Int J Adv Manuf Technol (2018) 95:1215–1231
https://doi.org/10.1007/s00170-017-1267-0

ORIGINAL ARTICLE

Identification of key design characteristics for complex product adaptive design

Xin Han [1] · Rong Li [1] · Jian Wang [1] · Shengfeng Qin [2] · Guofu Ding [1]

Received: 26 July 2017 / Accepted: 26 October 2017 / Published online: 8 November 2017

Abstract Key design characteristics (KDCs) are important information related to the product and part designs, which significantly influence on the product's functions, performances, and quality. Identifying KDCs for a complex product will help designers to focus on key design parameters during the design process and rapidly obtain design schemes based on their close relationships to the product's functions, performances, and quality. Although there are some researches on key characteristic (KC) identification, most of them are focused on key process characteristics (KPCs) and few on KDCs. There also lacks a KDC identification framework to support KDC identification with better completeness and diverse usages. Adaptive design is the most important pattern of complex product design. Therefore, this paper presents a systematic method to identify KDCs for complex product adaptive design, in which KDCs can be determined by two related phases. Firstly, a product design specification (PDS)-KDC Candidates Network (PKCN) is constructed by using existing product instance data, cluster analysis, KC flow-down, and network analysis approaches. Then, the result from the first phase is used as a basis to identify KDCs for adaptive design. Three KDC identification techniques: similarity reasoning technique, breadth-first search (BFS), and the gray relational analysis approach are applied to find out KDCs from the PKCN, which are the most sensitive to the variation of a PDS. These identified KDCs can help designers to understand the relationships between KDCs and PDS and rapidly develop a design scheme. The effectiveness and the feasibility of the proposed method are verified by a case study via the development of an electric multiple unit (EMU)'s bogie.

Keywords Identification of key design characteristics · Systematic approach · Complex products · Adaptive design · Network analysis · Gray relational analysis

✉ Rong Li
bogiey@163.com

✉ Shengfeng Qin
sheng-feng.qin@northumbria.ac.uk

[1] Institute of Advanced Design and Manufacturing, Southwest Jiaotong University, Chengdu 610031, People's Republic of China

[2] School of Design, Northumbria University, Newcastle upon Tyne NE1 8ST, UK

1 Introduction

The manufacturing mode has transferred from mass production to mass customization under the global competition. Mass customization aims to provide customer satisfaction with increasing variety and customization without a corresponding increase in cost and lead time [1]. For enterprises engaged in complex engineering products, to achieve this goal, they need to (1) capture and reuse best practices and (2) to create strong links with the suppliers. However, complex products show complexities in the customer demands, product structures, embedded techniques, etc., which involve in various design characteristics and complex relations among them. This leads to a challenge that the enterprises cannot easily reuse the existing design knowledge to generate a design scheme and effectively communicate with suppliers to develop a supply plan.

Some enterprises and scholars used a method called key characteristic (KC) control to alleviate the above problems, such as using KCs to capture the most similar instances [2–5], or using KCs as an efficient medium for communication with suppliers [6, 7]. However, at present, there is no mutual definition for a KC. Thornton [8] referred to the previous definitions of KCs and defined KCs as the product, subassembly, part, and process features that significantly impact the real cost [23, 24], performance, or safety of a product when the KCs vary from nominal. According to the main phases of product development, KCs can be divided into design KCs and process KCs [9]. This paper focuses on the product design applications; thus hereafter, we use the key design characteristics (KDCs) instead of the general term KCs to describe the product and part design information that significantly influence on product functions, performances, and quality. KDCs can better support adaptive design that is the most important pattern of complex product design, which helps designers focus on a small set of critical design characteristics to rapidly develop an adaptive design scheme by reusing KDC-based design knowledge. Therefore, how to identify KDCs becomes an important issue in the complex product adaptive design.

At present, there are some researches on the application of KCs [2–7, 10–15], but few on the systematic identification. In addition, most of these studies focus on the KPCs, less on KDCs. KPCs are identified by establishing and analyzing the relations between process characteristics and cost/quality [23, 24], which are not completely applicable to the identification of KDCs, because the KDCs are not only related to the quality and cost but also mainly related to the functions and performances. From the perspective of research methods, research on the identification of KCs has focused on either qualitative analysis for KC acquisition [18–20] or quantitative analysis for KC priority [21–28]. However, both have some defects (see details in next section).

Therefore, this paper presents a systematic method to identify KDCs for complex product adaptive design. Our contributions have twofold. Firstly, in theory, we propose a framework with two related phases to (1) construct product design specification (PDS)-KDC Candidates Network (PKCN) and (2) identify KDCs for application. Secondly, in application, we develop a set of KDC identification techniques to support developments at each phase. Techniques used in phase 1 include the use of cluster analysis, KC flow-down, and a network analysis approach. In phase 2, similarity reasoning technique, breadth-first search (BFS), and a gray relational analysis approach are used to identify KDCs that are the most sensitive to the changes of PDS [29], which in turn help designers to focus on these KDCs and rapidly develop a design scheme. Thirdly, this research makes a complementary contribution to key process characteristic (KPC) identification by adding KDCs, toward a whole design and process KC identification for future rapid product life cycle development.

The remaining sections of this paper are organized as follows. Section 2 gives a brief review of the related work. Section 3 presents a systematic method to identify KDCs and related implementation techniques. Section 4 shows an example to illustrate the proposed method. In the final section, the conclusions are drawn.

2 Related work

2.1 The definition of KC and KDC

As mentioned before, there is no mutual definition for KCs presently. Some typical definitions are given as follows.

In Boeing's advanced quality system standard D1-9000, it defines a KC as a feature whose variation has the greatest impact on the fit, performance, or service life of the finished product from the perspective of the customer [10].

General Motors in its key characteristic designation system defines a KPC as a special characteristic where the loss function shows that reasonably anticipated variation within specification could significantly affect customer satisfaction with a product [11].

Besides the enterprises, some researchers who studied on the KCs also gave their definitions [7, 8]. Although the KC definitions may vary from corporations to researchers, the KC's methods have a common goal that is to identify a small set of critical features for an organization to focus on during design and manufacturing [21].

According to the main phases of product development, KCs are classified as key design characteristic and key process characteristics, as shown in Fig. 1. The KDCs can be further divided into functional design characteristics and structural design characteristics. The KPCs can be further divided into manufacturing process characteristics and assembly process characteristics.

This paper focuses on the problem areas of product design; thus, we define KDCs as the product and part design information that significantly influence on product function, performance, and quality. In this paper, the identification of KDCs is our main focus.

2.2 The application of KCs

The concept of KCs can be applied in product design and manufacturing [6]. In the area of product design, KCs (or KDCs) mainly have the following two functions: (1) using KCs as an important enabler to select similar cases and (2) using KCs as an efficient medium for coordination and communication. Firstly, since KCs are key product features that reflect typical properties of a product, they are usually utilized to find the most useful cases to help solve a target problem by calculating the similarity of KCs in available cases. Peng et al. [2], Kocsis et al. [3], and Zhu et al. [4] used KCs as an enabler to select a similar case during an engineering design phase. While Romli et al. [5] found the optimal from all possible solutions based on KCs to support sustainable product design. Secondly, effective communication among collaborators involved in product development is critical to the realization of rapid product development. Yang et al. [6] used KCs as an efficient medium for coordination among contractors,

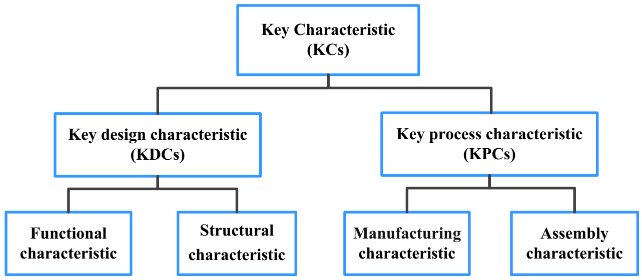


Fig. 1 KC classification

subcontractors, and partners, to set up a feasible and efficient model to facilitate the quality assurance in the supply chain. Dantan et al. [7] used product and process KCs as well as products and manufacturing resources to establish an information model for supporting complex product collaborative design.

In the area of product manufacturing, KCs (or KPCs) are mainly used for (1) process planning and (2) product quality control. Firstly, process planning includes manufacturing process and assembly process planning. For manufacturing process planning, Chin et al. [12] proposed an approach that combined QFD with FMEA to determine a process alternative with an adequate process capability based on KCs during a rough-machining process planning. For assembly process planning, Mathieu [13] proposed an approach to select an optimal assembly sequence based on KCs and assembly-oriented graphs. Zheng et al. [14] presented a novel algorithm for best assembly posture fit based on KCs for large component assembly to assure the assembly quality. Secondly, in the area of product quality control, Boeing [10] and General Motors [11] developed a control plan based on KPCs and used statistical process controls (SPCs) to monitor the variation of KPCs during the production process. Once the KPCs exceeded the control range, SPC would find the sources of variation and determine a solution to ensure the quality and performance of a product. Recently, Dai et al. [15] established a reliability model of manufacturing processes based on product KCs, material KCs, operation KCs, and equipment KCs, to reduce the risk of manufacturing process and improve product quality by calculating the reliability requirements of KCs with respect to different manufacturing process scenarios.

2.3 The KC identification techniques

At present, research on the identification of KCs has focused on either qualitative analysis for KC acquisition or quantitative analysis for KC priority.

2.3.1 Qualitative analysis for KC acquisition

Most KC identification methods are based on qualitative analysis for KC acquisition that uses the concept of a KC flow-down [16, 17], such as in Boeing [10] and GM Motors [11]. A KC flow-down is a hierarchical approach to tracing/propagating a key characteristic for an assembly or product down to key characteristics on its subassemblies, details, and processes believed to affect the variation of the top-level key requirement KCs [10]. The KC flow-down provides a systematic view of potential variation propagation of KCs and captures a design team's collective knowledge about variation and its contributors. Figure 2 shows an example of the KC flow-down for a car door [8].

In Fig. 2, one key customer requirement is the quality perception of the car door. Several product KCs of the car door (e.g., the evenness of the seams, etc.) influence the customer's perception of quality. Each product KC is linked to several contributing subsystem KCs (e.g., outer perimeter of the door, etc.). These, in turn, flow down to the part KCs (e.g., the door panel shape, etc.) and process KCs (e.g., the fixtures and stamping processes, etc.).

A variety of tools have been used to capture a KC flow-down, such as datum flow chain (DFC), assembly-oriented graph (AOG), and cause and effect diagram (CEA). For instance, Whitney [18] defined KCs as assembly-level dimensions related to design intention, and the delivery of those KCs was through a DFC. Mathieu et al. [19] adopted the AOG methodology to formulate an assembly model. A propagation chain that was an error accumulation route of the KCs was developed through analysis of the AOG. Sivasakthive et al. [20] used CEA methodology to identify the components' KCs related to the product performance.

However, KC flow-down method still has some defects: (1) it is a qualitative method for KC acquisition and lacks ability to prioritize the identified KCs, and (2) it lacks a detailed instruction of the processes of KC acquisition in the present study; as a result, it is not easy to implement and may lead to the incompleteness of the identified results.

2.3.2 Quantitative analysis for KC priority

KC flow-down is a qualitative method to identify KCs, which lacks the ability to prioritize the identified KCs. Therefore, some researchers have studied quantitative analysis approaches to determine which KCs have the most influence on the quality and cost of a product.

A variety of tools have been used for quantitative analysis of KC priority, such as Taguchi loss function, variation model, variation mode and effect analysis (VMEA), and variation risk management (VRMM). For instance, Tang et al. [21] used the Taguchi loss function to calculate the influence degree of the variations of characteristic candidates (CCs) on the product quality. The influence degree could determine the relative importance of CCs, and in turn, the relative importance could help identify new KCs. Lee and Thornton [22] proposed a variation model to calculate the importance degree of a part

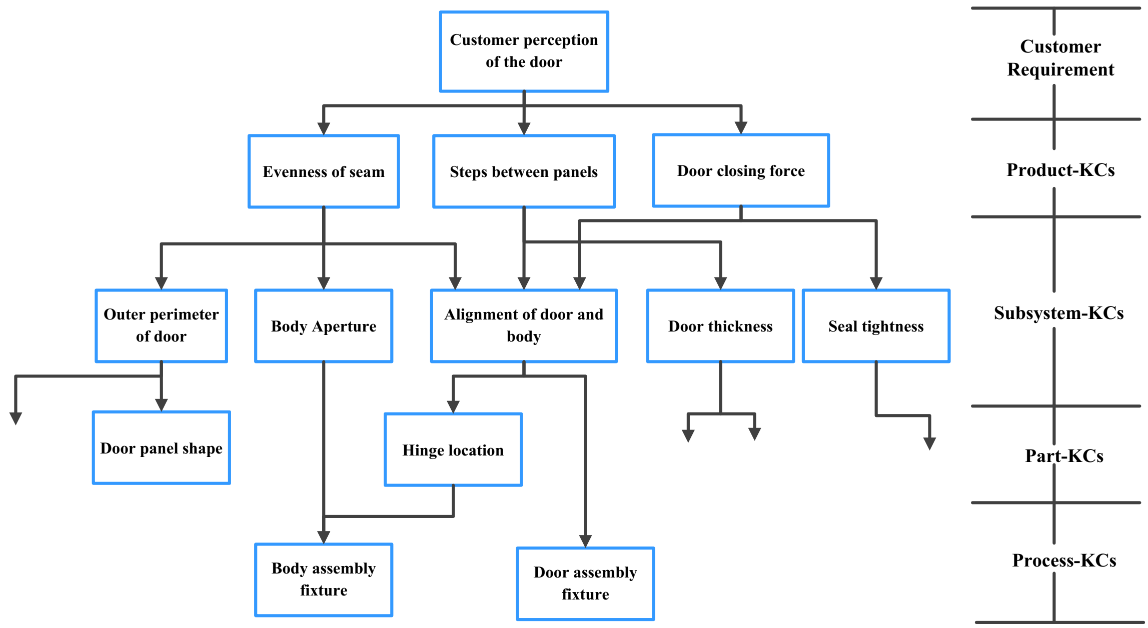


Fig. 2 KC flow down [8]

KC, which was dependent on the sensitivity to the variation of product quality. Recently, Estrada et al. [23] created a variation model to calculate the rework cost for KCs as a variable in function of the expected amount of material to be removed. This cost plus scrap cost was used to prioritize KCs running with low capability. The identified critical KCs could help engineers to develop solutions to eliminate what is causing KCs running with low capability. Etienne et al. [24] established a cost model for variation management to identify key process characteristics, which could support tolerance design, computer-aided process planning (CAPP), and computer-aided inspection planning (CAIP). As for VMEA, Chakhunashvili et al. [25] and Johansson et al. [26] used VMEA to analyze the sensitivity of product performance or quality to the variation of KCs, and the relative importance degree of KCs could be determined by the sensitivity degree. Similar to the VMEA method, Ibrahim et al. [27, 28] recently proposed a VRMM methodology to prioritize KCs and quantify their associated risk of variation.

The quantitative analysis of KCs can calculate the importance degrees of KCs, which helps designers to find which KCs have the most influence on the quality and cost of a product, and then focus on these KCs in the manufacturing production to improve the quality of the product. However, most of these methods are actually a process that uses a quantitative analysis tool to analyze and prioritize the potential KCs and further finalize the real KCs, but it does not discuss how to obtain the potential KCs (KC candidates) before evaluating them.

In addition, both the KC qualitative and quantitative analysis methods are mainly focused on the KPCs, but less on KDCs. In view of the above problems, it is necessary to study a systematic method to identify KDCs for complex product development.

3 Proposed methodology

3.1 The framework for identifying KDCs

Figure 3 presents our framework for the identification of KDCs for complex product designs. It includes two main phases: (1) construction of PKCN and (2) identification of KDCs for application.

In the first phase, previous design knowledge, case studies, and data usages are main sources for the construction of PKCN. The construction quality of PKCN not only depends on the construction techniques but also on the reference data quality. PKCN is a network structure, which includes PDS, KDC candidates, and the relations between them. At the end of phase 1, the PKCN is listed and distributed in a database, which can be represented in Fig. 4 as the PDS attribute table (a), KDC candidate attribute table (b), and characteristic (PDS and KDC candidates) relations table (c).

The PKCN contains potential KDCs for a wide range of application scenarios and forms the basis of the identification of KDCs for a new product development. Therefore, in the second phase of identification of KDCs for an application, the PDS is used as a searching and retrieving item to find a small set of KDCs from the PKCN for guiding a new product design. In Fig. 3, when a new project PDS is created, first check whether the PDS item is a new one within a PDS attribute table (a); if so, a new design pattern is required; if not, continue check whether its value is in the planned value range; if so,

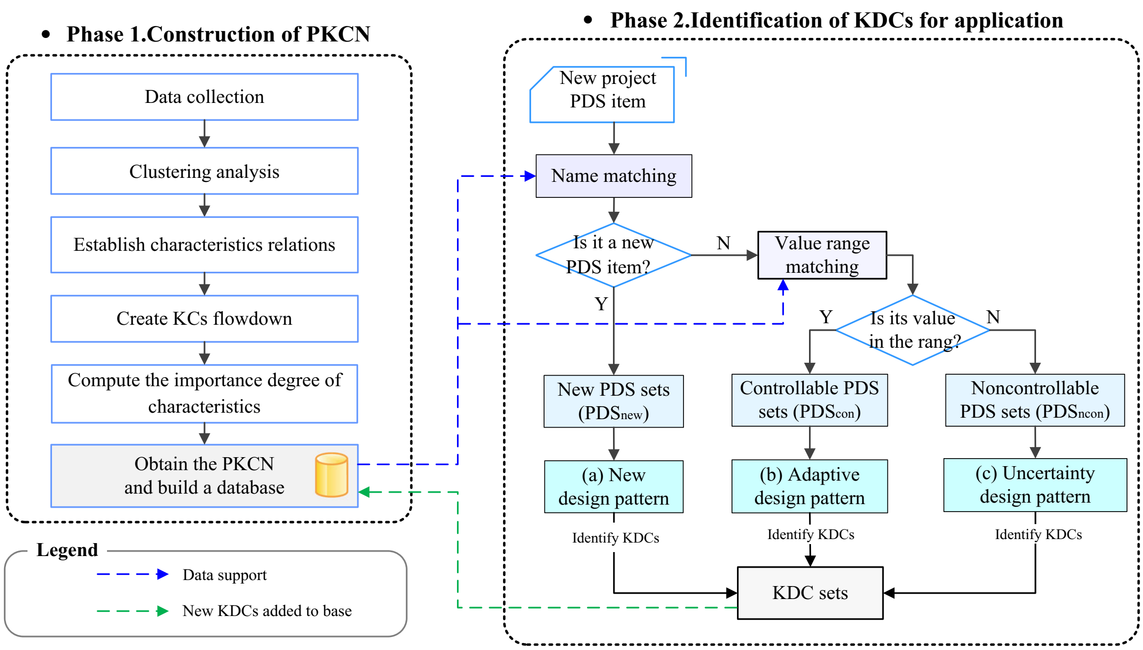


Fig. 3 The framework for the identification of KDCs

an adaptive design pattern is required. Otherwise, an uncertainty design pattern is required. The three design patterns will identify the KDCs associated with PDS, and finally, a set of KDCs will be obtained by getting union of the three identified KDCs. The three design patterns are shown in Fig. 5, and their characteristics are as follows.

(a) New design pattern

In this pattern, the new project PDS item (PDS_{new}) is a new one that does not exist in the PDS attribute table. Some new KDCs will be identified based on the theory of system design [30], axiomatic design [31], and FBS [32], which usually produces a new conceptual scheme. This pattern is generally accounted for 20% of the total design [33], which is a difficult one because there is no similar instances that can be used for reference. This paper will not discuss it in detail.

(b) Adaptive design pattern

In this pattern, the new project PDS item (PDS_{con}) exists in the PDS attribute table and its value is in the planned range, which can be controlled. Some existing KDCs which are the most sensitive to PDS changes need to be identified from the PKCN, and the evolution rules of PDS to KDCs need to be analyzed based on the existing instance data and expert knowledge. Then, designers focus on these identified KDCs

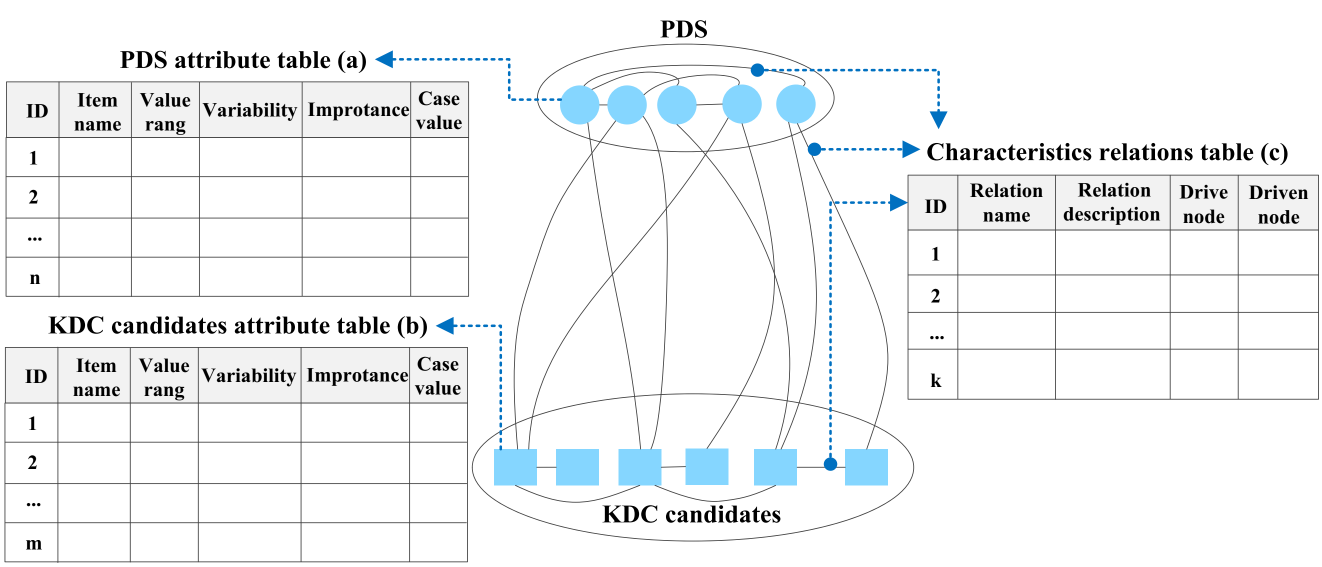


Fig. 4 The PDS-KDC Candidates Network (PKCN)

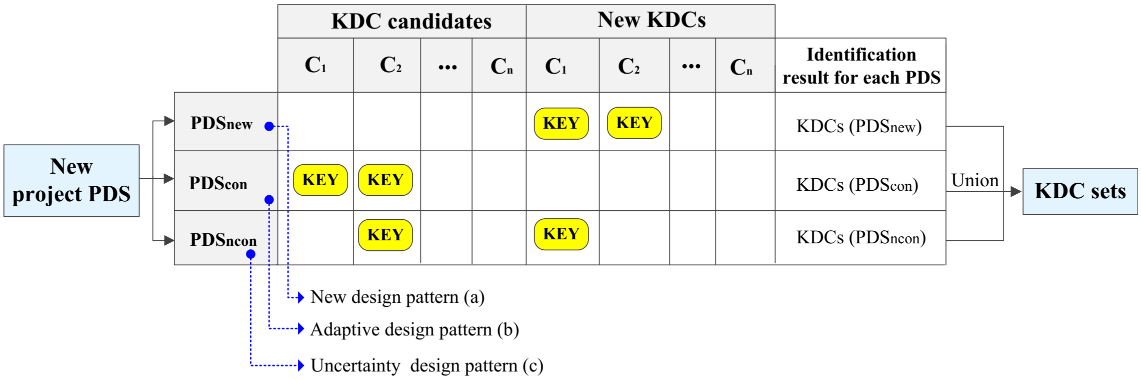


Fig. 5 The new, adaptive, and uncertainty design patterns

and determine their values to form a design scheme. This pattern is generally accounted for 80% of the total design [33], which is a common case. This paper will focus on this pattern, and the details are shown in Sect. 3.3.

(c) Uncertainty design pattern

In this pattern, the new project PDS item (PDS_{ncon}) exists in the PDS attribute table but its value is not in the planned range, which cannot be controlled. The identified KDCs in this pattern may be new KDCs (by a new design pattern), or some may be existing KDCs (by an adaptive design pattern). This pattern is a quite difficult one, and this paper will not discuss it in detail.

3.2 Construction of PKCN

The PKCN is the basis for the identification of KDCs for complex product development. It mainly includes the following five steps:

- Step 1: data collection

Before collecting data, we need to determine what data need to be collected. The product data is produced in the product development process, as shown in the Fig. 6.

This paper focuses on the product design; thus, the following data need to be collected in our research, as shown in Table 1.

After determining what data need to be collected, it is necessary to solve the problem of how to collect the data. A standardized data collection form (DCF) is used to solve this problem, as shown in Fig. 7. Based on DCF, the product and part designers use the Excel tool to collect data in line with their experience, knowledge, and existing instance data, respectively.

In the DCF, the "item name" is determined by experienced product/part designers based on their actual work experience; the "case value" and "unit" are determined by referring to the existing product family instances; the "value range" is the planned product family design scope by referring to the minimum and maximum values of product family instances; the "variability" is determined after completion of step 2, which shows whether the characteristic (PDS and DC) is changeable or not within the planned value range. It can help the designers to establish the characteristic relations in step 3; the "importance" is determined after completion of step 5, which shows the importance degree of each characteristic in a characteristics network.

- Step 2: clustering analysis

Based on the collected DCF, using cluster analysis tool to identify variant and invariant PDS and DCs (based on how far between characteristics), helps designers to know which characteristics change within the planned value range. The clustering analysis method can refer to [34–36]. In this step, a calculation program can be written based on the cluster analysis method, which can automatically calculate the distance between the characteristics when importing the collected excel table, then, updating

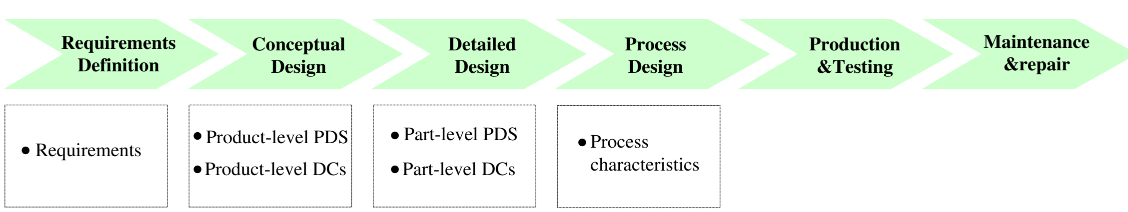


Fig. 6 Product data in the product development process

Table 1 The collected data in our research

Item	Description	Phase
Product-level PDS	The design specification of a product design, which entails design inputs, design objectives, and design constraints.	Conceptual design
Product-level DCs	A set of product-level design characteristics, which entail performance characteristics (to highlight a product's overall performance) and structural characteristics (to highlight a product's overall structure and layout features).	Conceptual design
Part-level PDS	The design specification of a part design, which comes from three aspects: product-level PDS, product-level DCs, other part-level DCs.	Detailed design
Part-level DCs	A set of part-level design characteristics, which entails performance characteristics (to highlight a part's performance) and structural characteristics (to highlight a part's geometry features).	Detailed design

the column (variability) of product/part DCF based on the calculated results, respectively.

- Step 3: establishing characteristic relations

If only collecting PDS and DCs without analyzing and establishing the relations between them, it will not be able to find out which DCs are sensitive to PDS changes and have the greatest influence on other DCs; as a result, it will be unable to effectively support subsequent product design. Therefore, based on the DCF, the product and part designers need to establish the relations among PDS, the relations between PDS and DCs, and the relations among DCs, respectively. Characteristic relations matrix (CRM) is used to solve this problem, as shown in Fig. 8.

The items under the term of PDS and DCs come from the DCF. Experienced designers determine the relations by function equations, charts, and semantic descriptions. The following three questions can be used to ask the designers when they encounter the variant characteristics (PDS and DCs) that are determined in step 2: (1) why these characteristics need changes, (2) how to achieve the needed changes, and (3) what will be affected by these changes. Answers to these questions could help designers to establish the characteristic relations.

After establishing the characteristic relations, product and part designers can transform the CRM into a characteristic relations network (CRN), as shown in Fig. 9. The CRN can visually show the characteristics and their relations on a graph, which helps designers to more easily check the established characteristic relations (in step 4). In this step, network analysis tools (e.g., Pajek [37], etc.) can be used to establish a CRN.

- Step 4: creating KC flow-down

The product/part-level CRN in the above steps is constructed by the designers with field expert knowledge. However, complex products involve many disciplines and have a lot of interaction between product and parts; whether the product/part-level CRN is accurate and complete or not still needs to be further verified, which requires KC flow-down, as shown in Fig. 10.

Firstly, a cross-functional team consisting of customers, product designers, part designers, suppliers, and college

Data collection form (DCF)

Type	Type	Item name	Unit	Value range	Variability	Importance	Value		
							Case1	···	Case k
PDS	Design inputs	PDS 1							
		PDS 2							
	Design objectives	PDS 3							
		PDS 4							
	Design constraints	...							
		PDS n							
Design characteristics (DCs)	Performance characteristics	DCs 1							
		DCs 2							
	Structural characteristics	...							
		DCs m							

Fig. 7 The data collection form (DCF)

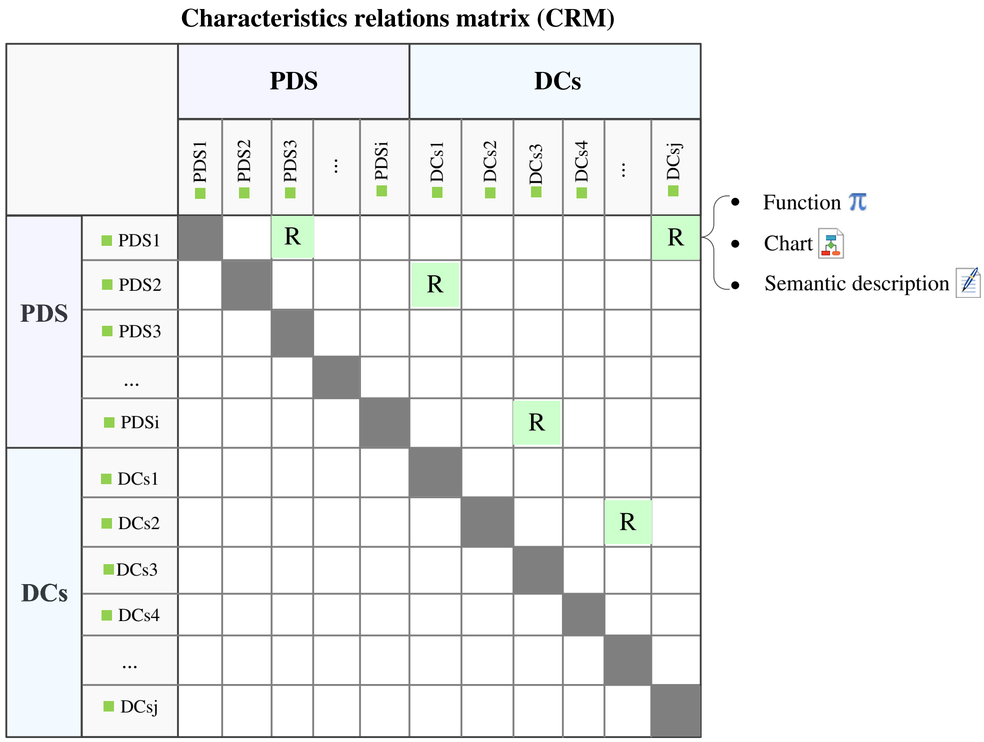


Fig. 8 The characteristic relations matrix (CRM)

professors is established. Then based on product/part-level CRN, we can use a team approach to create KC flow-down, which in turn determines PDS, product-level KDC candidates, and part-level KDC candidates. The determination of PDS can be done by referring to all product-level PDS and some part-level PDS; product-level KDC candidates can be determined by referring to all product-level DCs and some part-level PDS, and similarly, part-level KDC candidates can be found by referring to all part-level DCs and some other part-level PDS. It is essentially a process that product/part-level CRN is analyzed and verified by the cross-functional team and finally forms a relatively complete and accurate PKCN of the

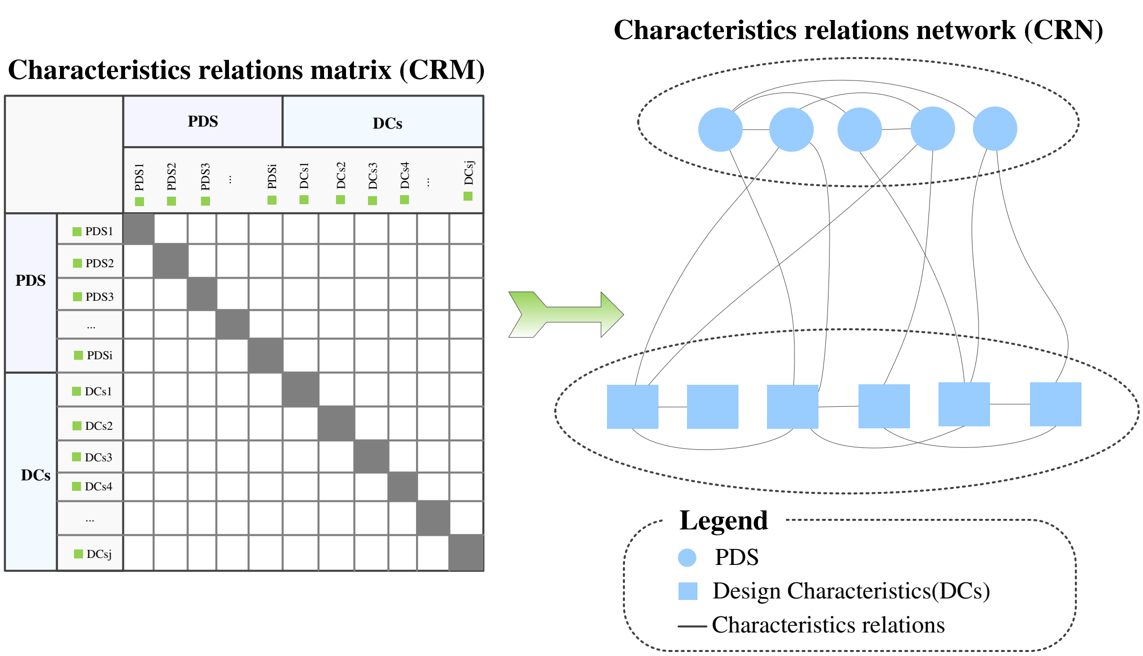


Fig. 9 The conversion of CRM to CRN

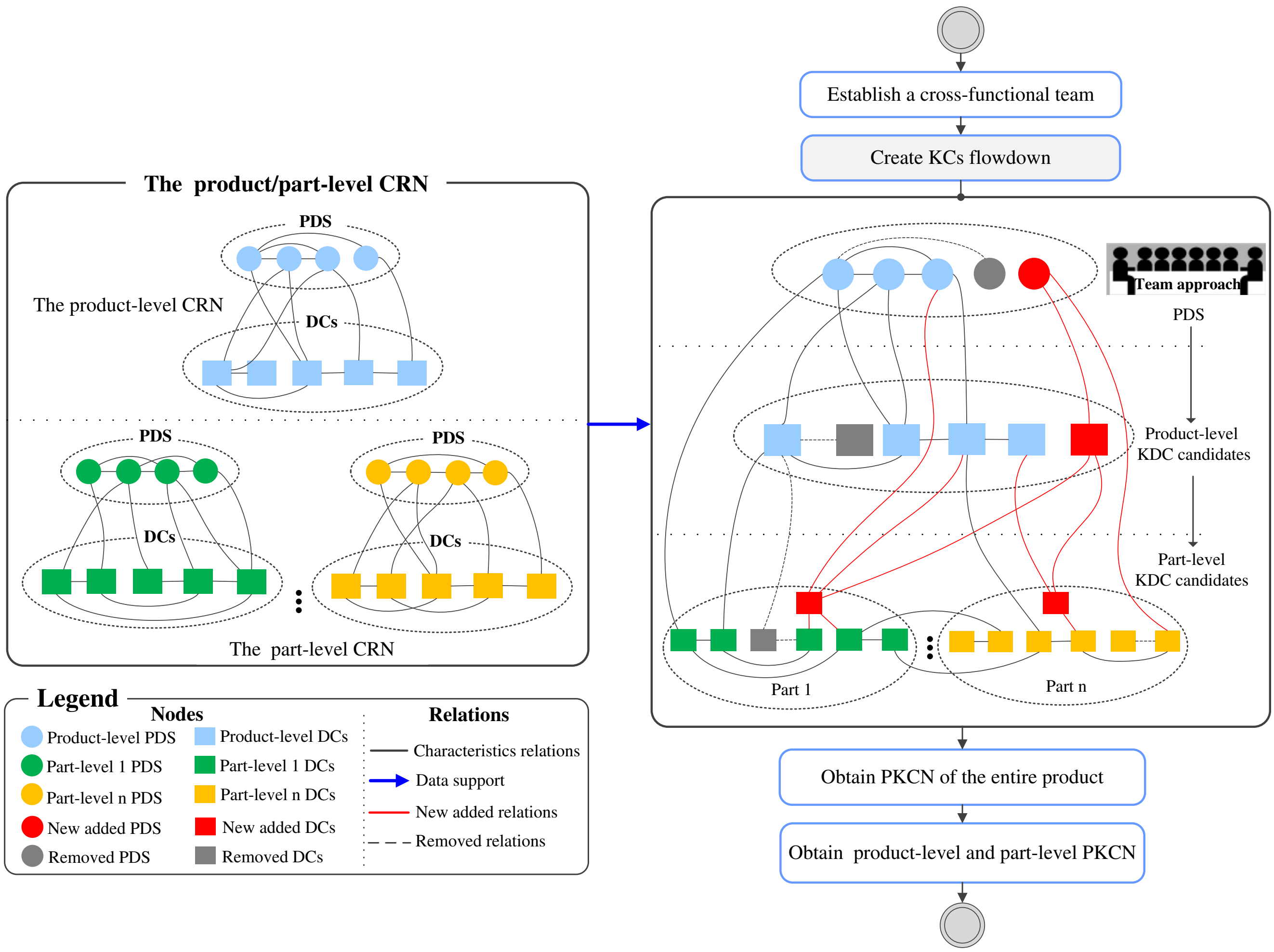


Fig. 10 KC flow-down approach

entire product. In this process, the data check should be considered in the following two aspects:

4 Is the item name consistent?

Due to a complex product involving many disciplines, it may lead to the heterogeneity of the name of items (PDS and DCs); as a result, it will affect the communication of designers in various fields. Therefore, it is necessary to eliminate the heterogeneity among the collected data and form a consistent item name.

5 Is the item correct?

Because the collected data and established relations are done independently by the designers without checking with other designers, the constructed product/part-level CRN may be incorrect, which reflects in the following two aspects: (1) omission and (2) unnecessary. For the former, for instance, a PDS_i of a part-level CRN comes from a DCs_i of the product-level CRN, while the product-level CRN does not have this DCs_i; thus, the product-level CRN needs to add this DCs_i when the cross-functional team confirms it. As for the unnecessary aspect, for instance, there is a PDS_i in a product-level CRN, while the product/part-level design does not need this PDS_i; thus, the product-level CRN could remove this PDS_i when the cross-functional team confirms it.

Through the data check by the cross-functional team, the designers modify incorrect information and finally form a relatively complete and accurate PKCN of the entire product that includes three layers: PDS, product-level KDC candidates, and part-level KDC candidates. Then, the product and part designers update the DCF, CRM, and CRN based on the checked results, respectively, and then obtain the product-level PKCN and part-level PKCN. The product-level PKCN is a database to support product-level design, which includes two layers: product-level PDS and product-level KDC candidates. While the part-level PKCN is a database to support part-level design, which includes two layers: part-level PDS and part-level KDC candidates.

- Step 5: computing the importance degree of characteristics

PKCN is a network structure, in which each characteristic (a PDS or KDC candidate) is inter-related in a net and shows the network properties, such as small world [38] and scale free [39]. However, not all characteristics are equally important, and some may have strong connections to other characteristics, while others may have weak connections. In order to allow designers to focus on a small set of critical design characteristics in the product development, it needs to analyze the importance degree of characteristics.

Through literature study, network analysis approach is suitable for solving the calculation of importance degree of characteristics. It not only can calculate the importance degree of each characteristic but also can handle a large number of data, which is convenient for computerization. In literature, the network analysis approach has emerged as a key method for analyzing a wide variety of complex systems such as social science [40], information engineering [41], and biological science [42]. Recently, researchers have applied the network theory into the development of mechanical products. For instance, Sosa et al. [43] and Fan et al. [44] built a component network and used it to guide the module division through calculating the degree of modularity of components. Batallas et al. [45], Braha et al. [46], and Dan et al. [47, 48] constructed a product development network to identify a "core team" by analyzing the information flow of design team and then to assign the core team to carry out the work for improving the efficiency of multi-disciplinary design.

From the network theory, we know that the network analysis is a quantitative analysis approach based on statistical theory, and the importance degree of nodes in the PKCN of an entire product can be calculated to provide an objective reference to designers in the subsequent KDC identification process. The formulas are known as centrality measures and described in the Table 2 [45].

In this step, the importance degree of characteristics (PDS and KDC candidates) of PKCN of the entire product can be calculated automatically by using the tools from Pajek [37]. Then, the product and part designers need to update the column (Importance) of product/part DCF based on the calculated results, respectively.

Finally, the product/part-level PKCNs need to be represented in a database, which includes a PDS attribute table, a KDC candidate attribute table, and a characteristic relations table. The establishment of PDS and KDC candidates' attribute table can refer to the DCF, and the characteristic relations table can refer to the CRM.

5.1 Identification of KDCs for adaptive design

Adaptive design pattern is a common one in the product development. Its core is to identify these KDCs that are sensitive to PDS changes and then focus on them to rapidly develop an adaptive scheme. Figure 11 shows the identification process of KDCs for product level (the process for part level is similar), including the following three steps.

- Step 1: determine which PDS need to be analyzed

Firstly, obtain a set of controllable PDS that have no new PDS item, and their item values are in the planned value ranges by name and value range matching. Then, designers determine which controllable PDS needs to be analyzed, by considering the following two aspects:

6 Is it similar to existing PDS?

A new project PDS (controllable PDS) is first used as a searching item to match PDS attribute table and calculate the similarity between the controllable PDS and existing PDS instances using similarity reasoning technique [49]. Then determine whether there is a case that its similarity exceeds the given threshold (e.g., 0.8) in the database; if so, it can be chosen as an initial design scheme; if not, a case with the highest similarity should be chosen. These controllable PDS not similar to existing ones should be further analyzed. Then, these KDCs which are sensitive to the variation of the chosen PDS need to be identified, and designers focus on them to develop an adaptive design scheme.

7 Is it important?

Not all PDSs are equally important, and some may have strong connections to KDC candidates, while others may have weak connections. Therefore, we can give priorities to these PDS with higher importance degree for further analysis.

- Step 2: determine which KDC candidates need to be analyzed

Input controllable PDS sets use product/part PKCN as database and then apply the BFS to identify KDC candidates associated with PDS. The BFS technology can refer to [50].

In general, a PDS will affect a number of KDC candidates, if each KDC candidate needs to be analyzed in detail, which will be time-consuming and laborious. Therefore, we can choose these KDC candidates with higher importance degree for further analysis, because these KDC candidates have strong connections to PDS and other KDC candidates.

Table 2 The calculation formulas of importance degree

Indexes	Description	Formula
Degree centrality	Degree centrality can be measured as the number of outlinks connecting a node to its neighbors or as the number of inlinks that a certain node is receiving from adjacent nodes.	$D(n_i) = \frac{\sum_{\forall j \neq i} x_{ij}}{n-1}$ $D'(n_i) = \frac{D(n_i)}{\frac{1}{n}\sum_{i=1}^{n} D(n_i)}$ $x_{ij} = \begin{cases} 1, if\ i is\ \ indicent\ \ to\ j. \\ 0, if\ i is\ \ not\ \ indicent\ \ to\ j. \end{cases}$ $D'(n_i)$ is standardized degree centrality of node i. n is the number of nodes.
Closeness centrality	Closeness centrality reflects how close an actor is to other actors in a network, which can be measured as a function of geodesic distance that is a shortest path between two nodes.	$C(n_i) = \frac{n-1}{\left(\sum_{j=1, i\neq j}^{n} d(n_i, n_j 1)\right)}$ $C'(n_i) = \frac{C(n_i)}{\frac{1}{n}\sum_{i=1}^{n} C(n_i)}$ $C'(n_i)$ is standardized closeness centrality of node i. $d(n_i, n_j)$ is geodesic between i and j.
Betweenness centrality	Betweenness centrality focuses on these nodes that lie in the path between other nodes, which have control over knowledge flow since information must travel through them.	$B(n_i) = \frac{\sum_{j<k, i\neq j, i\neq k} \frac{g_{jk}(n_i)}{g_{jk}}}{\frac{(n-2)(n-1)}{2}}$ $B'(n_i) = \frac{B(n_i)}{\frac{1}{n}\sum_{i=1}^{n} B(n_i)}$ $B'(n_i)$ is standardized betweenness centrality of node i. $g_{jk}(n_i)$ is the number of geodesics linking j and k that contains i in between. G_{jk} is the total number of geodesics linking j and k.
Important degree	The important degree of nodes is the weighted sum of the degree centrality, closeness centrality, and betweenness centrality.	$I(n_i) = w_1 D'(n_i) + w_2 C'(n_i) + w_3 B'(n_i)$ $w_1 + w_2 + w_3 = 1$ w_1 is weight of degree centrality. w_2 is weight of closeness centrality. w_3 is weight of betweenness centrality.

- Step 3: identify the most sensitive KDC candidates to PDS changes as final KDCs

A selected KDC candidate based on the importance degree is an "important node" (with higher degree in closeness and betweenness) in the PKCN, but whether it is sensitive to PDS changes needs further analysis and evaluation, because there are some KDC candidates which are closely linked with PDS and other KDC candidates, but they are not sensitive to PDS changes. Thus, we need

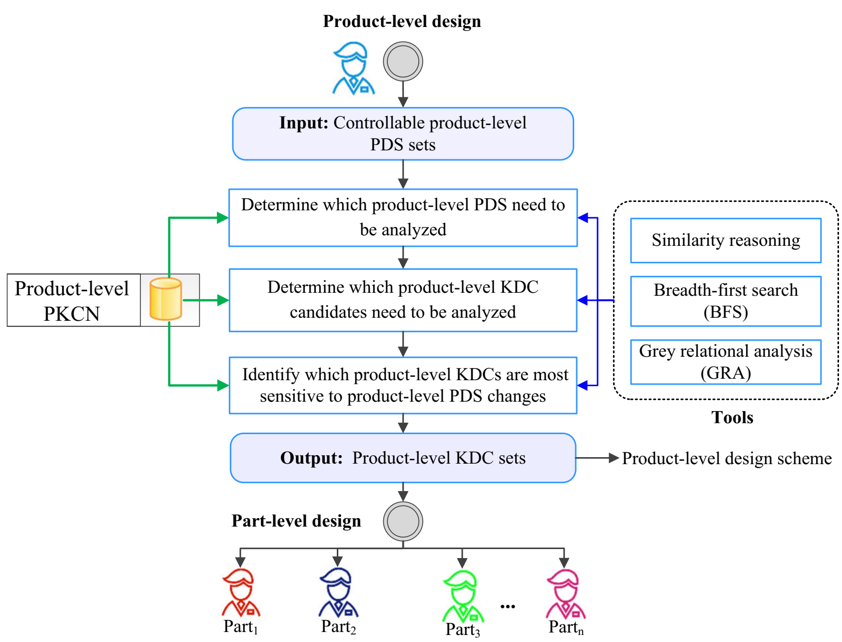


Fig. 11 The identification process of KDCs for adaptive design pattern

to pay more attention to these KDCs which not only have a higher importance degree but also are sensitive to PDS changes.

Complex products involve multi-disciplines; thus, their design is usually difficult to establish sensitivity analysis model. Through literature study, the gray relational analysis model, a kind of order relation model, can describe the strength of relations between factors by the gray relational degree [51, 52]. It can be used to deal with engineering problems such as factor analysis and is suitable for dealing with poor, incomplete, and uncertain information systems [53–55]. Therefore, based on the selected KDC candidates, this research uses the gray relational analysis approach to further identify these KDCs that are sensitive to PDS changes. The steps are as follows.

Firstly, use the selected PDS in step 1 as the target sequence, that is, $X(i) = \{X_1, X_2,..., X_n\}$, n is the number of the PDS items, and use the selected KDC candidates in step 2 as the comparable sequence, that is, $Y(j) = \{Y_1, Y_2,..., Y_m\}$, m is the number of the KDC candidates.

Secondly, calculate the gray relational degree of each KDC candidate (Y_j) to the ith PDS (X_i), that is, $\gamma(X_i,Y_j)$, and the calculation formulas of gray relational degree can refer to [51, 52].

Thirdly, calculate the weighted sum of the gray relational degree of the each KDC candidate (Y_j) to the all PDS, that is, $\zeta(X,Y_j)$ which can be calculated by Eq. (1).

$$\xi(X, Y_j) = \sum_{i=1}^{n} w_i \gamma(X_i, Y_j) \tag{1}$$

In Eq. (1), w_i is the weight of the ith PDS which is determined by experienced experts by reference to the importance degree of PDS, $\sum_{i=1}^{m} w_i = 1$.

The gray relational degree of a KDC candidate reflects the influence degree of the variation of PDS on it. If a KDC candidate has a relatively high gray relational degree, it shows that it is sensitive to the variation of PDS. Therefore, in this way, we can identify both sensitive and insensitive KDC candidates based on the gray relational degree.

Finally, take these sensitive KDC candidates as the final KDCs which their gray relational degree is larger than a threshold δ. δ is a threshold determined by the expert's experience. In this paper, δ is 0.7.

Based on the KDC identification results, we can make a corresponding design: (1) For the sensitive KDCs, the evolution rules of PDS to KDCs need to be analyzed based on the existing instances and expert knowledge, and then, designers focus on these KDCs to determine their values. (2) For the insensitive KDCs, because they are stable enough, we can temporarily reuse the value of existing similar instances.

8 Case study

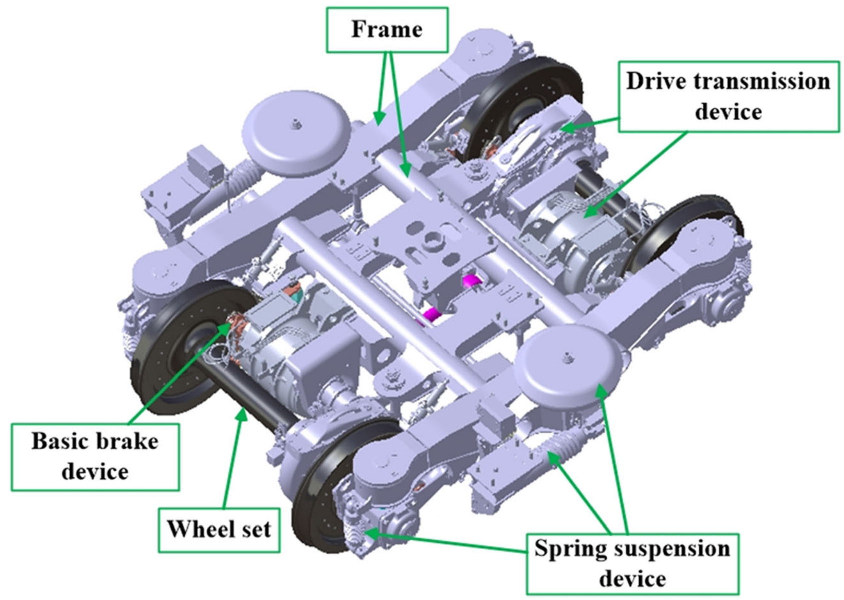
A bogie is a running unit of electric multiple unit (EMU) of a train; it is equivalent to car's chassis and wheels and has the functions of guide, bearing, vibration, traction, and braking. The bogie is composed of a frame, wheel sets, spring suspension device, drive transmission device, and basic brake device, as shown in Fig. 12. The rationality of the design of EMU's bogie determines the performance and quality of the whole vehicle, and thus, it is a key component in the development of EMU. The development of a new bogie is always started from the existing ones and makes an adaptive design. However, when a new project PDS is put forward, such as maximum speed is decreased from 250 to 220 km/h, deceleration emergency braking (EB) is increased from 0.9 to 0.95 m/s^2, inexperience designers often do not know which parts should be changed, and cannot predict what the changes will affect. Besides, when focusing on a product/part, they do not know what key characteristic should be used as key design variables. They can effectively lead to an adaptive design or rapidly develop a technical document (including the key requirements of performance, structure, interface, etc.) for suppliers; thus, it is difficult for enterprises to control the delivery lead time and further the entire R&D cycle of a bogie.

In view of this, it needs to construct a KDC-based design knowledge base to guide the development of bogie, so that when a PDS has changed, the designers will be able to quickly find the bogie's KDCs, which will be affected by the PDS changes and focus on designing them. We conducted a case study within a bogie department of OEMs of EMU in China, by building a KDC-based design knowledge base to support the rapid development of a bogie. Here, we only discuss the KDC identification of a bogie.

8.1 Construction of PKCN

Firstly, based on the planned product family of EMU, we collected the existing six product instances that located in the target segment. Then, we worked with the bogie product-level designers and part-level designers (e.g., frame, wheel sets, brake, etc.) for 2 weeks, guided them to collect PDS and DCs, and then completed the DCF based on the existing six instances, respectively. Subsequently, we clustered the collected PDS and DCs into variant or invariant ones by using cluster analysis tool and then updated the DCF of product and its parts based on the calculated results, respectively. After that, based on the DCF, the product-level and part-level designers established the characteristic relations and completed the CRM and CRN for 2 weeks. Upon that, we spent 2 weeks to create KC flow-down, which in turn determined PDS and product-level and part-level KDC candidates. In this process, first, the product-level and part-level CRNs were analyzed and verified by the cross-functional

Fig. 12 The units of bogie



team via brainstorming. Second, designers modified incorrect data and finally formed a relatively complete and accurate PKCN of the entire bogie. The partial results are shown in Fig. 13. The importance degree of the nodes (PDS and KDC candidates) of PKCN was calculated by using the method of Sect. 3.2. The partial results of the calculation are shown in Table 3; the third column is the degree centrality of each node; the fourth column is the betweenness centrality of each node; the fifth column is the closeness centrality of each node, and the sixth column is the importance degree of each node.

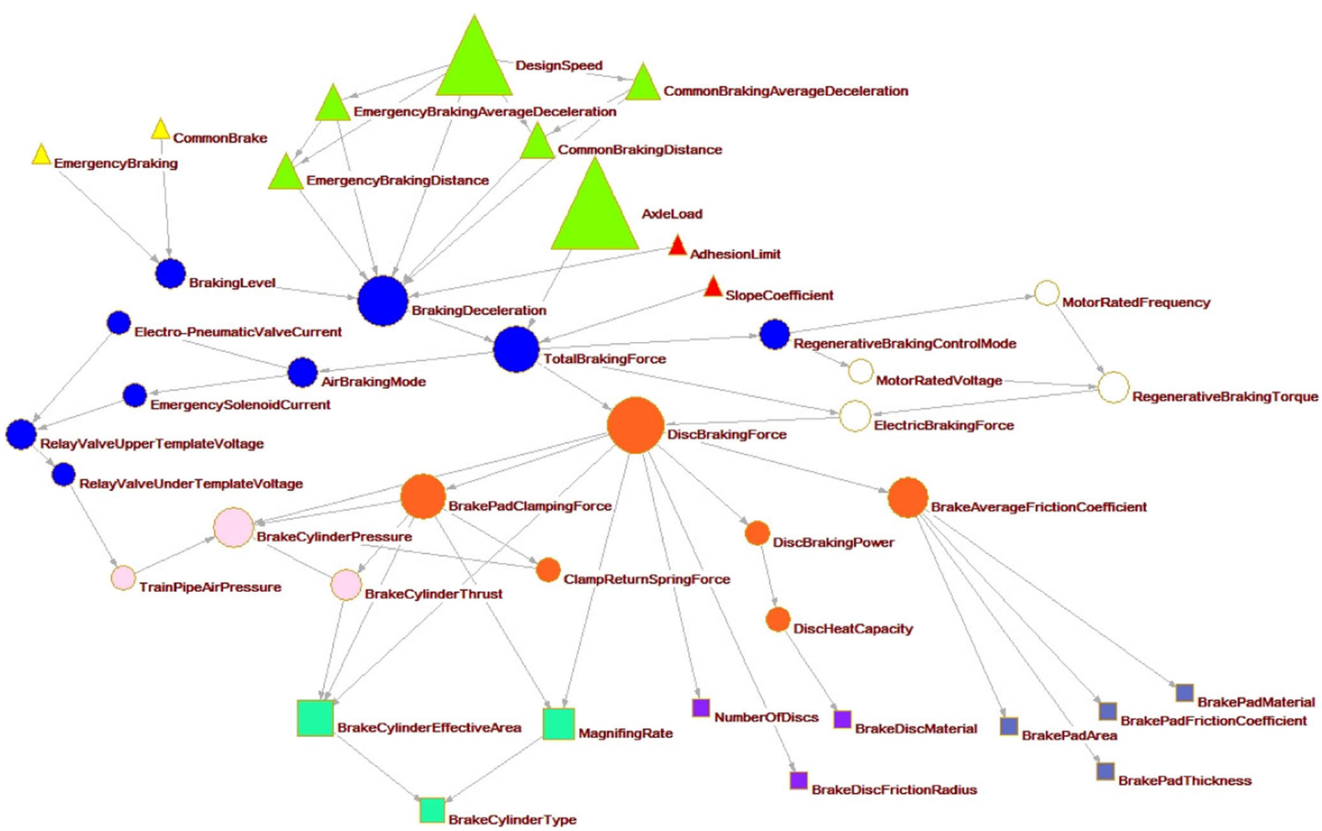


Fig. 13 The partial PKCN of the bogie

Table 3 The importance degree of the partial bogie's KDC candidates

Types	Node name	Degree	Betweenness	Closeness	Importance sum
PDS	Maximum speed	20.92	0	1.48	8.81
	Axle load	20.00	0	1.61	8.48
	Deceleration EB	2.15	19.41	1.31	7.08
	Starting acceleration	1.23	8.71	1.03	3.41
	Track gauge	1.23	0	0.58	0.67
Product-level KDC candidates	Disc braking force	3.39	23.38	1.10	8.70
	Tractive force	2.46	18.60	1.07	6.66
	Brake pad clamping force	1.85	4.04	0.87	2.21
Part-level KDC candidates	Brake cylinder effective area	1.23	0.85	0.87	1.01
	Magnifying power	0.92	1.06	0.87	0.95
	Traction motor power	1.85	2.17	1.38	1.81
	Traction motor torque	1.54	1.00	1.37	1.33
	Wheel seat diameter	1.23	0.06	1.16	0.86

8.2 Identification of KDCs for adaptive design

Use the new project PDS to match the constructed PKCN, and identify its design patterns among new design, adaptive design, and uncertainty design. It is with adaptive design, and therefore, the identification of product-level KDCs starts (part-level KDCs are identified similarly, which is not discussed in this case study).

9 Determine which PDS needs to be analyzed

In the development of a new project, the new project PDS (controllable PDS) including items "maximum speed = 220 km/h" and "deceleration EB = 0.95 m/s^2" was not similar to existing PDS instances, but the two items were in the planned value range (maximum speed 160–350 km/h and deceleration EB 0.85–1.0 m/s^2). In addition, the two PDS items had a higher importance degree; thus, they were chosen for further analysis.

10 Determine which KDC candidates need to be analyzed

With PDS (maximum speed and deceleration EB) as input and product-level PKCN as database, use BFS to identify KDC candidates associated with PDS, which are shown in Table 4.

In order to help product designers to focus on a small set of KDC candidates to save their time and energy, these KDC candidates with higher importance degree were chosen for further analysis. Based on the experience of product designers, this case study selected five product-level KDC candidates associated with maximum speed and deceleration EB to analyze. The five product-level KDC candidates are disc braking force, brake cylinder pressure,

Table 4 The product-level KDC candidates associated with PDS

Types	Node name	Importance sum
PDS	Maximum speed	8.81
	Deceleration EB	7.08
Product-level KDC candidates	Disc braking force	8.70
	Brake pad clamping force	2.21
	Brake cylinder pressure	0.87
	Brake cylinder thrust	0.59
	Clamp return spring force	0.63
	Train pipe air pressure	0.46

Table 5 The data for gray relational analysis

Types	Node name	Case 1	Case 2	Case 3	Case 4	Case 5	Case 6
PDS	Maximum speed (km/h)	140	160	200	250	300	350
	Deceleration EB (m/s^2)	1.2	1.12	1.0	0.9	0.8	0.75
Product-level KDC candidates	Disc braking force (kN)	85.68	80.6	77.2	72.59	68.4	60.63
	Brake pad clamping force (kN)	46.06	44.2	41.1	38.5	36.1	34.4
	Brake cylinder pressure (kPa)	456.6	438.2	418.5	386.9	355.3	330.5
	Brake cylinder thrust (kN)	5.38	5.02	4.68	4.33	4.04	3.72
	Clamp return spring force (kN)	630	630	630	630	630	630

brake pad clamping force, brake cylinder thrust, and clamp return spring force. Most of them are performance-related design parameters.

11 Identify the most sensitive KDC candidates to PDS changes as final KDCs

The selected five KDC candidates based on the importance degree were important node, but whether they were sensitive to PDS change was further analyzed and evaluated by using the gray relational analysis approach.

The data for gray relational analysis are shown in Table 5. Two PDS items are maximum speed and deceleration EB. Five product-level KDC candidates are disc braking force, brake cylinder pressure, brake pad clamping force, brake cylinder thrust, and clamp return spring force.

The gray relational degrees of the five product-level KDC candidates were calculated by using the method given in Sect. 3.3. The calculation results are shown in Table 6; the second column is the gray relational degree of the five product-level KDC candidates for the maximum speed; the third column is the results for deceleration EB, and the fourth column is the sum of the gray relational degree of each product-level KDC candidates.

From Table 6, according to the gray relational degrees of product-level KDC candidates, we ranked them in the following order: brake cylinder thrust > disc braking force > brake pad clamping force > brake cylinder pressure > clamp return spring force.

The gray relational degrees indicated the sensitivity of KDC candidates to the PDS changes. From them, we identified these sensitive KDC candidates (sum > 0.7 the threshold) as the final KDCs based on the gray relational degrees, which are brake cylinder thrust, disc braking force, brake pad clamping force, and brake cylinder pressure. The identification results were agreed by the experienced engineers.

After identifying the KDCs, designers made an adaptive design based on the results: (1) For the sensitive KDCs (e.g., brake cylinder thrust, disc braking force, brake pad clamping force, and brake cylinder pressure), the evolution rules of PDS to KDCs were analyzed resulting in some functions and empirical formulas, and then, designers focused on these KDCs to determine their values based on the evolution rules. (2) For the insensitive KDCs (e.g., clamp return spring force), because it is stable enough, designers temporarily reused the value (630 kN) of existing instances.

12 Conclusions

This paper presents a systematic identification framework to support KDC identifications with better completeness and diverse usages. Firstly, a PKCN is established by using the existing product instance data, a cluster analysis tool, KC flow-down, and a network analysis approach, which will be used as a basis for identifying KDCs for application. Then, the KDC identification for adaptive design pattern is developed in detail. Similarity reasoning technique, BFS, and gray relational analysis approach are used to identify which KDCs are most

Table 6 The results of gray relational analysis of product-level KDC candidates

Node name	Maximum speed	Deceleration EB	Sum	Result
Brake cylinder thrust (kN)	0.750	0.825	0.780	Sensitive KDCs
Disc braking force (kN)	0.710	0.755	0.728	
Brake pad clamping force (kN)	0.708	0.734	0.718	
Brake cylinder pressure (kPa)	0.707	0.731	0.717	
Clamp return spring force (kN)	0.579	0.543	0.565	Insensitive KDCs

sensitive to PDS changes, which helps designers to focus on these KDCs and rapidly develop a design scheme.

The case study of the KDC identification for an EMU's bogie shows that the proposed method is feasible and effective. This research mainly focuses on the identification of KDCs in the design phase, which makes a complementary contribution to the identification of KPCs in the manufacturing and maintenance phases. In future research, we will study a systematic method of identification of KDCs and KPCs to support product life cycle development. The latest studies on identification of KPCs published by Estrada et al. [23] and Etienne et al. [24] can be used for reference.

So far, we have only studied the identification method of KDCs; the development of a computer-aided design tool for KDC identification is needed; therefore, as one of our future works, we will develop a KDC-based computer-aided design tool to better support the complex product adaptive design. Specifically, the tool will include a data management module for administrators and an adaptive design module for designers. In the module of data management, the JDBC programming technology will be used with an Oracle database to sort and manage the constructed product/part-level PKCN. Besides, cluster analysis and network analysis tools will be integrated into this module, which can realize the function of data clustering and data importance calculation. In the module of adaptive design, the Java programming technology is planned to be used to develop an operation interface to support multi-disciplinary coordination design. Similarity reasoning, BFS, and gray relational analysis tools will be integrated into this module, which can realize the demand-oriented complex product adaptive design.

Funding information This work is partially supported by the National Natural Science Foundation of China (Grant No. 51575461). It is also supported by The National Key Research and Development Program of China (Grant No. 2016YFB1200506).

References

1. Tseng MM, Jiao J, Merchant ME (1996) Design for mass customization. CIRP Ann-Manuf Technol 45(1):153–156
2. Peng G, Mao H, Wang H et al (2016) BOM-based design knowledge representation and reasoning for collaborative product development. J Syst Sci Syst Eng 25(2):159–176
3. Kocsis T, Negny S, Floquet P et al (2014) Case-based reasoning system for mathematical modelling options and resolution methods for production scheduling problems: case representation, acquisition and retrieval. Comput Ind Eng 77:46–64
4. Zhu GN, Hu J, Qi J et al (2015) An integrated feature selection and cluster analysis techniques for case-based reasoning. Eng Appl Artif Intell 39:14–22
5. Romli A, Setchi R, Prickett P et al (2016) Eco-design case-based reasoning tool: the integration of ecological quality function deployment and case-based reasoning methods for supporting sustainable product design Proceedings of the Institution of Mechanical Engineers Part B: Journal of Engineering Manufacture 0954405416668928
6. Yang F (2011) A key characteristics-based model for quality assurance in supply chain. Industrial Engineering and Engineering Management (IE&EM), 2011 I.E. 18Th International Conference on. IEEE 1428–1432
7. Dantan JY, Hassan A, Etienne A et al (2008) Information modeling for variation management during the product and manufacturing process design. Int J Interact Des Manuf 2(2):107–118
8. Thornton AC (1999) A mathematical framework for the key characteristic process. Res Eng Des 11(3):145–157
9. Zheng LY, McMahon CA, Li L et al (2008) Key characteristics management in product lifecycle management: a survey of methodologies and practices. Proc Inst Mech Eng B J Eng Manuf 222(8):989–1008
10. Boeing Commercial Airplane Group (1998) AQS D1-9000: advanced quality system tools
11. General Motors (1998) Key characteristics designation system reference manual
12. Chin KS, Zheng LY, Wei L (2003) A hybrid rough-cut process planning for quality. Int J Adv Manuf Technol 22(9–10):733–743
13. Mathieu L, Marguet B (2001) Integrated design method to improve producibility based on product key characteristics and assembly sequences. CIRP Ann-Manuf Technol 50(1):85–88
14. Zheng L, Zhu X, Liu R et al (2013) A novel algorithm of posture best fit based on key characteristics for large components assembly. Procedia CIRP 10:162–168
15. Dai W, Maropoulos PG, Zhao Y (2015) Reliability modelling and verification of manufacturing processes based on process knowledge management. Int J Comput Integr Manuf 28(1):98–111
16. Madrid J, Forslund A, Söderberg et al (2017) A welding capability assessment method (WCAM) to support multidisciplinary design of aircraft structures. Int J Interact Des Manuf 1–19
17. Rosyidi CN, Irianto D, Cakravastia A et al (2014) Optimization models for deriving optimum target of key characteristics. J Adv Manuf Syst 13(02):89–101
18. Whitney DE (2006) The role of key characteristics in the design of mechanical assemblies. Assem Autom 26(4):315–322
19. Mathieu L, Marguet B (2001) Integrated design method to improve producibility based on product key characteristics and assembly sequences. CIRP Ann-Manuf Technol 50(1):85–88
20. Sivasakthivel T, Murugesan K, Thomas HR (2014) Optimization of operating parameters of ground source heat pump system for space heating and cooling by Taguchi method and utility concept. Appl Energy 116:76–85
21. Tang W, Li Y, Yu J et al (2014) A systematic top-down approach for the identification and decomposition of product key characteristics. Proceedings of the Institution of Mechanical Engineers Part B. J Eng Manuf 228(10):1305–1313
22. Lee DJ, Thornton AC (1996) The identification and use of key characteristics in the product development process. ASME Design Engineering Technical Conference
23. Estrada G, Shunk DL, Ju F (2017) Systematic continuous improvement model for variation management of key characteristics running with low capability. Int J Prod Res:1–18
24. Etienne A, Mirdamadi S, Mohammadi M et al (2017) Cost engineering for variation management during the product and process development. Int J Interact Des Manuf 11(2):289–300

25. Chakhunashvili A, Johansson PM, Bergman BLS (2004) Variation mode and effect analyze. Reliability and Maintainability, 2004 Annual Symposium-RAMS. IEEE:364–369
26. Johansson P, Chakhunashvili A, Barone S et al (2006) Variation mode and effect analyze: a practical tool for quality improvement. Qual Reliab Eng Int 22(8):865–876
27. Ibrahim IH, Chassapis C (2016) An interactive variation risk management environment to assess the risk of manufacturing variations. Int J Interact Des Manuf 11(3):597–608
28. Ibrahim IH, Chassapis C (2017) A variation risk management methodology for an interactive pharmaceutical design and manufacturing environment. Int J Interact Des Manuf:1–12
29. Zhang H, Han X, Li R et al (2016) A new conceptual design method to support rapid and effective mapping from product design specification to concept design. Int J Adv Manuf Technol:1–15
30. Pahl G, Beitz W, Feldhusen J et al (2007) Engineering design: a systematic approach. Nasa Sti/recon Technical Report A 89(133): 63–64
31. Suh N (2001) Axiomatic design: advances and applications. MIT-Pappalardo Series in Mechanical Engineering
32. Umeda Y, Ishii M, Yoshioka M (1996) Supporting conceptual design based on the function-behavior-state modeler. Artif Intell Eng Des Anal Manuf 10(10):275–288
33. Rezayat M (2000) Knowledge-based product development using XML and KCs. Comput Aided Des 32(5):299–309
34. Arabie P, Hubert LJ, Soete GD (1996) Clustering and classification. World Sci 3(53):109–128
35. Mirkin B (1998) Mathematical classification and clustering: From how to what and why. Classification, data analysis, and data highways. Springer Berlin Heidelberg 172–181
36. Hansen P, Jaumard B (1997) Cluster analysis and mathematical programming. Math Program 79(1):191–215
37. Batagelj V, Mrvar A (2014) Pajek. Springer New York
38. Dan B, Bar-Yam Y (2007) The structure and dynamics of complex product design. Understanding Complex Systems:40–71
39. Albert R, Barabási AL (2002) Statistical mechanics of complex networks. Rev Mod Phys 74(1):47
40. Wasserman S, Faust K (2010) Social network analysis methods and applications. Contemp Sociol:219–220
41. Faloutsos M, Faloutsos P, Faloutsos C (1999) On power-law relationships of the internet topology. ACM SIGCOMM computer communication review. ACM 29(4):251–262
42. Jeong H, Tombor B, Albert R et al (2000) The large-scale organization of metabolic networks. Nature 407(6804):651–654
43. Sosa ME, Agrawal A, Eppinger SD et al (2005) A network approach to define modularity of product components. ASME 2005 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference 435–446
44. Fan B, Qi G, Hu X et al (2015) A network methodology for structure-oriented modular product platform planning. J Intell Manuf 26(3):553–570
45. Batallas DA, Yassine AA (2006) Information leaders in product development organizational networks: social network analysis of the design structure matrix. IEEE Trans Eng Manag 53(4):570–582
46. Braha D, Bar-Yam Y (2004) The topology of large-scale engineering problem-solving networks. Social Science. Electron Publ 69(1 Pt 2):383–391
47. Dan B, Bar-Yam Y (2004) Information flow structure in large-scale product development organizational networks. Ind Organ 19(407012):244–253(10)
48. Dan B, Bar-Yam Y (2005) The statistical mechanics of complex product development: empirical and analytical results. Manag Sci 53(510005):1127–1145
49. Liao TW, Zhang Z, Mount CR (1998) Similarity measures for retrieval in case-based reasoning systems. Appl Artif Intell 12(4): 267–288
50. Cormen TH (2009) Introduction to algorithms. MIT press
51. Deng JL (1982) Control problems of grey systems. Syst Control Lett 1(5):288–294
52. Deng JL (1989) Introduction to grey system theory. J Grey Syst 1(1):1–24
53. Vinayagamoorthy R, Xavior MA (2014) Parametric optimization on multi-objective precision turning using grey relational analysis. Procedia Eng 97:299–307
54. Lin JL, Lin CL (2002) The use of the orthogonal array with grey relational analysis to optimize the electrical discharge machining process with multiple performance characteristics. Int J Mach Tools Manuf 42(2):237–244
55. Kuo Y, Yang T, Huang GW (2008) The use of grey relational analysis in solving multiple attribute decision-making problems. Comput Ind Eng 55(1):80–93