JOURNAL OF ENGINEERING DESIGN, 2015
http://dx.doi.org/10.1080/09544828.2015.1083539

Taylor & Francis
Taylor & Francis Group

An automated approach to quantifying functional interactions by mining large-scale product specification data

AQ1

Sung Woo Kang[a] and Conrad Tucker[b]

AQ2

~~[a]Department of Industrial Engineering~~, ~~Penn State University~~, ~~State College~~, PA, USA; [b]Department of Engineering Design, ~~Industrial Engineering~~, ~~Penn State University~~, ~~State College~~, PA, USA

ABSTRACT

The authors of this work hypothesise that the semantic relationship between modules' functional descriptions is correlated with the functional interaction between the modules. A deeper comprehension of the functional interactions between modules enables designers to integrate complex systems during the early stages of the product design process. Existing approaches that measure functional interactions between modules rely on the manual provision of designers' expert analyses, which may be time consuming and costly. The increased quantity and complexity of products in the twenty-first century further exacerbates these challenges. This work proposes an approach to automatically quantify the functional interactions between modules, based on their textual technical descriptions. Compared with manual analyses by design experts who use traditional design structure matrix approaches, the text-mining-driven methodology discovers similar functional interactions, while maintaining comparable accuracies. The case study presented in this work analyses an automotive climate control system and compares the functional interaction solutions achieved by a traditional design team with those achieved following the methodology outlined in this paper.

ARTICLE HISTORY
Received 15 March 2014
Accepted 12 August 2015

KEYWORDS
Functional interaction; text mining; design structure matrix

1. Introduction

To be successful in today's global market, companies try to offer competitive and highly differentiated products by analysing and developing product functions that satisfy customers' needs (Umeda et al. 2005). A product's *function* represents its operational purpose that meets customers' requirements (Umeda et al. 2005). This high-level product function, which directly interfaces with the customer, can be composed of multiple modules that perform each sub-function. A module performs a specific function by controlling the interactions of the functions of components (Jose and Tollenaere 2005). Analysing a product's functional characteristics is the initial step of the design process and precedes the definition of other aspects such the form and material (Bohm and Stone 2004; Bryant et al. 2005; Stone, Wood, and Crawford 1999; Umeda et al. 2005). Therefore, engineering designers need to understand both the functional interactions between each module and how these interactions impact the overall product. These functional interactions indicate the degree of

CONTACT Conrad Tucker ctucker4@psu.edu

modularity among the attached modules and enable the designers to create new modules for next-generation products by integrating/maintaining current modules within a product family/product portfolio (Dahmus, Gonzalez-Zugasti, and Otto 2001; Gershenson, Prasad, and Zhang 2003; Schilling 2000). To design a product, the designers must analyse the degree of functional interactions between modules based on their expertise/domain knowledge (Browning 2001; Danilovic and Browning 2007; Helmer, Yassine, and Meier 2010; Pimmler and Eppinger 1994; Sharman and Yassine 2004; Sosa, Eppinger, and Rowles 2003, 2004; Yassine and Braha 2003). However, expert manual analyses (e.g. analyses by designers that quantify functional interactions between modules) may be a time-consuming and costly process (Liang, Tan, and Ma 2008; Mudambi and Schuff 2010; Rockwell et al. 2008; Yanhong and Runhua 2007; Yoon and Park 2004). These challenges are further exacerbated by the constant increase in product quantity and complexity, that are primarily driven by customers' increasing desires for customisable products (Alizon, Shooter, and Simpson 2009; Christensen, Cook, and Hall 2005; Tucker and Kang 2012). For example, at the start of the twentieth century, 92 modules were required to construct a complete car. However, more than 3500 modules currently exist in a modern-day vehicle (Ford Motor Company 1989; Groote 2005). Over 30,000 new consumer products launched into the market each year (Christensen, Cook, and Hall 2005). Therefore, the complexity of managing modular product designs and their inherent functional interactions becomes cumbersome (Dahmus, Gonzalez-Zugasti, and Otto 2001).

This work measures the functional interactions between modules by analysing the semantic relationships between the modules' functional descriptions. The methodology presented in this work quantifies the functional interactions between modules by employing text-mining algorithms that analyse modules' functional descriptions, represented textually through technical manuals pertaining to each module. The authors of this work hypothesise that the semantic relationship between modules' functional descriptions is correlated with the functional interaction between the modules. A statistical analysis that compares the results of the text-mining methodology with experts' manual analysis of functional interactions (Browning 2001; Pimmler and Eppinger 1994; Pimmler 1994) is presented. This text-mining-driven methodology achieves results in a timely and efficient manner that are comparable to the designers' manual analyses.

This paper is organised as follows. This section discusses the research motivation; Section 2 describes works related to the research; Section 3 outlines the proposed methodology; Section 4 presents an automotive climate system case study that demonstrates the feasibility of the methodology; the results of the case study are discussed in Section 5; and Section 6 provides the conclusion and future-related work.

2. Related works

The literature review begins by discussing the functional model used during the early stages of the product development process (Section 2.1). Then, the literature regarding semantic analyses in the engineering design fields is presented (Section 2.2). Section 2.3 reviews the formation of a module on the basis of functional interactions between modules. In Section 2.4, the literature related to manual approaches for measuring functional interactions articulates the need for an automated methodology that analyses these interactions.

2.1. *Functional modelling in engineering design*

A functional model in engineering design is a structured representation of standardised functions and the flows between these functions within the formalised design space. The functional model generates a chain of functions as a process connected by energy, signal, and material flows – the essential requirements for operating each function, hereby developing a conceptual product design (Baxter, Juster, and De Pennington 1994; Bonjour et al. 2009; Hirtz et al. 2002; Kurtoglu et al. 2009; Stone and Wood 2000). The functions and flows are defined on a functional basis, which designers describe with standardised technical terminology (Stone and Wood 2000).

In product design, functional modelling is a crucial step in defining a product's architecture, wherein the architecture indicates the product's functional structure through which the product's function is allocated to physical modules. Designers have created a functional architecture for a next-generation product on the basis of a functional model and the functional interactions between the candidate modules (Kurtoglu et al. 2009; Sangelkar and McAdams 2013; Sen, Summers, and Mocko 2010). A quantitative functional model that captures both the product functionality and customer requirements has been proposed (Stone, Wood, and Crawford 1999). This model employs modular theory and requires an assessment tool to build a product repository for grouping products based on functionality and customer requirements.

Analysing the interdependency of a product's functions can be performed independent of the analysis of a product's form (Kurtoglu et al. 2009; McAdams, Stone, and Wood 1999). This allows designers to make an explicit connection between modules by measuring the interdependency of functions. These connections are usually inherent in the product development process. The functional model enables product development and manufacturing on a mass customised scale (Kahn, Castellion, and Griffin 2005).

A function dividing process (FDP) has been proposed to obtain sub-system-level (e.g. module) functions from system-level (e.g. product) functions (Taura and Nagai 2013). A FDP divides a system-level function in two ways: the decomposition-based dividing process and the casual-connection-based dividing process. The causal-connection-based dividing process addresses the functional interdependency among a group of modules, wherein the functions of a module are similar to the adjacent functions of a product at the same level. The decomposition-based dividing process addresses functional independency when the product function is realised with the functions of each independent module. The model generates new module functions by integrating existing sub-system functions with a functional model.

In the context of systems engineering, the need for a standard design process has arisen due to the international trade of system products and services. Therefore, EIA 632 has been established for standardising functions of systems (Martin 2000). The ISO/IEC 15288 standard has been introduced in the engineering design community to provide a common/comprehensive design framework for managing system development projects (Arnold and Lawson 2004). However, as the quantity and diversity of modules continue to increase, designers are presented with the challenge of searching through the entire design space for novel design knowledge that can lead to next-generation products.

To overcome these challenges, the methodology presented in this paper, employs semantic analysis techniques that discover functions from a large set of text data sets that describe modules. Functional interactions between modules are then automatically quantified in order to aid designers during next generation product design by informing them of the modules that are tightly or loosely coupled.

2.2. *Knowledge extraction via semantic analysis*

Semantic analysis techniques that discover knowledge from large-scale textual data sets have been proposed across a wide range of science and engineering disciplines. The utilisation of these techniques in the science and engineering fields enables researchers to access an immense number of textual data sets by mining statistically significant terms. For example, Bollen, Mao, and Zeng (2011) have predicted changes in the stock market by analysing the moods inherent in large-scale Twitter feeds. Eysenbach has quantified the social impact of scholarly articles based on a semantic analysis of buzz on Twitter (2011). Asur and Huberman (2010) have successfully predicted box office revenues by mining information from tweets. Ginsberg et al. (2009) have presented a methodology that tracks influenza epidemics in a population by analysing a large number of Google search queries that were semantically related to the term 'influenza'. Huang, Liu, and Zhou (2010) identify complex phenotypes and demonstrate a disease-drug connectivity map by analysing semantic relationships across multiple diseases query expression profiles . Tuarob et al. (2013) have retrieved health-related information from social media data by analysing the semantics of heterogeneous features. Paul and Dredze (2014) were able to identify ailments along with the terms that represent the symptoms of each ailment by mining public health information from social network services such as Twitter. These studies demonstrate that it is possible to automatically discover semantic information that attains statistically significant correlations with ground truth data, despite using minimal human supervision.

In the engineering design fields, semantic analysis techniques have been employed to extract design knowledge from text-based product data, including customer feedback and product technical descriptions, in order to design products that better meet customers' needs (Ghani et al. 2006; Menon et al. 2003; Romanowski and Nagi 2004). Researchers have extracted product functions from their functional descriptions, such as patents or official manuals, through text-mining techniques (Ghani et al. 2006; Kang et al. 2013; Tseng, Lin, and Lin 2007; Tucker and Kang 2012; Tuarob and Tucker 2014). Menon et al. (2003) employed a vector space document representation technique to derive useful product development information from customer reviews. Tuarob and Tucker (2015a, 2015b) proposed the method to identify lead users and product demand by mining product attributes from a large scale of social media networks. Zhou, Jianxin Jiao, and Linsey (2015) have extracted latent customer needs from customer product reviews through semantic analysis, which identifies the hidden analogical reasoning of customers' preferences. Gu et al. (2012) have employed a semantic reasoning tool to represent functional knowledge as function-cell pairs, where the cell is defined as a conceptual structure denoting the structure category that interacts with similar functions. Ghani et al. (2006) have extracted semantics as product attributes on the basis of textual product descriptions by employing a generative model with the expectation maximisation (EM) technique.

Tucker and Kang have extracted semantics as functions and behaviours of products from textual descriptions in order to discover cross-domain knowledge among multiple product domains (2012). Kang et al. (2013) have employed a text-mining technique that quantifies functional similarities between end-of-life (EOL) product descriptions. Product modules that exhibited low functional similarity were deemed strong candidates for EOL value creation. These modules were then combined to form a product with enhanced functional capabilities made from EOL products.

Existing semantic-based techniques in the engineering literature have focused on employing text-mining techniques either at the early stages of customer needs analyses or at the end of life decision-making stages. This work hypothesises that semantic relationships between modules' functional descriptions are correlated with functional interactions between the modules. This work demonstrates the feasibility of employing text-mining algorithms to automatically quantify the functions of a module and measure the functional correlations between modules. To quantify the functional interactions of each module, an automatic interaction measurement (AIM) is proposed in this work that extracts functions and converts those functions into a vector space on the basis of the semantic relationships of each module.

2.3. *Modularity based on functional interaction*

Modules that make up a product interact with each other through relevant engineering principles and knowledge (Batallas and Yassine 2006; Cascini and Russo 2007; Hong and Park 2014; Jiao, Simpson, and Siddique 2007). Each interaction type (material, energy, information, and spatial) between modules allows the product to operate its functions in the correct manner (Sharman and Yassine 2004). Therefore, engineers assemble a product with modules that share similar materials, energies, information, and even shapes. Designers describe a product's mechanisms in its official descriptions (e.g. patent, textbook, and manual) with engineering terms or conceptually similar terms that indicate the interactions and operation processes (Eckert, Martin, and Christopher 2005; Kim, Manley, and Yang 2006; Murphy et al. 2014). Hence, the official textual descriptions of modules contain its functional descriptions and knowledge about the potential interactions between other modules.

The research on modularity is derived from Suh's design axiom, which establishes an understanding of the interactions between modules (1984). A module is a group of components having strong functional interactions that proceed to perform a specific function (Fujita and Ishii 1997; Gershenson, Prasad, and Zhang 2003; Jose and Tollenaere 2005). It is difficult to integrate a module with other modules existing within a product when there are no functional interactions among them (Sharman and Yassine 2004). The definition of modularity is further revised by considering interdependent/independent interactions between modules (Gershenson, Prasad, and Zhang 2003; Gershenson, Prasad, and Zhang 2004). A module represents a unit of a product that independently performs a specific function. Product design methodologies based on modularity use modules as the standard units to construct products to increase the efficiency of both the product design and manufacturing processes (Huang and Kusiak 1998). For example, Volkswagen utilises a platform which is modularised with a floor panel, chassis, etc. for their products (Thevenot and Simpson 2006). The Ford Motor Company produces a climate control system to

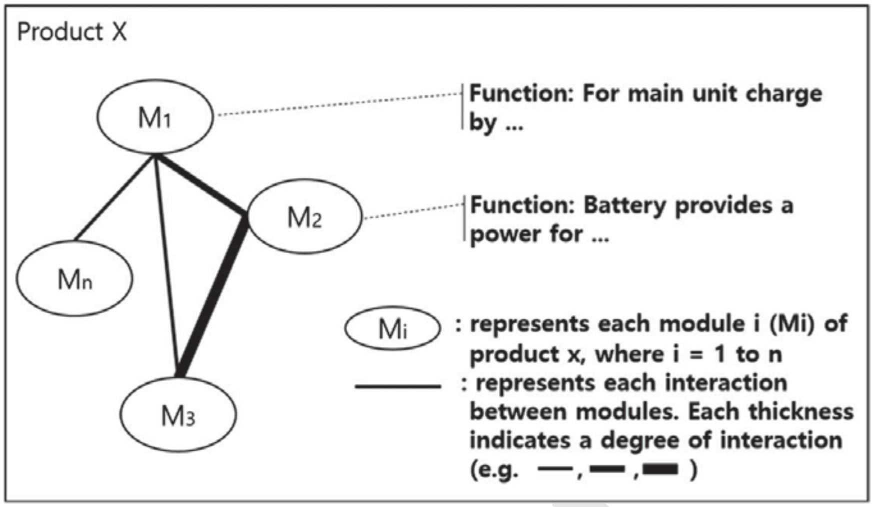

Figure 1. A modular product consisting of modules.

provide both heating and cooling for their customers by integrating 16 different modules (Pimmler and Eppinger 1994). Huang and Kusiak (1998) designed a digital circuit module containing end users' needs as functions by integrating electric components. Modules constitute a product and its functional interactions that achieve the product's primary function. Understanding the functional interactions between modules enables designers to identify which modules can be integrated when creating new modules or enhancing existing ones (Dahmus, Gonzalez-Zugasti, and Otto 2001; Gershenson, Prasad, and Zhang 2003; Schilling 2000). A modular product is constructed with multiple modules along with their functional interactions, as shown in Figure 1.

Each module performs a specific function(s), and these modules are connected to each other with different levels of functional interactions, as shown in (Figure 1). Although the identification of a functional interaction is an important factor for developing both modules and products, quantifying the degree of functional interaction (e.g. the line thickness in Figure 1) between modules has, until now, primarily relied extensively on manual feedback, which can be costly and time consuming, especially as products become more complex.

The methodology presented in this work automatically quantifies the degree of functional interaction of each module on the basis of each module's functional description. This work supports designers by creating an automated algorithm that discovers which modules can be integrated into a new module or which modules are not suitable for functional modifications.

2.4. *Functional interaction analysis through a design structure matrix*

In the context of engineering design, many researchers have employed a design structure matrix (DSM) to visualise the functional interactions among modules on the basis of experts' knowledge (Dahmus, Gonzalez-Zugasti, and Otto 2001; Jiao, Simpson, and Siddique 2007; Pimmler and Eppinger 1994; Sosa, Eppinger, and Rowles 2003, 2004). Steward coined the term 'DSM', a matrix-based approach to analysing system design structures (1981). The DSM's row index and the column index are described as system elements represented as modules, with the cells of the matrix representing the interactions between the elements. The initial stage of creating both matrices defines each *element* that is the object of interaction. Pimmler represented these system elements as product modules, proposing a taxonomy of functional interactions (spatial, energy, information, and material) with a quantification scheme to facilitate the means by which experts measure each interaction between modules (1994). Pimmler and Eppinger proposed a module-based DSM taxonomy to reorganise design teams along the lines of the functional interactions between modules (1994). Eppinger extended the DSM to integrate manufacturing systems by mapping both the functional interactions of power train modules and the interactions of their manufacturing processes (1997). Sosa, Eppinger, and Rowles (2003) identified new modular and integrative systems to develop complex products by clustering the quantified interface between modules and systems. They extended the research to analyse the functional misalignments of the product architecture and the organisational structure in complex product development by clustering the functional alignment of modules (Sosa, Eppinger, and Rowles 2004). Karniel and Reich proposed a 'DSM net' technique as a multi-level process model to create new products (2012). The DSM net is composed of design and process activities capable of checking process implementations on the basis of the functional interactions between modules. Existing DSM methodologies that measure functional interactions between modules are based on experts' analyses. However, it may be difficult for these manual-analyses-based approaches to quantify the functional interactions between modules because modern engineering products, such as vehicles, require more complicated modules than did earlier products. For example, approximately 3400 more modules are required for constructing vehicles today than at the start of the twentieth century (Ford Motor Company 1989; Groote 2005). The methodology presented in this work automatically quantifies the functional interactions across a wide range of modules, thereby reducing the time and costs associated with manual analysis techniques.

In the early stages of the product design and development process, designers have been supported by automatic approaches or platforms in each step, as shown in Figure 2. Although designers are supported in each step by automated approaches or

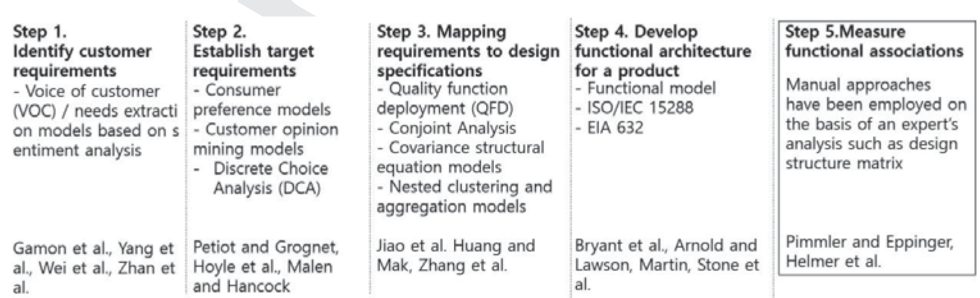

Figure 2. The beginning phase of the new product development process (Arnold and Lawson 2004; Bryant et al. 2005; Gamon et al. 2005; Helmer, Yassine, and Meier 2010; Huang and Mak 1999; Jiao and Chen 2006; Martin 2000; Petiot and Grognet 2006; Pimmler and Eppinger 1994; Stone et al. 2008; Wei et al. 2009; Yang, Wei, and Yang 2009; Zhan, Loh, and Liu 2009; Zhang et al. 2012).

platforms, Step 5 is still heavily reliant on manual processes. The objective of Step 5 is to quantify the functional interactions between candidate modules for selecting appropriate modules that satisfy both the design specifications and their interactions.

The methodology presented in Section 3 employs a topic model algorithm and a cosine measure to quantify the functional interactions among modules, with the purpose of moving towards automated methods that help to minimise the manual processes of Step 5 in the engineering design process (Figure 2).

3. Methodology

The methodology quantifies the degree of functional interactions of each module based on the module's functional descriptions by employing a topic model technique from natural language processing, thereby enabling designers to automatically model a product's functional architecture, and, as a result, minimise the manual analyses. This entire process, outlined in Figure 3, is defined as the AIM.

Step 1 describes the function data-acquisition process for creating a database containing products' module information. In this work, the module descriptions are assumed to be represented textually. Each function extracted from a module's functional description is converted into a vector space in Step 2. Step 3 quantifies the functional interactions between the modules by measuring their vector similarities. Finally, the methodology can automatically measure the degree of functional interactions between modules and can serve as a guide for designers aiming to understand the complexities of the functional interactions within modular products.

3.1. *Create a database of modules*

The first step in the methodology is to construct a database that consists of a product's modules and contains its functional descriptions. A module's function data can be acquired from textual descriptions, such as patents or official manuals (Brunetti and Golob 2000; Sheldon 2009; Sheremetyeva, Nirenburg, and Nirenburg 1996; Umeda et al. 2005), as shown in Figure 4. It is assumed that each module has a unique identification number (ID) that will be used to automatically search for and discover the functional interactions between the modules.

A product's database composed of each module's functional descriptions is created, as shown in Figure 4. Because engineering documents such as a functional description of a

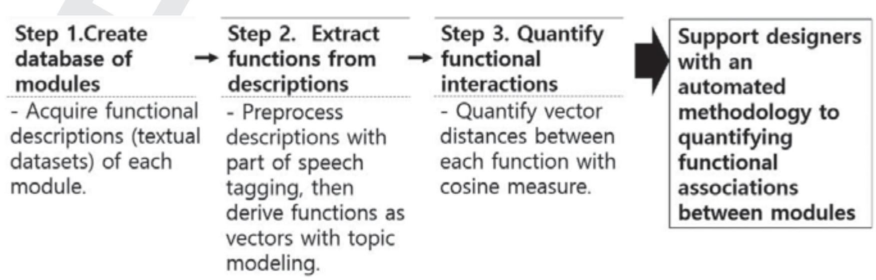

Figure 3. Flow diagram of the AIM.

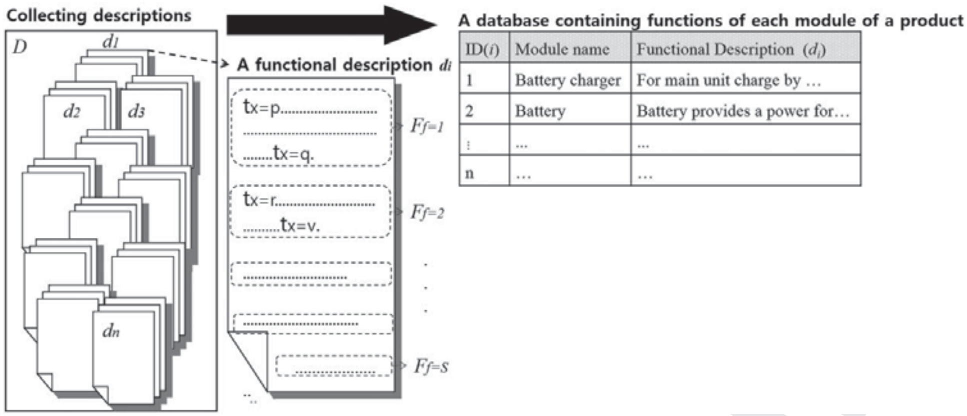

Figure 4. Process for extracting the functional descriptions from textual data. Notes: t_x represents the textual terms of a product's functional description (D). D represents a product's functional description, which is composed of the functional description of each module (e.g. $D = \{d_1, d_2, \ldots d_n\}$). d_i represents each module's functional description. F_f is the f^{th} paragraph of a functional description (d_i). S represents the total number of functions of the module that has the most functions.

module are laid out in a structured way, a paragraph of the description may contain one topic, which can be regarded as a function in this research (Nagle 1996). The next section describes how the text-mining algorithm extracts the functions from the modules' textual descriptions.

3.2. *Extracting functions based on topic modelling*

In the context of engineering design, official specifications, technical manuals, and patents can be regarded as the functional descriptions that include topics that can be represented as functions, as shown in Figure 4. Because a functional description is usually written in natural language, it includes many terms that do not provide any important information. Therefore, unnecessary terms such as linking verbs (e.g. is, and, etc.) are eliminated to reduce noise (Munková Daša, Michal, and Martin 2014; Murphy et al. 2014). To extract functions from the descriptions, this work employs natural language-processing techniques: a part of speech tagger and the Latent Dirichlet Allocation (LDA) algorithm (Blei, Ng, and Jordan 2003). Because functional terms mainly use verbs and nouns in functional descriptions, a part-of speech (POS) tagger algorithm, which identifies verbs and nouns, is employed in this work (Ahmed, Kim, and Wallace 2007; Toutanova and Manning 2000). Once the POS tagger pre-processes the functional descriptions, LDA extracts topics (e.g. functions) from the descriptions. LDA is a generative probabilistic model for compilations of text corpora, which can be regarded as functional descriptions with infinite mixtures over intrinsic topic groups (Blei, Ng, and Jordan 2003). Because each paragraph of the description may describe each product's function with terms, the LDA algorithm postulates that the description is a finite mixture of the number of functions and that each term's establishment is due to one of the functions from the description (Sheldon 2009). LDA provides the mixing proportions of functions through a generative

Table 1. Quantified functional data set.

ID(i)	Module name	Function 1	Function 2	…	Function S
1	Battery charger	$p(F_1\|d_1)$	$p(F_2\|d_1)$	…	$p(F_S\|d_1)$
2	Battery	$p(F_1\|d_2)$	$p(F_2\|d_2)$	…	$p(F_S\|d_2)$
⋮	⋮	⋮	⋮	⋮	⋮
n	…	$p(F_1\|d_n)$	$p(F_2\|d_n)$	…	$p(F_S\|d_n)$

probabilistic model on the basis of the Dirichlet distribution, as shown in the following equation:

$$p(t_x|d_i) = \sum_{f=1}^{S} p(t_x|F_f)p(F_f|d_i), \tag{1}$$

where t_x represents the textual terms of a product's functional description $D = \{d_1, d_2, \ldots d_n\}$. d_i represents each module's functional description. F_f is the ~~fth~~ function of a product's functional description (D). S represents the total number of functions of the module with the most functions. The total number of functions (S) is the same as the number of paragraphs (specifications) in the functional description of the module that has the most functions.

From the set of modules of a product, each function is sequentially extracted with topic probabilities from the entire functional description ($D = \{d_1, d_2, \ldots d_n\}$), as shown in Table 1, that is, Function 1 $p(F_1|D), \ldots,$ Function S $p(F_S|D)$. These functions represent abstracts of each functional description in terms of contextual semantics. From Equation (1), $p(F_f|d_i)$ measures the probability of function (F_f) being a topic of a technical description (d_i). To compare the functional interaction between modules, each functional probability of the ~~ith~~ module ($p(F_f|d_i)$) represents the functional descriptive vector in matrix form, as shown in Table 1.

3.3. *Quantifying functional interactions*

To search for functional interactions across textual data sets, the cosine measure has been employed using the LDA results. The cosine measure is employed in this work to quantify the degree of functional interaction between each module. The functional interaction between modules can be quantified by inputting the functional descriptive vectors from Table 1 into the cosine measure. For instance, the functional interaction between the functional vector 'Battery charger' and 'Battery' can be quantified with the following equation:

$$\cos(V_{i=1}|V_{i=2}) = \frac{V_{i=1} \cdot V_{i=2}}{||V_{i=1}||\ ||V_{i=2}||}, \tag{2}$$

where

$$V_{i=1} \cdot V_{i=2} = \sum_{f=1}^{S} p(F_f|d_1)p(F_f|d_2) \tag{3}$$

$$||V_{i=1}|| = \sqrt{\sum_{f=1}^{S} p(F_f|d_1)^2} \tag{4}$$

$$||V_{i=2}|| = \sqrt{\sum_{f=1}^{S} p(F_f|d_2)^2} \tag{5}$$

Each variable $V_{i=1}$ and $V_{i=2}$ represents a vector coordinate of the *Battery charger's* ($i = 1$) and *Battery's* ($i = 2$) functions in Table 1. The cosine measure is 1 when the angle between the two vectors is 0 degrees, while the cosine measure is 0 when the angle between the two vectors is 90 degrees. Therefore, the functional interaction increases when the cosine metric between the functional descriptive vectors is close to 1, whereas 0 means that there are no interactions between the functions. Modules that have strong functional interactions with one another can be integrated into a new module, while modules with low functional interactions may be independently updated/enhanced with minimal impact on the other modules (Browning 2001; Hirtz et al. 2002). In this work, if the cosine measure between the functional descriptive vectors results in a value of 1, it is assumed

```
Input: d_i: a functional description for i^th module
        Set i= 1
        Loop
                Run POS
                Record POS result
                                If i > n
End; POS end
Import record POS result (i = 1 to n)
                Set i = 1
                Loop
                                Set f = 1
                                                Loop
                                                Run LDA
                                                Record LDA result
                                                                Set f = f + 1
                                                                If f > S
                                                End
                                Set i = i+1
                                If i > n
End; LDA end
Import record LDA result (i = 1 to n)
Set i =1
                Loop
                Set j= i + 1
                                Loop
                                Run cosine measure (i, j)
                Set t = cosine measure results
                        If t  < 1
                                                Print out t
                                Set j = j + 1
                                If j > n
                                End
                Set i = i+1
                If i > n
End; cosine measure end
```

Figure 5. Algorithm flow of the AIM.

that the corresponding modules are identical. Sophisticated engineering products, ranging from automobiles to aircrafts, may include multiple identical modules in one system. For instance, most cars have two headlights that perform independently of one another. Because the interactions between one headlight and another module (e.g. wheel) would result in the same cosine similarity value, regardless of whether it is the right headlight or the left headlight, directly searching functional interactions across identical modules is redundant and therefore not considered in the methodology.

In the early stages of the product development process, designers consider functional interactions between different modules for functional architecture modelling. Figure 5 presents the algorithmic flow of the AIM. The AIM imports data sets from a database and then cleanses the textual data by employing the POS tagger. LDA extracts functions from the POS results, and then the cosine measure quantifies the functional interactions across the modules on the basis of the LDA results.

In contrast to traditional DSM approaches, the methodology outlined in this work analyses functional interactions in an automated manner, with minimal manual input from designers. This is particularly important as the number of modules and functional interactions increase in complex products. To evaluate the AIM, the next section introduces a DSM study as a case study for comparing the interaction analysis.

4. Application

The case study analyses an automotive climate control system that combines 16 modules that have 120 functional interactions. The case study is introduced to verify the feasibility of the AIM presented in Section 3 by comparing it to manual analyses performed by design experts. Pimmler and Eppinger extracted functional interactions between the automotive climate control system's modules through a taxonomy of functional interactions and a manual quantification process, as shown in Table 2 (1994). Their study analysed modules with four different interaction types (spatial, energy, information, and material) based on five different scores (Required/Desired/Indifferent/Undesired/Detrimental) (Pimmler and Eppinger 1994). Functional interactions are quantified by four different generic relationship types with values of −2, −1, 0, 1, and 2, as shown in Table 2. In Pimmler and Eppinger's study, a DSM was generated by conducting interviews with experts from the Ford Motor Company; the original DSM is described in the appendix (1994). Although the manual DSM analysis provides reliable outputs, it may be difficult to extract functional interactions, as the quantity and complexity of modules continues to increase in today's twenty-first century product space.

Table 2. Interaction types and quantification of the DSM (Pimmler and Eppinger 1994).

Type	Interaction values
Spatial	Needs for adjacency or orientation between two modules. Required(+2)/Desired(+1)/Indifferent(0)/Undesired(−1)/Detrimental(−2)
Energy	Needs for energy transfer/exchange two modules. Required(+2)/Desired(+1)/Indifferent(0)/Undesired(−1)/Detrimental(−2)
Information	Needs for data or signal exchange between two modules. Required(+2)/Desired(+1)/Indifferent(0)/Undesired(−1)/Detrimental(−2)
Material	Needs for material exchange between two modules. Required(+2)/Desired(+1)/Indifferent(0)/Undesired(−1)/Detrimental(−2)

The methodology presented in this work mitigates this challenge by automatically analysing functional interactions on the basis of modules' functional descriptions. To verify the feasibility of the methodology, this case study compares the interactions from the AIM to those of the DSM generated by manual expert feedback that has been thoroughly studied in the engineering design community (Browning 2001; Pimmler and Eppinger 1994).

Sosa, Eppinger, and Rowles (2003, 2004) have proposed the 'design interface strength' to measure the degree of the overall functional interactions between modules on the basis of each interaction type and its scale value from Pimmler and Eppinger's research shown in Table 2.

The interaction scale ranges from 0 (e.g. all values of each generic interaction are 0 or a negative value) to 8 (e.g. all values of each generic interaction are 2) by integrating the four generic interaction types: spatial $|2|$ + energy $|2|$ + information $|2|$ + material $|2|$ for a maximum of 8. To distinguish modules in the system, Sosa, Eppinger, and Rowles (2004) categorised the strength of the functional interactions into 'low' (less than 4) and 'high' (greater than 4).

In this paper, because the interactions are measured on a {0, 1} scale, the interaction values of the DSM need to be normalised to the same scale. The manually analysed interaction values of the automotive control system are normalised to a {0, 1} scale, as shown in Table 3.

In Table 3, 'M' stands for the module, module 1 represents the radiator, module 2 represents the engine-fan, module 3 represents the heater core, module 4 represents the heater hoses, module 5 represents the condenser, module 6 represents the compressor, module 7 represents the evaporator case, module 8 represents the evaporator core, module 9 represents the accumulator, module 10 represents the refrigerator controls, module 11 represents the air controls, module 12 represents the sensors, module 13 represents the command distribution, module 14 represents the actuator, module 15 represents the blower controller, and module 16 represents the blower-motor.

Referring to Sosa, Eppinger, and Rowles (2004), interactions can be divided into high or low values on the basis of the average value across the scale. Therefore, each degree of interaction is divided into high (H) and low values (L) in this paper, based on whether a functional interaction is greater or less than 0.5 (given a scale from 0 to 1).

Table 3. Normalised DSM functional interactions on the basis of a manual analysis.

M	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1		.50	.00	.00	.50	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
2			.00	.00	.50	.00	.00	.00	.00	.00	.00	.00	.13	.00	.00	.00
3				.13	.00	.00	.25	.00	.00	.00	.00	.00	.00	.00	.00	.25
4					.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00	.00
5						.50	.00	.50	.00	.00	.00	.00	.00	.00	.00	.00
6							.00	.50	.38	.25	.25	.00	.13	.00	.00	.00
7								.25	.00	.00	.00	.00	.00	.25	.25	.50
8									.38	.00	.00	.00	.00	.00	.00	.25
9										.13	.00	.00	.00	.00	.00	.00
10											.25	.00	.13	.00	.00	.00
11												.25	.13	.25	.25	.00
12													.13	.00	.00	.00
13														.13	.13	.13
14															.00	.00
15																.50
16																

Table 4. Transformed functional interactions on the basis of the normalised interactions.

Module	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1		H	L	L	H	L	L	L	L	L	L	L	L	L	L	L
2			L	L	H	L	L	L	L	L	L	L	L	L	L	L
3				L	L	L	L	L	L	L	L	L	L	L	L	L
4					L	L	L	L	L	L	L	L	L	L	L	L
5						H	L	H	L	L	L	L	L	L	L	L
6							L	H	L	L	L	L	L	L	L	L
7								L	L	L	L	L	L	L	L	H
8									L	L	L	L	L	L	L	L
9										L	L	L	L	L	L	L
10											L	L	L	L	L	L
11												L	L	L	L	L
12													L	L	L	L
13														L	L	L
14															L	L
15																H
16																

Table 4 describes the transformed functional interactions, which are based on the functional interaction degrees from Table 3. The automotive climate control system was analysed by design experts, whereas this work automatically quantifies the degree of interactions by automatically analysing the functional descriptions based on the LDA algorithm (Ramage and Rosen 2009).

4.1. *Create a database of the automotive climate control systems' modules*

To perform the experiment, this research follows each step of the AIM (referring to Figure 3 of Section 3). Each module's functional description has been collected from Daly's document as shown in Figure 6 (2006).

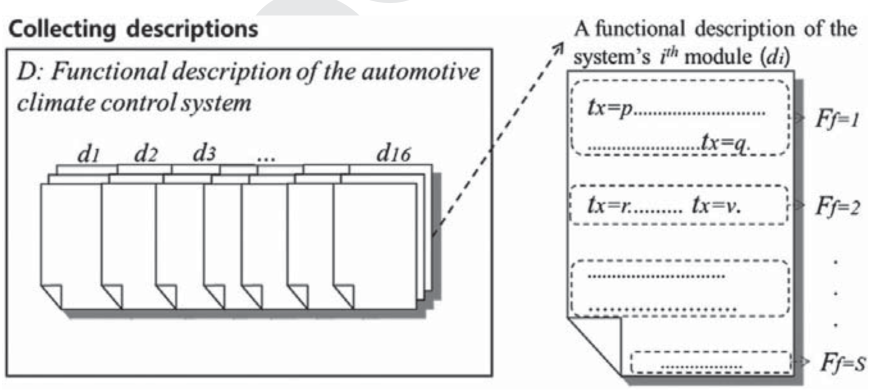

Figure 6. Functional description extraction process from textual data. Notes: D represents the automotive climate control system's functional description, which is composed of the functional description of each module (e.g. $D = \{d_1, d_2, \ldots d_{16}\}$). d_i represents each module's functional description. t_x represents the textual terms of the automotive climate control system's functional description (D). F_f is the ~~fth~~ paragraph of a functional description (d_i). S represents total number of functions of the module that has the most functions (e.g. $S = 10$ from d_8; the other descriptions have less than 10 paragraphs).

Table 5. A database containing functions of each module of the climate control system.

ID(i)	Module	Functional description (d_i)
1	Radiator	The radiator dissipates excess engine heat, …
2	Engine-fan	The engine-fan draws outside air into the engine, …
3	Heater-core	The heater-core transfers heat energy via forced, …
⋮	⋮	⋮
16	Blower-motor	The blower-motor moves fresh or vehicle interior air, …

On the basis of the data-collection process, the automotive climate control system's database, composed of each module's functional descriptions, is created, as shown in Table 5.

4.2. *Extracting functions from the functional descriptions*

The collected descriptions have been preprocessed by the POS tagger to extract nouns and verbs before performing LDA (Table 6). Both the preprocessing and function extraction processes are taken into account in step 2 of the methodology (Figure 3). These natural language-processing techniques are based on the Stanford Natural Language Processing platform (Ramage and Rosen 2009).

The AIM presented in this work measures the degrees of functional interactions between modules on the basis of functional descriptions. Given the functional descriptions of the 16 modules, the evaporator core (module 8) has the most functions among all the modules: $S = 10$, referring to variable S in Equation (1).

Therefore, 10 functions (e.g. heat, absorbs, transfers, coolant, air, energy, provides, temperature, exchanger, and accumulator) are extracted from the preprocessed automotive climate control system's entire functional description (D) by LDA, as shown in Table 7.

After each term representing the function of the climate control system is extracted, LDA quantifies the probabilities of each function being a topic of each module's technical description. Each functional probability ($p(F_f|d_i)$) of the climate control systems' modules represents the functional descriptive vector in matrix form, as shown in Table 8.

Quantifying the functional interactions between components is the final step (Step 3 in Figure 3 of Section 3) of the methodology presented in this research. The next section describes 120 interactions that are quantified by the AIM on the basis of the values

Table 6. Pre-process functional descriptions by the POS tagger.

ID ($I = 1$–16)	Module	Description : Pre-process by the POS tagger (n: noun, v: verb)
1	Radiator	The radiator dissipates excess engine heat, … : Radiator (n) dissipates (v) engine (n) heat (n), …
2	Engine-fan	The engine-fan draws outside air into the, … : Engine-fan (n) draws (v) air (n), …
⋮	⋮	⋮
16	Blower-motor	The blower-motor moves fresh air or, … : Blower-motor (n) moves (v) air (n), …

Table 7. Extracted functions from the automotive climate control system.

	F_1 = heat	F_2 = absorbs	F_3 = transfers	F_4 = coolant	…	F_{10} = accumulator
$P(F_f|D)$	0.14257	0.09431	0.09209	0.08999	…	0.00421

Table 8. Quantified functional data set for each module's description.

ID ($i=1$ to 16)	Module	F_1 = heat	F_2 = absorbs	F_3 = transfers	F_4 = coolant	…	F_{10} = accumulator
1	Radiator	0.03615	0.01503	0.09474	0		0.10043
2	Engine-fan	0.01274	0.10572	0.08347	0	…	0.02443
⋮	⋮	⋮	⋮	⋮	⋮	⋮	⋮
16	Blower-motor	0	0	0.00045	0.02899	…	0

from the LDA results. To verify the feasibility of the AIM, these interactions are compared with those of the manually analysed DSM (Table 4) on the basis of the statistical verification models presented in Section 5.

5. Results and discussion

The manually analysed DSM has been shown to be effective for analysing interactions between objects by providing designers with valuable results. Therefore, comparable results of the manually analysed DSM and the AIM presented in this work will demonstrate the feasibility of quantifying functional interactions in an automated manner so that designers can focus more on idea generation, rather than on functional mapping.

The degrees of the functional interactions are then quantified by the cosine measure (Equation (2)). The quantified interactions of each module are presented in Table 9. Each module (e.g. 1–16) represents the same modules presented in Table 4.

The interaction values from the AIM are transformed to binary values (high or low) using the same scale as the manual process, as shown in Table 10. Because the functional interaction degrees (i.e. manual analysis and the AIM) have been transformed to the same scale, a paired *t*-test and a confusion matrix are generated to provide statistical evidence of the similarity of the results from the AIM (Table 10) and the manual analysis (Table 4).

5.1. *Statistical verification: paired t-test*

Because a manual DSM has been shown to be effective for analysing functional interactions, a paired *t*-test is performed to determine whether there is a statistically significant

Table 9. Functional interaction by the AIM.

M	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1		.41	.47	.00	.75	.00	.00	.35	.00	.00	.00	.00	.14	.00	.00	.00
2			.00	.00	.22	.00	.08	.00	.00	.02	.28	.00	.00	.00	.01	.18
3				.28	.73	.40	.44	.93	.14	.01	.00	.00	.12	.00	.01	.28
4					.08	.00	.54	.03	.00	.01	.00	.00	.00	.00	.01	.00
5						.58	.08	.58	.21	.01	.01	.00	.13	.00	.01	.01
6							.00	.37	.36	.35	.00	.00	.00	.00	.01	.00
7								.41	.03	.21	.20	.00	.10	.40	.41	.50
8									.13	.01	.00	.00	.09	.00	.00	.39
9										.14	.00	.00	.00	.04	.01	.03
10											.01	.00	.01	.17	.42	.01
11												.32	.45	.00	.01	.08
12													.00	.00	.00	.00
13														.00	.36	.00
14															.34	.78
15																.01
16																

Table 10. Transformed functional interactions from the AIM results.

Module	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
1		L	L	L	H	L	L	L	L	L	L	L	L	L	L	L
2			L	L	L	L	L	L	L	L	L	L	L	L	L	L
3				L	H	L	L	H	L	L	L	L	L	L	L	L
4					L	L	H	L	L	L	L	L	L	L	L	L
5						H	L	H	L	L	L	L	L	L	L	L
6							L	L	L	L	L	L	L	L	L	L
7								L	L	L	L	L	L	L	L	H
8									L	L	L	L	L	L	L	L
9										L	L	L	L	L	L	L
10											L	L	L	L	L	L
11												L	L	L	L	L
12													L	L	L	L
13														L	L	L
14															L	H
15																L
16																

difference between the baseline results (e.g. manual DSM results) and the results generated by the AIM. This is achieved by comparing the degrees of functional interactions quantified using the proposed model (Table 10) and the manual DSM (Table 4). The paired *t*-test's null hypothesis assumes that the mean difference of paired values is 0. The paired values are the degrees of the functional interactions of each module from the manual DSM (Table 4) and the AIM (Table 10) results. The values from each analysis are paired if they have the same indices in both tables (the row and column indices of Tables 4 and 10 represent the same modules). The paired *t*-test is performed for 120 paired values (excluding values of identical row-column indices) to statistically determine whether the proposed automated and manual DSM results are significantly different. If the test does not reject the null hypothesis, the AIM can be regarded as a valid model for analysing the functional interactions for this case study. Based on the paired *t*-test results (N = 120), the mean difference of the *t*-test is 0. The results of this analysis indicate that the null hypothesis is not rejected, with a *t*-value of 0.00, a *p*-value of 1.000, and an α of 0.05. Because the null hypothesis is strongly supported by having *p*-value (1.000) greater than α (0.05), there is no significant difference between the functional interactions of each module from the proposed automated approach and the manual DSM. This test statistically verifies that the AIM provides functional interactions similar to the manually analysed interactions.

5.2. *Statistical verification: confusion matrix*

The confusion matrix (Table 11) shows that the AIM has 94% accuracy when benchmarked against the manual DSM generation; of a total of 120 instances, 114 (low: 110, high: 4) instances from the predictive model (the AIM) are matched with the actual model

Table 11. Confusion matrix: DSM vs. the AIM.

		Predictive class (the AIM)	
		L (Low)	H (High)
Actual class (DSM)	L (Low)	110	2
	H (High)	4	4

(DSM). In this case study, the AIM analyses low interactions among the modules with 98% precision and 96% recall.

Low interaction can be regarded as a functional independence (modularity) that affects how designers construct a system architecture with unique modules. These modules can be assembled for serving their own independent functions within the system. However, the AIM analyses high interactions among the modules with 50% precision and 67% recall, thereby providing designers with insufficient information regarding how modules should be integrated when creating a new module for next-generation products. Although the AIM presented in this work performed less accurately for extracting high interactions between modules, it discovers functionally detachable modules and guides designers in terms of which modules can be potentially detached, revised, or enhanced, with minimal impact on other sub-systems.

A text-mining technique may provide a more efficient means of quantifying functional interactions (especially for low functional interactions) between modules when compared with a manually generated DSM analysis, because the number of modules continues to increase along with their functional descriptions. To support designers with an analysis that is compatible with experts' manual analyses, the methodology needs to be improved for extracting high functional interactions in future work.

6. Conclusions and future work

This work includes an extensive literature review of research in the engineering design and text-mining fields. The literature review has shown that engineering design methodologies continue to introduce more automated approaches that support designers at both the customer needs and end-of-life product stages of the design process. Semantic analyses have been employed in the engineering design field to discover design information from customer reviews and functional descriptions. This work hypothesises that semantic relationships between modules' functional descriptions are correlated with functional interactions between the modules. To support designers in integrating/maintaining modules during the concept generation process, this work automatically measures the functional interactions between modules. By employing the LDA algorithm and the cosine metric, the methodology presented in this work discovers functional interactions between modules on the basis of semantic relationships between textual data sets that describe the modules' functions. Furthermore, the AIM has been validated using a case study involving a DSM analysis of an automotive climate system. The case study is conducted on a limited data set. The results achieved indicate the methodology's working prospects and scope. The authors would like to emphasise that this work explores a correlation (not a causal relationship) between modules' functional descriptions and modules' functional interactions.

The functional interactions between modules allow designers to efficiently create a product's architecture by integrating the modules' functions (Fixson 2007; Gershenson, Prasad, and Zhang 2003; Huang and Kusiak 1998; Sosa, Eppinger, and Rowles 2003). Functional interactions typically indicate the degree of modularity among modules at the beginning of the product development process, thereby enabling designers to make decisions such as to extend, upgrade, or maintain existing modules. The AIM algorithm presented in this work performed less accurately for extracting high interactions between modules than deriving low interactions. Thus, improving the methodology to accurately extract high functional interactions from functional descriptions may enable designers to discover modules that can be integrated during the creation of new modules for next-generation products.

Disclosure statement

No potential conflict of interest was reported by the authors.

ORCID

Conrad Tucker http://orcid.org/0000-0001-5365-0240

References

Ahmed, S., S. Kim, and K. M. Wallace. 2007. "A Methodology for Creating Ontologies for Engineering Design." *Journal of Computing and Information Science in Engineering* 7 (2): 132–140. doi:10.1115/1.2720879.

Alizon, F., S. B. Shooter, and T. W. Simpson. 2009. "Assessing and Improving Commonality and Diversity within a Product Family." *Research in Engineering Design* 20 (4): 241–253. doi:10.1007/s00163-009-0066-5.

Arnold, S., and H. W. Lawson. 2004. "Viewing Systems from a Business Management Perspective: The ISO/IEC 15288 Standard." *Systems Engineering* 7 (3): 229–242. doi:10.1002/sys.20006.

Asur, S., and B. A. Huberman. 2010. "Predicting the Future with Social Media." Web Intelligence and Intelligent Agent Technology (WI-IAT), IEEE/WIC/ACM International Conference, Vol. 1, 492–499. doi:10.1109/WI-IAT.2010.63. AQ3

Batallas, D., and A. Yassine. 2006. "Information Leaders in Product Development Organizational Networks: Social Network Analysis of the Design Structure Matrix." *Engineering Management, IEEE Transactions* 53 (4): 570–582. doi:10.1109/tem.2006.883706.

Baxter, J. E., N. P. Juster, and A. De Pennington. 1994. "A Functional Data Model for Assemblies Used to Verify Product Design Specifications." *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 208 (4): 235–244. doi:10.1243/PIME_PROC_1994_208_084_02.

Blei, D. M., A. Y. Ng, and M. I. Jordan. 2003. "Latent Dirichlet Allocation." *The Journal of Machine Learning Research* 3: 993–1022. Retrieved from http://dl.acm.org/citation.cfm?id=944919.944937.

Bohm, M. R., and R. B. Stone. 2004. "Representing Functionality to Support Reuse: Conceptual and Supporting Functions." ASME Design Engineering Technical Conference and Computers and Information in Engineering Conference, 411–419. Salt Lake City. doi:10.1115/DETC2004-57693. AQ4

Bollen, J., H. Mao, and X. Zeng. 2011. "Twitter Mood Predicts the Stock Market." *Journal of Computational Science* 2 (1): 1–8. doi:10.1016/j.jocs.2010.12.007.

Bonjour, E., S. Deniaud, M. Dulmet, and G. Harmel. 2009. "A Fuzzy Method for Propagating Functional Architecture Constraints to Physical Architecture." *Journal of Mechanical Design* 131 (6). doi:10.1115/1.3116253.

Browning, T. R. 2001. "Applying the Design Structure Matrix to System Decomposition and Integration Problems: A Review and new Directions." *IEEE Transactions on Engineering Management* 48 (3): 292–306. doi:10.1109/17.946528.

Brunetti, G., and B. Golob. 2000. "A Feature-Based Approach Towards an Integrated Product Model Including Conceptual Design Information." *Computer-Aided Design* 32 (14): 877–887. doi:10.1016/S0010-4485(00)00076-2.

Bryant, C., R. Stone, D. McAdams, T. Kurtoglu, and M. Campbell. 2005. "Concept Generation from the Functional Basis of Design." International Conference on Engineering Design, ICED (Vol. 5). Melbourne. AQ5

Cascini, G., and D. Russo. 2007. "Computer-aided Analysis of Patents and Search for TRIZ Contradictions." *International Journal of Product Development* 4 (1): 52–67. doi:10.1504/IJPD.2007.011533.

Christensen, C. M., S. Cook, and T. Hall. 2005. "Marketing Malpractice: The Cause and the Cure." *Harvard Business Review* 83 (12): 74–83. https://hbr.org/2005/12/marketing-malpractice-the-cause-and-the-cure.

Dahmus, J. B., J. P. Gonzalez-Zugasti, and K. N. Otto. 2001. "Modular Product Architecture." *Design Studies* 22 (5): 409–424. doi:10.1016/S0142-694X(01)00004-7.

Daly, S. 2006. *Automotive Air-Conditioning and Climate Control Systems*. Butterworth-Heinemann. AQ6

Danilovic, M., and T. R. Browning. 2007. "Managing Complex Product Development Projects with Design Structure Matrices and Domain Mapping Matrices." *International Journal of Project Management* 25 (3): 300–314. doi:10.1016/j.ijproman.2006.11.003.

Eckert, C. M., S. Martin, and E. Christopher. 2005. "References to Past Designs." In *Studying Designers 5.2005*, edited by J. S. Gero and N. Bonnardel, 3–19. Sydney: Key Centre of Design Computing & Cognition.

Eppinger, S. D. 1997. "A Planning Method for Integration of Large-scale Engineering Systems." International Conference on Engineering Design, ICED, 199–204. Tampere. AQ7

Eysenbach, G. 2011. "Can Tweets Predict Citations? Metrics of Social Impact Based on Twitter and Correlation with Traditional Metrics of Scientific Impact." *Journal of Medical Interest Research* 13 (4). doi:10.2196/jmir.2012.

Fixson, S. K. 2007. "Modularity and Commonality Research: Past Developments and Future Opportunities." *Concurrent Engineering* 15 (2): 85–111. doi:10.1177/1063293X07078935.

Ford Motor Company. 1989. *Ford Service Model T Practical Methods for Repairing and Servicing Ford Cars 1919–1927*. Ford Motor Company. AQ8

Fujita, K., and K. Ishii. 1997. "Task Structuring Toward Computational Approaches to Product Variety Design." Proceedings of the ASME International Design EngineeringTechnical Conferences. New York: ASME. AQ9

Gamon, M., A. Aue, S. Corston-oliver, and E. Ringger. 2005. "Pulse: Mining Customer Opinions from Free Text." IDA'05 Proceedings of the 6th international conference on Advances in Intelligent Data Analysis, 121–132. Springer Berlin Heidelberg. doi:10.1007/11552253_12. AQ10

Gershenson, J. K., G. J. Prasad, and Y. Zhang. 2003. "Product Modularity: Definitions and Benefits." *Journal of Engineering Design* 14 (3): 295–313. doi:10.1080/0954482031000091068.

Gershenson, J. K., G. J. Prasad, and Y. Zhang. 2004. "Product Modularity: Measures and Design Methods." *Journal of Engineering Design* 15 (1): 33–51. doi:0.1080/0954482032000101731.

Ghani, R., K. Probst, Y. Liu, M. Krema, and A. Fano. 2006. "Text Mining for Product Attribute Extraction." *ACM SIGKDD Explorations Newsletter* 8 (1): 41–48. doi:10.1145/1147234.1147241.

Ginsberg, J., M. H. Mohebbi, R. S. Patel, L. Brammer, M. S. Smolinski, and L. Brilliant. 2009. "Detecting Influenza Epidemics Using Search Engine Query Data." *Nature* 457 (7232): 1012–1014. doi:10.1038/nature07634.

Groote, S. De. 2005. "Construction of an F1 car." http://www.f1technical.net/articles/1271.

Gu, C.-C., J. Hu, Y.-H. Peng, and S. Li. 2012. "FCBS Model for Functional Knowledge Representation in Conceptual Design." *Journal of Engineering Design* 23 (8): 577–596. doi:10.1080/09544828.2011.629318.

Helmer, R., A. a. Yassine, and C. Meier. 2010. "Systematic Module and Interface Definition Using Component Design Structure Matrix." *Journal of Engineering Design* 21 (6): 647–675. doi:10.1080/09544820802563226.

Hirtz, J., R. B. Stone, D. A. McAdams, S. Szykman, and K. L. Wood. 2002. "A Functional Basis for Engineering Design: Reconciling and Evolving Previous Efforts." *Research in Engineering Design* 13 (2): 65–82. doi:10.1007/s00163-001-0008-3.

Hong, E.-P., and G.-J. Park. 2014. "Modular Design Method Based on Simultaneous Consideration of Physical and Functional Relationships in the Conceptual Design Stage." *Journal of Mechanical Science and Technology* 28 (1): 223–235. doi:10.1007/s12206-013-0956-3.

Huang, C. C., and A. Kusiak. 1998. "Modularity in Design of Products and Systems." *IEEE Transactions on Systems, Man, and Cybernetics – Part A: Systems and Humans* 28 (1): 66–77. doi:10.1109/3468.650323.

Huang, H., C.-C. Liu, and X. J. Zhou. 2010. "Bayesian Approach to Transforming Public Gene Expression Repositories into Disease Diagnosis Databases." *Proceedings of the National Academy of Sciences of the United States of America* 107 (15): 6823–6828. doi:10.1073/pnas.0912043107.

Huang, G. Q., and K. L. Mak. 1999. "Web-based Collaborative Conceptual Design." *Journal of Engineering Design* 10 (2): 183–194. doi:10.1080/095448299261380.

Jiao, J. R., and C. Chen. 2006. "Customer Requirement Management in Product Development: A Review of Research Issues." *Concurrent Engineering* 14 (3): 1–25. doi:10.1177/1063293X06068357.

Jiao, J., T. W. Simpson, and Z. Siddique. 2007. "Product Family Design and Platform-Based Product Development: A State-of-the-art Review." *Journal of Intelligent Manufacturing* 18 (1): 5–29. doi:10.1007/s10845-007-0003-2.

Jose, A., and M. Tollenaere. 2005. "Modular and Platform Methods for Product Family Design: Literature Analysis." *Journal of Intelligent Manufacturing* 16 (3): 371–390. doi:10.1007/s10845-005-7030-7.

Kahn, K. B., G. Castellion, and A. Griffin. 2005. *The PDMA Handbook of new Product Development*. Wiley. **AQ11**

Kang, S. W., C. Sane, N. Vasudevan, and C. S. Tucker. 2013. "Product Resynthesis: Knowledge Discovery of the Value of end-of-Life Assemblies and Subassemblies." *ASME Journal of Mechanical Design* 136 (1). doi:10.1115/1.4025526.

Karniel, A., and Y. Reich. 2012. "Multi-level Modelling and Simulation of new Product Development Processes." *Journal of Engineering Design* 24 (3): 185–210. doi:10.1080/09544828.2012.720015.

Kim, K.-Y., D. G. Manley, and H. Yang. 2006. "Ontology-based Assembly Design and Information Sharing for Collaborative Product Development." *Computer-Aided Design* 38 (12): 1233–1250. doi:10.1016/j.cad.2006.08.004.

Kurtoglu, T., M. I. Campbell, C. B. Arnold, R. B. Stone, and D. A. Mcadams. 2009. "A Component Taxonomy as a Framework for Computational Design Synthesis." *Journal of Computing and Information Science in Engineering* 9 (1): 011007. doi:10.1115/1.3086032.

Liang, Y., R. Tan, and J. Ma. 2008. "Patent Analysis with Text Mining for TRIZ." 4th IEEE International Conference on Management of Innovation and Technology, 1147–1151. IEEE. doi:10.1109/ICMIT.2008.4654531. **AQ12**

Martin, J. N. 2000. "Processes for Engineering a System: An Overview of the ANSI / EIA 632 Standard and Its Heritage." *Systems Engineering* 3 (1): 1–26. doi:10.1002/(SICI)1520-6858(2000)3:1<1::AID-SYS1>3.0.CO;2-0.

McAdams, D. A., R. B. Stone, and K. L. Wood. 1999. "Functional Interdependence and Product Similarity Based on Customer Needs." *Research in Engineering Design* 11 (1): 1–19. doi:10.1007/s001630050001.

Menon, R., L. H. Tong, S. Sathiyakeerthi, A. Brombacher, and C. Leong. 2003. "The Needs and Benefits of Applying Textual Data Mining Within the Product Development Process." *Quality and Reliability Engineering International* 20 (1): 1–15. doi:10.1002/qre.536.

Mudambi, S. M., and D. Schuff. 2010. "What Makes a Helpful Online Review? A Study of Customer Reviews on Amazon.com." *MIS Quarterly* 34 (1): 185–200. http://dl.acm.org/citation.cfm?id=2017447.2017457.

Munková Daša, M. Michal, and V. Martin. 2014. "Influence of Stop-words Removal on Sequence Patterns Identification within Comparable Corpora." In ~~*Advances in Intelligent Systems and Computing*~~, edited by V. Trajkovik and M. Anastas (Vol. 231), 66–76. Springer. doi:10.1007/978-3-319-01466-1_6. **AQ13**

Murphy, J., K. Fu, K. Otto, M. Yang, D. Jensen, and K. Wood. 2014. "Function Based Design-by-analogy: A Functional Vector Approach to Analogical Search." *Journal of Mechanical Design* 136 (10): 101102. doi:10.1115/1.4028093.

Nagle, J. G. 1996. *Handbook For Preparing Engineering Documents: From Concept To Completion*. John Wiley. AQ14

Paul, M. J., and M. Dredze. 2014. "Discovering Health Topics in Social Media using Topic Models." *PloS One* 9 (8). doi:10.1371/journal.pone.0103408.

Petiot, J.-F., and S. Grognet. 2006. "Product Design: A Vectors Field-based Approach for Preference Modelling." *Journal of Engineering Design* 17 (3): 217–233. doi:10.1080/09544820500275032.

Pimmler, T. U. 1994. *A Development Methodology for Product Decomposition and Integration*. Massachusetts Institute of Technology (Dept. of Mechanical Engineering). AQ15

Pimmler, T. U., and S. D. Eppinger. 1994. "Integration Analysis of Product Decompositions." Proceedings of the 1994 ASME Design Theory and Methodology Conference (DTM'94), 342–351. AQ16

Ramage, D., and E. Rosen. 2009. "Stanford Topic Modeling Toolbox." Retrieved from http://nlp.stanford.edu/software/index.shtml.

Rockwell, J. A., P. Witherell, I. Grosse, and S. Krishnamurty. 2008. "A Web-Based Environment for Documentation and Sharing of Engineering Design Knowledge." ASME Design Engineering Technical Conference and Computers and Information in Engineering Conference, 671–683. Brooklyn, New York, USA. doi:10.1115/DETC2008-50086. AQ17

Romanowski, C. J., and R. Nagi. 2004. "A Data Mining Approach to Forming Generic Bills of Materials in Support of Variant Design Activities." *Journal of Computing and Information Science in Engineering* 4: 316. doi:10.1115/1.1812556.

Sangelkar, S., and D. A. McAdams. 2013. "Mining Functional Model Graphs to Find Product Design Heuristics With Inclusive Design Illustration." *Journal of Computing and Information Science in Engineering* 13 (4): 041008. doi:10.1115/1.4025469.

Schilling, M. A. 2000. "Toward a General Modular Systems Theory and Its Application to Interfirm." *Academy of Management Review* 25 (2): 312–334. doi:10.5465/AMR.2000.3312918.

Sen, C., J. D. Summers, and G. M. Mocko. 2010. "Topological Information Content and Expressiveness of Function Models in Mechanical Design." *Journal of Computing and Information Science in Engineering* 10 (3). doi:10.1115/1.3462918.

Sharman, D. M., and A. A. Yassine. 2004. "Characterizing Complex Product Architectures." *Systems Engineering* 7 (1): 35–60. doi:10.1002/sys.10056.

Sheldon, J. G. 2009. *How to Write a Patent Application*. Practising Law Institute. AQ18

Sheremetyeva, S., S. Nirenburg, and I. Nirenburg. 1996. "Generating Patent Claims from Interactive Input." Proceedings of the Eighth International Workshop on Natural Language Generation, 61–70. AQ19

Sosa, M. E., S. D. Eppinger, and C. M. Rowles. 2003. "Identifying Modular and Integrative Systems and Their Impact on Design Team Interactions." *Journal of Mechanical Design* 125 (2): 240. doi:10.1115/1.1564074.

Sosa, M. E., S. D. Eppinger, and C. M. Rowles. 2004. "The Misalignment of Product Architecture and Organizational Structure in Complex Product Development." *Management Science* 50 (12): 1674–1689. doi:10.1287/mnsc.1040.0289.

Steward, D. V. 1981. "Design Structure System: A Method for Managing the Design of Complex Systems." *IEEE Transactions on Engineering Management* 28 (3): 71–74. doi:10.1109/TEM.1981.6448589.

Stone, R. B., R. Kurtadikar, N. Villanueva, and C. B. Arnold. 2008. "A Customer Needs Motivated Conceptual Design Methodology for Product Portfolio Planning." *Journal of Engineering Design* 19 (6): 489–514. doi:10.1080/09544820802286711.

Stone, R. B., and K. L. Wood. 2000. "Development of a Functional Basis for Design." *Journal of Mechanical Design* 122: 359. doi:10.1115/1.1289637.

Stone, R. B., K. L. Wood, and R. H. Crawford. 1999. "Product Architecture Development with Quantitative Functional Models." Proceedings of the ASME 1999International Design EngineeringTechnical Conferences. AQ20

Suh, N. P. 1984. "Development of the Science Base for the Manufacturing Field Through the Axiomatic Approach." *Robtics & Computer-Integrated Manufacturing* 1 (3): 397–415. doi:10.1016/0736-5845(84)90030-9.

Taura, T., and Y. Nagai. 2013. "Synthesis of Functions: Practice of Concept Generation (3)." In *Concept Generation for Design Creativity*. London: Springer. doi:10.1007/978-1-4471-4081-8. AQ21

Thevenot, H., and T. Simpson. 2006. "A Comprehensive Metric for Evaluating Component Commonality in a Product Family." ASME International Design Engineering Technical Conferences and Computers and Information in Engineering Conference, 823–832. Philadelphia: ASME. doi:10.1115/DETC2006-99268. AQ22

Toutanova, K., and C. D. Manning. 2000. "Enriching the Knowledge Sources used in a Maximum Entropy Part-of-speech Tagger." Proceedings of the 2000 Joint SIGDAT conference on Empirical methods in natural language processing and very large corpora: Held in conjunction with the 38th Annual Meeting of the Association for Computational Linguistics-Volume 13, 63–70. doi:10.3115/1117794.1117802. AQ23

Tseng, Y., C. Lin, and Y. Lin. 2007. "Text Mining Techniques for Patent Analysis." *Information Procsseing & Management* 43 (5): 1216–1247. doi:10.1016/j.ipm.2006.11.011.

Tuarob, S., and C. S. Tucker. 2013. "Fad or here to Stay: Predicting Product Market Adoption and Longevity Using Large Scale, Social Media Data." ASME International Design Engineering Conferences and Computers and Information in Engineering Conference, Portland, USA. AQ24 doi:10.1115/DETC2013-12661. AQ25

Tuarob, S., and C. S. Tucker. 2014. "Discovering Next Generation Product Innovations by Identifying Lead user Preferences Expressed through Large Scale Social Media Data." ASME International Design EngineeringTechnical Conferences & Computers and Information in Engineering Conference. Buffalo. doi:10.1115/DETC2014-34767. AQ26

Tuarob, S., and C. S. Tucker. 2015a. "Automated Discovery of Lead Users and Latent Product Features by Mining Large Scale Social Media Networks." *Journal of Mechanical Design* 137 (7): 071402. doi:10.1115/1.4030049.

Tuarob, S., and C. S. Tucker. 2015b. "Quantifying Product Favorability and Extracting Notable Product Features using Large Scale Social Media Data." *Journal of Computing and Information Science in Engineering* 15 (3): 031003. doi:10.1115/1.4029562.

Tuarob, S., C. S. Tucker, M. Salathe, and N. Ram. 2013. "Discovering Health-related Knowledge in Social Media using Ensembles of Heterogeneous Features." Proceedings of the 22nd ACM international conference on Conference on information & knowledge management – CIKM'13, 1685–1690. doi:10.1145/2505515.2505629. AQ27

Tucker, C. S., and S. W. Kang. 2012. "A Bisociative Design Framework for Knowledge Discovery Across Seemingly Unrelated Product Domains." Proceesings of ASME 2012 International Design Engineering Conferences and Computers and Information in Engineering Conference. Chicago, USA. doi:10.1115/DETC2012-70764. AQ28

~~Tucker, C. S., and H. M. Kim. 2011. "Trend Mining for Predictive Product Design." *Journal of Mechanical Design* 133 (11): 111008. doi:10.1115/1.4004987.~~ ~~AQ29~~

Umeda, Y., S. Kondoh, Y. Shimomura, and T. Tomiyama. 2005. "Development of Design Methodology for Upgradable Products Based on Function–Behavior–State Modeling." *AIE EDAM* 19 (3): 161–182. doi:10.1017/S0890060405050122.

Wei, C.-P., Y.-M. Chen, C.-S. Yang, and C. C. Yang. 2009. "Understanding What Concerns Consumers: A Semantic Approach to Product Feature Extraction from Consumer Reviews." *Information Systems and E-Business Management* 8 (2): 149–167. doi:10.1007/s10257-009-0113-9.

Yang, C.-S., C.-P. Wei, and C. C. Yang. 2009. "Extracting Customer Knowledge from Online Consumer Reviews: A Collaborative-filtering-based Opinion Sentence Identification Approach." Proceedings of the 11th International Conference on Electronic Commerce, 64–71. AQ30

Yanhong, L., and T. Runhua. 2007. "A Text-mining-based Patent Analysis in Product Innovative Process." In *Trends in Computer Aided Innovation* (Vol. 250) 89–96. MI. AQ31

Yassine, A. A., and D. Braha. 2003. "Complex Concurrent Engineering and the Design Structure Matrix Method." *Concurrent Engineering* 11 (3): 165–176. doi:10.1177/106329303034503.

Yoon, B., and Y. Park. 2004. "A Text-Mining-Based Patent Network: Analytical Tool for High-Technology Trend." *The Journal of High Technology Management Research* 15 (1): 37–50. doi:10.1016/j.hitech.2003.09.003.

Zhan, J., H. T. Loh, and Y. Liu. 2009. "Gather Customer Concerns from Online Product Reviews – A Text Summarization Approach." *Expert Systems with Applications* 36 (2): 2107–2115. doi:10.1016/j.eswa.2007.12.039.

Zhang, X. (Luke), T. W. Simpson, M. Frecker, and G. Lesieutre. 2012. "Supporting Knowledge Exploration and Discovery in Multi-Dimensional Data with Interactive Multiscale Visualisation." *Journal of Engineering Design* 23 (1): 23–47. doi:10.1080/09544828.2010.487260.

Zhou, F., R. Jianxin Jiao, and J. S. Linsey. 2015. "Latent Customer Needs Elicitation by Use Case Analogical Reasoning from Sentiment Analysis of Online Product Reviews." *Journal of Mechanical Design* 137 (7): 071401. doi:10.1115/1.4030159.

Appendix (Pimmler and Eppinger 1994)

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P
Radiator A		2 0 0 2			2 -2 0 0											
Engine Fan B	2 0 0 2				2 0 0 2								1 0 0 0			
Heater Core C				1 0 0 0			2 0 0 0	-1 0 0 0								0 0 0 2
Heater Hoses D			1 0 0 0						-1 0 0 0							
Condenser E	2 -2 0 0	2 0 0 2				0 2 0 2		-2 2 0 2								
Compressor F					0 2 0 2			0 2 0 2	1 0 0 2	0 0 2 0	0 0 2 0		1 0 0 0			
Evaporator Case G			2 0 0 0					2 0 0 0						2 0 0 0	2 0 0 0	2 0 0 2
Evaporator Core H			-1 0 0 0		-2 2 0 2	0 2 0 2	2 0 0 0		1 0 0 2							0 0 0 2
Accumulator I				-1 0 0 0		1 0 0 2		1 0 0 2		1 0 0 0						
Refrigeration Controls J						0 0 2 0			1 0 0 0		0 0 2 0		1 0 0 0			
Air Controls K						0 0 2 0				0 0 2 0		0 0 2 0	1 0 0 0	0 0 2 0	0 0 2 0	
Sensors L											0 0 2 0		1 0 0 0			
Command Distribution M		1 0 0 0				1 0 0 0				1 0 0 0	1 0 0 0	1 0 0 0		1 0 0 0	1 0 0 0	1 0 0 0
Actuators N							2 0 0 0				0 0 2 0		1 0 0 0			
Blower Controller O							2 0 0 0				0 0 2 0		1 0 0 0			2 0 0 2
Blower Motor P			0 0 0 2				2 0 0 2	0 0 0 2					1 0 0 0		2 0 0 2	

NOTE: BLANK MATRIX ELEMENTS INDICATE NO INTERACTION (FOUR ZERO SCORES).

Legend:

Spatial: S E :Energy
Information: I M :Materials