Industrial Management & Data Systems

Personalized configuration rules extraction in product service systems by using Local Cluster Neural Network

Jin Shen Bin Wu Li Yu

Personalized configuration rules extraction in product service systems by using Local Cluster Neural Network

Jin Shen and Bin Wu
School of Business, Shanghai Dianji University, Shanghai, China, and
Li Yu
Shanghai Key Laboratory of Financial Information Technology, Shanghai University of Finance and Economics, Shanghai, China

Received 24 March 2015
Revised 4 May 2015
9 June 2015
Accepted 7 July 2015

Abstract

Purpose – Configuration systems are used as a means for efficient design of customer tailored product service systems (PSS). In PSS configuration, mapping customer needs with optimal configuration of PSS components have become much more challenging, because more knowledge with personalization aspects has to be considered. However, the extant techniques are hard to be applied to acquire personalized configuration rules. The purpose of this paper is to extract the configuration rule knowledge in symbolism formulation from historical data.

Design/methodology/approach – Customer characteristics (CCs) are defined and introduced into the construction of configuration rules. Personalized PSS configuration rules (PCRs) are thereby proposed to collect and represent more knowledge. An approach combining Local Cluster Neural Network and Rulex algorithm is proposed to extract rule knowledge from historical data.

Findings – The personalized configuration rules with CCs are able to alleviate the burden of customers in expressing functional requirements. Furthermore, in the long-term relationship with a customer in PSS realization, PSS offerings can be reconfigured according to the changing CCs with the guide of PCRs.

Originality/value – The contribution of this paper lies in introducing the attribute of CCs into the antecedents of PCRs and proposing the neural networks-based approach to extracting the rule knowledge from historical data.

Keywords Product service systems, Customer characteristics, Local Cluster Neural Network, Rules extraction

Paper type Research paper

1. Introduction

Driven by the increasingly global competition, manufacturers are undergoing a transition from being product-focussed to service-based. Product service systems (PSS) are used to describe this transition. They are defined as a combination of products and services that are systemized to deliver the desired utility or function so as to satisfy customer needs (Baines *et al.*, 2007). Some of the successful examples of PSS are Rolls Royce, Xerox and Cannon (Baines *et al.*, 2007).

With the diversification of customer needs, highly personalized PSS will provide improved value for a wide range of customers (Wurtz *et al.*, 2013). Configuration systems are thus increasingly used as a means for efficient design of customer tailored PSS (Hvam *et al.*, 2013). PSS configuration aims to find a configuration solution from a given set of predefined components to satisfy the individual needs of customers without violating any constraint imposed on the components. No new component types

0263-5577
DOI 10.1108/IMDS-03-2015-0092

can be created and the interface of existing component types cannot be modified. The configuration of PSS can also be regarded as the searching process of all status set under the configuration rules (Wang *et al.*, 2014). And it has been argued that the major challenge in knowledge-based configuration system is the acquisition of rules (Sabin and Weigel, 1998).

Configuration rule is a kind of deductive knowledge which reflects mapping relationship between two concepts. There are two types of PSS configuration rules: the relationship between customer needs and PSS components, and the relationship among different PSS components (Wang *et al.*, 2014). Due to dynamically changing customer desires as well as rapid technology evolution, mapping customer needs with optimal configuration of components have become much more challenging than before (Wang, 2013). The development and application of these configuration rules help to transform the configuration manner from product-centric to customer centric (Ren and Zhang, 2011). The configuration rules discussed in this paper only refer to the mapping between customer needs and PSS components.

Customer needs can be reflected in the form of functional requirements (FRs) while PSS components are in the form of design parameters (DPs). In PSS configuration, more knowledge with personalization aspects has to be considered (Wurtz, Ardilio *et al.*, 2013). Even if different customers have the same expectation for the FRs, the actual PSS configuration solution may be different due to the influence caused by individual customer characteristics (CCs). Taken computer total solution as example, customers with different CCs "use purpose" usually have different understandings of the value "high" for FR "reliability." Thus, for the DP "Response speed for emergency repair," "one hour" should be designed for a stock exchange while "four hours" may be designed for a library, even the two customers selected the same value "high" for FR "reliability."

Since the customer becomes an active part in PSS realization, the individual behavior of a customer has to be analyzed in advance (Aurich *et al.*, 2009). CCs (such as age, occupation, interests, usage purpose, etc.) are thus defined and introduced into the construction of PSS configuration rules in this paper. Personalized PSS configuration rules (PCRs) are thereby proposed to collect and represent more knowledge which helps meeting or exceeding customer expectations in PSS configuration.

Despite the merits, the introduction of PCRs probably adds the difficulty in rules extraction. First, the hybrid antecedents which incorporate both FRs and CCs will formally result in massive coupled or interrelated rules and finally generate a "confusing" rule set. It is difficult to explore the nonlinear and complicated relationship among the multi-attributes of the rules (Yu and Chen, 2011). Second, the multi-attributes of the rules brings difficulties for companies to acquire knowledge from various stakeholders, such as sales/marketing, customers, PSS designers, configuration experts, by traditional knowledge acquisition approaches, like conjoint analysis, QFD and AHP (Fung *et al.*, 2002).

However, data mining has been considered to be an effective method to extract valuable knowledge from the historical cases (Yu and Chen, 2011). The historical cases in PSS configuration include customer information and successful transactional records. CCs can be obtained from customer information stored in systems like CRM. FRs and DPs can be analyzed through systems like configurators. To take the advantages of massive historical cases in PSS configuration, this paper adopts a data-based knowledge acquisition strategy based on neural networks (NN).

The rapid and successful proliferation of NN application has provided a clear testament to the capability of NN paradigm. The salient characteristics of NN include the direct and straightforward manner in which NN acquires knowledge about a given problem domain, the compact and flexible form in which the acquired knowledge is presented, and the ease and speed with which the knowledge can be accessed and deduced (Hagan *et al.*, 1996). It is argued that parallel reasoning process is more efficient than the activation verification in traditional symbolic reasoning (Chen and Wang, 2010). Although there is a traditional inability for NN to provide comprehensible explanation for the process, rule extraction from trained NN is considered as an addition of explanation (Andrews *et al.*, 1995).

In all, this paper proposes an approach based on Local Cluster Neural Network (LCNN) (Geva *et al.*, 1998) (one kind of NN) and Rulex algorithm (Andrews and Geva, 2002) (one kind of rule extraction techniques) to realize personalized configuration rules extraction in PSS. Since the study on PSS configuration is not abundant (Wang *et al.*, 2014), the contribution of this paper lies in introducing the attribute of CCs into the antecedents of PCRs and proposing the NN-based approach to extracting the rule knowledge from historical data. The PCRs with CCs are able to alleviate the burden of customers in expressing FRs. Furthermore, in the long-term relationship with a customer in PSS realization, PSS offerings can be reconfigured according to the changing CCs with the guide of PCRs.

The remainder of the paper is organized as follows. First, the review of related work is given. The formulation for PCRs is then presented in Section 3. Both the symbolism and connectionist representation approach are discussed. In Section 4, LCNN is adopted to address the acquisition and deduction of the rules, and Rulex algorithm is utilized for the explanation and refinement in symbolism formulation. A configurable PSS named "building solution" is taken as an illustrative example in Section 5. Finally, conclusions and discussion are drawn in Section 6.

2. Related work

The mass customization paradigm has been widely applied to manufacturing industry and many achievements have been made in product configuration. However, the study on PSS configuration is not abundant (Wang *et al.*, 2014). Wang *et al.* (2014) proposed a modular product service configuration method based on the structural knowledge of ontology. The configuration rules are interpreted and represented in Semantic Web Rule Language (SWRL), which is a rule language based on OWL. (Long *et al.*, 2013) built a multiclass support vector machine model for configuring a specific PSS that meets both functional needs and perception needs of customers. Pezzotta *et al.* (2013) proposed a Service Engineering framework that integrated a product service design modeling tool developed at the Tokyo Metropolitan University with a discrete event simulation test-bench, which enabled the comparison of several PSS configurations and the evaluation of both customer and internal performance. Shen *et al.*(2012) proposed an ontology-based approach to representing service configuration knowledge and developed a service configuration system. Dong *et al.* (2011) presented an ontology-based service product modeling approach for configuration. The configuration rules were also formalized in SWRL. Dong and Su (2011) modeled a configuration system of PSS based on ontology under mass customization. Shilov (2011) presented an approach based on efficient management of information services in the open information environment oriented to product-service system configuration. The approach was based on the technologies of ontology and context management. Aurich *et al.* (2009) presented a framework which comprised customer, manufacturer and product life cycle specific aspects to conduct a systematic configuration of PSS. Becker *et al.* (2009) presented a configurative service engineering tool Adapt(X) to generate customized service processes, organizational infrastructures and IT infrastructures. Baida *et al.* (2004) presented a service ontology to support a component-based structure of services. In their paper they used a case study from the Norwegian energy sector to describe how a component-based ontological description of services facilitates the automated configuration of a set of services. However, all of the work above focussed on the systematic configuration system where configuration rules were assumed to be acquired in advance.

As to configuration rule problem, Tidstam and Malmqvist (2015) proposed a systematic development process for product configuration rules. It included author (elicit, interpret, formalize, implement), evaluate (inspect, compute, test) and release. However, this was a deductive way which heavily depended on users' experiences. Ren and Zhang (2011) regarded the simulation data of FEA as the important knowledge source of configuration rules. Then, a methodology of integration between product configuration design and simulation analysis was proposed to realize the dynamic update of configuration rules. Shao *et al.* (2006) proposed a methodology which was an integration of popular data mining approaches and variable precision rough set. It focussed on the discovery of configuration rules. However, the proposed methodology depended on the customer groups indirectly rather than the individual customer. Soininen and Niemelä (1999) proposed a rule-based language for representing typical forms of configuration knowledge. It was equipped with a declarative semantics providing formal definitions for main concepts in product configuration. However, their work only addressed the representation problem of configuration rules.

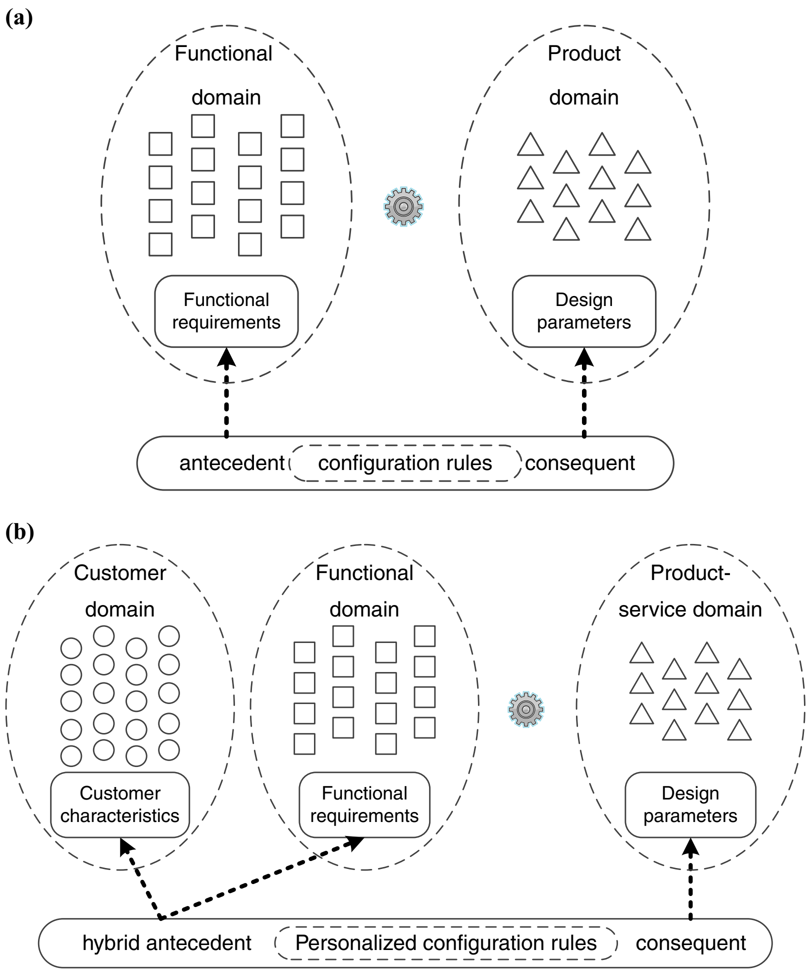
In the field of PSS configuration rules, Long and Wang (2012) proposed an approach based on dominance-based rough set to extract configuration knowledge. The configuration knowledge (rule) described the mapping relationship from PSS to customer experience perception. The customer experience perception they defined was subjective and obtained by the customer questionnaire. It was different from the CCs discussed in this paper which objectively exist before configuration. Geng *et al.* (2012) proposed an association rule mining and maintaining approach in dynamic database to obtain parameter translating rules aiding two domain mapping process in PSS. Yu and Chen (2011) proposed an a priori-based data mining method to transform the implicit knowledge into explicit association rules. Three criterions, support, confidence and interestingness, were applied for the evaluation of the extracted rules. These data mining-based approaches aim to reveal the relations between attributes, but companies still have to organize, simplify, and program them to form configuration rule set. All these transition activities will involve intensive human work and consume a lot of organizational resources (Chen and Wang, 2010).

3. Formulation of personalized configuration rules

3.1 Fundamentals of CCs and PCRs

According to axiomatic design theory (Suh, 1990), the specifications of the DPs are identified based on collecting and ranking FRs. Thus there are two domains in this framework: functional domain and product/service domain. Configuration rules are usually in the form of production rules "IF antecedent (condition) THEN consequent (result)" to describe the transformation from FRs to DPs. The antecedents and consequent of conventional configuration rules are mainly composed of FRs and DPs, respectively, as Figure 1(a) shown.

In this paper, for the purpose of improving the effectiveness of PSS configuration, CCs are introduced into the construction of configuration rules. CCs are attributes that



Notes: (a) Conventional configuration; (b) PSS configuration

Figure 1. Domain knowledge of configuration rules

indicate the customers' personality who involve in configuration activities. CCs include basic information, interests and preferences, usage pattern and environment, etc. (Shen *et al.*, 2012). Basic information describes a user profile by age, marital status, income, education, gender, profession, etc. for a single person; and by staff number, geographical location, company scale, etc. for a business customer. Obtaining and accumulating data about customer interests and preferences can improve learning about market trends and the trade-offs made in choosing services, which in turn help to find the optimal configuration for the customer. Information about usage pattern and environment also significantly affect the configuration solutions. Thereby, PCRs are proposed, aiming to improve the ability of differentiating personalized customer needs. The antecedents of PCRs consist of hybrid domain attributes which include functional domain and customer domain. The consequent of PCRs consists of product service domain attributes, as Figure 1(b) shown.

There are several advantages of proposing PCRs and modeling CCs as parts of the antecedent of PCRs:

(1) CCs are abstract description of the individual customer. There are not any principles of completeness, and only the core features that are closely related to the configuration are requisite. Thereby, there is great flexibility of the size of CCs portfolio.

(2) Customers are normally aware of their basic needs and often have no idea of configuration alternatives that a manufacturer can offer (Shao *et al.*, 2006). CCs are the attributes of customers themselves. Customers thus are more competent to determine this knowledge than to determine the FRs options that they are not familiar with.

Besides the traditional rule formulation of symbolism, this paper introduces a connectionist formulation. Although these two approaches are disparate, our research appropriately coordinates them into a mutually complementary model, which simultaneously addresses the expressiveness and deductiveness of PCRs and intelligently alleviates the difficulties in rule acquisition, representation and deduction.

3.2 Symbolism formulation for PCRs

PCRs focus on the mapping from functional domain to product service domain, and also incorporate customers' personalized characteristics into antecedents to reveal the distinct feature of PSS configuration. The symbolism formulation for PCRs is based on production rules "IF antecedent THEN consequent." The antecedents of PCRs consist of functional domain and customer domain attributes, while the consequent of the rule consists of product service domain attributes.

In functional domain, the functionality of a PSS is characterized by *s* set of FRs, $\mathbf{F} = \{f_1, \ f_2, \ldots, f_N\}$. Each FR, $f_i | \forall_{\boldsymbol{i}} \in \{1, \ldots, N\}$, possesses several values $\mathbf{F}'_i = \{f'_{i1}, f'_{i2}, \ldots, f'_{in}\}$. That is, $f_i =:: f'_{ij} \big| \forall f'_{ij} \in \mathbf{F}'_i$, where $j \in \{1, \ldots, n\}$. f'_{ij} denotes the *j*th possible values of *i*th FR. Therefore, the functionality of a particular PSS can be represented as a vector of certain values of those FRs. For example, $\mathbf{F}_s = \{f'_{11}, f'_{22}, \ldots, f'_{N2}\}$ denotes that for the functionality of *s*th PSS configuration solution, f_1 takes the first value. The second value are set to f_2 and f_N, respectively.

In customer domain, the personality of a customer is characterized by s set of CCs, $\mathbf{C} = \{c_1, c_2, \ldots, c_M\}$. Each CC, $c_p | \forall p \in \{1, \ldots, M\}$, possesses several levels of instances $\mathbf{C}'_p = \{c'_{p1}, c'_{p2}, \ldots, c'_{pm}\}$. That is, $c_p =:: c'_{pq} \big| \forall c'_{pq} \in \mathbf{C}'_p$, where $q \in \{1, \ldots, m\}$. c'_{pq} denotes the *q*th possible values of *p*th CC. Therefore, the personality of a particular customer can be represented as a vector of certain values of those CCs. For example, $\mathbf{C}_s = \{c'_{12}, c'_{23}, \ldots, c'_{M2}\}$ denotes that for *s*th PSS configuration solution, the corresponding customer has *m* characteristics, where c_1 takes the second value. The third value are set to c_2, c_N takes the second value.

In physical domain, a PSS configuration solution is characterized by s set of DPs, $\mathbf{D} = \{d_1, d_2, \ldots, d_K\}$. Each DP, $d_g | \forall g \in \{1, \ldots, K\}$, may take on one out of a finite set of values $\mathbf{D}'_g = \{d'_{g1}, d'_{g2}, \ldots, d'_{gk}\}$. That is, $d_g =:: d'_{gh} \big| \forall d'_{gh} \in \mathbf{D}'_g$, where $h \in \{1, \ldots, k\}$. d'_{gh} denotes the *h*th possible values of *g*th DP. Therefore, the DP of a particular PSS configuration solution can be represented as a vector of certain values of those DPs.

In all, a basic PCR can be described as $f'_{ij} \wedge c'_{pq} \Rightarrow d'_{gh}$, where a mapping symbol "⇒" indicates an inference from the antecedents to the consequent, which mean "IF the *i*th FR takes the *j*th value AND the *p*th CC takes the *q*th value THEN the *g*th DP takes the *h*th value." More sophisticated PCRs could be logically compounded by basic rules which address the same consequent, as Table I shown. The first case in this table indicates that "IF the third FR takes the first value AND the first CC takes the second value THEN the first DP takes the first value." Still taken computer total solution as example, this could be a rule "IF the FR (reliability) takes the value of high AND CC (use purpose) takes the value of library THEN the first DP (Response speed for emergency repair) takes the value of four hours."

3.3 Connectionist formulation for PCRs

The symbolism rules clearly indicate specific domain attributes as well as the relations among them. However, the complexity of PCRs challenges the work of rule acquisition and deduction. To deal with the problems, as Figure 2 illustrates, a connectionist formulation for PCRs based on NN is proposed in this paper.

To formulate a set of PCRs, a network module is developed which takes FRs and CCs as the inputs and the target DPs as the outputs. The deductive reasoning of a PCR can be regarded as a problem of function approximation. Since the instantiation of DPs is determined by several PCRs, the reasoning behavior (i.e. PCR) from FRs and CCs to a specific DP is simulated by the approximation of several functions.

IF	$f'_{ij} \wedge c'_{pq}$	THEN	d'_{gh}	
	f'_{31} AND c'_{12}		d'_{11}	d_1
	$<f'_{11}, f'_{22}>$ AND c'_{13}		d'_{14}	
	$<f'_{11}, f'_{21}>$ AND $<c'_{11}, c'_{12}>$		d'_{12}	
	$<f'_{11}, f'_{22}>$ AND c'_{31}		d'_{13}	
	$<f'_{12}, f'_{21}>$ AND $<c'_{11}, c'_{12}>$		d'_{15}	
	…		…	d_2
	…		…	…
	…		…	d_K

Table I.
Cases for PCRs

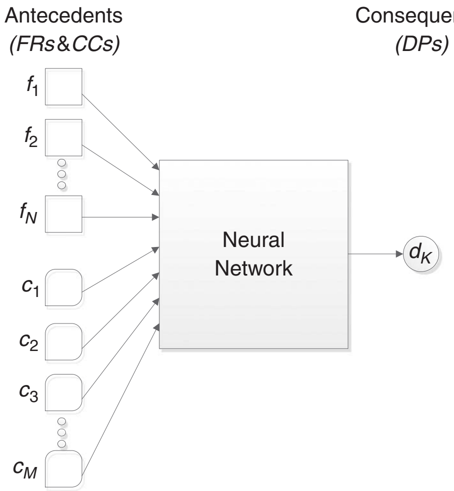


Figure 2.
NN-based PCRs construction

To simulate this function, LCNN is adopted in this paper. LCNN is a class of multilayer perceptron which utilizes sigmoid functions to generate a specific multi-dimensional local function (Geva *et al.*, 1998). Local function is almost zero everywhere except in the ridge region, which has a similar function property with the "response" behavior of configuration rules. LCNN has been suggested that it trains as fast as radial basis functions network and is well suitable for multi-dimensional function approximation. Moreover, it well facilitates the further rule extraction and symbolic explanation (Andrews *et al.*, 1995).

The rationale of local function construction of LCNN lies in taking the difference between two appropriately parameterized, parallel and displaced sigmoid functions. Consider the ridge function l which is almost zero everywhere except in the region between the steepest parts of two sigmoid functions (Geva *et al.*, 1998), as Figure 3(a) shown:

$$l(\mathbf{w},\mathbf{r},\mathbf{x}) = l^{+}(\mathbf{w},\mathbf{r},\mathbf{x}) - l^{-}(\mathbf{w},\mathbf{r},\mathbf{x}) = \sigma\left(k_1, \mathbf{w}^T(\mathbf{x}-\mathbf{r})+1\right) - \sigma\left(k_1, \mathbf{w}^T(\mathbf{x}-\mathbf{r})-1\right) \quad (1)$$

where $\sigma(k,h) = 1/(1+e^{-kh})$, $\mathbf{w}$ is the weight matrix, $\mathbf{x}$ is the input vector, $\mathbf{r}$ is a reference vector, the value of k_1 determines the shape of the ridge. Adding n ridges in different dimensions but a common center produces a function f which peaks at the common center with the component ridges radiating on all dimensions (Geva *et al.*, 1998), as Figure 3(b) shown:

$$f(\mathbf{w},\mathbf{r},\mathbf{x}) = \sum_{i=1}^{n} l(\mathbf{w}_i,\mathbf{r},\mathbf{x}) \quad (2)$$

As Figure 3(c) shown, to make this function local, the component ridges must be "cut off" without introducing discontinuities into the derivatives of the local functions (Geva *et al.*, 1998), which is achieved by:

$$L(\mathbf{w},\mathbf{r},\mathbf{x}) = \sigma_0(k_2, f(\mathbf{w},\mathbf{r},\mathbf{x}) - d) \quad (3)$$

where d is selected to ensure the maximum value of function f coincides with the center of the linear region of the output sigmoid. The parameter k_2 determines the steepness of the sigmoid function.

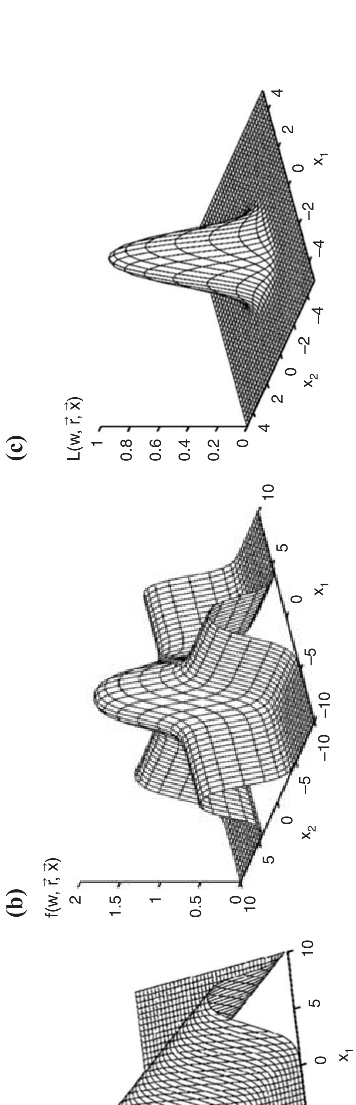
After all, the output of the NN is finally approximated by linear combination of T local clusters distributing over the domain with n dimensions (Geva *et al.*, 1998):

$$d_i(\mathbf{x}) = \sum_{j=1}^{T} v_j L\left(\mathbf{w}_j, \mathbf{r}_j, \mathbf{x}\right) \quad (4)$$

where v_j is the output connection weight associated with particular local function. With the advantages of LCNN, the resulting connectionist formulation provides inherent facilities for intelligent acquisition and efficient deduction.

4. Rule extraction by using LCNN and Rulex

The approach of PCRs extraction mainly includes two steps: acquisition and explanation. First, PCRs are acquired and deduced from knowledge hidden in historical data by training LCNN. Then, PCRs are reasonably explained in symbolism form by using Rulex algorithm. Besides, networks partition and refinement are utilized to combine LCNN with Rulex.



Source: Geve *et al.* (1998)

Figure 3.
Multi-dimensional local function based on sigmoid functions

4.1 Acquisition and deduction of PCRs in connectionist formulation

To acquire PCRs, the connectionist formulation is required to be further trained. First, historical data should be organized as pairs $\mathbf{R} = <\mathbf{F} \cup \mathbf{C}, \mathbf{D}>$, where $\mathbf{F} \cup \mathbf{C} = [f_1, f_2, \ldots, f_N, c_1, c_2, \ldots, c_M]$, $\mathbf{D} = [d_1, d_2, \ldots, d_K]$. The data are collected from transactional records of sales, customer profiles and configuration documents. Any effective pair of such historical data are an instantiation of $\mathbf{R}$, and indicates a specific instance of functional and physical description of a PSS configuration solution, and the personalized characteristics of the corresponding customer. The historical data are then further decomposed into K sub-sets, $\mathbf{R}^u = <\mathbf{F} \cup \mathbf{C}, \mathbf{D}^u>$, where $u \in \{1, \ldots, K\}$. Therefore, each pair of $\mathbf{R}^u$ implies an instance of the mapping from the antecedents to a specific consequent.

Second, to train a specific NN, a set of compatible training samples should be established for $\mathbf{R}^u$. Because LCNN cannot directly handle nominal inputs and outputs, the instances of FRs, CCs and DPs should be encoded into numerical values before network training. For example, the nominal instances, {A, B, C}, of an attribute can be encoded as {1, 2, 3}, where "1" stands for "A," "2" stands for "B" and "3" stands for "C." The encoding rules should be recorded to support further PCRs explanation in symbolic formulation. Besides, because the variables may involve different metrics and ranges of values, the inputs and outputs also need to be transformed into standard form. To avoid the dependence on the choice of different metrics of certain variables over others, those values are normalized to be dimensionless by max-min normalization method.

Finally, the training method for LCNN in this paper is based on a variation of gradient descent algorithm with individual adaptive learning rates for each adjustable parameter. With the limitation of the paper length, more details of network training can be referred to (Geva *et al.*, 1998).

Given a well-trained LCNN, the inputs, i.e. instantiations of a specific set of FRs and CCs, are referred to as a "stimulus," and the value of output, i.e. DP, is referred to as a "response" achieved by the trained LCNN. Therefore, the acquisition and deduction of PCRs is transformed to an automatic stimulus-response process guided by Equation (4).

4.2 Explanation of PCRs in symbolism formulation

Although LCNN has the capability of rule acquiring and deducing, there is an inability for LCNN to provide good explanation for the process through which the given output has been reached. To address the problem, the rule extraction technique is considered. Rulex (Andrews *et al.*, 1995), which is one of the existing techniques for rule extraction from NN, is adopted in this paper to provide symbolic explanations for connectionist formulation. Rulex has been proved to have the capability of accurately extracting high-quality rules through the tests on benchmark problems. It is able to directly interpret the weight parameters into the rules like "IF $\forall 1 < i < n{:}x_i \in [x_{i-lower}, x_{i-upper}]$, THEN the output takes the target value," where $[x_{i-lower}, x_{i-upper}]$ represents the effective range in the ith input dimension. Then, let $k_i = k_1 w_i$, $b_i = 1/w_i$, $m = e^{-(x_i - r_i)k_i}$, $n = e^{-b_i k_i}$, the expressions to determine $[x_{i-lower}, x_{i-upper}]$ can be established as:

$$x_{i-lower} = r_i - \ln\left(\frac{p - q - \sqrt{p^2 + q^2 - 2(\alpha^2 + 1)}}{2\alpha}\right) k_i^{-1} \tag{5}$$

$$x_{i-upper} = r_i + \ln\left(\frac{p-q+\sqrt{p^2+q^2-2(\alpha^2+1)}}{2\alpha}\right)k_i^{-1} \tag{6}$$

where $\alpha = 1/(1+mn)-1/\left(1+m/n\right), p = (1-\alpha)e^{b_i k_i}, q = (1+\alpha)e^{-b_i k_i}$

By the assistance of Rulex, the reasoning behavior of a well-trained LCNN could be interpreted as a set of symbolic rules, which are exactly the extracted PCRs. Additionally, the directly extracted rule set may contain redundant rules, individual rules with redundant conditions, and pairs of rules where conditions can be combined. Rulex also provides facilities to simplify the redundancy to improve the comprehensibility without compromising the accuracy. Since the paper length is limited, more details of Rulex can be referred to (Andrews *et al.*, 1995).

4.3 Networks partition and refinement

Based on the original historical data $\mathbf{R} = <\mathbf{F} \cup \mathbf{C}, \mathbf{D}>$, a formally full-connected network topology NET_0 can be established. NET_0 fully connects all the inputs and outputs. Then, based on the data set $\mathbf{R}^u = <\mathbf{F} \cup \mathbf{C}, \mathbf{D}^u>$, NET_0 can be further divided into K individual network modules, namely NET_i, where $i \in \{1, \ldots, K\}$. NET_i connects all the input attributes and the output d_i. For each well-trained NET_i, the structure of the network module will be further refined as NET_i^R to simplify the representation and improve the training efficiency.

To refine the network module, the inputs are classified into different categories with different policies, as Table II shown. The refined NET_i^R is trained by the data in $\mathbf{R}^{u'} = <\mathbf{F}_A \cup \mathbf{C_r}, \mathbf{D}^u>$. The effort of network refinement leads to compact network modules with less input attributes and simplified connection weights.

5. Case study

With the high-speed development of construction in China, automation and control products in buildings are widely installed. An anonymous leading company H who

Category	Description	Policy
$\mathbf{F_C}$	FRs that are obviously unrelated to the instantiation of d_i	$\mathbf{F_C}$ is directly removed from the input attributes of NET_i
$\mathbf{F_B}$	FRs that are related to the instantiation of d_i but could be replaced by CCs through network training	$\mathbf{F_B}$ is primitive FRs for NET_i, and then the explanations of well-trained NET_i will indicate the reduction of $\mathbf{F_B}$
$\mathbf{F_A}$	FRs that are decisive to the instantiation of d_i	$\mathbf{F_A}$ is the final decisive input attributes for the refined network module NET_i^R, which is figured out through the explanations of well-trained NET_i
$\mathbf{C_r}$	CCs that are related to the instantiation of d_i	$\mathbf{C_r}$ is the final decisive input attributes for the refined network module NET_i^R, which is figured out through the explanations of well-trained NET_i
$\mathbf{C_n}$	CCs that are unrelated to the instantiation of d_i	$\mathbf{C_n}$ is primitive CCs for NET_i, and then the explanations of well-trained NET_i will indicate the reduction of $\mathbf{C_n}$

Table II. Classification of inputs

traditionally only sold the tangible products are now transforming to provide customers with customized PSS called "Building Solution." It aims to keep the buildings safe, secure, comfortable and energy-efficient. It serves a wide range of customers, including commercial and government buildings, health care facilities, airports, and schools. Significant revenues can be extracted from servicing the installed base of physical products to manipulate decreasing profits from product sales. The service also provides a differentiating factor to lock the customer into a long-term relationship.

"Building Solution" is a typical kind of product-oriented PSS. It is composed of several tangible product parts like access control, reader, field equipment, monitor, etc., and intangible services like spare parts supply, installation, emergency maintenance, recycling, etc. "Building Solution" is customized according to diverse customer needs. The configurator rules are important during this customization process.

To clearly explain the extraction process of PCRs in "Building Solution," the design parameter "Service Level" of Emergency Maintenance is selected as the target design parameter (s_1). First, based on the analysis of the historical data, the nominal range of s_1, i.e. {Engineer, Technician} is set, and coded by {1, 2}. Second, after analyzing the configuration options, FRs are confirmed, which include reliability, expertise, technical supporting level, field service and convenience. According to the definition of primitive FRs, $\mathbf{F_C}$, i.e. Convenience, is removed, and $\mathbf{F_A} \cup \mathbf{F_B}$ is developed, as Table III shown. Then, with the permission of customer privacy, some CCs are acquired from customer documents and set as parts of PCRs, as Table IV shown. Finally, the primitive training data are organized as $\mathbf{R}^1 = <[f_1, f_2, f_3, f_4, c_1, c_2], s_1>$, and the primitive training network NET_1 is developed as Figure 4 shown.

To train NET_1, 231 historical transaction records are totally collected. They are encoded with the rules shown in Table V. To ensure the effectiveness of the extracted PCRs, 11 times independent training activities are performed based on different initial local functions that are randomly generated. When each network is converged, Rulex algorithm is then utilized to realize symbolic explanation of the network, as Figure 5 shown.

There are 11 different sets of rules based on the converged $NET1$. They can be regarded as different forms of explanation of the same information. Therefore, the weights of 11 groups of network are then combined and further extracted to simplified rules by using Rulex. Afterwards, since there are only four input attributes in the antecedent of simplified rules, NET_1 can further be refined as NET_i^R with the training structure of $\mathbf{R}^{1'} = <[f_1, f_2, f_3, c_1], s_1>$. Based on the same 231 records, training process

Table III. Primitive functional requirements

FRs	FR ID	Range of FR	Code
Reliability	f_1	{High, medium, meaningless}	{1, 2, 3}
Expertise	f_2	{High, medium, meaningless}	{1, 2, 3}
Technical supporting level	f_3	{High, medium, meaningless}	{1, 2, 3}
Field service	f_4	{Yes, No}	{1, 2}

Table IV. Customer characteristics

CCs	CC ID	Range of CC	Code
Usage purpose	c_1	{commerce, office, healthcare, residence}	{1, 2, 3, 4}
Geographical location	c_2	{downtown, suburban}	{1, 2}

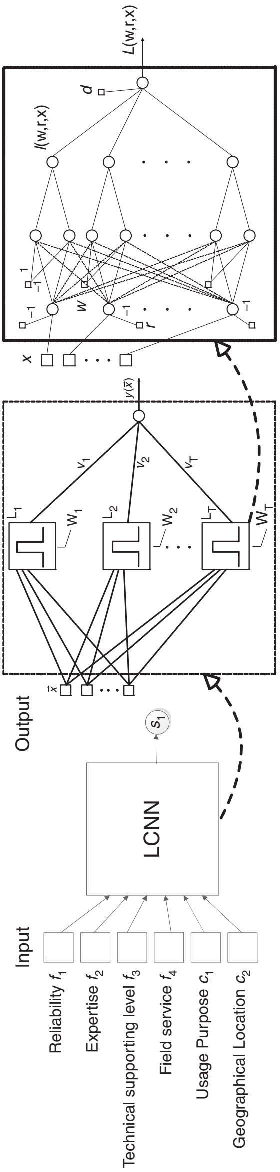


Figure 4.
Refined network module

is re-performed and NET_i^R is symbolic explained with Rulex. Finally, the PCRs of this case are extracted as Table VI shown.

To evaluate the performance of the proposed PCRs extraction method, two important performance issues are further studied: the learning accuracy of the connectionist approach (LCNN) and the comprehensibility of the symbolic PCRs abstracted by RULEX.

Learning accuracy is referred to the capability of classifying a set of previously unseen examples from the problem domain. In the case study, the authors have collected 231 effective records as the data set. To verify the learning accuracy, a certain number (denoted as train-set-size) of the records are randomly selected in each verification trial as training samples. The remaining samples are then used for testing. The network model is then trained and tested ten trials with the given train-set-size. The final error rate with the train-set-size is determined by the average error rate for classifying the testing samples in ten trials. To analyze the learning accuracy with different train-set-sizes, the authors run the trials with the train-set-sizes ranging from 20 to 220, i.e. the size of training/testing data are 20/211, 40/191, 60/171, …, 220/11. The average error rates and standard deviations (STD) in training trails are illustrated in

Table V. Samples for the network module NET_1

Record ID	Input $= [f_1, f_2, f_3, f_4, c_1, c_2]$	$\Rightarrow$	Output $= s_1$
1	[2, 2, 1, 1, 3, 1]	$\Rightarrow$	1
2	[3, 2, 2, 1, 2, 1]	$\Rightarrow$	1
3	[2, 1, 2, 1, 1, 1]	$\Rightarrow$	1
…	…	$\Rightarrow$	…
231	[3, 3, 3, 1, 2, 2]	$\Rightarrow$	2

```
DIRECTLY EXTRACTED RULE SET
==========================
RULE 1
IF    Reliability   IS Medium OR Meaningless
AND   Expertise     IS Meaningless
AND   Remote        IS No
THEN Level-Technician(2)

RULE 2
IF    Reliability   IS Medium OR Meaningless
AND   Technical     IS Meaningless
AND   FieldService  IS Yes
AND   Usagelife     IS '0-2' OR '2-5' OR '5-10'

IF    Reliability   IS Meaningless
AND   Expertise     IS High OR Medium
AND   Technical     IS Meaningless
THEN Level-Technician(2)

RULE 6
IF    Reliability   IS High
THEN Level-Engineer(1)

RULE 7
IF    Reliability   IS Medium OR Meaningless
```

Figure 5. Parts of training results of No. 10 trial

Table VI. Formulations of the extracted PCRs for s_1

Rule ID	Antecedent (IF)	Consequent (THEN)
1	Expertise (f_2) = NOT High AND Technical supporting level (f_3) = NOT High	s_1 = Technician
2	Reliability (f_1) = High AND Expertise (f_2) = High AND Usage Purpose (c_1) = healthcare	s_1 = Engineer
3	Reliability (f_1) = High AND Expertise (f_2) = NOT Meaningless AND Technical supporting level (f_3) = NOT Meaningless	s_1 = Engineer

Figure 6, respectively. The results show high learning capability of LCNN in configuration. When train-set-size is smaller than 40, the accuracy improves dramatically from 0.92 to 0.48. When train-set-size is larger than 40, LCNN could achieve very high accuracy and the results of STD indicate stable training processes.

The comprehensibility of the symbolic PCRs abstracted by Rulex is characterized by two indices: the average number of PCRs and the average number of antecedents per PCR, which is illustrated in Figure 7 with different train-set-sizes, respectively. Except for 20/211, the average numbers of PCRs in trials do not vary greatly. Therefore, the authors suggest that the number of rules acquired in PCRs development is primarily related to the complexity of the configuration knowledge hidden in the historical data and the training process of LCNN is able to outline the set of PCRs even with small train-set-size. Second, the average number of antecedents per PCR is generally inversely related to train-set-size. Therefore, the authors suggest that PCRs could be abstracted more comprehensively if more sufficient and consistent training data are provided.

6. Conclusions

Configuration systems are used as a means for efficient design of customer tailored PSS. Since more knowledge with personalization aspects has to be considered in PSS, mapping customer needs with optimal configuration of PSS components have become much more challenging. Traditional knowledge acquisition approaches, like conjoint

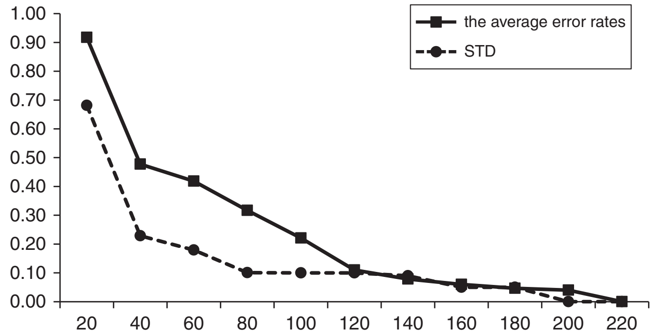


Figure 6. Learning performance of LCNN

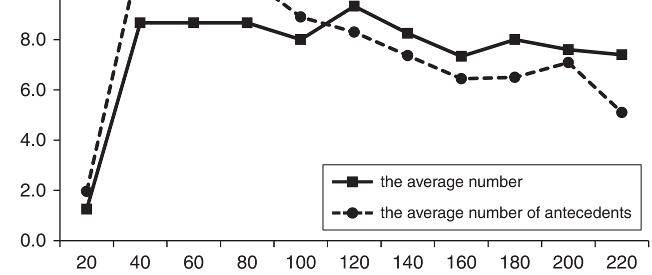


Figure 7. Complexity analysis of PCRs

analysis, QFD and AHP are hard to be applied, because of the multi-knowledge domain caused by various stakeholders. However, data mining has been considered to be an effective method to extract valuable knowledge from the historical cases. The rapid and successful proliferation of NN application has provided a clear testament to the capability of this kind of data mining techniques. Therefore, this paper aims to extract the configuration rule knowledge in symbolism formulation from historical data by using a NN-based approach.

In this paper, first CCs are defined and introduced into the construction of configuration rules. Personalized PCRs are thereby proposed to collect and represent more knowledge. Second, symbolism formulation and connectionist formulation are provided, respectively. Finally, rule extraction is realized by using an approach combining LCNN and Rulex algorithm. Connectionist formulation is adopted to address the acquisition and deduction of the rules based on LCNN, while the rule extraction technique Rulex is adopted for the explanation and refinement in symbolic formulation. Networks partition and refinement are also utilized to combine LCNN with Rulex.

With the case study "Building Solution," the feasibility and effectiveness of the approach is demonstrated. Both the learning accuracy of the network and the comprehensibility of the symbolic rules are satisfied. The PCRs with CCs are able to alleviate the burden of customers in expressing FRs. Furthermore, in the long-term relationship with a customer in PSS realization, PSS offerings can be reconfigured according to the changing CCs with the guide of PCRs.

However, there are two important assumptions in our research. First, a configuration implies a set of entities that are combined in a structured manner, therefore with reference to a PSS, both the physical product as well as the services are required to designed in components with responding DPs; second, the data sources like CRM and PDM should be available. Thus, the approach cannot be applied to a new PSS without any transaction records.

In a word, the contribution of this paper lies in introducing the attribute of CCs into the antecedents of PCRs and proposing the combined LCNN and Rulex approach to extracting the rule knowledge from historical data. For the further research, the other kind of configuration rules mapping among different PSS components are considered.

Acknowledgements

This research is supported by the National Natural Science Foundation of China (No. 71401099 and No. 71101084), Humanities and Social Sciences Research Youth Foundation Project of Ministry of Education of China (No. 13YJC630129 and No. 10YJC630274), Shanghai University Young Teachers Training Scheme (No. ZZSDJ13003), and Leading Academic Discipline Project of Shanghai Dianji University (No. 10XKJ01). The authors would like to express their sincere thanks to Professor Geva, Dr Chen Zhaoxun and three anonymous reviewers for their help to the research.

References

Andrews, R., Diederich, J. and Tickle, A.B. (1995), "Survey and critique of techniques for extracting rules from trained artificial neural networks", *Knowledge-Based Systems*, Vol. 8 No. 6, pp. 373-389.

Andrews, R. and Geva, S. (2002), "Rule extraction from local cluster neural nets", *Neurocomputing*, Vol. 47 Nos 1-4, pp. 1-20.

Aurich, J.C., Wolf, N., Siener, M. and Schweitzer, E. (2009), "Configuration of product-service systems", *Journal of Manufacturing Technology Management*, Vol. 20 No. 5, pp. 591-605.

Baida, Z., Gordijn, J., Saele, H., Morch, A.Z. and Akkermans, H. (2004), "Energy services: a case study in real-world service configuration", in Persson, A. and Stirna, J. (Eds), *Advanced Information Systems Engineering, Proceedings*, Vol. 3084, pp. 36-50.

Baines, T.S., Lightfoot, H.W. *et al.* (2007), "State-of-the-art in product-service systems", *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, Vol. 221 No. 10, pp. 1543-1552.

Becker, J., Beverungen, D., Knackstedt, R. and Matzner, M. (2009), "Configurative service engineering – a rule-based configuration approach for versatile service processes in corrective maintenance", *Proceedings of the 42nd Annual Hawaii International Conference on System Sciences, HICSS, Waikoloa, HI*.

Chen, Z. and Wang, L. (2010), "Personalized product configuration rules with dual formulations: a method to proactively leverage mass confusion", *Expert Systems with Applications*, Vol. 37 No. 1, pp. 383-392.

Dong, M. and Su, L. (2011), "Modeling a configuration system of product-service system based on ontology under mass customization", *Advanced Science Letters*, Vol. 4 Nos 6/7 pp. 2256-2261.

Dong, M., Yang, D. and Su, L. (2011), "Ontology-based service product configuration system modeling and development", *Expert Systems with Applications*, Vol. 38 No. 9, pp. 11770-11786.

Fung, R., Tang, J., Tu, Y. and Wang, D. (2002), "Product design resources optimization using a non-linear fuzzy quality function deployment model", *International Journal of Production Research*, Vol. 40 No. 3, pp. 585-599.

Geng, X., Chu, X. and Zhang, Z. (2012), "An association rule mining and maintaining approach in dynamic database for aiding product-service system conceptual design", *International Journal of Advanced Manufacturing Technology*, Vol. 62 Nos 1-4, pp. 1-13.

Geva, S., Malmstrom, K. and Sitte, J. (1998), "Local cluster neural net: architecture, training and applications", *Neurocomputing*, Vol. 20 Nos 1-3, pp. 35-56.

Hagan, M.T., Demuth, H.B., Beale, M.H. and Jesus, O.D. (1996), *Neural Network Design*, PWS, Boston, MA.

Hvam, L., Haug, A., Mortensen, N.H. and Thuesen, C. (2013), "Observed benefits from product configuration systems", *International Journal of Industrial Engineering-Theory Applications and Practice*, Vol. 20 Nos. 5/6 pp. 329-338.

Long, H. and Wang, L. (2012), "A rough set based approach to knowledge acquisition for product service system configuration", *Applied Mechanics and Materials*, Vols 220-223, pp. 2534-2539.

Long, H.J., Wang, L.Y., Shen, J., Wu, M.X. and Jiang, Z.B. (2013), "Product service system configuration based on support vector machine considering customer perception", *International Journal of Production Research*, Vol. 51 No. 18, pp. 5450-5468.

Pezzotta, G., Pirola, F., Akasaka, F., Cavalieri, S., Shimomura, Y. and Gaiardelli, P. (2013), "A service engineering framework to design and configure product-service systems", *IFAC Proceedings Volumes* (IFAC-PapersOnline).

Ren, B. and Zhang, S. (2011), "Knowledge acquisition from simulation data to product configuration rules", *Advanced Design Technology, Pts 1-3. j. GAO,* Vols 308-310, pp. 77-82.

Sabin, D. and Weigel, R. (1998), "Product configuration frameworks – a survey", *IEEE Intelligent Systems and Their Applications*, Vol. 13 No. 4, pp. 42-49.

Shao, X.Y., Wang, Z.H., Li, P.G. and Feng, C.X.J. (2006), "Integrating data mining and rough set for customer group-based discovery of product configuration rules", *International Journal of Production Research*, Vol. 44 No. 14, pp. 2789-2811.

Shen, J., Wang, L. and Sun, Y. (2012), "Configuration of product extension services in servitisation using an ontology-based approach", *International Journal of Production Research*, Vol. 50 No. 22, pp. 6469-6488.

Shilov, N. (2011), "Product-service system configuration in SOA-based environment", *Lecture Notes in Business Information Processing*, Vol. 97, pp. 184-195.

Soininen, T. and Niemelä, I. (1999), "Developing a declarative rule language for applications in product configuration", *Lecture Notes in Computer Science (Including Subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, Vol. 1551, pp. 305-319.

Suh, N.P. (1990), *The Principles of Design*, Oxford University Press, New York, NY.

Tidstam, A. and Malmqvist, J. (2015), "A systematic process for developing product configuration rules", *International Journal of Product Lifecycle Management*, Vol. 8 No. 1, pp. 46-64.

Wang, C.-H. (2013), "Incorporating customer satisfaction into the decision-making process of product configuration: a fuzzy kano perspective", *International Journal of Production Research*, Vol. 51 No. 22, pp. 6651-6662.

Wang, P.P., Ming, X.G., Wu, Z.Y., Zheng, M.K. and Xu, Z.T. (2014), "Research on industrial product-service configuration driven by value demands based on ontology modeling", *Computers in Industry*, Vol. 65 No. 2, pp. 247-257.

Wurtz, G., Ardilio, A., Lasi, H. and Warschat, J. (2013), "Towards mass individualization: life-cycle oriented configuration of time-variable product-service systems", *2013 Proceedings of PICMET 2013: Technology Management in the IT-Driven Services*.

Yu, L. and Chen, Y. (2011), "An apriori-based knowledge mining method for product configuration design", in Zhang, L.C., Zhang, C.L. and Shi, T.L. (Eds), *Manufacturing Engineering and Automation I, Pts 1-3*, Vols 139-141, Trans Tech Publications, 1490-1493.

Corresponding author
Dr Jin Shen can be contacted at: shenjin2002@sina.com