Available online at www.sciencedirect.com

ScienceDirect

Procedia CIRP 44 (2016) 55 – 60

www.elsevier.com/locate/procedia

6th CIRP Conference on Assembly Technologies and Systems (CATS)

Optimum Overall Product Modularity

Mohamed Kashkoush and Hoda ElMaraghy*

Intelligent Manufacturing Systems (IMS) Centre, University of Windsor, 401 unset Avenue, Windsor, ON, N9B 3P4 Canada

* Corresponding author. *E-mail address:* hae@uwindsor.ca

Abstract

Modularity in product architecture is beneficial to both product development and manufacturing. Several methods exist for clustering product components into modules all of which, with few exceptions, do not consider the hierarchical structure of the product. Products architecture consists of a number of hierarchical levels, which add a useful dimension to modularity analysis. Designing products architecture that maximizes modularity over all levels of the product structure (i.e. overall modularity) is the main objective of this work. Interactions between various product components are represented using a Design Structure Matrix (DSM). The product architecture is represented by product structure tree in the form of a binary rooted tree. A novel Mathematical Programming Model is developed to construct the corresponding product structure tree for a given product which ensures optimal modularity at all hierarchical levels, without prior knowledge of their number. The proposed optimal modular product architecture design method is demonstrated using a real product case study. Optimal overall modularity leads to better management of product changes and variety and more cost-effective product development and manufacturing.

Peer-review under responsibility of the organizing committee of the 6th CIRP Conference on Assembly Technologies and Systems (CATS)

Keywords: Modularity; Product Architecture; Granularity; Design Structure Matrix; Integer Programming

1. Motivation

Constructing the product architecture is a critical design activity that affects all subsequent product development steps such as analysis, modeling and prototyping, manufacturing, assembly and supply chain. A widely accepted definition of product architecture is: “the organization of the functional components of a product into physical units and the interaction among them” [1]. Products with integral architecture are constructed with a combination of the lowest level (individual) components without intermediate sub-assemblies; while products with modular architecture are built of one or more levels of sub-assemblies [2]. A module in a modular product refers to a structurally independent element or a sub-assembly of a larger sub-assembly with clearly defined interfaces [3, 4]. Modularity of product architecture is not an absolute property; in fact, it is a relative property that can be quantitatively assessed and analyzed.

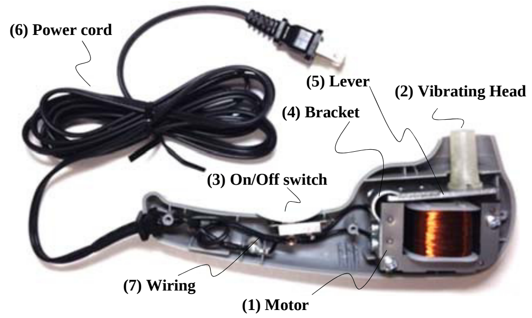
Modular product architecture has become an important product development topic in the last few decades [5]. It allows manufacturers to cost-effectively handle and develop complex products by decomposing them into simpler sub-units or modules [6]. The benefits of modularity regarding product functionality, product design, maintenance, services, testing and verification, production, supply chain, and other activities have been discussed by many researchers in various fields [5, 7, 8]. A comprehensive discussion of benefits of modularity is provided by Ulrich [1]. They include: (1) efficient management of product changes and updates as manufacturers can easily change some functions of subsequent product generations by simply changing some modules, (2) better product variety management as variety could be simply created by having various combinations of modules, and (3) enhanced product development process since design tasks can be properly decoupled and carried out concurrently.

2. Literature Review

Several methods exist for designing modular products [9] and can be classified into two main groups [10]: function-based and matrix-based methods. Function-based methods [11-13] identify modules by mapping the functional decomposition of a product to its physical architecture. These

Peer-review under responsibility of the organizing committee of the 6th CIRP Conference on Assembly Technologies and Systems (CATS)
doi:10.1016/j.procir.2016.01.023

methods are criticized for not being able to adequately address the interfaces between physical elements of the product [2]. Matrix-based methods [2, 14, 15], used in this research, identify modules by reconfiguring the product Design Structure Matrix (DSM) which represents the product architecture [3].



(a) Hand-held massager

		Motor	Vibrating Head	On/Off Switch	Bracket	Lever	Power Cord	Wiring
		1	2	3	4	5	6	7
Motor	1	1	0	0	1	0	0	1
Vibrating Head	2	0	1	0	0	1	0	0
On/Off Switch	3	0	0	1	0	0	1	1
Bracket	4	1	0	0	1	1	0	0
Lever	5	0	1	0	1	1	0	0
Power Cord	6	0	0	1	0	0	1	0
Wiring	7	1	0	1	0	0	0	1

(b) Massager DSM

Fig. 1. DSM developed for **Wahl 4120-600** hand-held body massager (http://massagers.wahl.com/)

The Design Structure Matrix (DSM), first introduced by Steward [16], is the most common tool used to model interaction between components of a given product [17] and is, therefore, used by many researchers to model and study product architecture [18]. Fig. 1 illustrates the DSM constructed for a hand-held body massager made of seven components in addition to the casing. Each row and column of the matrix represents one of the massagers' components. Cells filled with 1 in the matrix refer to the existence of interaction between the two components corresponding to that cell. The type of interaction modeled in this example is on/off spatial interaction (physical adjacency). Other types of interactions (e.g. energy and information) and different interaction coding schemes (e.g. quantified interaction) may also be used [19]. Product modules are identified by re-arranging the DSM into clusters along the matrix diagonal to maximize interaction within clusters and minimum interaction outside. Several methods have been used in literature to identify products modules by clustering the DSM based on different modularity measures.

Existing methods for designing modular product architecture provide a flat map for modules and their interaction without considering their hierarchical levels of sub-assemblies representing the structural arrangement of components within each module. The only exception to that finding is the work on granularity analysis of modular products and manufacturing systems architectures by AlGeddawy and ElMaraghy [20-22].

AlGeddawy and ElMaraghy [20] used cladistics analysis, which is a hierarchical classification method commonly used in Biology [23], in order to identify product modules based on DSM architectural representation. Along with the identified modules, the method further provides the hierarchical structure of the product in a binary rooted tree format (cladogram). The ultimate objective of that method is to find the optimal hierarchical level of the product architecture (granularity level) for product modularization. Although, the cladistics method has provided better modularity results compared to earlier research results in the subject; the main focus of the study was to find the optimal granularity level and not to optimize the product architectural tree. Finding the product structure tree (number of tree levels and clusters of components formed at each level) that leads to the optimal total modularity over all the tree levels was not an objective for the cladistics analysis-based method.

Inspired by the work AlGeddawy and ElMaraghy [20], this paper addresses product modularity from a new perspective to provide the product architecture, represented by the product structure tree in the form of binary rooted tree, for maximum overall modularity at all levels of the product structure. In this research, every potential module of a given product is viewed as if it was a product on its own the architecture of which needs to be designed for optimal modularity. This allows all modularity benefits discussed earlier to be propagated throughout the entire product at all its hierarchical levels.

3. Proposed Modular Product Architecture Design Methodology

Product architecture is modeled as shown in Fig. 2 using binary rooted trees (i.e. product structure tree or product tree for short). The tree in Fig. 2 represents the structure of a product with eight components. The product has four hierarchical levels and each level has its own modularity. For instance, at level 1 the product has only two modules; module 1 {6, 8, 5, 7, 3, 4} and module 2 {2, 1}. Whilst, at level 2, the product has three modules; module 1 {6, 8, 5, 7, 3}, module 2 {4} and module 3 {2, 1}. Modules # 1 and # 2 at level 2, when combined, they form module # 1 of level 1.

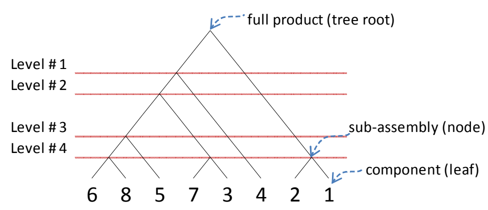


Fig. 2. Binary rooted tree representation of product structure

Therefore, each level of the tree provides more details related to components arrangements for modules at higher levels of the tree. Hence, the tree represents the components