On the Quantification of Interface Design Architectures

Pandremenos J.
Laboratory for Manufacturing Systems and Automation, University of Patras, Rio, Patras 26500, Greece

Chatzikomis C.
Laboratory for Manufacturing Systems and Automation, University of Patras, Rio, Patras 26500, Greece

Chryssolouris G.
Laboratory for Manufacturing Systems and Automation, University of Patras, Rio, Patras 26500, Greece

Abstract
A modular architecture presumes the assignment of a single function, per part, within the product and specifies decoupled interfaces among the components. Since the trend is for designs to be produced with a specific level of modularity, the necessity of quantifying this design approach has arisen. This study focuses on the interface modularity, since interfaces can affect significantly the design process by imposing technical constraints. This paper describes a method of quantifying and evaluating the interface structure with the help of Design Structure Matrices (DSM). This method is then applied to a case study for the examination of its application.

Keywords: *Modularity quantification, design, interface architecture*

1 Introduction

Modularity is a design approach that generates product designs of low complexity and consequently, of high flexibility. According to Chryssolouris [1], the higher the flexibility is, the lower the sensitivity of cost to change. Since the latter is the basic requirement for a product's personalisation, being the trend nowadays, it is explained why modularity is so widely examined over the last years.

A modular design architecture is achieved by assigning each of the product's functional requirements to a single component, and at the same time, by maintaining uncoupled interfaces among the parts. The exactly opposite design architecture is called integral and presents a coupled functions' mapping as well as interfaces [2]. Whilst in the past, the designers were trying to create products with the features of the one or the other architecture, nowadays, this binary logic has been surpassed and the designers are making products of combined architectures. Thus, in order for the modularity level of a design to be controlled, the necessity for quantifying modularity has arisen.

Quantification of modularity can be studied either from the functions mapping or the interface's structure, point of view. In the current work, modularity is investigated only in terms of the second requirement.

1.1 *Interface architecture*

The interface architecture indicates the connection among the parts of a product. When each component is connected with all the others, within a product, the architecture is considered Fully Integral. Furthermore, a product with the minimum number of interfaces is considered either Bus or Fully Modular. In the former case there is a common bus to which the other parts are connected whilst in the latter, the parts are connected sequentially. In all other cases, the interface architecture is not clear and this is one of the reasons that quantification measures are developed.

1.1.1 Design Structure Matrix

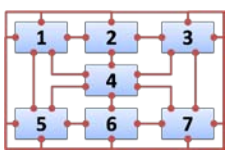
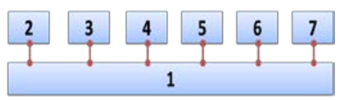
The Design Structure Matrix (DSM) is used for the better representation of a system's interface architecture. Through this visualization facility, the designer is capable of better controlling the modularity of the product, with regard to the interface complexity.

These matrices are binary in general, square and contain the name of the system's elements down the side (as row headings) and across the top (as column headings). Whilst a link exists between node i to node j, the value of the ij element is unity or it is marked with X, otherwise, the element value is zero or it is left empty. Finally, the diagonal elements of such matrices have usually a zero value or they are left empty as well, since they do not play any role within the matrix [3]. Hölttä-Otto and de Weck [4] describe the DSMs of a fully "integral", a "bus-modular" and a fully "modular" system of seven components (Table 1).

DSMs are in general quite simple to be produced, however they may be time consuming for products consisting of many components.

Within the next sections of the paper, the interface modularity is investigated. The main existing quantification methods are reviewed, assessed and a new one is introduced. The application of the method is firstly shown in some examples and then in a case study such as a "state of the art" car body in white. The results are finally evaluated and conclusions on the method are drawn.

Table 1: DSMs for different interface architecture [4]

Interface architecture	**Fully integral**	**Bus-modular**	**Fully modular**
Graph			
DSM	$\begin{bmatrix} 0&1&1&1&1&1&1 \\ 1&0&1&1&1&1&1 \\ 1&1&0&1&1&1&1 \\ 1&1&1&0&1&1&1 \\ 1&1&1&1&0&1&1 \\ 1&1&1&1&1&0&1 \\ 1&1&1&1&1&1&0 \end{bmatrix}$	$\begin{bmatrix} 0&1&1&1&1&1&1 \\ 1&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0 \\ 1&0&0&0&0&0&0 \end{bmatrix}$	$\begin{bmatrix} 0&1&0&0&0&0&0 \\ 1&0&1&0&0&0&0 \\ 0&1&0&1&0&0&0 \\ 0&0&1&0&1&0&0 \\ 0&0&0&1&0&1&0 \\ 0&0&0&0&1&0&1 \\ 0&0&0&0&0&1&0 \end{bmatrix}$

2 Quantification of modularity

2.1 *Existing methods*

A number of methods, evaluating a product design through the quantification of the interface modularity, have been developed over the last years. An aspect that should be taken into consideration, during sorting out these methods, is the use of subjective or objective criteria for the evaluation.

2.1.1 Methods involving objective criteria

Hölttä-Otto et al. [5] used Singular Value Decomposition (SVD) on DSMs in order to derive the Singular Value Matrix and there from to extract the Singular value Modularity Index (SMI), an index that measures modularity, based on the internal connections of the product. The same authors developed the formula for SMI [3] even further and also introduced the Packaging Factor, which is the ratio of the unused volume to the total volume of the product. A relatively simple quantification method has been proposed by Ulrich [4], the component to function ratio. Yu et al. [6] combined DSM with Genetic Algorithms and the Information theory, in order to optimize an initial DSM through clustering. The Minimum Length Description of each cluster was used as a criterion for the improvement of the clusters. DSM representation has also been used by Guo and Gershenson [7], in order to calculate their metric for modularity. Their method quantifies modularity by measuring the internal interaction of each module's components as well as the interaction of the module itself. Since the module boundaries appear to rely on the designer's judgment, this method could also fit in the category of the quantification methods with subjective criteria. Finally, Ericsson and Erixon [8], have defined the ideal number of modules as the square root of the number of assembly operations, taking into consideration the average assembly time for each operation.

2.1.2 Methods involving subjective criteria

Fixson [9] used interfacial information, like component interaction type and interaction intensity, along with disassembly easiness [6], in order to depict the level of modularity. The values of the factors that are used are clearly subjective. Hölttä [10] based her evaluation of the "design effort complexity", on the interviews she conducted with design engineers as a means of interfacial complexity. Asan et al. [11] quantified the desired amount of modularity, by analyzing the feedback from questionnaires they had prepared, with direct and in-direct questions. The direct questions were referring to the design rules of modularity and the indirect ones to the outputs of modularity. Finally, ElMaraghy et al. [12] developed a complexity quantification model that takes into consideration a number of product interfacial parameters. These parameters are weighted with the use of a self-defined ranking system.

2.1.3 Assessment of existing methods

The advantages and disadvantages of the existing methods are presented in Table 2. As it can be remarked, few objective methods exist and most of them are either inaccurate or difficult to be applied.

2.2 *Proposed method*

The method described hereafter, consists of a number of steps and is based only on objective criteria trying to fill the gaps of the existing methods. DSM has been extensively used in the method in order for information about the product's interface structure to be extracted.

Step 1: Product's DSM formation

Since the whole approach is DSM dependent, the initial step of the method includes the construction of the product's DSM.

Step 2: Identification of the max and min number of the product's interfaces

A very useful piece of information that can be derived while observing a DSM, is the minimum and maximum numbers of interfaces, for a design with a specific number of components:

$$I_{\max} = \frac{n^2 - n}{2} \tag{1}$$

$$I_{\min} = n - 1 \tag{2}$$

where I_{max} the maximum number of interfaces, I_{min} the minimum number of interfaces and n the number of components.

For instance, for a DSM with $n = 7$ (Table 1), I_{max} will be 21 and I_{min} 6. It can be easily noticed that a design with I_{max} is Fully Integral while a design with

Table 2: Evaluation of existing quantification methods

<table>
<tr><th>Method</th><th>Advantages</th><th colspan="2">Disadvantages</th></tr>
<tr><td colspan="4">Objective Methods</td></tr>
<tr><td>Singular value Modularity Index (Hölttä-Otto et al)</td><td>Offers consistent results</td><td colspan="2">In some cases, the results contradict to logic (e.g. Figure 1: DSM f appears more modular than b)</td></tr>
<tr><td>Packaging Factor (Hölttä-Otto and de Weck)</td><td>Easy to apply</td><td colspan="2">Simplistic, not accurate approach</td></tr>
<tr><td>Component to Function Ratio (Ulrich)</td><td>Easy to apply</td><td colspan="2">Simplistic, not accurate approach</td></tr>
<tr><td>DSM – Genetic Algorithms – Information Theory (Yu et al.)</td><td>Very effective for clustering</td><td colspan="2">- Cannot compare different DSMs
- Difficult to apply</td></tr>
<tr><td>Guo and Gershenson’s metric</td><td>Offers consistent results</td><td colspan="2">Subjective evaluation of module boundaries</td></tr>
<tr><td>Ericsson and Erixon’s “ideal” number of modules</td><td>Easy to apply</td><td colspan="2">Simplistic, not accurate approach</td></tr>
<tr><td colspan="4">Subjective methods</td></tr>
<tr><td>Fixson’s DSMs</td><td>Considers the various coupling natures</td><td></td><td rowspan="4">- Time consuming
- Expert design engineers required
- Subjective evaluation</td></tr>
<tr><td>Design Effort Complexity (Hölttä and Otto)</td><td>Depicts the various interactions’ flows</td><td rowspan="2">Qualitative evaluation</td></tr>
<tr><td>Evaluation of degree of modularity (Asan et al.)</td><td>Determines the ideal level of modularity for a design</td></tr>
<tr><td>Product Complexity Index (ElMaraghy and Urbanic)</td><td>Offers quantitative data</td><td></td></tr>
</table>

I_{min} can be either Bus or Fully Modular. It should be noted that equations (1) and (2) are valid with the assumption that all the parts of a product have at least one interface.

Step 3: Calculation of the “Modularity Performance”

The “Modularity Performance” (*MP*) is introduced as a quantification measure of the interface modularity. *MP* derives from the normalization of I_{max} and I_{min} values in order for the same boundaries to be always maintained, between 0 and 1 (I_{max} and I_{min} values depend on *n* and therefore, are not constant).

$$MP = 1 - \frac{I - I_{\min}}{I_{\max} - I_{\min}} \quad (3)$$

Where *I* the number of interfaces.

It is obvious that when $I = I_{max}$, $MP = 0$ (Integral design) and when $I = I_{min}$, $MP = 1$ (Modular design).

Step 4: Identification of the standard deviation of the number of interfaces per part

Although the *MP* can measure the level of modularity effectively, it cannot indicate the type. For example, in the case of $MP = 1$ where two different DSMs may exist (with the same *n* and *I*), the measure cannot define which one is the Bus and which is the Fully Modular. Therefore, an additional measure is required, capable of evaluating the modularity type of the DSM. In order for this evaluation to be performed, the dispersion of the number of interfaces per part should be identified. The measure that calculates this dispersion is the “Standard Deviation” (*STD*):

$$STD = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(p_i - \overline{p})^2} \quad (4)$$

$$\bar{p} = \frac{1}{n}\sum_{i=1}^{n} p_i \qquad (5)$$

Where p_i the number of interfaces in the ith part and $\bar{p}$ the arithmetic mean of the number of interfaces per part.

Step 5: Calculation of STD_{max} and STD_{min}

The calculation of the maximum and minimum *STD* of the number of interfaces per part, for a product with specific *n* and *I*, can be determined by creating two DSMs *n* x *n* with *I* interfaces. In the first DSM all the interfaces will be equally distributed among the parts (interfaces close to the diagonal of the matrix) and this will give the STD_{min} (Fully Modular oriented). The other DSM will consist only of bus-parts where all the interfaces will be placed among these parts. This DSM will give the STD_{max} (Bus Modular oriented).

Step 6: Evaluation of the modularity type

The evaluation of the modularity type occurs by the comparison of the *STD* with the STD_{max} and STD_{min}. If *STD* is close to STD_{max} then the product presents a Bus Modular oriented architecture. On the other hand, if *STD* is close to STD_{min} then the product will be dominated by a Fully Modular architecture.

STD value can be normalized according to equation (3) in order for STD_{max} to give 0 and STD_{min} 1.

2.2.1.Example

Figure 1, presents the DSMs of different interface architectures of a product with *n = 7* parts. For each DSM the total number of interfaces per part is summed and the *I*, *MP* and *STD* are calculated. The results from a, b and c matrices have been already discussed in previous paragraphs. However, comparing the d, e and f matrices, the following observation may be made:

- All matrices have the same *I* and therefore *MP*
- The d matrix, which consists of two main bus-parts (3 and 5), has the higher *STD* (STD_{max}) and correctly presents a Bus-Modular lay out
- The e matrix, with one main bus-part (1) and the other interfaces, randomly placed in the matrix, has a smaller *STD* compared with that of the d matrix
- The f matrix, with all the interfaces concentrated round the main diagonal of the matrix, has the lower *STD* (STD_{min}) and it obviously presents a Fully Modular oriented architecture.

a. FULLY MODULAR

	1	2	3	4	5	6	7
1		1					
2	1		1				
3		1		1			
4			1		1		
5				1		1	
6					1		1
7						1	
SUM	1	2	2	2	2	2	1
I							6
MP							1
STD							0,451753951

b. BUS MODULAR

	1	2	3	4	5	6	7
1			1				
2			1				
3	1	1		1	1	1	1
4			1				
5			1				
6			1				
7			1				
SUM	1	1	6	1	1	1	1
I							6
MP							1
STD							1,749635531

c. FULLY INTEGRAL

	1	2	3	4	5	6	7
1		1	1	1	1	1	1
2	1		1	1	1	1	1
3	1	1		1	1	1	1
4	1	1	1		1	1	1
5	1	1	1	1		1	1
6	1	1	1	1	1		1
7	1	1	1	1	1	1	
SUM	6	6	6	6	6	6	6
I							21
MP							0
STD							0

d. RANDOM I

	1	2	3	4	5	6	7
1			1		1		
2			1		1		
3	1	1		1	1	1	1
4			1		1		
5	1	1	1	1		1	1
6			1		1		
7			1		1		
SUM	2	2	6	2	6	2	2
I							11
MP							0,666666667
STD							1,807015806

e. RANDOM II

	1	2	3	4	5	6	7
1		1	1	1	1	1	1
2	1				1		
3	1					1	
4	1					1	1
5	1	1					
6	1		1	1			1
7	1			1		1	
SUM	6	2	2	3	2	4	3
I							11
MP							0,666666667
STD							1,355261854

f. RANDOM III

	1	2	3	4	5	6	7
1		1	1				
2	1		1	1			
3	1	1		1	1		
4		1	1		1	1	
5			1	1		1	1
6				1	1		1
7					1	1	
SUM	2	3	4	4	4	3	2
I							11
MP							0,666666667
STD							0,832993128

Figure 1: DSMs of different interface architectures.

3 Case study

A “state of the art” car body in white was selected as a test case in order for the proposed quantification method to be applied. The interface structure of the main body parts is represented in the DSM of Figure 2. A couple of assumptions were made during the formulation of the DSM, so as for the number of parts to be reduced and thus, to be made smaller. First of all, the parts that are symmetrical in the body (e.g. the parts of the body side section) were excluded from the matrix. Additionally, many parts were grouped together and were represented as one part in the DSM. These assumptions led to a reduction from 127 to 38 parts. Finally, the total number of interfaces (*I*) was calculated to 108.

The quantification method was then applied to the car body in white DSM. Moreover, the SMI of Hölttä-Otto et al. [5] was calculated and normalized according to eq.(3), for comparison with the MP. The reason SMI was selected, is that this index measures modularity from exact the same perspective as MP.

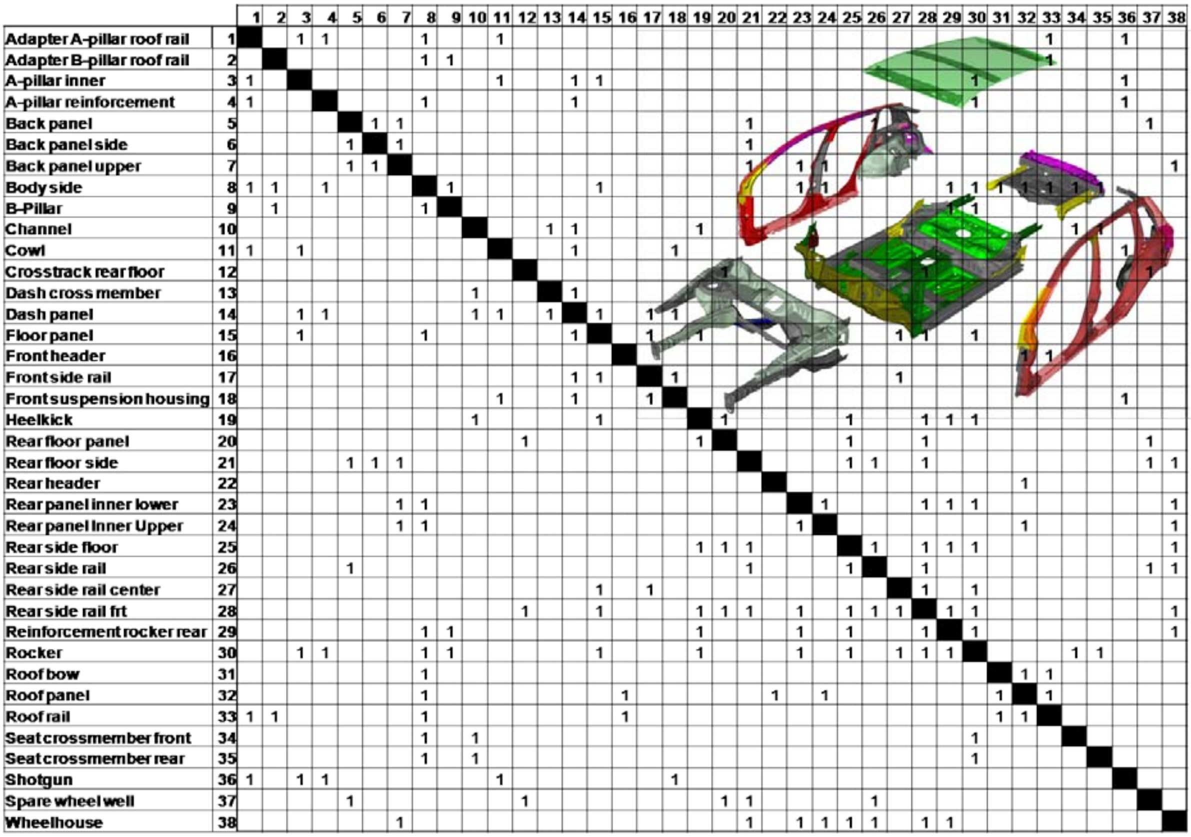
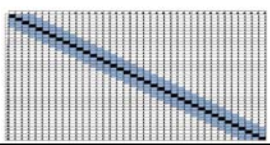
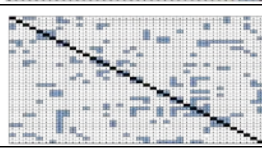
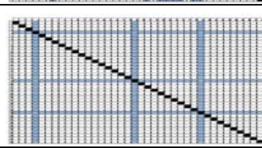


Figure 2: DSM of a car's body in white parts

Table 3: Quantification measures of the car body in white DSM

Measure	Value
n	38
I	108
I_{max}	703
I_{min}	37
MP	0.89
SMI	0.91
STD	2.84

Table 4: STD, STD_{max} and STD_{min} for car body in white DSM(n = 38 and I = 108)

DSM	STD	Normalized STD
	0.8	1
	2.84	0.24
	9.17	0

From the results presented in Table 3, a quite modular design can be noticed since both *MP* and *SMI* converge to a close-to-one value. In order to evaluate the calculated *STD*, the STD_{max} and STD_{min} for a design with n = 38 and I = 108 were determined. Moreover, all *STD* values were normalized according to equation (3) for easier assessment. The results from the aforementioned procedure are shown in Table 4. It can be easily observed that the car body presents the Bus Modular oriented architecture.

4 Conclusions – Future work

The review of existing interface modularity quantification methods, revealed the absence of an efficient objective approach. The method proposed in this paper, takes into consideration the number of interfaces within a product as well as the standard deviation of the number of interfaces per part. Excluding the time required for the formulation of the DSM, which generally is ample for large DSMs, the application of the method is relatively simple and fast. The implementation of the measures into a couple of examples and into a test case as well as the comparison of *MP* with the *SMI* (which showed only a 2.24% deviation), has proven their reliability. Furthermore, it can be easily concluded that there is no point in calculating *STD* for DSMs with low *MP*.

The implementation of the method to a "state of the art" car body in white, revealed a high level of modularity and specifically, a Bus Modular oriented architecture. Such an architecture was expected for the car body design, since the current trend in the automotive industries is the utilization of platform (Bus) parts [13]. They can therefore produce many model variants quickly and at the same time, at low cost, by utilizing common parts between them.

The authors' intention is to further investigate and validate the proposed method by applying it to more products. Moreover, a limit for *MP* will be examined under which the calculation of *STD* would be pointless. Finally, an automated approach integrated within a CAD software, capable to extract the DSM of a product design would facilitate the method.

Acknowledgments

The work was partially supported by the EU funded project: FP6-2004-NMP-NI-4-026621, Multi-FUnctional maTerials and related prodUction technologies integRated into the Automotive industry of the future –FUTURA".

References

[1] Chryssolouris G., 2006, *Manufacturing Systems: Theory and Practice 2nd edition, Springer,* New York.

[2] Ulrich K. T., 1995, The Role of Product Architecture in The Manufacturing Firm, *Research Policy,* 24:419-440.

[3] Hölttä-Otto K. and de Weck O., 2007, Degree of Modularity in Engineering Systems and Products with Technical and Business Constraints, *Concurrent Engineering*, 15(2):113-126.

[4] Ulrich K. T. and Eppinger S. D., 2003, *Product Design and Development*, 3rd Edition, McGraw-Hill, New York.

[5] Hölttä-Otto K., Suh E.S. and de Weck O., 2005, *Tradeoff between Modularity and Performance for Engineered Systems and Products*, Proceedings of ICED 05, Melbourne.

[6] Yu T., Yassine A. and Goldberg D., 2007, An Information Theoretic Method for Developing Modular Architectures using Genetic Algorithms, *Research in Engineering Design*, 18:91-109.

[7] Guo F. and Gershenson J. K., 2004, *A Comparison of Modular Product Sesign Methods on Improvement and Iteration*, In Proceedings of the 2004 ASME Design Engineering Technical Conferences, Salt Lake City, UT.

[8] Ericsson A. and Erixon G., 1999, *Controlling Design Variants*, Society of Manufacturing Engineers, Dearborn, Michigan.

[9] Fixson S., 2005, Product Architecture Assessment: A Tool to Link Product, Process, and Supply Chain Decisions, *Journal of Operations Management*, 23:345-369.

[10] Hölttä K. and Otto K., 2004, Incorporating Design Effort Complexity Measures in Product Architectural Design and Assessment, *Design Studies*, 5:463-485.

[11] Asan U., Polat S. and Serdar S., 2004, An Integrated Method for Designing Modular Products, *Journal of Manufacturing Technology Management*, 15(1):29-49.

[12] ElMaraghy W. H. and Urbanic R. J., 2003, Modelling of Manufacturing Systems Complexity, *CIRP Annals - Manufacturing Technology*, 52(1):363-366.

[13] Pandremenos J, Paralikas J., Salonitis K. and Chryssolouris G., 2009, Modularity Concepts for the Automotive Industry: A Critical Review, *CIRP Journal of Manufacturing Science and Technology*, 1:148-152.

