ICED13
INTERNATIONAL CONFERENCE ON ENGINEERING DESIGN, ICED13
19-22 AUGUST 2013, SUNGKYUNKWAN UNIVERSITY, SEOUL, KOREA

GINI INDEX BASED ATTRIBUTE SELECTION FOR PRODUCT CONFIGURATOR DESIGN

Yue WANG, Mitchell TSENG
The Hong Kong University of Science and Technology, Hong Kong S.A.R. (China)

ABSTRACT

Product configurator system is an important tool to bridge customer needs and company's offering. It has been widely accepted in industry to facilitate customized product design. However the efficiency of product configuring process has been a challenging issue, especially when the product is complicated. Current configuration systems often perform in a deterministic manner. They cannot adapt to each individual customer preferences by leveraging on the preferences information captured in previous configuration steps. This paper present a Gini index based attribute selection approach for configurator design. Product configuring is modeled as a sequential query-answer process. In each configuring step, Gini index is deployed to quantify the clarity of designer's belief about the customer's needs. The attribute which contributes most to the clarity will be select for the customer to configure. As a result, designers get clear about the customer's needs and preferences in an accelerated manner. An example is presented to test the viability of the method.

Keywords: user centred design, decision making, product configurator, gini index

Contact:
Dr. Yue Wang
Hong Kong University of Science and Technology
Advanced Manufacturing Institute
Hong Kong
NA
Hong Kong S.A.R. (China)
yacewang@ust.hk

1 INTRODUCTION

Product design starts with the identification of customer needs and preferences. It has been acknowledged that the success of product design depends heavily on the understanding of voice of the customer (Griffin and Hauser, 1993). In the more and more competitive global marketplace, customers become increasingly diversified and empowered. The fulfillment of customer preferences and needs becomes a key factor for customer's purchase decision. Therefore, customer centric product design has been accepted as a viable strategy for companies to survive in the increasingly diversified and competitive marketplace (Chen et al, 2009). Recent years have witnessed a growing trend of custom products coming into market, ranging from consumer products like consumer electronics, sneakers, apparels and automobiles to industrial products like escalator, airplane, etc. Federal Reserve of Dallas reported that since 1970s, the number of product variety has increased sharply, PC from 0 to 400, car models from 140 to 260, car style from 654 to 1212 by 1998 (Cox and Alm, 1998). Customers can gain more utility from customized products than the best standard product obtainable.

While the general gist of customized product design is to satisfy individual customers' needs, the prevailing practice of customization, e.g., product family design, is to identify patterns of customer needs, along with common building blocks of product fulfillment, and in turn to reuse existing design elements for offering customer-perceived product variety. It essentially entails a configure-to-order paradigm that is built upon known design solutions and can fulfill customer needs. The fundamental assumption is that customers can realize what they want and then purchase the product which can satisfy their needs. Therefore, product configuration system can plays critical roles in this situation. One of the most cited examples is Dell Computer's online component selection system. When browsing the page, customers just select the desired components and at the end they can get the configured product. By doing so, Dell is able to deliver customized personal computers and notebooks within one week with prices lower than its mass producing competitors. Dell Computer has gained the so-called first-mover advantage and maintained high profitability and growth in a hyper-competitive industry for a long period (Magretta, 1988).

In engineering, product configuration system has been an important tool to elicit customer needs. Product configurators can be classified as rule-based, model-based and case-based according to the reasoning techniques used. The first generation of configurators is mostly rule-based. Examples include R1 (McDermott, 1980), Cossack (Frayman et al, 1987), BLADES (Elturky et al, 1986) and MICON (Birmingham, 1988). But if the configuration system is complicated, they often suffer from the maintenance issues due to the lack of separation between domain knowledge and control strategy. Mittal and Frayman first proposed configurator as a Constraint Satisfaction Problems (CSP) by assigning values to all the variables without violating any constraints (Mittal et al, 1989). Later this approach is extended to handle different scenarios such as conditional CSP (Gelle et al, 2003), Dependent CSP (Xie et al, 2004) etc. Case-based configurator system retrieves a similar product configuration from a case base and then adjusts it to customers' particular preferences and requirements. Thus the key issue is how to retrieve the best configuration from the database and identify aspects which cause violation of constraints or requirements (Wielinga, 1997). Critiquing is also a case-based configuration and recommendation systems by leveraging on customers' feedback information (Burke et al, 1997, 2002). Customers only need to indicate a directional preference for a feature instead of inputting detailed feature values. The potentially accepted configurations will be retrieved and adapted to the preferences direction. Recent research has also been focused on improving the efficiency of the communication with customers (McSherry, 2004; Wang et al., 2011).

However there are some limitations about product configuring process. Customers need to have certain expertise about the product to accurately express their needs in design parameter domain. Huffman and Kahn investigated customer choice behavior when facing product with high variety in configuring process (1998). They found that customers can get frustrated when they are confused by the amount of product variants or not familiar with the choices. In addition, most current product configurator systems contain a fix query sequence. It cannot adapt to the active customer's specification. Therefore it passively receives preferences information from customers. All these factors hinder the efficiency of product configuration process, especially when the product is complicated.

Towards this end, this paper presents a new attribute selection approach for product configurator design. The objective is to get customers' preferences and needs information more efficiently by product configurators. Gini index is used to measure the "pureness" of the unspecified attributes set.

We want that after a customer gives one more specification to an attribute, the designer is much clearer about the customer's needs and preferences. In each configuration step, the product attribute which can lead to the highest expected pureness improvement will be selected for the customer to configure. In this way, the configuring process is not a one-way information flow from customer to designer, but a bi-direction information flow process. The selection of new attributes depends not only on the prior knowledge of general customer preferences pattern, but also the active customer's specifications to previous attributes.

The paper is organized as follows; Gini index based attribute selection is introduced in section 2. An illustrative example is shown in section 3 to validate the proposed approach. Section 4 will conclude the whole paper.

2 GINI INDEX BASED ATTRIBUTE SELECTION FOR CONFIGURATOR

2.1 Introduction

As it has been acknowledged that customers may not be patient enough to specify a long list of attributes, the main consideration in product configuration process is how to elicit customer needs efficiently and with fewer burdens to customers (Wang et al., 2012). The attributes to be configured differ a lot in terms of the usefulness they can provide. Therefore if we can select the most relevant attribute for the customer to configure at each stage, the configuration efficiency will be improved. We suppose that in each round only the most relevant item is proposed from unspecified components pool for the customer to configure. Thus the product specification definition process corresponds to a sequence of Q&A procedure.

2.2 Gini index

The Gini index is a measure of statistical dispersion which quantifies the inequality among values of a frequency distribution (Gini, 1909). Since its introduction in early 1900, it has been applied in many areas such as sociology, economics, health science, ecology, chemistry, engineering and agriculture. In computer science, Gini index is applied in some decision tree learning algorithm to select the split nodes such as CART (Breiman et al, 1984). The problem of attribute selection for configurator design has great similarity with decision tree learning problem. Attribute selection can be considered as a decision making process during configuring procedure. Thus we can leverage on the Gini index to help select the appropriate attribute for customers to specify in a similar way as decision tree learning process. The main difference for attribute selection in this paper is that the decision process is performed for each customer, instead of building a configurator for all the customers like decision tree learning problem. Thus the configurator itself is customized based on the particular customer's preferences.

Gini index can be calculated by summing the probability of each attribute being chosen multiplies the probability of a mistake in selecting that item. Therefore Gini index measures how often a randomly chosen element from the set would be incorrectly labeled if it were randomly labeled according to the distribution of labels in the subset. Another explanation of Gini index is that it quantifies the "pureness" of the data set. Coming back to the attribute selection task for product configurator, we want to achieve the maximal improvement of the incorrect label after configure an attribute to make the customers' preferences more clear to designers.

To compute Gini index for a set of attribute, we need some prior knowledge about the probability that each end product will meet the customer's needs. If we have some existing configuration data, the corresponding probabilities can be estimated by the frequencies. i.e., we use the fraction of product i in the data set to represent the probability the product i will meet the customer' s needs at configuration stage t. In a formal way,

$$G_t = \sum_{i=1}^{n} p_{i,t}(1-p_{i,t}) = \sum_{i=1}^{n}(p_{i,t} - p_{i,t}^2) = 1 - \sum_{i=1}^{n} p_{i,t}^2 \tag{1}$$

where n is the number of product in the product family.

For the attribute selection task in configurator design, we wish to select the attribute which can maximize the reduction in Gini index for customer to configure. Therefore the attribute selection procedure can be summarized as follows;

For t=1:n

- At stage t, put all the unconfigured attributes into candidate set CS_t.
- Update the parameters $p_{i,t}$ which is the fraction of product *i* in the data set.
- Calculate the Gini index in the product family according to equation (1)
- For j=1:n
- Calculate the weighted sum of the corresponding Gini index if the jth attribute is configured
- Find the attribute which leads to maximal Gini index reduction and ask the customer to configure the corresponding attribute
- Get the customer's specification and remove the attribute from the candidate set CS_t.

3 AN ILLUSTRATIVE EXAMPLE

To demonstrate the validity of the approach, we use a simplified PC as an example to illustrate the idea. The set of components and their alternatives are listed in Table 1. Here we use a six-tuple to represent one PC configuration. For example, (1,2,2,3,2,2) stands for the configuration containing the components A1, B2, C2, D3, E2 and F2. A survey was conducted in an East Asian university and 69 customers' preferred configurations data were obtained. The corresponding probabilities $p_{i,t}$ which is the fraction of product *i* in the data set can be estimated. Due to the limit of pages, the detailed probabilities are omitted here.

Table 1. List of Components and their alternatives for PC

Component	**Code**	**Description**
Processor (A)	A1	Intel(R) Core(TM)2 Duo 3.16G
	A2	Intel(R) Core(TM)2 Duo 2.66G
	A3	Intel(R) Core(TM)2 Duo 2.8G
	A4	Intel(R) Pentium(R) Dual-Core 2.6G
	A5	Intel(R) Core(TM) 2 Quad Processor 2.5G
	A6	Intel(R) Core(TM) 2 Quad Processor 2.6G
Memory (B)	B1	2GB DDR2
	B2	4GB DDR2
	B3	6GB DDR2
	B4	8GB DDR2
Monitor (C)	C1	17' LCD
	C2	19' LCD
	C3	20' LCD
	C4	22' LCD or above
Hard Disk (D)	D1	160 GB
	D2	250 GB
	D3	500 GB
	D4	750 GB
Disk Driver (E)	E1	16X DVD+/-RW*
	E2	Blu-ray Disc
	E3	Blu-ray Disc + 16X DVD+/-RW*
Display Card (F)	F1	Intel(R) GMA 3100
	F2	512MB NVIDIA(R) GeForce(R) 9800GT
	F3	256MB ATI Radeon HD 3450 LE
	F4	256MB ATI Radeon HD 3650

	F5	512MB ATI Radeon HD 4670

Suppose a new customer's target configuration is (1,2,2,3,2,2) that is unknown to designers before the product configuring process. Then the reduction for Gini index of attribute A can be calculated as follows,

$$\Delta_A(Gini) = Gini_1(CS) - Gini_1(CS \mid A) =$$
$$(1-\sum_{i=1}^{69} 0.014\cdot 0.014) - \left(0.246\left(1-\sum_{i=1}^{17}0.059\cdot 0.059\right)+0.159\left(1-\sum_{i=1}^{11}0.091\cdot 0.091\right)+0.246\left(1-\sum_{i=1}^{17}0.059\cdot 0.059\right)+\right.$$
$$\left.0.087\left(1-\sum_{i=1}^{6}0.167\cdot 0.167\right)+0.159\left(1-\sum_{i=1}^{11}0.091\cdot 0.091\right)+0.101\left(1-\sum_{i=1}^{7}0.143\cdot 0.143\right)\right)$$
$$=0.0748$$

Similarly, we can get the reduction of Gini index for other attributes

$$\Delta_B(Gini) = Gini_1(CS) - Gini_1(CS \mid B) = 0.0455$$
$$\Delta_C(Gini) = Gini_1(CS) - Gini_1(CS \mid C) = 0.0445$$
$$\Delta_D(Gini) = Gini_1(CS) - Gini_1(CS \mid D) = 0.0445$$
$$\Delta_E(Gini) = Gini_1(CS) - Gini_1(CS \mid E) = 0.0300$$
$$\Delta_F(Gini) = Gini_1(CS) - Gini_1(CS \mid F) = 0.0589$$

Therefore, atttibute A can lead to the biggest Gini index reduction. A should be present for the customer to configure first. The customer selects alternative 1 because his target product is (1,2,2,3,2,2). Then the whole solution space is reduced to all the configurations with attribute A being the first choice.

In the next configuring step, the whole previous procedure is repeated. The corresponding reductions of Gini index for the remaining attributes are

$$\Delta_B(Gini) = Gini_2(CS \mid A=1) - Gini_2(CS \mid A=1,B) = 0.1176$$
$$\Delta_C(Gini) = Gini_2(CS \mid A=1) - Gini_2(CS \mid A=1,C) = 0.1765$$
$$\Delta_D(Gini) = Gini_2(CS \mid A=1) - Gini_2(CS \mid A=1,D) = 0.1765$$
$$\Delta_E(Gini) = Gini_2(CS \mid A=1) - Gini_2(CS \mid A=1,E) = 0.4412$$
$$\Delta_F(Gini) = Gini_2(CS \mid A=1) - Gini_2(CS \mid A=1,F) = 0.2353$$

Therefore the second attribute to be configure is E. The customer selects alternative 2 based on his/her preferences. Then the third attribute to be configured is selected to minimize $Gini_3(CS \mid A=1, E=2) - Gini_3(CS \mid A=1, E=2, X)$ where $X \in \{B, C, D, F\}$.

By following the calculation procedure and based on the customer's specification, the configuring sequence for this particular customer is AEFBCD. The calculation details are omitted here.

Customer preferences may differ from individual to individual. The Gini index based configurator can present the most suitable attribute for a customer to configure based on his specifications in previous steps. Thus the configuring sequence may vary a lot for different customers. The sequence is adaptive to customers' preferences and needs.

3 CONCLUSION

Customer centric design is gaining more attentions nowadays for companies to gain advantage in the more and more competitive global marketplace. To efficiently elicit customer needs, product configurators have been widely accepted. This paper is concerned with the efficiency of configurator design, particularly the order of attribute selected for customers to configure to reduce the communication process between customers and designers. A Gini index based attribute selection approach is proposed. The main contribution can be summarized as follow;

- The configuring process is considered as a sequential query and answer process. In each configuring step, Gini index is deployed to quantify how unclear the designer's is about the customer's needs. The attribute which can reduce the most "unclearness" will be select for the customer to configure. As a result, designers get clear about the customer's needs and preferences in an accelerated manner.

- Customized configurator query sequence adapted to each individual's needs is presented in the configuring process, meaning that different customers may have different set of questions and their sequences. Thus it is not a one-direction preference information flow process but a bi-directional process. Designers' expertise or prior knowledge on customer preferences will be incorporated in the preferences elicitation process.

REFERENCES

Birmingham W., B. A. and Siewiorek D. (1988) MICON: A single board computer synthesis tool. *IEEE Circuits and Devices Magazine*, Vol. 4, No.1, pp. 37-46.

Breiman L., J. Friedman, R. Olshen, and Stone C. (1984) *Classification and regression trees*. Wadsworth International Group

Burke, R. (2002) Interactive Critiquing for Catalog Navigation in E-Commerce. *Artificial Intelligence Review*, Vol. 18, No.3-4, pp. 245–267.

Chen, S. L., Wang, Y. and Tseng, M. M. (2009) Mass Customization as a Collaborative Engineering Effort. *International Journal of Collaborative Engineering*, Vol. 1, No. 2, pp. 152-167

Cox, W. M. and Alm R. (1998) The right stuff: America's move to mass customization, *1998 annual report of Federal Reserve Bank of Dallas, Dallas*, TX.

Elturky, F. and Nordin R. (1986) BLADES: an expert system for analog circuit design, *Proceeding of the IEEE Symposium on Circuit and System*, San Jose, CA, pp. 552-555.

Frayman F. and Mittal S. (1987) Cossack: A Constraint based expert system for configuration task. In: Sriram, D. and Adey, R. ed. *Knowledge-based Expert Systems in Engineering: Planning and Design*. Boston: Computational Mechanics Publication, pp. 143-166.

Gelle, E. and Faltings B. (2003) Solving mixed and conditional constraint satisfaction problems. *Constraint*, Vol. 8, No. 2, pp. 107-141.

Gini, C. (1909) Concentration and dependency ratios (in Italian). English translation in Rivista di *Politica Economica*, Vol. 87 (1997), pp. 769-789.

Griffin, A. and Hauser, J. (1993) The Voice of the Customer. *Marketing Science*, Vol. 12, No.1, pp. 1-27 (Winter).

Huffman, C. and Kahn B.E. (1998) Variety for Sale: Mass Customization or Mass Confusion? *Journal of Retailing*, Vol. 74, No.4, pp. 491-513

Magretta, J., (1988) The power of virtual integration: an interview with Dell Computer's Michael Dell. *Harvard Business Review*, Vol. 76, pp.73-84.

McDermott, J., (1980) R1: An Expert in the Computer Systems Domain, In: *Proceedings of the 1st Annual National Conference on Artificial Intelligence*, Stanford University, California: Stanford University, pp. 269-271.

McSherry, David, (2003) Minimizing dialog length in interactive case-based reasoning, *Proceeding IJCAI'01 Proceedings of the 17th international joint conference on Artificial intelligence*, pp 993-998

Mittal, S. and Frayman F. (1989) Towards a generic model of configuration task. *Proceedings of International Joint Conference on Artificial Intelligence*. Calilfornia: Morgan Kaufmann, pp. 1395-1401.

Wielinga, B. and Schreiber B. (1997) Configuration-Design Problem Solving. *IEEE Expert: Intelligent Systems and Their Applications.* Vol. 12, No. 2, pp. 49-56

Wang, Y., and Tseng M. M. (2011) "Adaptive Attribute Selection for Configurator Design via Shapley Value", Artificial Intelligence for Engineering Design, Analysis and Manufacturing, Vol 25 (1), 189–199.

Wang, Y., and Tseng M. M. (2012) "Customized products recommendation based on probabilistic relevance model". Journal of Intelligent Manufacturing, (Accepted, DOI 10.1007/s10845-012-0644-7)

Xie, H., P. Henderson, J. Neelamkavil, and J. Li, (2004) "A systematic search strategy for product configuration", Proceedings of the 17th international conference on Innovations in applied artificial intelligence. Springer, pp. 718-727