Research Article

Decision-Making Approach to Selecting Optimal Platform of Service Variants

Vladimir Modrak, 1 Slavomir Bednar, 1 and Pavol Semanco 2

1 Faculty of Manufacturing Technologies, Technical University of Kosice, Bayerova 1, 08001 Presov, Slovakia
2 Lear Corporation Seating Slovakia s.r.o., Solivariska 1/A, 08001 Presov, Slovakia

Correspondence should be addressed to Vladimir Modrak; vladimir.modrak@tuke.sk

Received 22 January 2016; Revised 9 May 2016; Accepted 22 May 2016

Academic Editor: Feng Yang

Copyright © 2016 Vladimir Modrak et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Nowadays, it is anticipated that service sector companies will be inspired to follow mass customization trends of industrial sector. However, services are more abstract than products and therefore concepts for mass customization in manufacturing domain cannot be transformed without a methodical change. This paper is focused on the development of a methodological framework to support decisions in a selection of optimal platform of service variants when compatibility problems between service options occurred. The approach is based on mutual relations between waste and constrained design space entropy. For this purpose, software for quantification of constrained and waste design space is developed. Practicability of the methodology is presented on a realistic case.

1. Introduction

In general, services are defined as every kind of activity that contributes to the satisfaction of customer needs [1]. They play an ever-increasing importance in industrialized economies due to the need for the right balance between industrial and service sector. Particularly in changing business environment, there is a challenge for service providers to strengthen their position on the competitive market through innovative business strategies [2]. They also involve the advancing trend of transition from standard services to mass customized service packages [3].

However, mass customization is mostly related to tangible products and to manufacturing processes, whereas customization in services is studied only marginally [4]. On the other hand, growing mass customization tendencies induce that practically every product today has a service component in it.

Service variants in terms of mass customized environment are usually created for particular user groups. The problem with variety is not variety itself or increasing size of variety. The problem arises in the design process with an elimination of variants that are impractical due to presence of constraint(s). Such unviable variants are unadvisable for customers, because they lose time and time is too valuable for them to be wasted. Consideration of conflicts among variants causes enormous problems in the product or service design [5]. This suggests that a more integrated approach to mass customization is needed, which considers equally both the architecture design process and the subsequent implementation into configuration systems. In this context we propose by using information entropy theory a novel decision-making approach to selecting optimal platform of service variants.

2. Related Work

2.1. Related Trends in Mass Customization of Services. According to Zhang [6], recent growing trend and pressure on mass customization practice make companies expand their product and service mix better address preference of consumers. ElMaraghy et al. [7] state that this trend has several reasons including customers’ demand for new product functions and features, different regional requirements, large number of market segments with different needs, and so forth. Another reason of this trend lays in the fact that
companies realized that effective way to achieve competitive edge leads not only through attractive products they produce but also through the services they provide [8]. Tien et al. [9] emphasized that a service mass customization strategy can be successfully adopted if the mass tailored services are delivered to customers at sufficiently short time scales. Liou et al. [10] focused on improving mass customization of airline services through determining dispensable ones associated with unimportant attributes. Particularity this approach makes a connection with our research, in which we focus on unviable service variants. Unviable service variants not only are dispensable but also have negative impact on customer perception [11, 12]. In this context, authors, for example, [13, 14], argue that infeasible configurations might be hidden by using algorithm-based product configurators.

2.2. Research on Optimal Platform of Service Variants. There are several approaches to examining the impact of service variety on service development process; see, for example, sources [15]. As the literature shows, unviable service variants reflect higher operational complexity and decrease in efficiency of design [16, 17]. In this context, Tseng and Piller [18] proposed a generic variety structure including also configuration constraints. Therefore, the right product or service composition and their component compatibility need to be ensured [19]. As service industry is intentionally becoming more complex, a key enabler to achieve this tailoring of services to consumer needs is the application of efficient computer-aided variation technologies [20]. Such technologies usually allow service designers and decision-makers to determine the optimum level of customization and to eliminate infeasible service variants. Our paper has an ambition to contribute to the tools that can be incorporated in such computer-aided variation technologies.

3. Methodological Framework

3.1. Theoretical Background of Decision-Making Approach. The very first notion of complexity was outlined in the work of Shannon [21] where information theory was introduced. Few years later, information became a key complexity element for the description and analysis of the systems and information entropy $H_d$ for the discrete case has been defined by probability $P_i$ of $n$-state occurrence as follows [21]:

$$H_d = \sum_{i=1}^{n} P_i \log_2 P_i. \quad (1)$$

Differential information entropy of the probability density function $p(x)$ for continuous signals ($H_C$) has been expressed as

$$H_C = \int_{-\infty}^{\infty} p(x) \log_2 (p(x)) \, dx. \quad (2)$$

Krus [22] adopted the design information entropy for multidimensional case ($H_X$) in the following form:

$$H_X = \int_{-\infty}^{\infty} p(x) \log_2 (p(x)S) \, dx, \quad (3)$$

where $D$ as the design space within the design $x$ is defined. $S$ is the size of the design space expressed as

$$S = \int_{D} x \, dx. \quad (4)$$

In the case of general multivariable design space, its information entropy $H$ can be expressed as follows:

$$H = \log_2 \frac{S}{s}, \quad (5)$$

where $s$ is the region of uncertainty for the final design of validated system architecture.

According to Krus [22], each particular design “$x$” with regard to its design space has information entropy $H_X$:

$$H_X = \log_2 n_x, \quad (6)$$

where $n_x$ is number of all possible unique design alternatives (representing the so-called complete design space) as a result of a combination of service options.

There are many real cases, in which some variants are impractical due to presence of constraint(s). Then information entropy of constrained design space $H_C$ can be enumerated as

$$H_C = \log_2 n_w, \quad (7)$$

where $n_w$ is number of viable design alternatives.

Because the higher number of all possible design variants has more positive impact on consumers than smaller constrained design space, then information entropy of constrained design space in terms of mass customization environment should be maximized. In this sense, entropy of constrained design space can be considered as positive entropy.

In this context, Krus [22] proposed to express a quality of a modular design through the rest of the design space that is outside the constrained design space by the term “waste” information entropy of design space and to quantify it using the following formula:

$$H_W = H_X - H_C. \quad (8)$$

Waste information entropy of the complete design space is considered as negative entropy.

3.2. Notation. The notation used in the proposed methodological framework is summarized as follows:

- $x$: particular design,
- $n_x$: number of all possible unique design alternatives (complete design space),
- $n_w$: number of viable design alternatives (restricted design space),
- $H_X$: entropy of complete $D$,
- $H_C$: entropy of constrained $D$,
- $H_W$: waste information entropy,
3. Proposed Decision-Making Method. Based on above preconditions, we propose the following decision-making procedure to select optimal platform of service variants by using mutual relations between \( H_W \) and \( H_C \).

Let us have a service company providing several parallel mass customized service activities (see Figure 1). As it is known, in order to maintain sustainability of their core activities, each of the activities might be continuously analyzed and subsequently optimized. Obviously, there are many aspects of service activities that should be analyzed. However, customers’ view has strong priority. A questionnaire is not always the most effective method as some general useful findings that could be easily implemented are unnoticed. Among them is a known fact that infeasible options are unwelcome. The following method shows one possible way to reduce or eliminate them in individual service activity.

We start by taking the so-called draft design platform \( D_0 \) defined by \( n_s \) and \( n_{s0} \). This starting platform can be either existing system of service activities or the first version of developed system of service activities, where \( n_{s0} \) presents a number of unique design alternatives as a result of a combination of service options and \( n_s \) is a number of viable design alternatives.

Let us further assume that we remove from service platform \( D_0 \) some components, which are in conflict with other ones. Then \( D_0 \) can be transformed into a new state with \( n_s \) and \( n_{s0} \), denoted as \( D_1 \). If we would continue in such a reduction of components, the design platform \( D_1 \) will be modified to \( \{D_2, D_3, \ldots, D_p\} \), while \( p \) represents a number of actions taken to reduce the system design space and waste entropy.

In order to compare two arbitrary design platforms, for example, \( D_0 \) and \( D_1 \), the following two measures are proposed:

\[
\Delta H_{W_{01}} = \left| \frac{H_W}{H_{W0}} - 1 \right|, \tag{9}
\]

\[
\Delta H_{W_{12}} = \left| \frac{H_C}{H_{C1}} - 1 \right|. \tag{10}
\]

Then if \( \Delta H_{W\{3,4\}} > \Delta H_{C\{1,2\}} \to \) design platform \( D_3 \) is more preferable for mass customization (MC) than \( D_0 \). To compare three alternative design platforms, the following subprocedure can be used. Let us suppose that design platforms \( D_1 \) and \( D_2 \) are more preferable for MC than \( D_0 \), based on the following criteria:

\[
\Delta H_{W_{13}} > \Delta H_{C_{14}}, \tag{11}
\]

\[
\Delta H_{W_{23}} > \Delta H_{C_{24}}. \tag{12}
\]

Then one can select more preferable design platform between \( D_1 \) and \( D_2 \) using these three criteria:

(I) If \( \Delta H_{W_{13}} - \Delta H_{C_{14}} > \Delta H_{W_{23}} - \Delta H_{C_{24}} \to \) design platform \( D_1 \) is more suitable than \( D_2 \).

(II) If \( \Delta H_{W_{13}} - \Delta H_{C_{14}} < \Delta H_{W_{23}} - \Delta H_{C_{24}} \to \) design platform \( D_2 \) is more suitable than \( D_1 \).

(III) If \( \Delta H_{W_{13}} - \Delta H_{C_{14}} = \Delta H_{W_{23}} - \Delta H_{C_{24}} \to \) both design platforms \( D_1 \) and \( D_2 \) are equally preferable for buyers.

Subsequently, proposed procedure for selection of optimal design platform is graphically depicted in Figure 2 and a simple software tool to select optimal design platform can be accessed online [23].

As seen in Figure 2, input parameters of the algorithm are \( n_s \), number of all possible unique design alternatives, in other words alternatives that may theoretically be chosen including incompatible ones, and \( n_{s0} \), number of all viable design alternatives. Therefore, Sections 3.4 and 3.5 describe procedures for their quantification. Note that if there are no restrictions, then \( n_s = n_{s0} \) and the method is irrelevant.

3.4. Service Variety Quantification without Restrictions. In this section, wider description for quantification of all possible design alternatives understood as the design space \( n_s \) will be treated.

This description refers to hierarchy of service ontology, as shown in Figure 3. To explain the hierarchy, the following working definitions can be useful. Service activity is here considered as a compact package of services divided into features \( \{f_1, \ldots, f_m\} \). Feature consists of a set of attributes \( \{A_{i1}, A_{i2}, \ldots, A_{ij}\} \).

Attributes represent characteristics of individual feature and will be denoted as \( A_{ij} \), where \( j = (1, \ldots, n) \).

Each attribute is represented by options \( O_q = \{O_1, O_2, \ldots, O_r\} \), but the number of options depends
on structure of attribute components which can be of different nature. We distinguish between three types of such components.

The simplest type of component is when customer can select from a portfolio of items on a voluntary basis. This means that all combinations of offered items and even no selection are possible. This type of option is denoted as voluntary option $C^V_{ij}$.

Delimited components $C^D_{ij}$ have limited selection possibilities specified by restriction rule $0 \leq l < k$, where $l$ is number of components to be chosen by customer from a set of components $C^D_{ij}$.

Compulsory components $C^C_{ij}$ have three possibilities when selecting desired service option or the so-called selection rules: $1 \leq l < k$.

Their practical importance, especially in the case of high numbers of available component, can be explained by the following example. Let us say that a structure of attribute components consists of all three component types. If we had three components $C^V_{ij}$ and two components $C^D_{ij}$, while selection of none and single component is allowed, and three compulsory components, where selection of at most two of them is allowed, then one would obtain in total 128 service options, as in Figure 4.

Enumeration of the final value of attribute options is based on the previously developed combinatorial metric of variability [24]:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} O_{ij} = \left(\sum_{j=1}^{n} C^C_{ij} \left(\frac{k!}{l!(k-l)!}\right)\right) \cdot \left(\sum_{j=1}^{n} C^D_{ij} \left(\frac{k!}{l!(k-l)!}\right)\right) = 2^3 \left(\frac{2!}{0!(2-0)!} + \frac{2!}{1!(2-1)!}\right) \cdot \left(\frac{3!}{1!(3-1)!} + \frac{3!}{2!(3-2)!}\right) = 2^3 \times (1 + 2) \times (3 + 3) = 8 \times 3 \times 6 = 128,$$

Figure 2: Procedure for the selection of optimal design platform.
where \( l \) is the number of components to be chosen from a set of all components \( k \) belonging to feature \( f_i \), component \( C_{ij}^D \) adopts the condition that \( 1 \leq l < k \), and component \( C_{ij}^C \) adopts the condition that \( 0 \leq l < k \).

Current service variety in terms of MC may involve high number of features and attributes. Then, each attribute can be represented by the different types of components. Services either have fixed features with voluntary attributes or are completely customizable. The problem of such services is in the quantification of all possible design alternatives that are understood as the design space \( n_s \). Therefore our aim is to propose a decision-making method to select optimal service platform where restrictions are predefined.

Constraints are specifying how two different options \( O_{ij} \) cooperate within single service activity architecture. A restriction rule \( R \) has two possible expressions. It can be either

(a) requiring rule; for example, \((O_{1})_{ij} \rightarrow (O_{1+r})_{ij}\), then selection of option \( O_1 \) automatically requires selection of option \( O_{1+r} \) in the same variant, or

(b) excluding rule; for example, \( r((O_{1})_{ij} \land (O_{1+r})_{ij}) \), then selection of option \( O_1 \) excludes selection of option \( O_{1+r} \) in the same variant.

Applying such restrictions within service architecture, system's design space \( n_s \) is constrained to viable design space \( n_v \).

Let us use the feature model from Figure 5 with four options on the first attribute \((O_1 \lor O_2 \lor O_3 \lor O_4) \rightarrow A_{11}\), two options at the second attribute \((O_1 \lor O_2) \rightarrow A_{12}\), and three attributes with determined options. Subsequently, we can enumerate number of variants that are very useful when designing a new service, as shown in Figure 5. Summary number of variants of the service activity is calculated as follows:

\[
    n_s = \sum_{i=1}^{m} \prod_{j=1}^{n} \left( \sum_{q=1}^{r} O_q \right)_{i,j}.
\]  

This calculation of number of variants will be further used when applying proposed decision-making algorithm.

3.5. Service Variety Quantification with Restrictions. Real world models are seldom without any restriction; they are constrained due to functionality and compatibility issues. Therefore our aim is to propose a decision-making method to select optimal service platform where restrictions are predefined.
four options at the third attribute \(\{(O_1 \lor O_2 \lor O_3 \lor O_4) \rightarrow A_{21}\}\), to determine the extent of both systems design and viable spaces \(n_s\) and \(n_v\). Three restrictions on design variants can be defined on the model for this purpose as follows:

\[
\begin{align*}
R\#1: & \rightarrow ((O_1)_{11} \land (O_1)_{12}), \\
R\#2: & \rightarrow ((O_1)_{21} \land (O_2)_{12}), \\
R\#3: & \rightarrow ((O_2)_{11} \land (O_2)_{12}).
\end{align*}
\]

Subsequently, we may create a matrix of relations among systems options using a design matrix (Figure 6), where restrictions are assigned.

Here, relations above the diagonal express all possible collaborations within a model and “\(X\)” relates to the three restriction rules \(R\#1-3\). To resolve this restricted model, pseudocode formulation of the procedure is applied (see Algorithm 1). User tool of proprietary software developed for creation of matrix and determination of \(n_s\) and \(n_v\) values can be reached online [25]. The procedure begins with input variables, namely, numbers of options per each system attribute. Subsequently, code generates an interrelation matrix \(\Sigma(O_0)_{ji} \times \Sigma(O_0)_{ij}\). At the same time, design space of the model is determined \((n_s = 32)\). Restricted model in turn provides only \(n_v = 23\) viable service variants.

### Algorithm 1: Pseudocode for quantification of design spaces \((n_s, n_v)\).

```plaintext
INPUT: \(S_n = \text{Size of each subgroup}, \ N = \text{number of subgroups}\)
\(M = \text{binary Service Model} [1 - \text{if } i\text{-th row links with } j\text{-th column}; 0 - \text{otherwise}],\)
\(GR = \text{general restriction - diagonal restriction},\)
\(SR = \text{specific restriction - off-diagonal restriction}\)

OUTPUT: \(\Sigma Var = \text{Sum of variation with specified restriction}\)

PROCEDURE Sum_Var(S_n, M, Iteration=0, Row=1, From=1)
  IF N<2 THEN RETURN \(\Sigma Var \leftarrow 0\)
  i = Row
  FOR j=1,\ldots,\(S_{Iteration}\) DO
    IF Iteration=N-2 THEN RETURN \(\Sigma Var\)
    \(\Sigma Var+=M[i,j] \times Product(M[j])\)
  UNTIL i<=\(S_{Iteration}\)
RETURN \(\Sigma Var\)
END PROCEDURE

INITIALIZATION
Sum_Var(S_n, M)
```

Each with different (potentially optimal) values of \(n_s\) and \(n_v\) of the same service activity. This is to simulate the decision-making approach proposed in Section 3.3 to determine the most optimal service variant platform on the basis of the waste entropy concept \(H_w\).

### 4. Realistic Case

In order to prove a practical importance of this research subject, it was decided to use realistic service problem situation to provide motivation for similar cases. This case model aims to demonstrate potential tool to decide about the optimal service design platform from customer perspective. Unlike simple example in Section 3.4, in realistic service activity, three service platforms \(D_{0-2}\) are gradually identified, each with different (potentially optimal) values of \(n_s\) and \(n_v\) of the same service activity. This is to simulate the decision-making approach proposed in Section 3.3 to determine the most optimal service variant platform on the basis of the waste entropy concept \(H_w\).

#### 4.1. Problem Description

The following service model is a real representation of a hotel-booking system consisting of two steps (features \(f_1\) for lodging and \(f_2\) for boarding), as seen in Figure 7. Let us present a starting platform \(D_0\), where each of the features has several compulsory attributes:

\[
\begin{align*}
\{[A_{11} \land A_{12} \land A_{13} \land A_{12}] & \rightarrow f_1\}, \\
\{[A_{21} \land A_{22}] & \rightarrow f_2\}.
\end{align*}
\]

Moreover, each of the attributes \(A_{ij}\) has several options, as a result of component combinations. In the platform \(D_0\), attribute \(A_{11}\) enables customer to book, for example, one of the four room sizes \(\{(O_1 \lor O_2 \lor O_3 \lor O_4) \rightarrow A_{11}\}\), or attribute \(A_{12}\) allows choosing smoking and nonsmoking accommodation option: \(\{(O_1 \lor O_2) \rightarrow A_{12}\}\).

Options of all individual attributes can be seen in Figure 7.

Complete design space of the platform \(D_0\) is \(n_v = 864\) variants. Platform \(D_0\), however, has nine initial restrictions \(R\#1-9\). They are formulated in the form of exclusion; for example, 4-bed room option \(O_{21}\) must not be in the same variant with superior room option \(O_{21}\). This rule restricts selection of service model variants and limits the design space \(n_s\) to \(n_v\). Then, platform \(D_0\) of case service activity offers “only” \(n_v = 288\) viable variants after application of all restrictions \(R\#1-9\).

Values of the service design spaces \(n_s\) and \(n_v\) have been obtained using the software tool described in Section 3.5.
In the second phase, alternative platform $D_1$ can be created. Let us assume that a service provider planned to optimize his service model by removing one of the attributes $A_{12}$ with two options in order to make model less differentiated but still customizable. Then the service design space has been reduced to $n_{v_1} = 432$ service variants and viable space to $n_{v_2} = 156$ service variants.

Finally, a removal of restriction of $R\#2$ has been proceeded to obtain the last platform $D_2$. After elimination of both the restriction $R\#2$ and attribute $A_{12}$, the service design space is $n_{v_2} = 432$ and $n_{v_2} = 168$ viable variants.

Then, using the software tool described in Section 3.5, we obtain values of the design spaces. Subsequently, we will apply the software tool for determination of optimal service platform, from Section 3.3, using the values of complete design spaces $n_{v_1} = 864$, $n_{v_1} = 432$, and $n_{v_1} = 432$ and viable design spaces $n_{v_2} = 288$, $n_{v_2} = 156$, and $n_{v_2} = 168$ as inputs. In the case of our service model, the most optimal platform is $D_2$.

From managerial point of view, platform $D_2$ presents more or less compromise between total reduction of infeasible variants and keeping all viable variants. Generally known, extreme solutions are not always the best. However, the method provides relevant alternative solutions with possible consequences and the final decision is up to service company management.

5. Conclusions

Effective and accurate execution of low-selling service components may bring relevant decrease of waste entropy as well as infeasible variants. Then more preferable platform of service variants can be identified. Naturally, presented decision-making algorithm for three platforms of service variants can be increased for unlimited number of platforms.

It was also proven in psychosocial domain (see, e.g., [26]) that any changes of long-term accepted rules in human behavior initiate disappointments or frustrations. These disappointments arise also when providers’ options involve infeasible variants. To avoid such customer reaction, the proposed decision-making tool could be helpful.

Competing Interests

The authors declare that there are no competing interests regarding the publication of this paper.

References


