innovation happen in the current platform design.

5.1.2 Node degree

The average node degree of each product family structure network has a range between 3.6 and 4.8 (see Table 4); the components of each smartphone product family approximately have four or five structural connections to other components. $\langle k \rangle$ tends to increase for newer product family structure networks, and this indicates that components in newer product families have more number of structural connections.

The degree distribution of each product family structure network shows that there are many nodes with a few links and a few nodes with many links (see Fig. 7). These highly connected nodes in each network serve as hubs. The hub nodes commonly identified in each smartphone product family network are the bare logic boards and rear cases/fames of the associated smartphone models in each product family network. A bare logic board is integrated with many chipsets

and connected with other components to realize their functions. Similarly, a rear case or frame holds many components to maintain their locations for structural durability. The logic board and rear case assemblies are the key modules of a smartphone product platform, and they form common interfaces shared across different smartphone product variants. Thus, the commonality and importance of the key design components on the product platform render the identified scale-free property in the product family structure networks.

The degree distributions shown in Fig. 7 were fitted with a power-law function. Table 5 and Fig. 16 in Appendix, respectively, shows the exponent of the fitted power-law function and the fitted power-law plot for each degree distribution. They have a range of $2 < \gamma < 3$, showing the scale-free network regime (Barabási 2016). Although most of the fitted power-law functions are statistically significant under $p > 0.01$, the product family structure networks for recent models (i.e., product family 12, 13, and 14) do not have statistically significant power-law degree distributions. Components having a few edges in the individual product networks of these product families tend to have additional edges in their product family structure networks due to many partly shared components in the product families. For this reason, the degree distributions for product families 12, 13, and 14 slightly deviate from power-law degree distributions.

5.1.3 Average path length

Table 4 shows that the product family structure networks have average path lengths in a range of $2 < \langle l \rangle < 5$. It is noted that the $\langle l \rangle$ values of product families 10 and 11 are for reachable pairs of nodes in the networks since each of the networks is separated due to the existence of a smartphone model that does not share all the components with other models in the product family. The average path length of each product family structure network seems to be close to that of its equivalent random network (see Fig. 8). This indicates that the product family structure networks also behave as small-world networks.

The $\langle l \rangle$ values in newer product family structure networks become greater than the expected $\langle l \rangle$ values in their equivalent random networks. A plausible explanation of the increase in $\langle l \rangle$ for the recent smartphone product families is structural separation among different generation products in the product families. A new generation smartphone model tends to have structural variations from previous generation models due to functional improvement and technological innovation. Although smartphone models at the same generation have high commonality, the structural separation between different generation models results in weak connection (or no connection) among network clusters within a smartphone product family network. This phenomenon happens since the 6th smartphone product family, showing a few or no common

Fig. 7 Degree distribution of smartphone product family structure networks

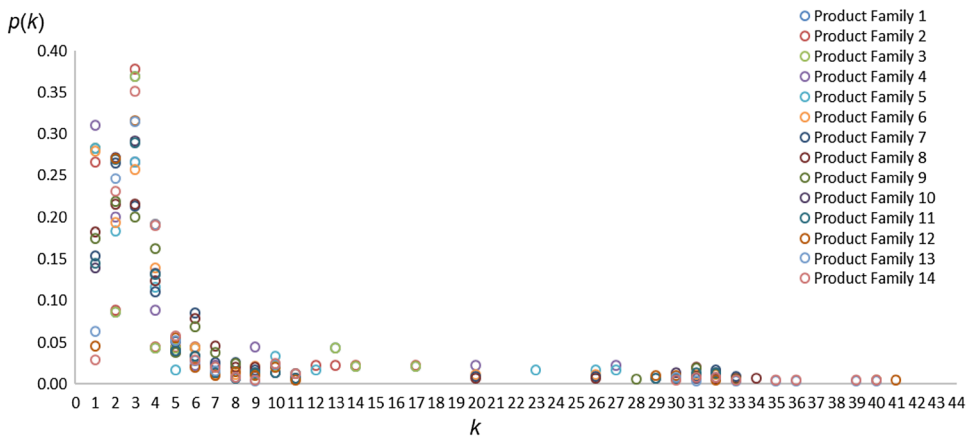
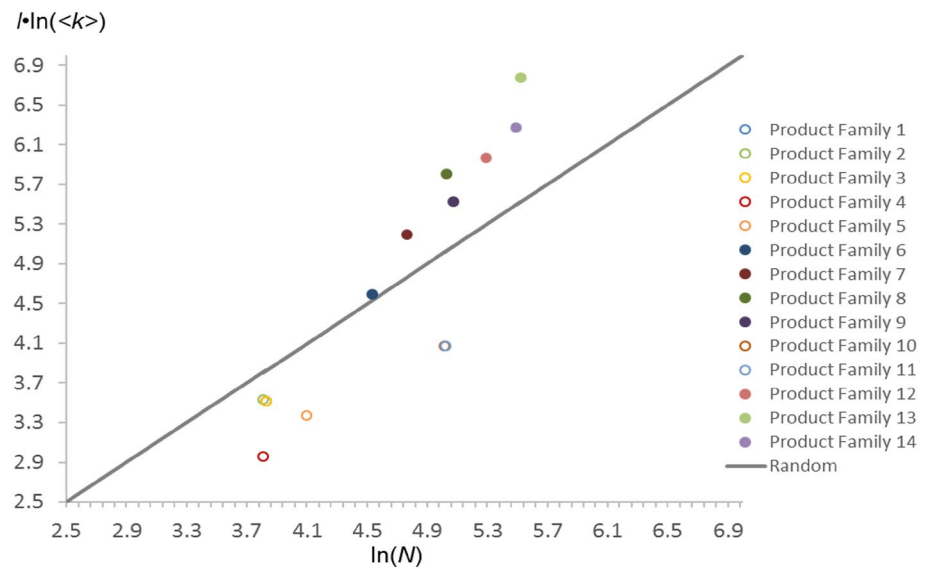


Table 5 Power-law exponents for smartphone product family structure networks

Product family #	1	2	3	4	5	6	7
γ	2.79	2.80	2.79	2.70	2.63	2.79	2.47
p value	0.05*	0.04*	0.13*	0.25*	0.18*	0.02*	0.49*
Product family #	8	9	10	11	12	13	14
γ	2.44	2.45	2.82	2.81	2.87	2.84	2.87
p value	0.422*	0.242*	0.091*	0.067*	0.002	0	0.002

* p value > 0.01

Fig. 8 Average path length of smartphone product family structure networks and their random networks



components in the product family networks (see Fig. 15 in Appendix); consequently, $\langle l \rangle$ becomes greater in the smartphone product family networks. The increase in $\langle l \rangle$ of the smartphone product family network as various old and new generation products are included in the smartphone product family represents an increase in the structural scale of the product family design over time.

5.1.4 Clustering coefficient

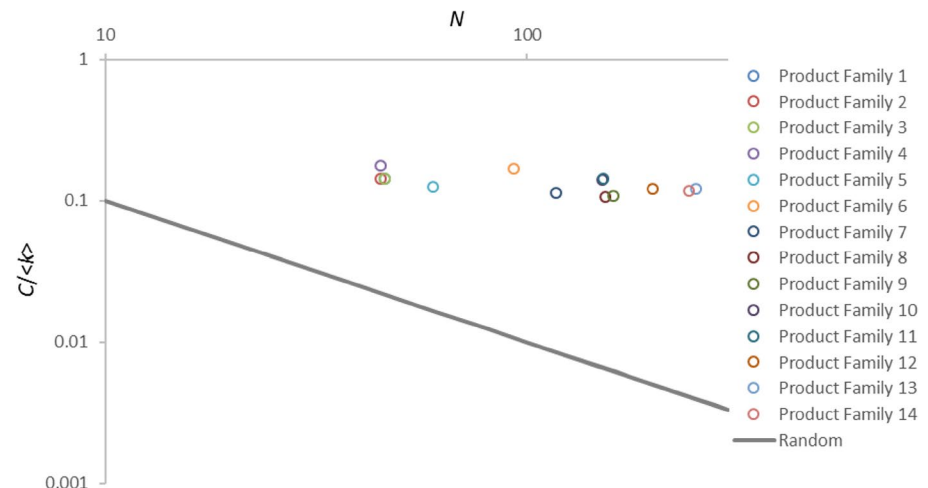
The C value of each product family structure network is greater than that of each equivalent random network, and the ratio $C/\langle k \rangle$ becomes independent with N (see Fig. 9). The results are consistent with the property of small-world networks. A low correlation (≈ -0.12) between N and C also supports that the clustering coefficient of the product family structure networks does not depend on the size of the networks.

The average clustering coefficient C of each smartphone product family structure network is maintained with an above moderate degree (>0.5). C tends to vary in the product family networks, showing resilience to product family design changes. For example, once C becomes greater (or lower) in a current product family network, then C becomes lower (or greater) in its next product family network. In this way, C tends to converge to the mean value of C (≈ 0.56) in the product family networks (see Fig. 10). This resilience of C in the smartphone product family network over time periods shows that the smartphone product family evolves to stability in the local connectedness of the structure.

5.2 Network motifs

All the node and edge types illustrated in Sect. 3.1 were considered for motif detection. Figure 11 shows the motifs obtained from the product family structure network in each

Fig. 9 Clustering coefficient of smartphone product family structure networks and their random networks



time period. For 3-node motifs, 19 different types of motifs are identified across the product family structure networks. The common formations of the motifs are clique and tree. In the early time periods (i.e., time periods 1–4) of the product family, completely connected three common components (i.e., Δ) and a centralized common component connected with a common component and a non-shared variant component (i.e., Δ) are typically observed. In the later time periods of the product family, frequently occurring motifs across the periods form a clique among partly shared components (i.e., Δ), a clique among non-shared components (i.e., Δ), and a centralized partly shared component connected with a partly shared component and a common component (i.e., Δ). After

the time period 5 of the product family, significant motifs including functional performance change between components are also frequently observed (i.e., Δ , Δ , and Δ).

For 4-node motifs, a higher variety of motifs are identified in the product family structure networks; 32 motifs are detected across the time periods. A two-triangle for common components (i.e., Δ), a triangle with a dangling edge for common components (i.e., Δ), and a three-star with three common components and one non-shared variant component (i.e., Δ) commonly appear in the early time periods. During the later periods, there are five motifs frequently occurring across the periods: (1) a triangle with a dangling edge for partly shared components (i.e., Δ), (2) a series of partly shared components where a starting (or an ending) node is functionally scaled from or to a non-shared component (i.e., Δ), (3) a series of non-shared components where a starting (or an ending) node is functionally scaled from or to a partly shared component (i.e., Δ), (4) a triangle with a dangling edge for non-shared components (i.e., Δ), and (5) a three-star with three partly shared components and one common component (i.e., Δ). In the recent time periods (time periods 12–14), trees of partly shared components without or with a functional scaling edge are distinctively observed (i.e., Δ , Δ , Δ , and Δ).

The identified motifs show that there is no one common motif over time periods but a group of shared motifs common in a subset of time periods. There are several time periods (e.g., time periods 5, 8, and 9) when specific motifs exclusively occur in the product family network. These results indicate that the product family network in each time

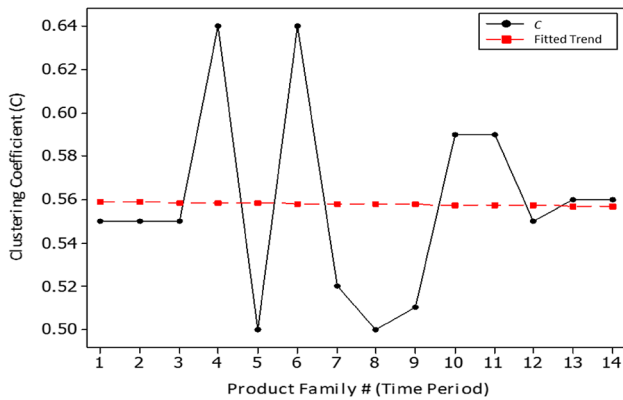


Fig. 10 Changes in clustering coefficient of smartphone product family structure network

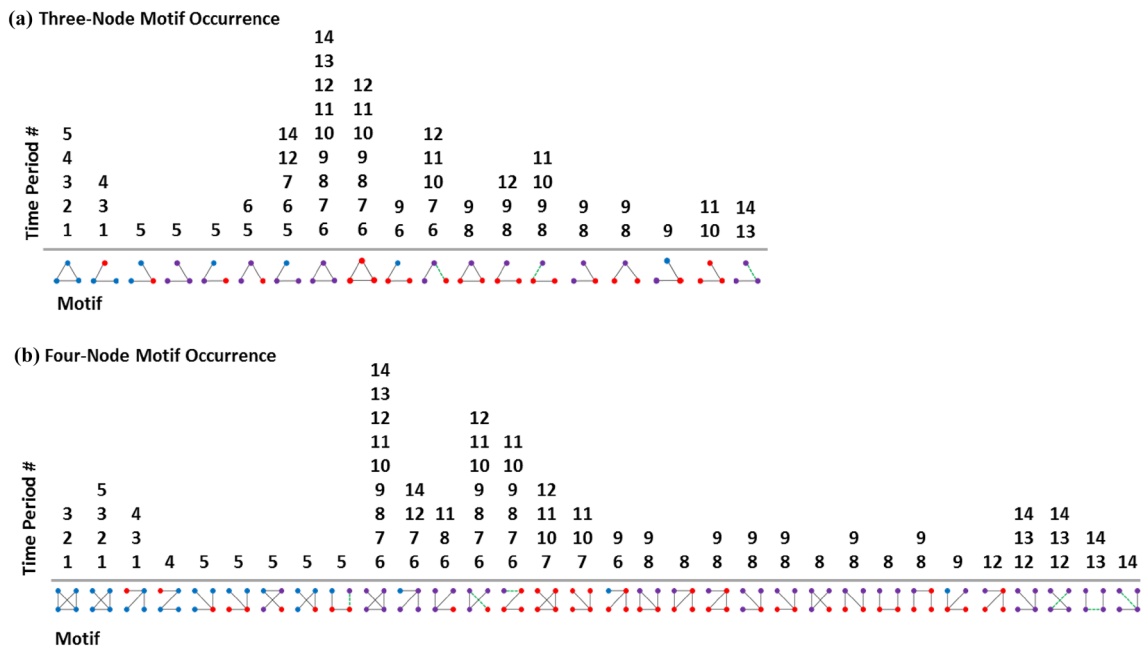


Fig. 11 Significant motifs and their occurring time periods

period is characterized with common and specific building blocks.

5.3 Topological robustness and structural complexity

The topological resilience of real networks can be considered in product family design. Figure 12 shows network fragmentation under random errors and targeted attacks on the smartphone product family in each time period.

The results show that the product family structure network in each time period is robust to random errors. The product family structure in each time period is maintained without significant fragmentation under random errors except for the case that hub nodes are eliminated by chance. On the other hand, each product family structure is rapidly fragmented as f increases when the most connected nodes (hubs) are removed. The attack vulnerability of each product family structure network seems to have a threshold of the node removal rate at which the network significantly breaks apart (see Fig. 12b). This threshold tends to decrease in the product family networks in later time periods, indicating that the networks are rapidly destroyed by the removal of a few hub nodes. For all the product family structure networks, the targeted removal of 10% of the total nodes critically impacts on the networks, leading to more than 50% of significant disruption of the networks.

The example of the network fragmentation under targeted node removal for the latest product family (i.e., product family 14) and the latest product in the product family is visualized in Fig. 13. In the product network, single nodes break off due to the removal of the largest hub ($r = 1$). Then, the network is fragmented into small clusters and single nodes when the two largest hubs ($r = 2$) are eliminated. However, the product family network is still robust to the removal of the most connected node, resulting in no impact on fragmentation (see Fig. 13). The main cluster breaks into another

large cluster and a few isolated single nodes when the second most connected node is removed. This is a different pattern from the single product network, where the removal of its two most connected nodes generates many isolated single nodes and small network pieces.

The survivability of the product family structure shows the following implications. If a company only holds a single product in a market, the company could not survive in the marketplace when the most connected component in the product design structure has a critical problem since there is no choice except for fixing the problem. However, a company holding a well-designed product family could avoid the risk of the overall product family failure in a market when a significant defect on the most connected component in the product family structure occurs. This is because components that substitute the vital component may exist in product variants. Similarly, a product family that is rigidly structured with partly shared components may have buffer product variants; product variants not affected by the failure on the most vital component still can survive in the marketplace. Thus, the network topology of a product family design structure supports why nowadays companies release various product variants in marketplaces to survive.

The behavior analysis of the product family structure networks, under topological perturbations, explains that structural robustness in product and product family design can increase by (1) adding protective components (e.g., brackets and covers) for vital components that have many structure connections and (2) achieving a sufficient degree of commonality. In addition, the topological tolerance of real networks provides inspiration for a design structure based on a scale-free network structure to be robust. Thus, a scale-free structure with well-protection of vital components and a sufficient degree of commonality would be important for robust product and product family design.

Figure 14 shows that both the complexity of structural connectivity and the complexity of structural variety

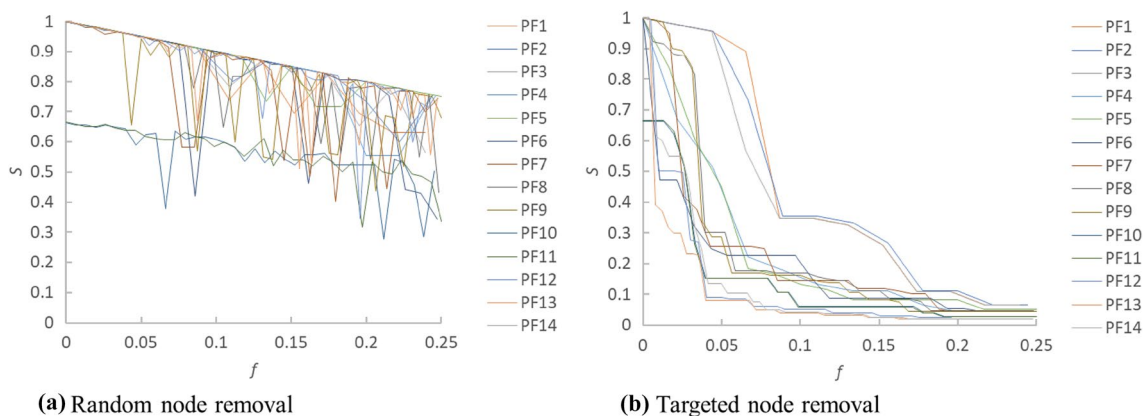


Fig. 12 Network fragmentation under random errors and targeted attacks

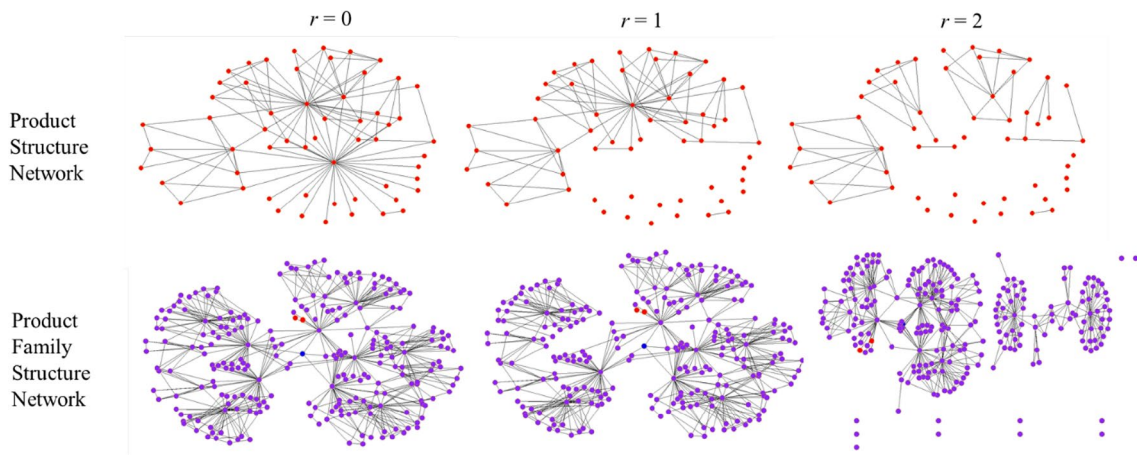


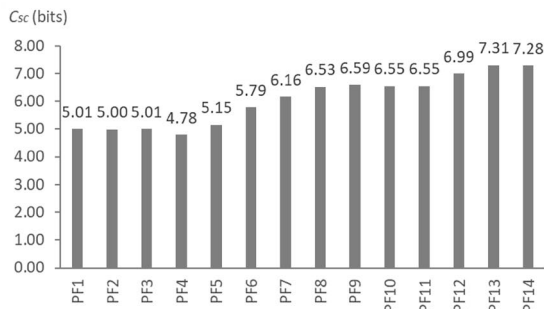
Fig. 13 Visualized network fragmentation under targeted node removal

increase in the product family over time. The results show that the design structure of the smartphone product family evolves with an increase in complexity. The complexity of structural connectivity and the complexity of structural variety seem to be closely related to each other in the smartphone product family case; the Pearson’s correlation coefficient between these two measures for the smartphone product family in each time period is 0.97, showing a strong positive linear relationship. This relationship implies that complexity caused from structural variety in a design structure can lead to complexity in connectivity; adding more structural variety into a product family can affect the design structure by increasing uncertainty in structural connectivity.

Structural complexity affects the robustness of a design structure. The complexity of structural connectivity (C_{sc}) depends on the topology of a network; the complexity measure becomes low if a network is highly centralized by a few nodes. This property provides a possible explanation of the topological behavior under errors and attacks found in empirical and theoretical scale-free networks in the literature. Scale-free

networks have a few highly connected nodes and many nodes with a few links. This topological characteristic of scale-free networks gives the networks less uncertainty in structural connectivity and thereby less complexity than other networks characterized with a specific node degree. Thus, scale-free networks show structural tolerance to random (uncertain) network perturbations and vulnerability to targeted (certain) network perturbations since their connectivity is more certain than that in other network types (i.e., random and small-world networks). The fact that the complexity of structural variety is closely related to the complexity of structural connectivity also implies that structural variety indirectly impacts on a network’s behavior under errors and attacks. The above implications show that the topological robustness of a design structure for a product family is influenced by structural connectivity and variety inherent in the design structure. A robust product family design structure with a scale-free network structure with a sufficient level of commonality would decrease structural complexity and thus increase structural robustness.

(a) Complexity of Structural Connectivity



(b) Complexity of Structural Variety

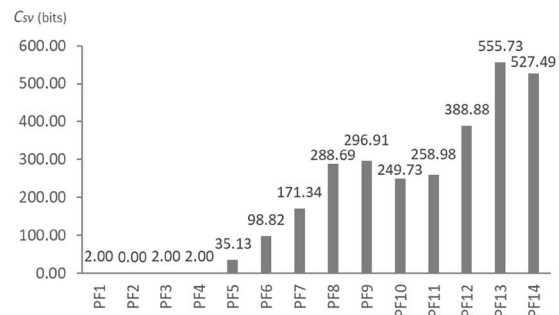


Fig. 14 Structural complexity in a smartphone product family in each time period

6 Conclusions and discussion

This study investigated structural properties and patterns in a product family across generations and time periods. A network science approach was employed to effectively capture the underlying dynamics of an evolving product family structure since a product family can be viewed as a complex network involving many interactions between physical components. This research explored network properties and patterns in a time-series of product family structure networks through their network topologies. Historical smartphone models released by a major company in the smartphone industry were used as a case study.

The results demonstrate that the product family structure network in each time period follows the universality of topological properties observed in other real networks. The network properties and significant motifs of the individual product family structure networks suggest that the product family product structure evolves as a scale-free and small-world network with an increase in the commonality of the product family. The product family network changes with common motifs shared in each specific range of time periods and specific motifs only appearing in specific time periods. The size of the smartphone product family structure network exponentially grows over time with an exponential increase in the number of shared components faster than that in the number of total components; the product platform of the smartphone product family becomes efficient with an increase in commonality over time. The resilience in the clustering coefficient of the smartphone product family network also shows that the smartphone product family evolves with stability in local connectedness.

The product family structure networks over a time horizon have both scale-free and small-world properties. First of all, the fitted power-law functions show that most of the product family structure networks are globally viewed as following the scale-free property. This seems to result from a few major components centralizing structural relationships with many other components in the product family structure; the logic board and rear case in the individual smartphone products of each product family play significant roles in design functionality and durability through structural connections with many other components. This commonality of a few key components across the product variants of the product family makes each product family network have a scale-free topology. The fact that the scale-free property is also observed in other software and hardware design networks (e.g., open source software networks and electronic circuit networks) (Braha and Bar-Yam 2007) shows that the scale-free property of the product family structure networks is not a special feature but

a common phenomenon in engineering design. However, several product family structure networks for the recent smartphone models do not have statistically significant scale-free topologies. This deviation from the scale-free property can be attributed to high commonality in the recent product families, resulting in additional structural connections for many components with a few connections.

From the local view of network topology, the product family structure networks also show the small-world property; the average path length is close to that in an equivalent random network and the clustering coefficient is much higher than that of an equivalent random network. The small-world property of the product family structure networks can be related to the modular structure of the product family. The historical smartphone products have similar types of key assemblies. Each assembly has a modular structure, which structurally interconnects associated components within a module. In addition, each product family has one or multiple product platforms with common components across product variants. These product characteristics lead to internally clustered physical components and weak connections among the component clusters, which are in turn characterized as the small-world property.

The topological characteristics identified from the historical smartphone product families are similar to those from the product development task networks of Braha and Bar-Yam's studies (Braha and Bar-Yam 2004a, b, 2007). Focusing on organizational relationships among product development tasks, they discussed that a design task network and its related product structure network may have similar topological properties. In this regard, the findings from this research support the close potential linkage between product architectures and task organizations. Product platform-based design drives a product development organization to have multi-functional teams where they can effectively handle flexibility and complexity of a product development process for the product platform (Muffatto and Roveda 2000). The core design components of a product platform shared across different product variants may require various information and knowledge for many other design tasks to be shared and connected to the core product development teams, where they handle central tasks for the key components of a product platform, to reduce inefficiency in a product development process for product variety (Calabrese 1997; Muffatto and Roveda 2000; Sosa et al. 2004). This can explain the scale-free topological pattern of both product family design and task networks centralized by a few key components and tasks, respectively. The small-world property of the product family structure networks resembling a modular design structure may be also observed in a relevant task network

for the product families in that different component design tasks can be grouped based on a modular product architecture (Gershenson et al. 2003; Ulrich 1995). The functional significance of the small-world property of task networks (Braha and Bar-Yam 2004a, b, 2007) and the importance of alignment between product architectures and organizational structures (Gokpinar et al. 2010; Sosa et al. 2004) support that design benefits from a modular design structure can be connected to organizational benefits. Thus, the small-world property observed in both the product family networks and the task networks show that they inherently evolve to improve efficiency in designs and tasks by decreasing dependencies between component groups in the product architectures and the organizational structures, respectively.

The coexistence of the scale-free and small-world properties in the historical product family structure networks implies the distinct nature of product family design. Product family design has both modularity and commonality as key concepts to effectively realize product variety in a product family (Jiao et al. 2007). Modularity focuses on decomposing components and functions into independent logical units, whereas commonality focuses on using similar components and functions across product variants (Jiao et al. 2007; Pirmoradi and Wang 2011). The modular structure of a product family enables product variants to be easily derived by adding, replacing, and/or removing one or multiple functional modules (Du et al. 2001). Since modularity is characterized as minimal interactions between modules and high interactions within a module (Ulrich 1995), a product family design inherently involves the small-world property in its evolution process. On the other hand, commonality is an important concept that can reduce complexity embedded in a product family design (Park and Okudan Kremer 2015). Thus, a common product platform for product family design at an appropriate degree of commonality is essential to achieve cost-effectiveness in realizing product variety (Nayak et al. 2002). Since a product family structure is based on core functional components shared on a product platform (Meyer and Utterback 1993; Simpson et al. 2006), commonality in a product family can lead to a scale-free topology centralized by core components. Consequently, product family design as a mixture of modularity and commonality renders a product family structure network that will evolve as a network having both the small-world and scale-free network properties. Indeed, the trade-off between product variety from modularity and cost savings from commonality (Pirmoradi and Wang 2011) shapes a product family structure network to have both high dependencies on core components and high interactions within each assembly. This is similar to the co-occurrence of the small-world and scale-free properties in task networks

due to a trade-off between task dependency and performance improvement (Braha and Bar-Yam 2004a, b, 2007).

Focusing on the universality of the network topology, topological robustness and complexity that were addressed for other theoretical and empirical networks in the literature can be interpreted in the context of product family design through this study. Product families having the scale-free network topologies can be tolerant to random failures and vulnerable to targeted attacks. In a product family, the overall design structure can be more robust than the structure of a single product under a certain targeted attack due to commonality in the product family structure. Moreover, complexity as conceptualized with its measures based on information theory (i.e., the complexity of structural connectivity and the complexity of structural variety) explicitly shows that a design structure based on a scale-free network topology and commonality can have topological robustness since it has less uncertainty in structural connectivity and variety. The fact that product development task networks have similar perturbation patterns (Braha and Bar-Yam 2007) may imply that both a product family architecture and its corresponding organizational structure should be co-optimized to maximize operational performance. Agile alignment between core components of a product platform and related central tasks may help improve the well-established design and development processes.

The primary intellectual contribution of this research is to lay a foundation for significant advancement opportunities in product design decision making through understanding evolution mechanisms of a product family. For example, the successive future configuration of a product family network over time would be able to be modeled and simulated with the identified evolution characteristics in this study and the relevant theoretical network models that reflect the universality of topological properties. In this regard, this study will provide various potential research opportunities and insights to extend conventional engineering design approaches into design science approaches. The methods and results from this research will serve as a foundation for the analysis of product family evolution, and this will encourage researchers to explore distinct evolution properties of a product family according to industry types, market trends, and technology changes. The methodology proposed, which interconnects network science and product design, will establish a new interdisciplinary approach to be a useful basis for the design of other management systems (e.g., supply chains and social networks).

Despite the benefits and contributions of the research, there are several issues that should be addressed in future work. First, more extensive product types should be analyzed to generalize the findings from this research. Second,

the co-evolution of market, technological, and manufacturing systems in line with product family evolution should be analyzed to characterize their underlying interactions. Third, the product family evolution should be addressed with an analytical model that can simulate the processes of product family evolution under dynamic external changes. It also should be further validated with other real case studies.

Product family evolution consistent with the universality of topological characteristics in other real networks shows that a product family evolves in the same nature with other systems. Indeed, product design should be understood as a continuum of the evolution paradigm of our world. This

perspective would give us new insights for development of next generation products and product families. The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgements This work was supported by Incheon National University (International Cooperative) Research Grant in 2018.

Appendix

See Figs. 15 and 16.

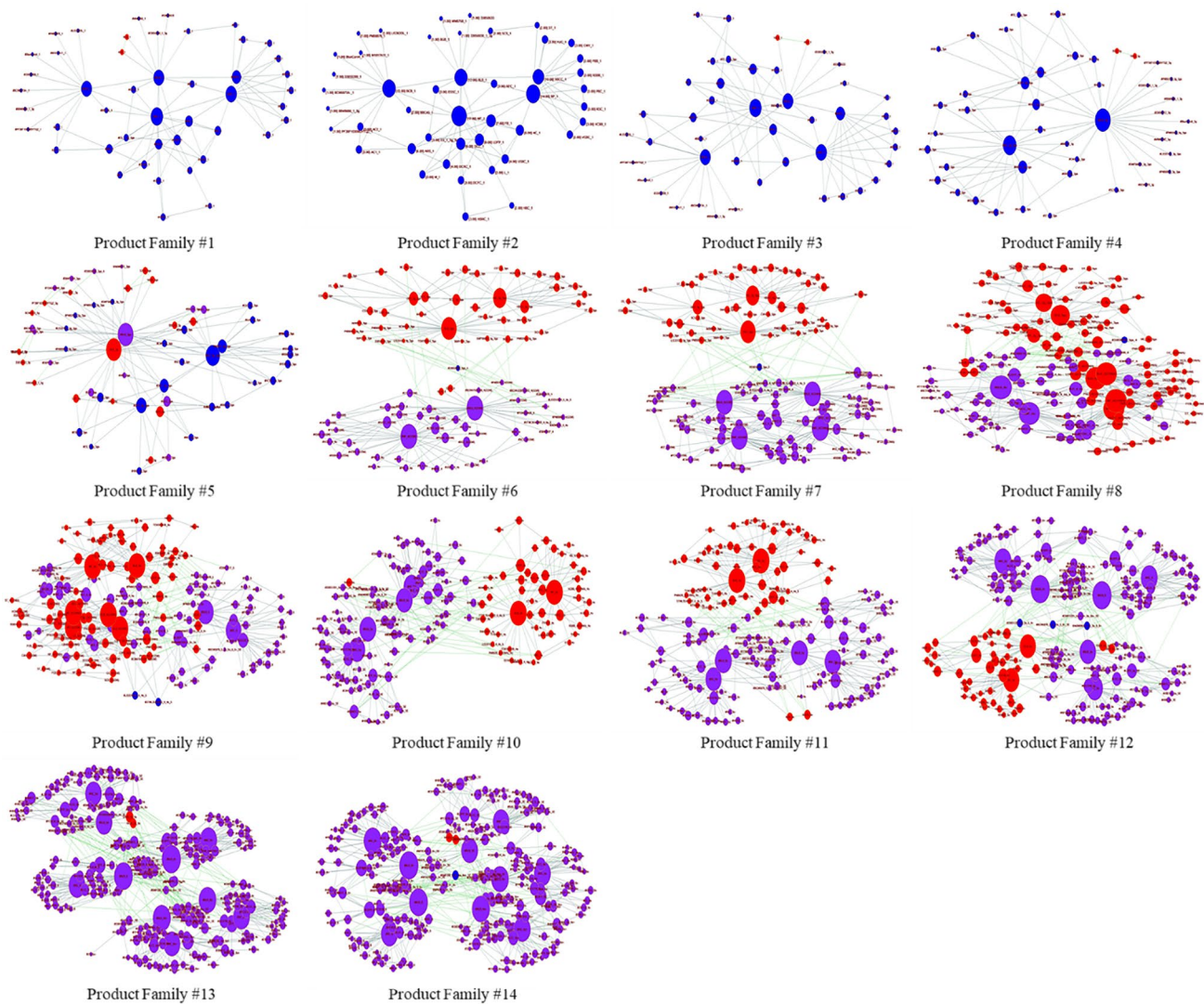


Fig. 15 Smartphone product family architecture networks

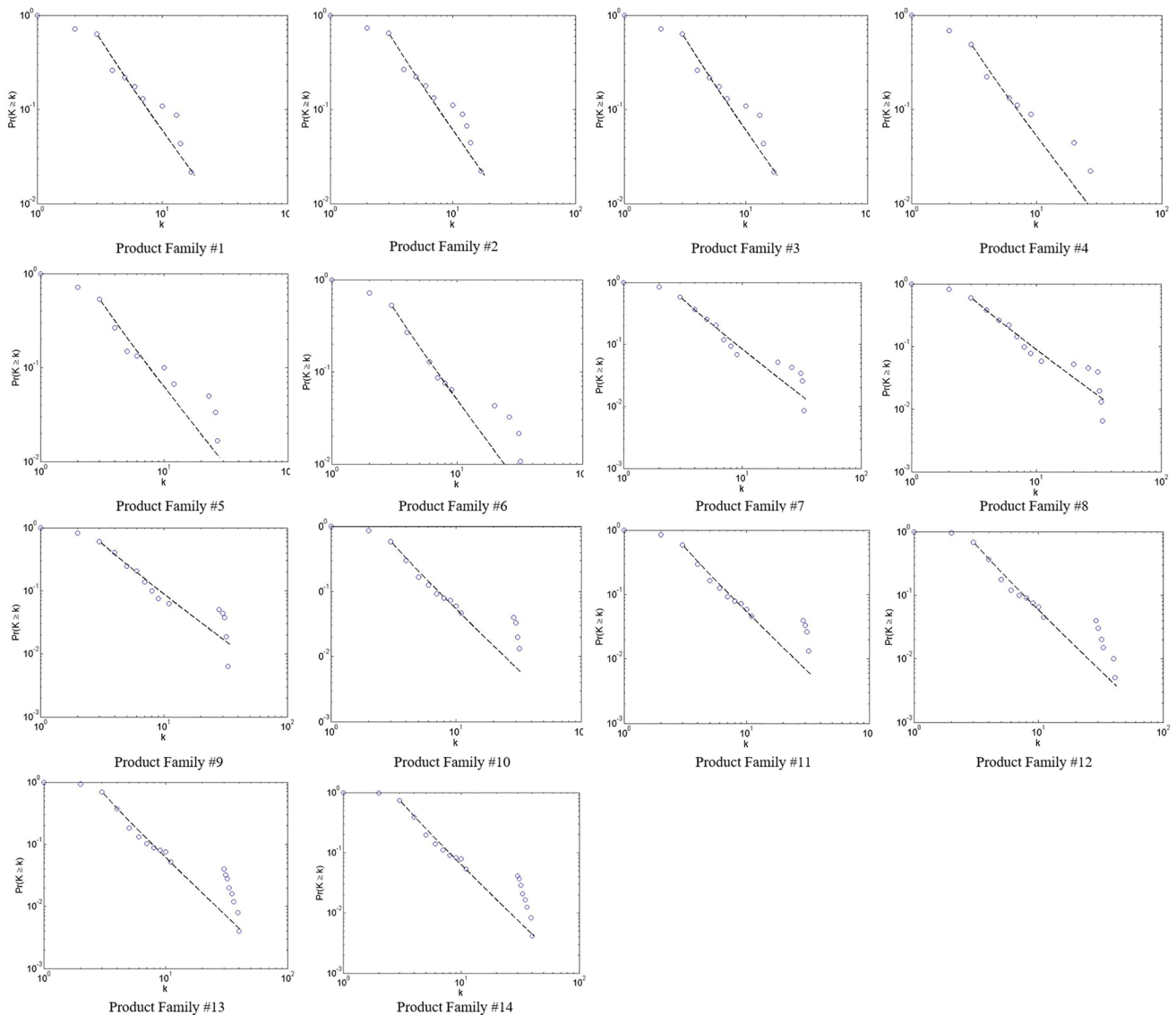


Fig. 16 Fitted power-law plot for smartphone product family structure networks

References

- Albert R, Barabási A-L (2000) Topology of evolving networks: local events and universality. *Phys Rev Lett* 85:5234–5237
- Albert R, Barabási A-L (2002) Statistical mechanics of complex networks. *Rev Mod Phys* 74:47–97
- Albert R, Jeong H, Barabási A-L (2000) Error and attack tolerance of complex networks. *Nature* 406:378–382
- Albert R, Albert I, Nakarado GL (2004) Structural vulnerability of the North American power grid. *Phys Rev E* 69:025103
- Alizon F, Khadke K, Thevenot HJ, Gershenson JK, Marion TJ, Shooter SB, Simpson TW (2007) Frameworks for product family design and development. *Concurr Eng* 15:187–199
- Ameri F, Summers JD, Mocko GM, Porter M (2008) Engineering design complexity: an investigation of methods and measures. *Res Eng Des* 19:161–179
- Anderson DM, Pine JB, Pinei BJ (1997) Agile Product development for mass customization: how to develop and deliver products for mass customization, niche markets, JIT, Build-to-order, and flexible manufacturing. McGraw-Hill, New York
- Barabási A-L (2009) Scale-free networks: a decade and beyond. *Science* 325:412–413
- Barabási A-L (2016) The scale-free property. In: Barabási A-L (ed) *Network science*. Cambridge University Press, Cambridge
- Barabási A-L, Albert R (1999) Emergence of scaling in random networks. *Science* 286:509–512
- Bonchev D (2003) Shannon's information and complexity. In: Bonchev D, Rouvray DH (eds) *Complexity in chemistry: introduction and fundamentals*. Taylor & Francis, London, pp 157–187
- Bonchev D, Buck GA (2005) Quantitative measures of network complexity. In: Bonchev D, Rouvray DH (eds) *Complexity in chemistry, biology, and ecology*. Springer, Boston, pp 191–235
- Braha D (2016) The complexity of design networks: structure and dynamics. In: Cash P, Stankovic T, Štorga M (eds) *Experimental design research*. Springer, Switzerland, pp 129–151

- Braha D, Bar-Yam Y (2004a) Information flow structure in large-scale product development organizational networks. *J Inf Technol* 19:244–253
- Braha D, Bar-Yam Y (2004b) Topology of large-scale engineering problem-solving networks. *Phys Rev E* 69:016113
- Braha D, Bar-Yam Y (2006) From centrality to temporary fame: dynamic centrality in complex networks. *Complexity* 12:59–63
- Braha D, Bar-Yam Y (2007) The statistical mechanics of complex product development: empirical and analytical results. *Manag Sci* 53:1127–1145
- Braha D, Bar-Yam Y (2009) Time-dependent complex networks: dynamic centrality, dynamic motifs, and cycles of social interactions. In: Gross T, Sayama H (eds) *Adaptive networks*. Springer, Berlin, pp 39–50
- Braha D, Maimon O (1998) The measurement of a design structural and functional complexity. *IEEE Trans Syst Man Cybern Part A Syst Hum* 28:241–277
- Bryan A, Hu SJ, Koren Y (2013) Assembly system reconfiguration planning. *J Manuf Sci Eng* 135:041005
- Calabrese G (1997) Communication and co-operation in product development: a case study of a European car producer. *R&D Manag* 27:239–252
- Centola D (2008) Failure in complex social networks. *J Math Sociol* 33:64–68
- Chu D (2011) Complexity: against systems. *Theory Biosci* 130:229–245
- Closs DJ, Jacobs MA, Swink M, Webb GS (2008) Toward a theory of competencies for the management of product complexity: six case studies. *J Oper Manag* 26:590–610
- Cui L, Kumara ST, Albert R (2010) Complex networks: an engineering view. *IEEE Circuits Syst Mag* 10:10–25
- Du X, Jiao J, Tseng MM (2001) Architecture of product family: fundamentals and methodology. *Concurr Eng* 9:309–325
- Duray R, Ward PT, Milligan GW, Berry WL (2000) Approaches to mass customization: configurations and empirical validation. *J Oper Manag* 18:605–625
- ElMaraghy H, Schuh G, ElMaraghy W, Piller F, Schönsleben P, Tseng M, Bernard A (2013) Product variety management. *CIRP Ann Manuf Technol* 62:629–652
- Eppinger SD, Whitney DE, Smith RP, Gebala DA (1994) A model-based method for organizing tasks in product development. *Res Eng Des* 6:1–13
- Ferguson S, Kasprzak E, Lewis K (2009) Designing a family of reconfigurable vehicles using multilevel multidisciplinary design optimization. *Struct Multidiscip Optim* 39:171–186
- Gershenson JK, Prasad GJ, Zhang Y (2003) Product modularity: definitions and benefits. *J Eng Des* 14:295–313
- Giovannini A, Aubry A, Panetto H, El Haouzi H, Canciglieri O, Pierrel L (2016) Knowledge representation, retrieval and reuse for product family design: an anti-logicist approach. *Comput Ind Eng* 101:391–402
- Gokpinar B, Hopp WJ, Irvani SM (2010) The impact of misalignment of organizational structure and product architecture on quality in complex product development. *Manag Sci* 56:468–484
- Halman JIM, Hofer AP, Van Vuuren W (2003) Platform-driven development of product families: linking theory with practice. *J Prod Innov Manag* 20:149–162
- Hou L, Wu Y-m, Lai R-s, Huang SH (2013) Evaluation and driver analysis in product family evolution. *Int J Adv Manuf Technol* 69:863–874
- Huang X, Kristal MM, Schroeder RG (2008) Linking learning and effective process implementation to mass customization capability. *J Oper Manag* 26:714–729
- iFixit (2015) iPhone 6 s Plus teardown. <http://www.ifixit.com/Teardown/iPhone+6s+Plus+Teardown/48171>. Accessed 8 Dec 2016
- iFixit (2016) Apple iPhone repair. <http://www.ifixit.com/Device/iPhone>. Accessed 8 Dec 2016
- Jacobs MA (2013) Complexity: toward an empirical measure. *Technovation* 33:111–118
- Jeong H, Mason SP, Barabási A-L, Oltvai ZN (2001) Lethality and centrality in protein networks. *Nature* 411:41–42
- Jiao J, Tseng MM (1999) A methodology of developing product family architecture for mass customization. *J Intell Manuf* 10:3–20
- Jiao J, Simpson TW, Siddique Z (2007) Product family design and platform-based product development: a state-of-the-art review. *J Intell Manuf* 18:5–29
- Johnson MD, Kirchain R (2010) Developing and assessing commonality metrics for product families: a process-based cost-modeling approach. *IEEE Trans Eng Manag* 57:634–648
- Liu H, Özer Ö (2009) Managing a product family under stochastic technological changes. *Int J Prod Econ* 122:567–580
- Liu F, Qi G, Yang Q (2006) Forecast method for used number of parts and components based on complex network. *Front Mech Eng China* 1:479–484
- Liu F, Song S, Deng X (2008) Research on evolving rule of part relation network of product family. *Int J Mater Struct Integr* 1:334–343
- Liu E, Hsiao S-W, Hsiao S-W (2014) A decision support system for product family design. *Inf Sci* 281:113–127
- Meyer MH (1997) Revitalize your product lines through continuous platform renewal. *Res Technol Manag* 40:17–28
- Meyer MH, Lehnerd AP (1997) *The power of product platforms: building value and cost leadership*. Free Press, New York
- Meyer MH, Utterback J (1993) The product family and the dynamics of core capability. *Sloan Manag Rev* 34:29–47
- Milo R, Shen-Orr S, Itzkovitz S, Kashtan N, Chklovskii D, Alon U (2002) Network motifs: simple building blocks of complex networks. *Science* 298:824–827
- Mitchell M (2009) *Complexity: a guided tour*. Oxford University Press, New York
- Muffatto M, Roveda M (2000) Developing product platforms: analysis of the development process. *Technovation* 20:617–630
- Nayak RU, Chen W, Simpson TW (2002) A variation-based method for product family design. *Eng Optim* 34:65–81
- Newman MEJ (2010) *Networks: an introduction*. Oxford University Press, New York
- Ohvanainen J, Hietikko E (2012) Building competitive advantage through platform-based product family thinking: case powerpacks. *J Ind Eng Manag* 5:180–197
- Ong SK, Xu QL, Nee AYC (2006) Design reuse methodology for product family design. *CIRP Ann Manuf Technol* 55:161–164
- Orfi N, Terpenney J, Sahin-Sariisik A (2011) Harnessing product complexity: step 1—establishing product complexity dimensions and indicators. *Eng Econ* 56:59–79
- Otto KN, Wood KL (1998) Product evolution: a reverse engineering and redesign methodology. *Res Eng Des* 10:226–243
- Park K, Okudan Kremer GE (2015) Assessment of static complexity in design and manufacturing of a product family and its impact on manufacturing performance. *Int J Prod Econ* 169:215–232
- Perc M (2009) Evolution of cooperation on scale-free networks subject to error and attack. *New J Phys* 11:033027
- Pirmoradi Z, Wang GG (2011) Recent advancements in product family design and platform-based product development: a literature review. In: *ASME 2011 international design engineering technical conferences and computers and information in engineering conference*, Washington, DC, August 28–31. pp ASME paper no. DETC2011-47959
- Rinne M (2004) Technology roadmaps: infrastructure for innovation. *Technol Forecast Soc Change* 71:67–80
- Roy R, Evans R, Low M, Williams D (2011) Addressing the impact of high levels of product variety on complexity in design

- and manufacture. *Proc Inst Mech Eng Part B J Eng Manuf* 225:1939–1950
- Salhieh SEM (2007) A methodology to redesign heterogeneous product portfolios as homogeneous product families. *Comput Aided Des* 39:1065–1074
- Salvador F, Forza C, Rungtusanatham M (2002) Modularity, product variety, production volume, and component sourcing: theorizing beyond generic prescriptions. *J Oper Manag* 20:549–575
- Sanderson S, Uzumeri M (1995) Managing product families: the case of the Sony Walkman. *Res Policy* 24:761–782
- Sawhney MS (1998) Leveraged high-variety strategies: from portfolio thinking to platform thinking. *J Acad Mark Sci* 26:54–61
- Seepersad CC, Mistree F, Allen JK (2002) A quantitative approach for designing multiple product platforms for an evolving portfolio of products. In: ASME 2002 international design engineering technical conferences and computers and information in engineering conference, Montreal, Canada, September 20–October 2. American Society of Mechanical Engineers, pp 579–592
- Shannon CE (2001) A mathematical theory of communication. *ACM SIGMOBILE Mob Comput Commun Rev* 5:3–55
- Shao WP, Hao YP, Zeng PF (2012) Study on product family architecture dynamic evolution based on product instances. *Adv Mater Res* 472:2612–2616
- Shen-Orr SS, Milo R, Mangan S, Alon U (2002) Network motifs in the transcriptional regulation network of *Escherichia coli*. *Nat Genet* 31:64–68
- Shibata T, Yano M, Kodama F (2005) Empirical analysis of evolution of product architecture: Fanuc numerical controllers from 1962 to 1997. *Res Policy* 34:13–31
- Shooter SB, Evans CM, Simpson TW (2007) Building a better ice scraper—a case in product platforms for the entrepreneur. *J Intell Manuf* 18:159–170
- Simpson TW (2004) Product platform design and customization: status and promise. *Artificial intelligence for engineering design. Anal Manufact* 18:3–20
- Simpson TW, Siddique Z, Jiao RJ (2006) Product platform and product family design: methods and applications. Springer Science & Business Media, New York
- Sivadasan S, Efstathiou J, Calinescu A, Huatucu LH (2006) Advances on measuring the operational complexity of supplier–customer systems. *Eur J Oper Res* 171:208–226
- Sköld M, Karlsson C (2013) Stratifying the development of product platforms: requirements for resources, organization, and management styles. *J Prod Innov Manag* 30:62–76
- Sosa ME, Eppinger SD, Rowles CM (2004) The misalignment of product architecture and organizational structure in complex product development. *Manag Sci* 50:1674–1689
- Steward DV (1981) The design structure system: a method for managing the design of complex systems. *IEEE Trans Eng Manag EM-28:71–74*
- Sui RB, Zhong GJ (2008) Research on degree distribution of networks of parts relationship for product family on scale-free networks. In: The 4th international conference on wireless communications, networking and mobile computing, Dalian, China, October 12–17, 2008, pp 1–4
- Thadakamaila HP, Raghavan UN, Kumara S, Albert R (2004) Survivability of multiagent-based supply networks: a topological perspective. *IEEE Intell Syst* 19:24–31
- Thevenot HJ, Simpson TW (2009) A product dissection-based methodology to benchmark product family design alternatives. *J Mech Des* 131:041002
- Thevenot HJ, Nanda J, Simpson TW (2005) A methodology to support product family redesign using genetic algorithm and commonality indices. In: ASME 2005 international design engineering technical conferences and computers and information in engineering conference, Long Beach, CA, September 24–28, pp 1009–1018
- Tseng MM, Jiao J, Merchant ME (1996) Design for mass customization. *CIRP Ann Manuf Technol* 45:153–156
- Tu Q, Vonderembse MA, Ragu-Nathan T (2001) The impact of time-based manufacturing practices on mass customization and value to customer. *J Oper Manag* 19:201–217
- Tucker CS, Kim HM (2008) Optimal product portfolio formulation by merging predictive data mining with multilevel optimization. *J Mech Des* 130:041103
- Ulrich K (1995) The role of product architecture in the manufacturing firm. *Res Policy* 24:419–440
- Wernicke S (2005) A faster algorithm for detecting network motifs. In: the 5th international workshop on algorithms in bioinformatics, Mallorca, Spain, October 3–6, pp 165–177
- Wernicke S, Rasche F (2006) FANMOD: a tool for fast network motif detection. *Bioinformatics* 22:1152–1153
- Wheelwright SC, Sasser WE Jr (1989) The new product development map. *Harv Bus Rev* 67:112–125
- Xu X, Gao L, Fang S (2008) Product family architecture evolution based on technology evolution theory of TRIZ. In: The 7th world congress on intelligent control and automation, Chongqing, China, June 25–27, pp 5662–5666
- Yeang C-H, Huang L-C, Liu W-C (2012) Recurrent structural motifs reflect characteristics of distinct networks. In: 2012 IEEE/ACM international conference on advances in social networks analysis and mining (ASONAM), Istanbul, Turkey, August 26–29, pp 551–557

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.