UNISINOS UNIVERSITY
PRODUCTION AND SYSTEMS ENGINEERING GRADUATE PROGRAM
MASTER OF SCIENCE DEGREE

LEANDRO GAUSS

MARKET-DRIVEN MODULARITY:
An Integrated Method to Conceptually Design Modular Product Families.

São Leopoldo
2020

LEANDRO GAUSS

MARKET-DRIVEN MODULARITY:

An Integrated Method to Conceptually Design Modular Product Families.

Dissertation presented to the UNISINOS University in partial fulfillment of the requirements for the Degree of Master of Science in Production and Systems Engineering

Advisor: Prof. Daniel P. Lacerda, D.Sc.

Co-advisor: Prof. Paulo. A. C. Miguel, Ph.D.

São Leopoldo

2020

G274m Gauss, Leandro.
Market-Driven Modularity : an integrated method to conceptually design modular product families / by Leandro Gauss. – 2020.
270 f. : il. ; 30 cm.

Dissertation (master of science degree) — UNISINOS University, Production and Systems Engineering Graduate Program , São Leopoldo, RS, 2020.
Advisor: Daniel P. Lacerda, D.Sc.
Co-advisor: Paulo. A. C. Miguel, Ph.D.

1. Design science research. 2. Product family design. 3. Modularity. I. Title.

CDU: 658.5

Catalogação na Publicação (CIP):
Bibliotecário Alessandro Dietrich - CRB 10/2338

LEANDRO GAUSS

MARKET-DRIVEN MODULARITY:

An Integrated Method to Conceptually Design Modular Product Families.

Dissertation presented to the UNISINOS University in partial fulfillment of the requirements for the Degree of Master of Science in Production and Systems Engineering

Approved in ___/___/2020

GRADUATE COMMITTEE

Prof. Daniel P. Lacerda, D.Sc. – Advisor
(UNISINOS)

Prof. Paulo. A. Cauchick Miguel, P.hD. – Co-advisor
(UFSC)

Prof. Miguel A. Sellitto, D.Sc.
(UNISINOS)

Prof. Maria Isabel Morandi, D.Sc.
(UNISINOS)

Prof. Régis K. Scalice, D.Sc.
(UFSC)

Prof. Ricardo A. Cassel, D.Sc.
(UFRGS)

ACKNOWLEDGMENTS

First, I would like to thank my wife Cintia and my two sons, Lucas and Miguel, for supporting me along this journey. Soon, you guys will understand the reasons behind all of this. I would also like to express my gratitude to my research advisors Daniel P. Lacerda, D.Sc. and Paulo A. Cauchick Miguel, P.hD. for dedicating their time and providing remarkable suggestions and recommendations that undoubtedly enhanced the quality of this research. Finally, I would like to thank the Coordination for the Improvement of Higher Education Personnel (CAPES) for funding.

ABSTRACT

This study uses design science research to integrate marketing, engineering, and economic aspects into a single approach to conceptually design lucrative product families. In this context, the traditional stages of design science research methodologies are decomposed into 32 steps to provide practical guidance on the artifacts' design and evaluation. By following these steps, a field problem gives rise to a method, entitled Market-Driven Modularity (MDM), which is validated through a series of practical applications and experts' judgments. The main output of this process, the MDM, consists of an integrated method to conceptually design modular product families that balance the fulfillment of market needs and the resulting profitability to achieve them. To do that MDM uses the discrete choice modeling for quantifying the customers' preferences, modularity as a mechanism to provide product variety, the product family as a strategy to manage the trade-off between variety and cost, and profit as a moderating variable to balance the level of accomplishment of the customers' needs. In order to provide a better understanding of the proposed method, this study also presents an illustrative application of the MDM within the development process of a family of collaborative robotic palletizers for multiple market segments. The results indicate, that even from a deterministic perspective and under a context of low data availability, the two MDM outcomes, a lucrative product family structure, and the decision on investment in the product family design, are reasonably stable.

Keywords: design science research; product family design; modularity.

RESUMO

Esta pesquisa utiliza a ciência do projeto para integrar variáveis de marketing, engenharia e economia em uma única abordagem para projetar famílias de produtos economicamente orientadas ao mercado. Neste contexto, os estágios tradicionais das metodologias de pesquisa em ciência de projetos são decompostos em 32 etapas de forma a prover orientações práticas sobre o projeto e avaliação de artefatos. Por meio desta sequência de passos, um problema de campo dá origem a um método, intitulado Modularidade Orientada ao Mercado (MOM), o qual é validado através de um conjunto aplicações práticas combinadas com opiniões de especialistas. A principal saída deste processo, o MOM, consiste em um método integrado para projetar conceitualmente famílias de produtos modulares que equilibrem o atendimento das necessidades do mercado com as vantagens econômicas de atendê-las. Para tal, o MOM, utiliza a modelagem de escolhas discretas para quantificar as preferências dos clientes, a modularidade como mecanismo para prover variedade, a família de produtos para equilibrar o "trade-off" entre variedade e custo, e o lucro como variável moderadora para balancear o nível de atendimento das necessidades dos clientes. Com o intuito de prover um melhor entendimento do método proposto, este estudo também apresenta uma aplicação ilustrativa do MOM no projeto de uma família de paletizadores robóticos colaborativos para múltiplos segmentos. Os resultados apontam, que mesmo de uma perspectiva determinística e em um contexto de baixa disponibilidade de dados, as duas saídas do método MOM, a estrutura da família de produtos economicamente orientada ao mercado, e a decisão sobre investir no projeto da família de produtos, se apresentam razoavelmente estáveis.

Palavras-chave: ciência do projeto; projetos de famílias de produtos; modularidade.

LIST OF FIGURES

LIST OF TABLES

NOMENCLATURE

AHP	Analytic hierarchy process
CN	Customer needs
DP	Design parameters
DSM	Design structure matrix
DSR	Design science research
FM	Functional module
FR	Functional requirements
FR*	Functional requirement instances
GBOM	Generic bill-of-material
IIA	Independence of irrelevant alternatives
MBPF	Module-based product family
MDM	Market-Driven Modularity
MI	Modularity index
PFA	Product family architecture
PM	Physical modules
SKU	Stock-keeping units
SLR	Systematic literature review

TABLE OF CONTENTS

1 INTRODUCTION

The ever-increasing diversity of customers' needs has been pushing companies to provide more product variants without sacrificing production efficiency (Jiao, Simpson and Siddique, 2007). In industry and academy alike, the negative impact of product variety on operational performance has been traditionally addressed by two complementary approaches: the product line planning and product family design (Miao *et al.*, 2017). The product line planning consists of optimally selecting the group of products to be marketed to one specific market (Kahn, 2012), while the product family design consists of designing a set of products sharing common elements yet target different market segments (Simpson *et al.*, 2014).

Although numerous product line planning methods in management science and marketing literature deal with the selection problem using various objectives derived from profit, few of them explicitly address product design details not directly perceived by customers (Jiao, Simpson and Siddique, 2007). These approaches normally assume that any combination of product attributes can somehow be attained by design engineers post hoc (Michalek *et al.*, 2011). In contrast, most existing product family design approaches are targeted at identifying an optimal commonality decision in order to minimize cost while meeting pre-specified performance tiers (Kumar, Chen and Simpson, 2009). As a result, these engineering approaches do not sufficiently examine broader business indicators such as demand and profit (Michalek *et al.*, 2011).

The product family design is challenging for many aspects, and addressing its front-end issues is a complex activity (Colombo *et al.*, 2019). In general, the front-end issues are subdivided into four prevalent classes of design problems: (i) product family positioning, (ii) market-driven product family design, (iii) product family modeling, and (iv) product family configuration. The first two classes account for the marketing-related issues, which include customer involvement, product portfolio design, product family positioning, and transition or mapping from customer needs to functional requirements. While the last two classes are grounded on engineering-related issues, which include product family configuration, product architecture, design of families and platforms, leveraging commonality and modularity, and optimization of the family and platform design (Simpson *et al.*, 2014).

A recent study, concerning 72 methods for designing module-based product families, has shown that 1.4% of methods address the four classes of design problems concurrently. Among those methods (41.7%) considering marketing-related issues in its formulation, less than 7% derive the desired attributes in a product straight from the customers. Still from this study, it is seen that only 15.3% of methods account for enterprise-level indicators in product family configuration (Gauss, Lacerda and Miguel, 2020) (see Section 3 – Article 1). Findings that comply with previous research indications of lacking methods integrating marketing, engineering, and economic aspects into product family design (Jiao, Simpson and Siddique, 2007; Kumar, Chen and Simpson, 2009; Colombo *et al.*, 2019).

The problem is that marketing and engineering issues are highly interdependent in product family design. Moreover, the coupled relationships between them imply that any change in one domain can potentially influence the outputs of the other, with both affecting the economic benefits of an enterprise (Chen, Hoyle and Wassenaar, 2013). Therefore, in the design of optimal or near-optimal product families, marketing, engineering, and economic aspects cannot be pursued separately or even sequentially (Luo, 2011).

1.1 Research Problem

In this context, the problem this research poses is the missing link between marketing and engineering domains into product family design; and the undesirable effect resulting from it is the development of non-profitable product families.

In general, the economic benefit of a product family j to an enterprise, typically profit (V), is defined as a function of demand (Q), price (P), and cost (C) of the i product variants compounding the family, as shows the Equation 1 (Dong, Shao and Xiong, 2011; Chen, Hoyle and Wassenaar, 2013).

$$V_j = \sum_{i=1}^{n} Q_i.(P_i - C_i) \qquad 1$$

These enterprise-level indicators have an implicit interdependence resulting from the elements of a modular product family structure according to illustrate Figure 1.

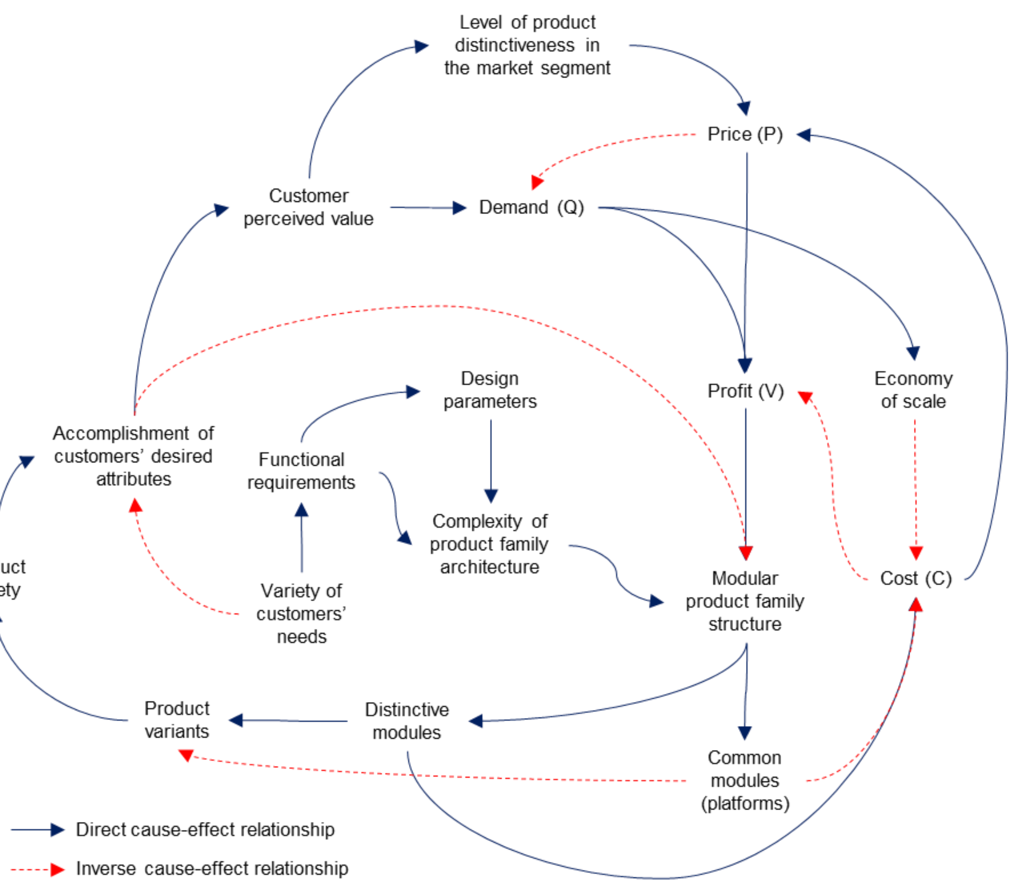


Figure 1. Systemic structure of product family design.

Where, the translation of customer needs into tangible specifications gives rise to the functional requirements (Yu *et al.*, 2015). The functional requirements not only integrate the product family architecture but also influences the formulation of the design parameters (Suh, 1998). The mapping between functional requirements and design parameters generates the product family architecture (Ulrich, 1995), that after decomposed, results in a set of common and distinctive modules compounding the modular product family structure (Otto et al., 2016). The combination of common modules, also referred to as platforms, along with the distinctive modules, make-up different product variants to fulfill the variety of products required by the market (Li, Huang and Newman, 2008). The distinctive modules positively influence the overall ability of a family member to meet the customer desired attributes (Jiao and Tseng, 1999a). The resulting customer perceived value impacts the demand and the level of distinctiveness of the product's offering in the market segment (Chen, Hoyle and Wassenaar, 2013). The level of product distinctiveness has an impact on price, which consequently influences back the demand (Dong, Shao and Xiong, 2011). The drawback of distinctive modules is the increase in cost, which can be balanced by the number of common modules used to compose the variants (Farrell and Simpson, 2008). The cost influences the price and is indirectly influenced by demand (Kumar, Chen and Simpson, 2009), and these three variables together determine the profit.

The systemic structure presented in Figure 1 shows the coupling relationships between marketing and engineering aspects into product family design, and how they affect broader business indicators such as cost, demand, price, and profit (Kumar, Chen and Simpson, 2009; Luo, 2011; Michalek *et al.*, 2011; Chen, Hoyle and Wassenaar, 2013). But integrating these three domains into product family design is not trivial (Colombo *et al.*, 2019).

Over the years, active work in developing methods to design product families has been done (Borjesson and Hoelttae-Otto, 2014; Otto *et al.*, 2016). Among those methods related to this research, the one encompassing four classes of design problems is the work of Jiang and Allada (2005). However, this method assumes the modules' set already exists, being deeply sensitive to the ability of extant modules in accomplishing the customer desired attributes. Besides that, the product family configuration is used to configure one variant at a time instead of building a product family structure. In like manner, other methods only entail the three first classes of design problems (Jiao and Tseng, 1999a; Asan, Polat and Serdar, 2004; Hsiao and Liu, 2005; Kazemzadeh *et al.*, 2009; Hsiao *et al.*, 2013; Sahin-Sariisik *et al.*, 2014; Ma and Kim, 2016; Pakkanen, Juuti and Lehtonen, 2016). But the main limitation of them lies in the inability to combine the designed modules into product family variants or even selecting the most adequate ones to compose the product family structure.

There is another group of methods, encompassing the product family modeling, which focuses on modules identification (Thevenot *et al.*, 2007; Arciniegas and Kim, 2011; Agard and Bassetto, 2013; AlGeddawy and ElMaraghy, 2013; Li *et al.*, 2013; Borjesson and Hoelttae-Otto, 2014; Aydin and Ulutas, 2016; Ma *et al.*, 2016; Hou *et al.*, 2017, 2018; Miao *et al.*, 2017). Within this group, a few methods, if any, perform the functional and physical decomposition concurrently. Besides that, these approaches occasionally measure the quality of the clustering solution, indicating in this way its open-loop nature. Still from this group, some approaches combine the product family positioning with product family modeling (ElMaraghy and AlGeddawy, 2012; Simpson *et al.*, 2012; Fan *et al.*, 2015; Miao *et al.*, 2017), while others combine the market-driven product family reasoning with the product family modeling (Dahmus, Gonzalez-Zugasti and Otto, 2001; Zhang, Tor and Britton, 2006; Du, Jiao and Tseng, 2006;

Krishnapillai and Zeid, 2006; Meng, Jiang and Huang, 2007; Stone *et al.*, 2008; Park *et al.*, 2008; Bonjour *et al.*, 2009; Yan and Stewart, 2010; Emmatty and Sarmah, 2012; Yang, Yu and Jiang, 2014; Wei *et al.*, 2015; Jung and Simpson, 2016; Cheng *et al.*, 2017; Bejlegaard *et al.*, 2018; Wang *et al.*, 2018). In both, less than a quarter, derive the customer desired attributes straight from the customers.

The last group of methods focuses on the product family configuration. More specifically in the process of mixing, matching, and scaling modules to generate product family variants (Tucker and Kim, 2008; Jiao, 2012; Pate, Patterson and German, 2012; Hanafy and Elmaraghy, 2015; Goswami, Daultani and Tiwari, 2017; Xiao *et al.*, 2018). In this group, the major part, solve the combinatorial and parametric problem through meta-heuristics and some use enterprise-level indicators to compound the objective function. Some methods also consider the product family design and configuration being performed together (Rai and Allada, 2003; Li, Huang and Newman, 2008; Li and Huang, 2009; Dong, Shao and Xiong, 2011; Chowdhury *et al.*, 2016; Baylis, Zhang and McAdams, 2018). However, they assume the modules' set already exists, and use the configuration process to generate product family variants instead of building product family structures. Additionally, nor a threshold to evaluate if the variants instantiated satisfy the desired attributes in a product, neither feedbacks leading to new modules' developments are found. Moreover, it is not explicit in these works, the product family configuration supporting or even playing the role of product line planning, an issue that has been traditionally dealt with in the management science and marketing literature (Jiao, Simpson and Siddique, 2007).

Synthesizing, there is a lack of integrated approaches modeling the customers' preferences and using it to design and configure gainful product family structures.

Therefore, the question this research poses is how to integrate marketing, engineering, and economic aspects into a single approach to design lucrative product families?

1.2 Objectives

In this sense, the primary objective of this research is to integrate marketing, engineering, and economic aspects into a single method to conceptually design lucrative product families. The specific objectives of this research include:

(1) Critically analyze the existing methods addressing modularity into product family design;

(2) Critically analyze the existing methods addressing market considerations into product family design;

(3) Design and evaluate the proposed method regarding pragmatic validity and practical relevance;

(4) Apply the proposed method in a complex made-up case or real situation.

1.3 Justification

As mentioned before, active work in developing methods to design product families has been performed over the past two decades (Borjesson and Hoelttae-Otto, 2014; Otto *et al.*, 2016). However, they have been developed independently of one another, and it can be daunting to try to compare the methods and understand which approach might be suitable when or how the methods might interlink, if at all (Simpson *et al.*, 2014). As a result, the transfer of these methods to industrial practice is inhibited by the seemingly broad array of material without a coherent organizing structure to compare development process tasks and the associated available methods, techniques, and tools (Otto *et al.*, 2016). Therefore, studies organizing these methods within the product development process are necessary to support future research in this field (Bonvoisin *et al.*, 2016). In addition to the theoretical aspects, the increasing adoption of modularisation in organizations requires more uniform and accurate definitions to characterize and study this phenomenon (Piran *et al.*, 2016; Frandsen, 2017). In this sense, the first contribution of this research lies in the integrative connection among existing methods to design module-based product families. An integration performed in the form of a functional model and structured classes of design problems, with both together serving as a meta-method for organizing the research in the field of module-based product design as well as a roadmap for implementing it in industry.

Equally important is the involvement of customer preferences into engineering design decisions, an issue that has received remarkable attention recently (Simpson *et al.*, 2014; Colombo *et al.*, 2019). While different product family variants may call for similar requirements, previous researches have shown that they might not be equally preferred by customers (Van Wie *et al.*, 2007). Therefore, procedures such as market segmentation (Farrell and Simpson, 2008), the transition or mapping from customers' desired attributes to engineering specifications (Stone *et al.*, 2008), customers' choice modeling (Jiao, 2012), among others, have been helpful in product attributes selection, family configuration, and portfolio optimization. In that direction, the second contribution of this work comprises the identification of market-related instances complementing the functional model and the structured classes of design problems on module-based product family design.

However, using only customer and competitor information to set targets without considering the engineering aspects or other enterprise-level objectives, such as market share and potential profit, can result in targets that can never be achieved in practice (Aungst, Barton and Wilson, 2003). Moreover, the costs incurred to create, sustain or use product families might not be worth the customization benefits (Ulrich, 1995; Hölttä-Otto and De Weck, 2007). For that reason, the level of accomplishment of customers' needs must be balanced by the economic advantages of meeting them (Chen, Hoyle and Wassenaar, 2013). Given the previous research indications of lacking approaches modeling the customers' preferences and using it to design and configure gainful product family structures, the third contribution of this research consists of the proposition of an integrated method to conceptually design modular product families that balance the fulfillment of market needs and the resulting profitability to achieve them. This is beneficial for two main reasons. First, the demand of potential product family variants can be compared against competing alternatives on the market, so that the economic benefits of a product family design can be assessed before making relevant investments on it, preventing in this way the development of non-profitable product families (Simpson *et al.*, 2014). Then, by highlighting the most valuable combinations, manufacturers can prioritize the modules to be developed at subsequent design stages (Colombo *et al.*, 2019), which directly implies the reduction of design and manufacturing costs as well as in shorter time-to-market (Ulrich and Tung, 1994).

Finally, the product family design, as well as other engineering disciplines, is typically concerned with construction problems related to not yet existing entities (van Aken and Romme, 2009; Vaishnavi, Kuechler and Petter, 2017). This conception complies with the goals of research performed under the design science paradigm, which seeks to produce knowledge to solve real problems or to design something that does not yet exist (Simon, 1996; van Aken, 2005). Despite conceptual coupling between product family design and design science, besides the works of Koh, Caldwell, and Clarkson (2013) and Andre and Elgh (2018), no other study has been conducted by design science research in this field. However, these studies lack practical guidance on artifact's design and evaluation. Issues not derived from the research on product family design, or any other field, but from the design science research methodologies that only approach the research conduction from higher abstraction levels.

Different methods for conducting research based on design science exist in the literature (Bunge, 1980; Nunamaker, Chen and Purdin, 1990; Takeda *et al.*, 1990; Eekels and Roozenburg, 1991; Walls, Wyidmeyer and Sawy, 1992; Cole, 2005; Gregor and Jones, 2007; Peffers *et al.*, 2007; Baskerville, Pries-Heje and Venable, 2009; Alturki, Gable and Bandara, 2011; van Aken, Berends and van der Bij, 2012; Dresch, Lacerda and Antunes Jr, 2015). Despite the differences in methods' steps, they share the same outcome, which is the well-tested, well-understood, and well-documented innovative generic design that has been field-tested to establish pragmatic validity (van Aken, Chandrasekaran and Halman, 2016). According to Kvale and Brinkman in Van Burg (2011), the pragmatic validity has to do with "the extent to which the research creates guidelines that generate the desired outcomes when those guidelines are applied". However, the extant design science research literature does not provide sufficient instruction on the artifact's design (Gacenga *et al.*, 2012). Moreover, there is little or no guidance on how or why one can or should choose among the different paradigms or methods to achieve design science research evaluation goals (Venable, Pries-Heje and Baskerville, 2016; Coetzee, 2019; Gassel, Reymen and Maas, 2019). In this sense, regarding the methodological aspects, the last contribution of this research lies in providing practical guidance on the artifact's design and evaluation, enhancing in this way the pragmatic validity of design science research methodologies.

1.4 Research Structure

This research consists of an article-based dissertation, structured in seven sections. In the sequence of this first section, which presents initial research considerations, Section 2 describes the research approach along with relevant methodological issues. The results retrieved from the procedures adopted in Section 2 are presented in Sections 3, 4, 5, and 6, wherein each section consists of an article. Table 1 depicts the relationship between dissertation objectives, sections, and articles.

Section 3 (Article 1) presents a systematic literature review and meta-synthesis of 72 articles (1999-2019) published in peer-reviewed journals concerning the module-based product family design. As a result, a functional model synthesizing the methods to design module-based product families along with its respective structured classes of design problems have been formulated. Section 4 (Article 2) follows the same pattern but encompassing 21 articles regarding the market-driven product family design.

Section 5 (Article 3) decomposes the traditional stages of design science research methodologies into 32 steps to provide practical guidance on the artifacts' development and evaluation. By following these steps, a field problem gives rise to a method, entitled Market-Driven Modularity (MDM), which is validated through a series of practical applications and experts' judgments. Section 6 (Article 4), in turn, presents an illustrative application of the MDM within the development process of a family of collaborative robotic palletizers for multiple market segments. Finally, Section 7 provides the research contributions and limitations as well as its future directions.

Table 1. The relationship between dissertation objectives, sections, and articles.

Primary objective	Specific objective	Section and scope	Main contributions	Article	Journal to be submitted
Integrate marketing, engineering, and economic aspects into a single method to conceptually design lucrative product families.	Critically analyze the existing methods addressing modularity into product family design.	Section 3 - Systematic literature review and meta-synthesis of 72 articles (1999-2019) published in peer-reviewed journals concerning the module-based product family design.	▪ The functional model synthesizing the methods to design module-based product families; ▪ The structured classes of design problems; ▪ The construction heuristic to build and assess functional models and classes of design problems.	Article 1 - Module-Based Product Family Design: Systematic Literature Review and Meta-Synthesis.	Journal of Intelligent Manufacturing (IF: 3.535), which features articles on new models, solutions, methodologies, algorithms, and tutorials on product development, with no limitations concerning the number of words, figures, and tables.
	Critically analyze the existing methods addressing market considerations into product family design.	Section 4 - Systematic literature review and meta-synthesis of 21 articles (1999-2019) published in peer-reviewed journals concerning the market-driven product family design.	▪ The functional model synthesizing the methods to design market-driven product families; ▪ The structured classes of design problems; ▪ The identification of market-related instances complementing the functional model and the structured classes of design problems regarding the module-based product family design.	Article 2 - Market-Driven Product Family Design: Systematic Literature Review and Meta-Synthesis.	Research in Engineering Design (IF:2.000), which features articles on design theory and methodology in engineering, with no limitations concerning the number of words, figures, and tables.
	Design and evaluate the proposed method regarding pragmatic validity and practical relevance.	Section 5 - Design and evaluation of the proposed method, entitled Market-Driven Modularity (MDM), under the design science paradigm.	▪ The MDM itself; ▪ The MDM's construction and contingency heuristics; ▪ Practical guidance on the artifact's design and evaluation; ▪ The quantitative approach to measure pragmatic validity and practical relevance.	Article 3 - Market-Driven Modularity: A Design Method Developed Under Design Science Paradigm.	Journal of Operations Management (IF: 7.776), which features articles on design science research strategy for operations management issues, with no limitations concerning the number of words, figures, and tables.
	Apply the proposed method in a complex made-up case or real situation.	Section 6 - Detailed application of the proposed method in a complex made-up case performed from an engineering design perspective.	▪ The MDM in its functional state.	Article 4 - Market-Driven Modularity: An Integrated Method to Conceptually Design Modular Product Families.	Journal of Intelligent Manufacturing (IF: 3.535), which features articles on new models, solutions, methodologies, algorithms, and tutorials on product development, with no limitations concerning the number of words, figures, and tables.

2 RESEARCH DESIGN

The product family design, as well as other engineering disciplines, is typically concerned with construction problems related to not yet existing entities (van Aken and Romme, 2009; Vaishnavi, Kuechler and Petter, 2017). This conception complies with the goals of research performed under the design science paradigm, which seeks to produce knowledge to solve real problems or to design something that does not yet exist (Simon, 1996; van Aken, 2005). Given this conceptual coupling, the present work followed the design science research methodology proposed by Dresch, Lacerda, and Antunes (2015). The 12 stages originally conceived by them were decomposed into 32 steps for better guiding the process of the artifact's design and evaluation.

Figure 2 synthesizes the approach adopted in this research, which started by identifying the problem to be solved in step 1.1. At initial iteration cycles between steps 1.1 and 2.1, the problem was just a "potential problem", and after understanding it in-depth, at step 2.1, it became the research problem to be studied at subsequent stages. This phase of awareness on the problem was supported by the Systems Thinking (Senge, 1990), and its expected outcomes were the understanding of the problem, the formalized aspects of the problem, and the problem-related topics to be investigated in the third stage, the systematic literature review.

Since two topics of investigation were defined, the systematic literature review was subdivided into two parallel flows, one for each topic. The steps from 3.1 to 3.5, and from 3.6 to 3.10 followed the same procedures. The difference between them lied in the topic under investigation and the input flow coming from step 3.4 to 3.8.

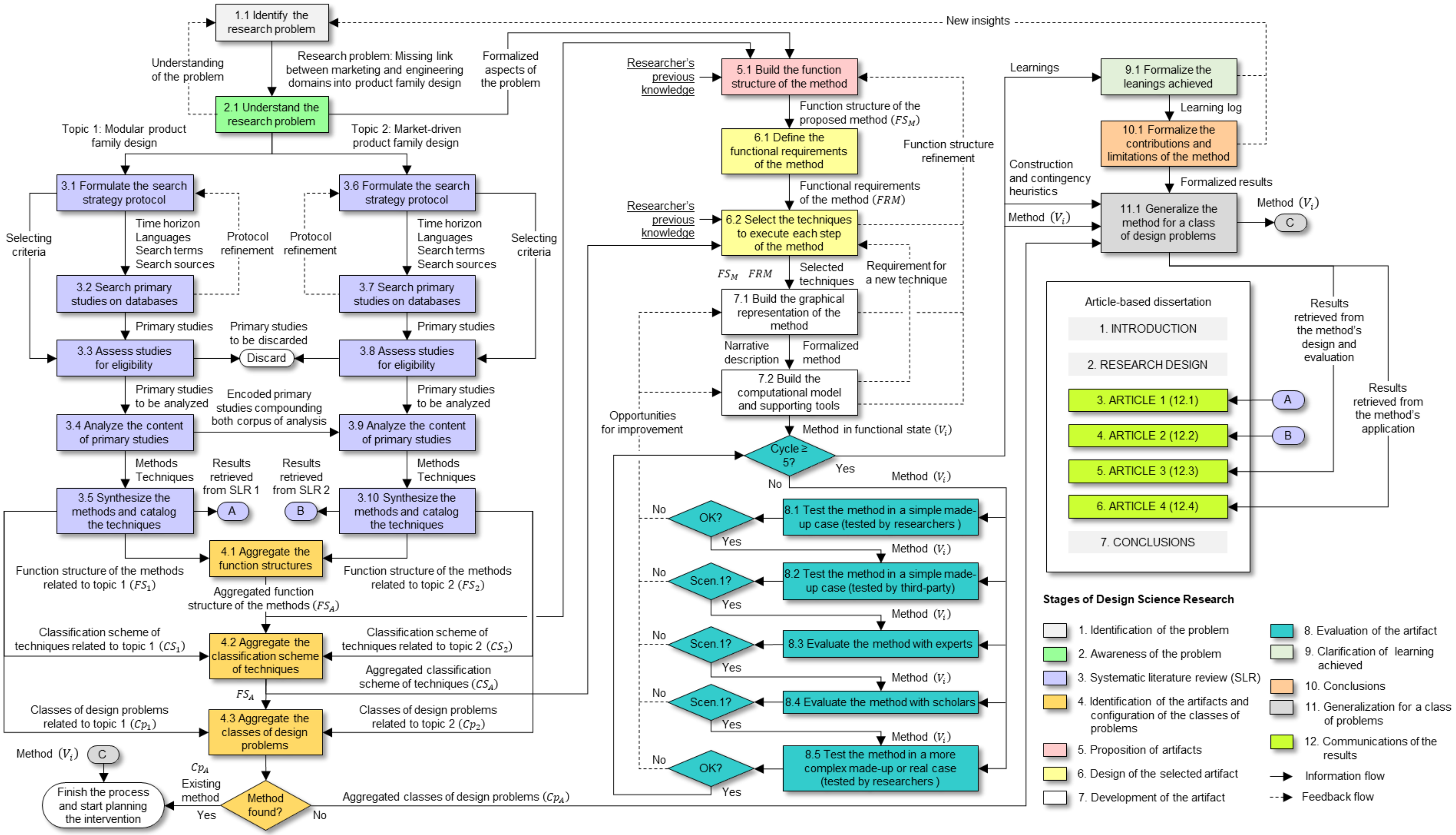


Figure 2. Research approach.

In this stage, the process began with the formulation of a search strategy protocol, as suggested by Table 2. The protocol gave rise to the outputs of steps 3.1 and 3.6, which were used for searching and selecting the primary studies, as indicated by Figure 2. The primary studies found in steps 3.2 and 3.7 were assessed for eligibility in steps 3.3 and 3.8. Those studies in compliance with the selecting criteria were chosen for review and those who not, were discarded and had the excluding statistics recorded according to illustrate Figure 3 and Table 3.

Table 2. The search strategy protocol adapted from Morandi and Camargo (2015).

Conceptual framework	Concepts that led to conducting the systematic review. May include a summary of the problem situation that is the focus of the review, as well as the known concepts and results.
Context	The context in which the research is being conducted: may include but is not limited to an industry, a sector or a location. For example, machine manufacturers located in Germany.
Time horizon	The time horizon being considered for the review. For example, studies published since 1990.
Theoretical currents	A strategy may or may not limit the theoretical currents to be searched for. For example, platform-based product family design.
Languages	Languages to be considered in the searching process
Research question	The question to be answered by the systematic review. Might be the review question itself or derived from it.
Review strategy	Aggregative or Configurative.
Selecting criteria	Criteria that will serve to determine the inclusion or exclusion of primary studies.
Search terms	Terms that will be used to search the databases. Consider not only the terms themselves but also the Boolean and proximity operators (AND, OR, NOT, NEAR, WITHIN, ADJ).
Search sources	Databases: EBSCO, Web of Science, Scopus. Proceedings: ASME, IoTSMS, ICII, ETFA Internet: Google scholar. Others.

In steps 3.4 and 3.9, the content of each selected study was analyzed in-depth (Bardin, 1993), and the artifacts resulting from this process came to integrate the solution field of this research. This work refers to an artifact as being a method or a technique intended to solve product design problems. In general, a method can be understood as a group of systematic steps (sub-functions) needed to accomplish specific product design objectives while a technique consists of a set of related procedures required to execute each step of the method (March and Smith, 1995). In this sense, at steps 3.5 and 3.10, the sub-functions compounding the structure of methods found were combined into a functional model, and its respective techniques were organized and cataloged. The outputs of these steps were the function structure of the methods (FS_i), the classification scheme of techniques (CS_i), and the classes of design problems (Cp_i) related to each parallel flow that travelled through stage 3.

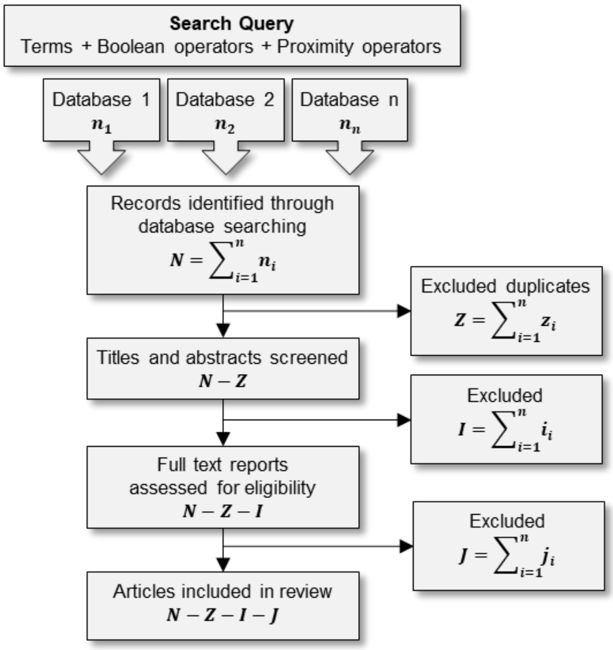


Figure 3. Generic results of the search and eligibility.

Table 3. Example of excluding statistics.

No. of exclusions	*Percentage*	*Excluding criteria*
$ne_1 = \sum_{i=1}^{n} z_i + i_i + j_i$	$[ne_1/(Z - I - J)]\%$	criterion 1
$ne_2 = \sum_{i=1}^{n} z_i + i_i + j_i$	$[ne_2/(Z - I - J)]\%$	criterion 2
$ne_n = \sum_{i=1}^{n} z_i + i_i + j_i$	$[ne_n/(Z - I - J)]\%$	criterion n
$Z - I - J$	100 %	Total

The function structure consists of a graphical form of a functional model where its overall function is represented by a collection of sub-functions connected by the flows on which they operate (Stone and Wood, 2000). Figure 4 illustrates a generic function structure (FS_i), wherein each sub-function (S_i) corresponds to an action intended to solve a particular design problem (Pb_i). A sub-function can be performed by one or more techniques (T_i) as shows the nonblank entries of the classification scheme (CS_i) presented in Table 4, i.e. $[CS]_{mxn} = [S]_m[T]_n$ (Pahl *et al.*, 2007). The execution order of each sub-function follows the causal relationship between the steps of the methods (M_i) identified during the content analysis. The dashed lines around the sub-functions represent the classes of design problems (Cp_i), defined here as a set of design problems sharing common characteristics and containing useful artifacts for their solution, i.e. $Cp_i = \{Pb_i, T_i, M_i\}$ (Dresch, Lacerda and Antunes Jr, 2015). The process of building functional models and structuring classes of design problems is detailed in Sections 3 and 4.

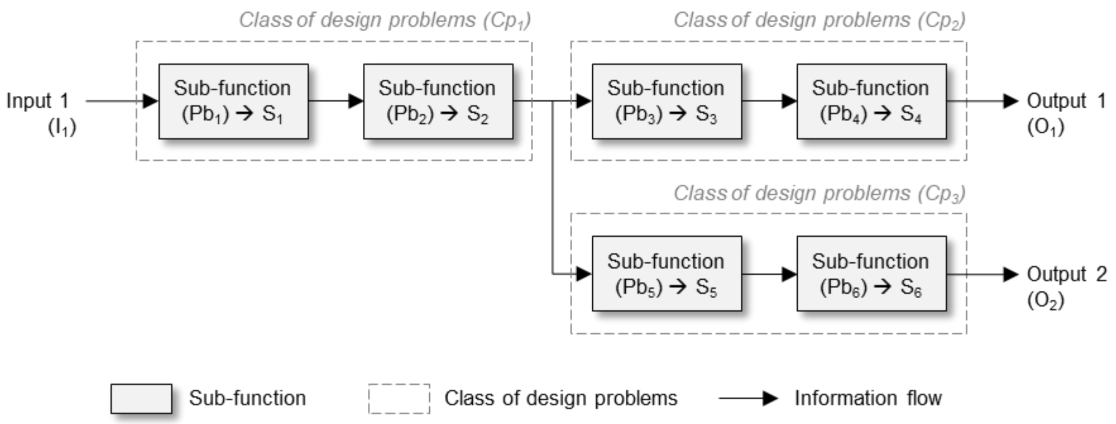


Figure 4. Generic function structure (FS_i).

Table 4. Generic classification scheme of techniques (CS_i).

Sub-functions (FS_i)	Techniques				
	T_1	T_2	T_3	T_4	T_5
$(Pb_1) \rightarrow S_1$	1	1			
$(Pb_2) \rightarrow S_2$			1		
$(Pb_3) \rightarrow S_3$		1			
$(Pb_4) \rightarrow S_4$			1		
$(Pb_5) \rightarrow S_5$				1	
$(Pb_6) \rightarrow S_6$					1

At stage 4, the issue was to aggregate the outcomes of the same nature coming from the two parallel flows of stage 3. This process started by combining the function structures FS_1 and FS_2 into a single model, i.e. $FS_A = FS_1 \cup FS_2$. This task was performed in step 4.1, and the technique used for that purpose, Aggregating Function Chains Into Functional Models (Stone and Wood, 2000), is synthesized in Figure 5.

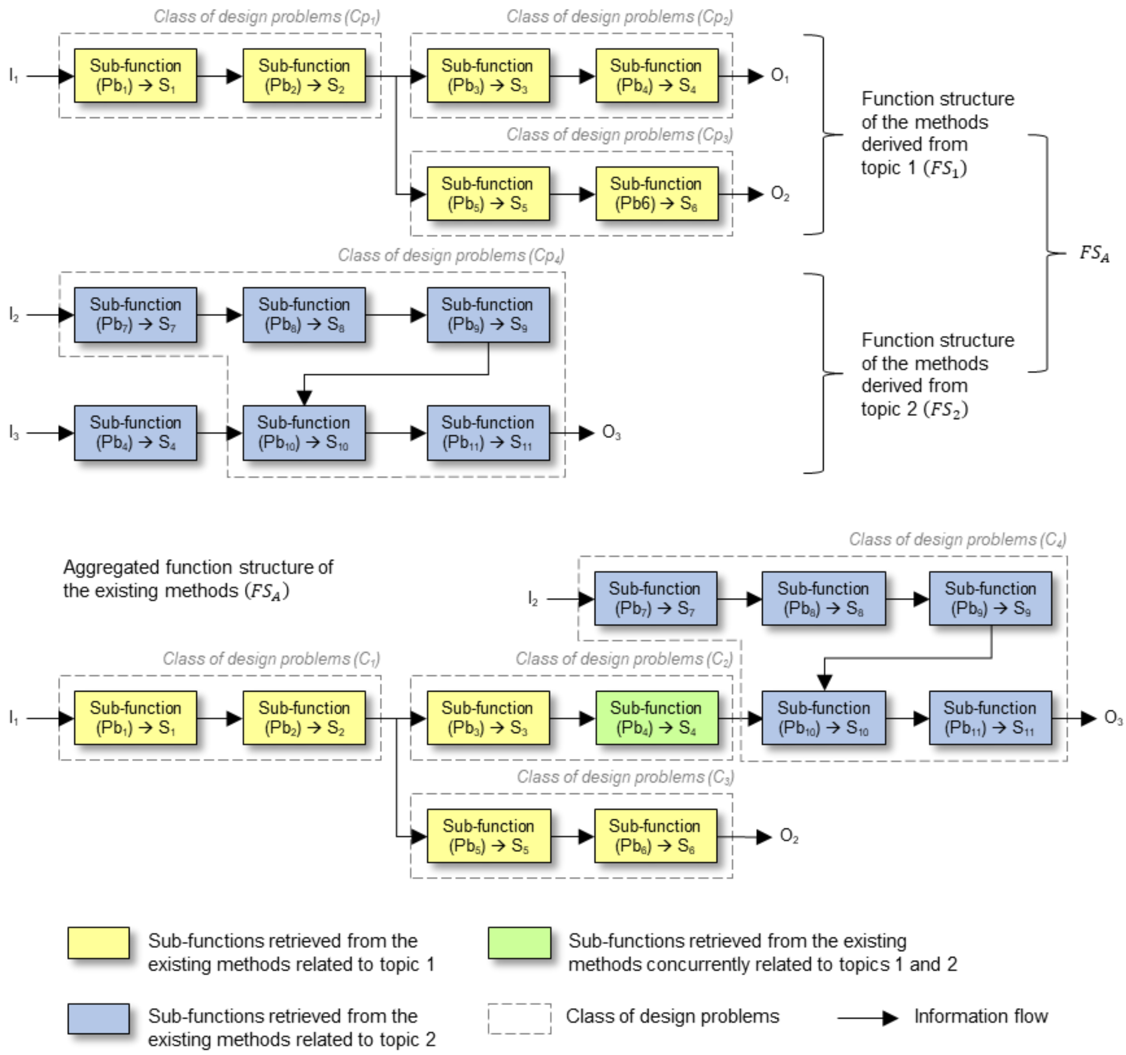


Figure 5. Formulation of the aggregated function structure of the methods (FS_A).

Following the same reasoning, in step 4.2, the classification schemes CS_1 and CS_2 were arranged into a single entity, i.e. $CS_A = CS_1 \cup CS_2$, wherein the sub-functions co-occurring in both schemes were merged as indicated by the green fill in Figure 6. In step 4.3, the classes of design problems Cp_1 and Cp_2, were grouped and reorganized based on the co-occurrence among the design problems (Pb_i), techniques (T_i), methods (M_i), evaluation approaches (Et_i), products classification (Pt_i), and primary studies (R_i). In this research context, the evaluation approach consists of the strategy adopted to test the artifact (method or technique) while the product classification relates to the type of goods used during the evaluation. The final product of this step was the aggregated classes of design problems (Cp_A) illustrated in Table 5. If at the end of this stage, a method that fully meets the needs to solve the problem was found, the process should be finished and the intervention, out of this research scope, should be planned. Otherwise, it should continue to step 5.1.

Classification scheme CS_1

Sub-functions (FS_1)	T_1	T_2	T_3	T_4	T_5
$(Pb_1) \rightarrow S_1$	1	1			
$(Pb_2) \rightarrow S_2$			1		
$(Pb_3) \rightarrow S_3$		1			
$(Pb_4) \rightarrow S_4$			1		
$(Pb_5) \rightarrow S_5$				1	
$(Pb_6) \rightarrow S_6$					1

Equals

Classification scheme CS_2

Sub-functions (FS_2)	T_3	T_6	T_7	T_8
$(Pb_4) \rightarrow S_4$	1			
$(Pb_7) \rightarrow S_7$		1		
$(Pb_8) \rightarrow S_8$		1		
$(Pb_9) \rightarrow S_9$			1	
$(Pb_{10}) \rightarrow S_{10}$				1
$(Pb_{11}) \rightarrow S_{11}$			1	1

Aggregated classification scheme of techniques CS_A

Sub-functions (FS_A)	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8
$(Pb_1) \rightarrow S_1$	1	1						
$(Pb_2) \rightarrow S_2$			1					
$(Pb_3) \rightarrow S_3$		1						
$(Pb_4) \rightarrow S_4$			1					
$(Pb_5) \rightarrow S_5$				1				
$(Pb_6) \rightarrow S_6$					1			
$(Pb_7) \rightarrow S_7$						1		
$(Pb_8) \rightarrow S_8$						1		
$(Pb_9) \rightarrow S_9$							1	
$(Pb_{10}) \rightarrow S_{10}$								1
$(Pb_{11}) \rightarrow S_{11}$							1	1

CS_1 resulting from FS_1

CS_2 resulting from FS_2

Figure 6. Formulation of the aggregated classification scheme of techniques (CS_A).

Table 5. Example of aggregated classes of design problems (Cp_A).

Classes of problems	Design problems	Artifacts		Evaluation approach	Products classification	Primary studies
		Techniques	Methods			
Cp_1	Pb_1	T_1	M_1	Et_1	Pt_1	R_1
			M_2	Et_2		R_2
		T_2	M_3	Et_1		R_3
	Pb_2	T_3	M_2	Et_2		R_2
			M_4	Et_3	Pt_2	R_4
Cp_2	Pb_3	T_2	M_3	Et_1	Pt_1	R_3
	Pb_4	T_3	M_2	Et_2		R_2
			M_{12}		Pt_4	R_{12}
Cp_3	Pb_5	T_4	M_5	Et_4	Pt_3	R_5
	Pb_6	T_5	M_6		Pt_4	R_6
Cp_4	Pb_7	T_6	M_7	Et_1	Pt_1	R_7
	Pb_8		M_8			R_8
	Pb_9	T_7	M_9	Et_1		R_9
	Pb_{10}	T_8	M_{10}	Et_2	Pt_4	R_{10}
	Pb_{11}				Pt_3	
		T_7	M_{11}	Et_3	Pt_2	R_{11}

Based on the boundary conditions imposed by the research problem, in step 5.1, the aggregated function structure of the methods (FS_A) was refined by abductively adding or removing sub-functions, or even function chains (FS_n), from its architecture, i.e., $FS_M = FS_1 \cup FS_2 \cup FS_n$. The output here was the function structure of the proposed method (FS_M), which combines the existing knowledge and new propositions to address the problem under study, as shown in Figure 7.

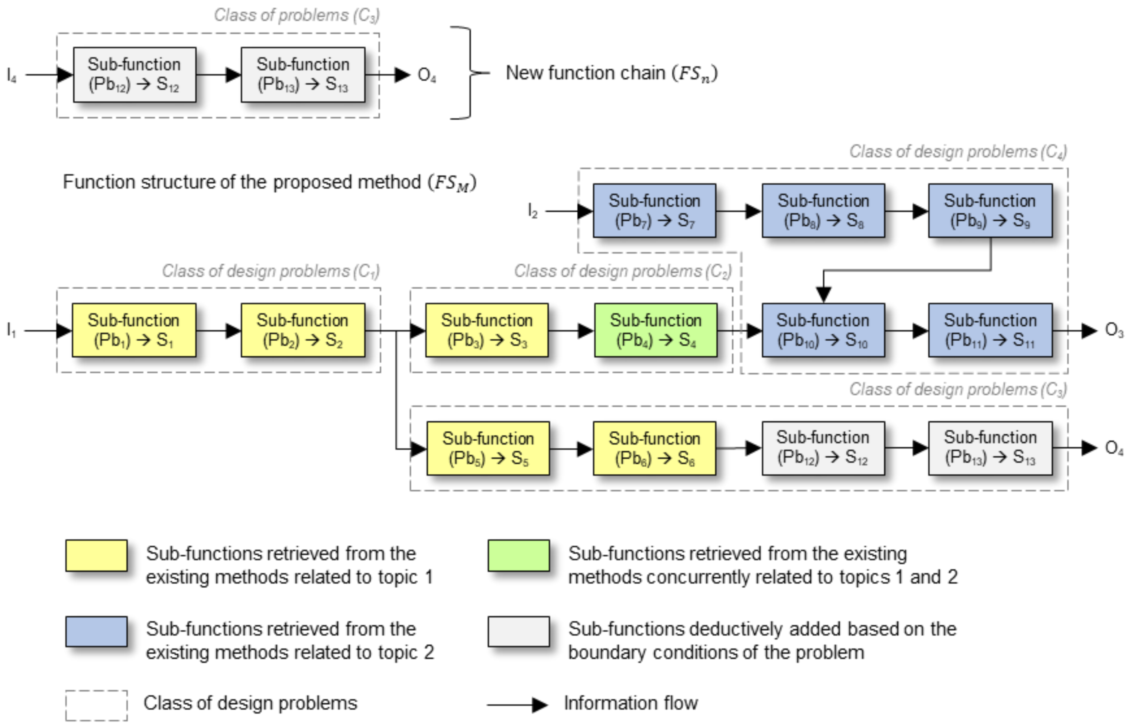


Figure 7. Example of the function structure of the proposed method (FS_M).

With the function structure of the proposed method (FS_M) defined, the next issue was to select the most suitable techniques to execute its corresponding sub-functions. This process took place at stage 6 and started by formulating the list of functional requirements (FRM) that each sub-function of the method should fulfill to solve the problem. Table 6 gives an example of the FRM defined at step 6.1.

Then, in step 6.2, those techniques integrating the aggregated classification scheme (CS_A) were assessed against the functional requirements, among other criteria.

Since the techniques are only known qualitatively, the selecting procedure employed in this research was the Elimination and Preference (Pahl *et al.*, 2007). From which, all unsuitable techniques were eliminated, and among the remaining ones, those that were patently better than the rest were given preference.

Table 6. Example of the functional requirements of the method (FRM).

Classes of problems	Design problems	Inputs	Sub-functions (FS_M)	Outputs	Functional requirements (FR)	Techniques*
C_1	Pb_1	I_1	S_1	OS_1	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_1\}$	$\{T_1,T_2\}$
	Pb_2	OS_1	S_2	OS_2	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_2\}$	T_3
C_2	Pb_3	OS_2	S_3	OS_3	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_3\}$	T_2
	Pb_4	OS_3	S_4	OS_4	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_4\}$	T_3
C_3	Pb_5	OS_2	S_5	OS_5	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_5\}$	T_4
	Pb_6	OS_5	S_6	OS_6	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_6\}$	T_5
	Pb_{12}	OS_6	S_{12}	OS_{12}	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_{12}\}$	$\{T_6,T_8\}$
	Pb_{13}	OS_{12}	S_{13}	O_4	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_{13}\}$	AT_1
C_4	Pb_7	I_2	S_7	OS_7	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_7\}$	T_6
	Pb_8	OS_7	S_8	OS_8	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_8\}$	T_6
	Pb_9	OS_8	S_9	OS_9	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_9\}$	T_7
	Pb_{10}	OS_4, OS_9	S_{10}	OS_{10}	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_{10}\}$	T_8
	Pb_{11}	OS_{10}	S_{11}	O_3	$\{FR_i \mid \forall FR_i \in \mathrm{Pb}_{11}\}$	$\{T_7,T_8\}$

* Column added after the step 6.2

In this sense, the use of a schematic selection chart provided a clear overview of the decision-making, wherein the unsuitable techniques were eliminated by the three first criteria applied in the sequence presented in Figure 8. Criteria A and B are suitable for yes/no decisions, and their application posed relatively few problems. Criterion C is grounded in the pragmatic validity, and its assessment required a deeper understanding of each technique. A preference was justified if, among the number of possible techniques, some reached better rates in the last three criteria (D, E, and F). Along the selection process, some techniques appeared to be inadequate for executing its corresponding sub-functions. When it happened, alternative techniques (AT_i), or even the development of new ones (DT_i) were proposed, as illustrated in Figure 8. It was also noted that, depending on the technique selected, it could influence back the functional model. In such situations, the function structure of the proposed method (FS_M) was revised as indicated by the feedback between steps 6.2 and 5.1.

Created by:	Date:	SELECTION CHART	Pg.
Gauss	*07/02/2019*	*Example related to FS_M*	1

Technique (T) evaluated by SELECTION CRITERIA

+ Yes
- No
? Lack of information
! Check functional requirements

DECISION

+ Pursue technique
- Eliminate technique
? Collect information
! Check functional requirements for changes

A: Does it fulfill the demands of functional requirements?
B: Is it compatible with the neighbor techniques?
C: Is it technically feasible?
D: Does it encompass different classes of products (consumer, intermediate and capital goods)?
E: Are there available tools to perform the technique?
F: Is it preferred by the researcher?

Enter the sub-function and its execution technique [S,Ti]	A	B	C	D	E	F	G	Remarks (Indications, Reasons)	DECISION
S1, T1	+	+	+	+	+	+		*T1 has been extensively applied to solve this kind of problem.*	+
S1, T2	+	-						*The outputs of the T2 is not compatible with the inputs of the T3.*	-
...	...	...	...	...	...	...		...	...
S8, T6	+	+	+	-	+	-		*Does not encompass the design of intermediate goods.*	-
S8, AT2	+	+	+	+	+	+		*Encompasses the three classes of products and has free tools available.*	+
...	...	...	...	...	...	...		...	...
S10, T8	+	+	-					*A very theoretical application only (does not present pragmatic validity).*	-
S10, DT1	!	?	?	?	?	+		*T7 can be adapted to execute the S10, however it must be developed.*	+
S11, T7	+	+	+	+	-	+		*A script can be written in R to execute the technique.*	+
S11, T8	+	+	+	-	-	-		*Does not encompass the design of intermediate goods.*	-

Figure 8. A schematic selection chart adapted from Pahl et al. (2007).

At stage 7, the internal environment of the artifact was defined (Simon, 1996). This process started by building the graphical representation of the method in step 7.1. Here, some conditional steps and feedbacks were included to refine the execution order of the method, as illustrated in Figure 9. Also, a narrative description was created to formalize the proper functioning of the internal environment of the method as well as to describe how it interacts with the external environment. Besides the graphical representation and the narrative description, the construction of the method required different approaches such as computational algorithms, prototypes, among others (Dresch, Lacerda and Antunes Jr, 2015). This task of building computational models and other supporting tools was performed in step 7.2, where the expected outcome was the method in its functional state (V_i). At the end of the steps 7.1 and 7.2, some function structure refinements and new techniques were required as indicated in Figure 2.

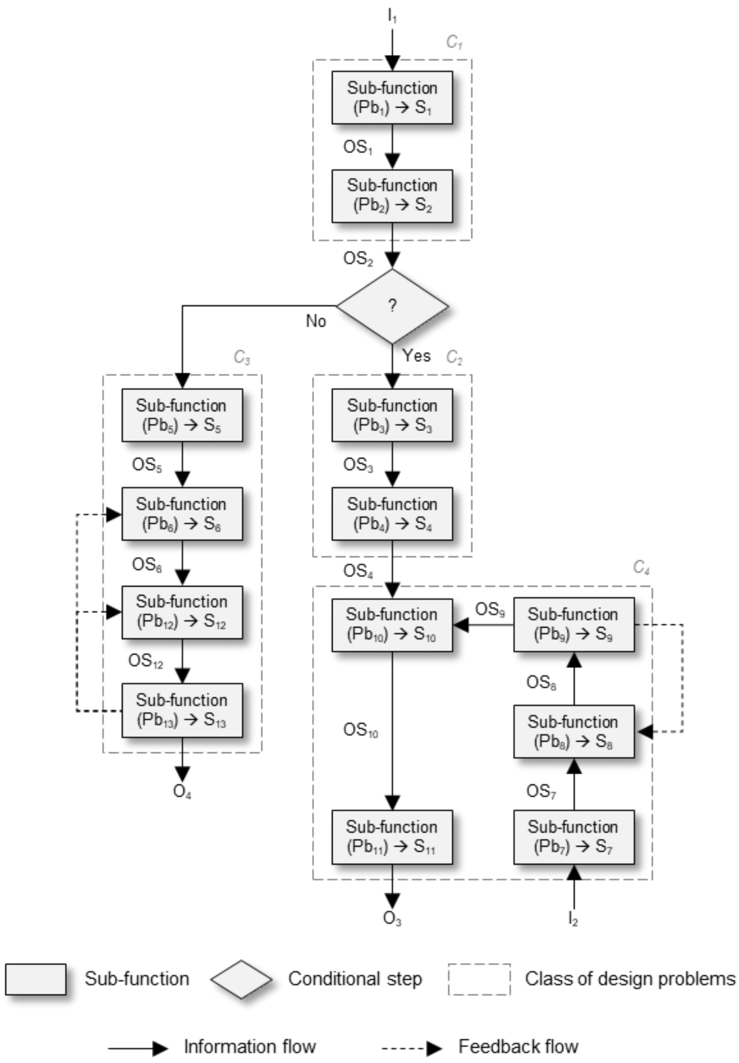


Figure 9. Graphical representation of the proposed method.

The seven stages described so far accounted for the method's development. The next issue was to evaluate its pragmatic validity and practical relevance. This process took place at stage 8 and began in step 8.1 with the method being tested by the researchers themselves in a simple made-up case. Those opportunities for improvement emerged from this first evaluation cycle, lead the process back to the steps 7.1 or 7.2.

In step 8.2, the method was tested by third-party practitioners in simple made-up cases. At the end of this second cycle, the participants' opinions have been captured through a questionnaire composed of closed and open questions, as illustrated by Figure 10 (Malhotra and Birks, 2007).

Q11. Do you agree that the MDM method can be incorporated in the early stages of the product development process, such as planning, conceptual design, and system-level design? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q12. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Figure 10. Example of closed and open questions used in the questionnaire.

After collecting the data, it was checked against inconsistent responses (Floyd and Fowler, 2014). Then the Median $(\tilde{x})$ (Montgomery and Runger, 2011) and the Free-Marginal Multirater Kappa $\left(k_{free}\right)$ (Randolph, 2005) were calculated for each closed question. For the open questions, in turn, the Content Analysis was used to derive the moderating variables (MV_i) and its respective frequencies (f) (Bardin, 1993). In this context, the $\tilde{x}$ measures the amplitude of agreement, k_{free} measures the level of agreement among the respondents, and f measures how often the moderating variables (MV_i), supposed to reduce the amplitude of agreement, occur. Table 7 gives an example of the results of participants' opinions, wherein the hierarchy between the top terms, constructs, dimensions, and moderating variables is depicted. In this table, the underlined numbers represent those values below the acceptable threshold adopted in this research, i.e. $k_{free} < 0{,}41$ or $\tilde{x} < 3$. Besides that, the characters within parentheses indicate to which question the dimension is related to. The results of participants' opinions led to four scenarios and two actions, as shown in Table 8.

Table 7. Example of the results of participants' opinions.

Top term / Construct / Dimension / Moderating variable	Resp. (n=10) $\tilde{x}$	k_{free}	f
1.0 Pragmatic validity	**3**	**0,66**	**-**
1.1 External environment	*3*	*0,75*	-
1.1.1 (Q07) Company size	3	0,63	-
...			
1.1.6 (Q17) Multiple market segments	3	0,63	-
MV.01 Aesthetics requirements	-	-	1
1.2 Internal environment	*3*	*0,59*	-
1.2.1 (Q19) Steps' sufficiency	3	1,00	-
MV.28 Complex products	-	-	1
...			
1.2.4 (Q25) Applicability of techniques	3	<u>0,20</u>	-
MV.06 Other existing techniques	-	-	1
2.0 Practical relevance	**3**	**0,70**	**-**
2.1 General utility	*3*	*0,70*	-
2.1.1 (Q34) Customers' choice modeling	3	<u>0,36</u>	-
MV.05 Uncertainty of estimated data	-	-	1
...			
2.1.6 (Q44) Utility	3	0,63	-
MV.09 Method's complexity	-	-	2

Table 8. Scenarios and actions resulting from participants' opinions.

Id.	Conditions	Action
1	$k_{free} \geq 0{,}41$ and $\tilde{x} = 3$	No changes in the method are required, and the process should go-ahead to the next step.
2	$k_{free} \geq 0{,}41$ and $\tilde{x} < 3$	Changes in the method are required, and the process should go back to step 7.1 or 7.2.
3	$k_{free} < 0{,}41$ and $\tilde{x} = 3$	Changes in the method are required, and the process should go back to step 7.1 or 7.2.
4	$k_{free} < 0{,}41$ and $\tilde{x} < 3$	Changes in the method are required, and the process should go back to step 7.1 or 7.2.

In scenarios 2, 3, and 4, the moderating variables (MV_i) were accessed to understand which part of the method should be changed. If, for any reason, it was impractical to implement all changes required, the most frequent MV_i were given preference.

The same reasoning was employed with slight differences in steps 8.3 and 8.4. The first difference was that the method was presented to the experts and scholars instead of being tested by them. The second difference lied in the fact that, besides the open questions of the questionnaire, audio records and responses by e-mail were used as an additional source to obtain the moderating variables (MV_i).

The last cycle consisted of testing the method by the researchers themselves in a complex made-up case or real situation. As well as in the first cycle, those opportunities for improvement emerged in this step, lead the process back to the steps 7.1 or 7.2.

Details on these five evaluation cycles such as cases of application, questionnaire, participants sampling and characterization, the learnings achieved, among other factors, are given in Section 5.

The outputs of stage 8 are the duly evaluated method along with its construction and contingency heuristics (Dresch, Lacerda and Antunes Jr, 2015). The learnings achieved during the method's design and evaluation were identified in step 9.1 and then cataloged as illustrated in Table 9 (Cole, 2005; van Aken, Berends and van der Bij, 2012). In step 10.1, the research contributions, limitations as well its future directions were formalized (Vaishnavi, Kuechler and Petter, 2017). While in step 11.1, the proposed method, together with its construction and contingency heuristics, was generalized for a particular class of problems (Venable, 2006; Gregor, 2009).

Table 9. Example of the learning log.

Id.	Entered by	Cycle	DSR step	Subject	Situation	Recommendations & Comments	Implemented in
The number used to identify learning.	The name of the individual who identified the learning.	The testing cycle where the learning happened.	The step of the research method where the learning occurred.	A brief headline describing the subject of the learning.	A detailed description of the situation learned from.	Recommendations and comments regarding the action taken, to help guide future research.	Where the recommendation has been implemented, i.e. research strategy or artifact.

Finally, the knowledge generated from the research process was compiled into four articles at stage 12. The first two articles encompassed the procedures and results retrieved from the two problem-related topics investigated in the systematic literature review stage. The third article entailed the entire process of design and evaluation of the proposed method. The fourth article covered the method in its functional state applied to a complex made-up case. As mentioned before, the results retrieved from the procedures adopted in Section 2 are presented in Sections 3, 4, 5, and 6, wherein each section consists of an article.

3 ARTICLE 1 - MODULE-BASED PRODUCT FAMILY DESIGN: SYSTEMATIC LITERATURE REVIEW AND META-SYNTHESIS [1]

[1] Article accepted for publication in the Journal of Intelligent Manufacturing (JIM).

Abstract: Increased demand for a greater variety of products has forced many companies to rethink their strategies to offer more product variants without sacrificing production efficiency. In this context, research has found that such a trade-off can be properly managed by exploiting the module-based product family (MBPF) design. Over the years, active work in developing methods to design MBPFs has been done. Nevertheless, many of them have been created, and consequently exist, in isolation from one other. As a result, the adoption of these methods in industry and academy alike is inhibited by the seemingly broad array of material without a coherent organizing structure. To bridge this gap, this paper aims at developing a meta-synthesis of 72 articles concerning MBPF design developed over the past 20 years through a systematic literature review. The research findings are synthesized in the form of a functional model and structured classes of design problems, wherein the existing methods to design MBPFs and their respective instances are connected. These entities together serve as a meta-method for organizing the research in the field of MBPF design as well as a roadmap for implementing MBPFs in the industry. The main contributions of this work include: (i) constructing a functional model that connects the design methods for MBPFs; (ii) suggesting structured classes of design problems that complement the functional model by cataloging the techniques meant to execute each sub-function of the model; (iii) proposing a construction heuristic to build and assess functional models and classes of design problems.

Keywords: modularity; product family design; systematic literature review; meta-synthesis; functional model; structured classes of design problems.

3.1 Introduction

In today's competitive global business environment, product variety can help manufacturing companies to increase sales and profits (Zhu, Li and Feng, 2017). However, increased variety can also lead to higher design and production costs as well as longer lead times for new variants (Simpson *et al.*, 2014). In this sense, product variety more than doubled between 1997 and 2012, while product life cycles reduced by about 25% (Andersen *et al.*, 2017). The direct consequence of product variety on production is an exponentially increased number of process variations, such as different machines, tools, fixtures, set-ups, cycle times, and labor (Jiao, Simpson and Siddique, 2007). As a result, the unit cost rises more than 20% when the variety of manufactured items doubles (Antunes *et al.*, 2008).

Under those circumstances, previous research stated that product family design is an effective strategy to provide variety at reduced costs (Simpson *et al.*, 2014). In general, a product family refers to a set of products derived from a standard product platform to satisfy various market applications (Meyer and Lehnerd, 1997). Platforms, in turn, are intellectual and material assets shared across a family of products, to minimize manufacturing complexity (Erens and Verhulst, 1997). Coupled with that, authors advocate the modularity is the core of many supporting techniques for product family design (Simpson, Siddique and Jiao, 2006; Kong *et al.*, 2009; Otto *et al.*, 2016). Usually, there are two strategic objectives in applying modularity to a new product family (Simpson *et al.*, 2014): (i) a technology strategy to increase product configurability and reduce engineering effort within individual product lines by eliminating duplicate and competitive technical solutions that are used to fulfill the same customer value, and (ii) a manufacturing strategy to reduce the number of different product lines by establishing a module system that embodies more customer value-driving variance. In this context, the prominent approach to product family design is through the development of module-based product families (MBPF), wherein product family members are instantiated by mixing and matching functional modules from the platform (Ulrich, 1995; Du, Jiao and Tseng, 2001). An alternative approach, considered as a subset of the former (Fujita and Yoshida, 2004), is through the development of a scale-based product family, which consists of scaling one or more variables to change the platform specifications while common parameters remain constant (Simpson, 2004).

Over the years, active work in developing methods to design MBPFs has been done (Borjesson and Hoelttae-Otto, 2014; Otto *et al.*, 2016). However, they have been developed independently of one another, and it can be daunting to try to compare the methods and understand which approach might be suitable when or how the methods might interlink, if at all (Simpson *et al.*, 2014). As a result, the transfer of these methods to industrial practice is inhibited by the seemingly broad array of material without a coherent organizing structure to compare development process tasks and the associated available methods, techniques, and tools (Otto *et al.*, 2016). Therefore, studies organizing these methods within the product development process are necessary to support future research in this field (Bonvoisin *et al.*, 2016). In addition to the theoretical aspects, the increasing adoption of modularisation in organizations requires more uniform and accurate definitions to characterize and study this phenomenon (Piran *et al.*, 2016; Frandsen, 2017).

Some literature reviews have been developed in that direction. For instance, Gershenson, Prasad, and Zhang (2003) present an overview of existing research on the definition of modular product design and its benefits. The same authors expand their first work by presenting another study of existing research on measures of product modularity and methods to achieve modularity in product design (Gershenson, Prasad and Zhang, 2004). In like manner, Jose and Tollenaere (2005) provide a review of the platform concept with a particular interest in modular design methodologies. Jiao, Simpson, and Siddique (2007) present a comprehensive review of the state-of-the-art research on the product family design. Fixon (2007) analyses sources on modularity and commonality concerning the subjects they have studied, the performance effects they have investigated, and the tools they have applied in doing so. Simpson et al. (2014) present a novel state-of-the-art review on product family design focusing on research published after their first work. Bonvoisin et al. (2016) summarize published literature to introduce a common language in the field of product modularization and to build the theoretical basis of a multi-purpose approach. Piran et al. (2016) organizes modularization studies in a conceptual structure and classifies the articles analyzed into a specific modularization taxonomy concerning the study's objective. Otto et al. (2016) link the different strands of platform research into logical sequences that can be practically used for product platform development. Frandsen (2017) employs a bibliometric analysis to identify the structures, the evolution of the literature, and the emerging research areas.

Given the depth that these research formats permit, it is difficult to fully understand each work (Gershenson, Prasad and Zhang, 2003). Moreover, it is still not clear if and how various methods could be used jointly (Otto *et al.*, 2016). In this sense, to best of our knowledge, there are some questions regarding MBPF design that remain: (i) which methods address modularity into the design of product families? (ii) what kind of design problems do these methods account for? (iii) for which kind of products have these methods been developed? (iv) how has the performance of these methods been assessed? (v) what are the main steps of these methods? (vi) what is the execution order of these steps? (vii) which techniques are used to execute each step of these methods? (viii) is there a common underlying structure among these methods?

This paper aims at answering these questions through a systematic literature review in addition to a meta-synthesis of 72 articles (1999-2019) published in peer-reviewed journals concerning MBPF design. The novelty of this research lies in the integrative connection among existing works on MBPF design. Besides that, its main contributions consist of (i) constructing a functional model that connects the design methods for MBPFs; (ii) suggesting structured classes of design problems that complement the functional model by cataloging the techniques meant to execute each sub-function of the model; (iii) proposing a construction heuristic to build and assess functional models and classes of design problems.

The remainder of this paper is structured as follows. Section 3.2 contains the research approach and relevant research methodological issues. Section 3.3 presents a functional model and the structured classes of design problems resulting from the literature mapping and analysis, followed by Section 3.4 that critically analyses the research findings. Finally, the last section draws some concluding remarks and main implications of this work as well as the next steps of this research.

3.2 Systematic Literature Review

Engineering is typically concerned with construction problems related to not yet existing entities (van Aken and Romme, 2009). This conception is in agreement with the goals of research performed under the design science paradigm, which seeks to produce knowledge to solve real problems or to design something that does not yet exist (Simon, 1962; van Aken, 2005). In a higher level of abstraction, that is the research scope of the MBPF design, which aims to formalize or to create artifacts that do not yet exist to solve real engineering problems concerning the product development. This work refers to an artifact as being a method or a technique intended to solve design problems. In general, a method can be understood as a group of systematic steps (sub-functions) needed to accomplish specific design objectives while a technique consists of a set of related procedures required to execute each step of the method (March and Smith, 1995).

While design science is the epistemological basis, design science research is the method that operationalizes research in this context (Lacerda *et al.*, 2013). For that reason, a six-step systematic literature review adapted to design science research by Morandi and Camargo (2015) were adopted: (i) question definition and conceptual framework, (ii) research strategy, (iii) search, eligibility and coding, (iv) quality assessment, (v) synthesis of results, and (vi) study presentation.

First, the modularity into product family design was defined as the central topic of this research. Then, to clarify the research question and to limit its scope, a conceptual framework was developed based on the fundamental references about (i) modularity, (ii) product family design, and (iii) product development process. As a result, the following research questions and boolean search terms - ("Modularity" OR "Modular") AND "Design" AND ("Product family" OR "Product platform") - were formulated.

- Which methods address modularity into the design of product families?
- What kind of design problems do these methods account for?
- For which kind of products have these methods been developed?
- How has the performance of these methods been assessed?
- What are the main steps of these methods?
- What is the execution order of these steps?
- Which techniques are used to execute each step of these methods?
- Is there a common underlying structure among these methods?

The search was conducted in two major databases: the Web of Science and Scopus. Those have been chosen because they provide quick access to the principal citation databases worldwide and have smart tools to track, analyze, and visualize research (Morandi and Camargo, 2015). Additionally, they cover over 21.000 titles and more than 73 million records of research production in natural sciences, health sciences, engineering, computer science, and materials sciences, with coverage dating back to 1900 (*Content - How Scopus Works - Scopus - | Elsevier solutions*, 2017; Clarivate Analytics, 2019). Besides that, to ensure the quality of the primary studies, only articles published in peer-reviewed international journals have been considered. Consequently, the English language was used as an inclusion criterion. Concerning the period and subject area, the articles published up to 2020 that encompassed the research in engineering, production, and operations management were consulted. Appendix A (Table A1) shows additional criteria in the search strategy protocol.

With the research strategy defined and based on a search limited to the article title, abstract, and keywords, the primary studies were found. Then they were checked for duplicates, followed by an inspection of the titles and abstracts (Brunton, Stansfield and Thomas, 2012). After, the potentially relevant studies were analyzed in-depth, as recommended elsewhere (Adler and van Doren, 1972), and those in compliance with the research scope were selected for review as indicates Figure 11. Table 10 presents the excluding statistics, while Table 11 provides the list of 72 primary studies included in the review.

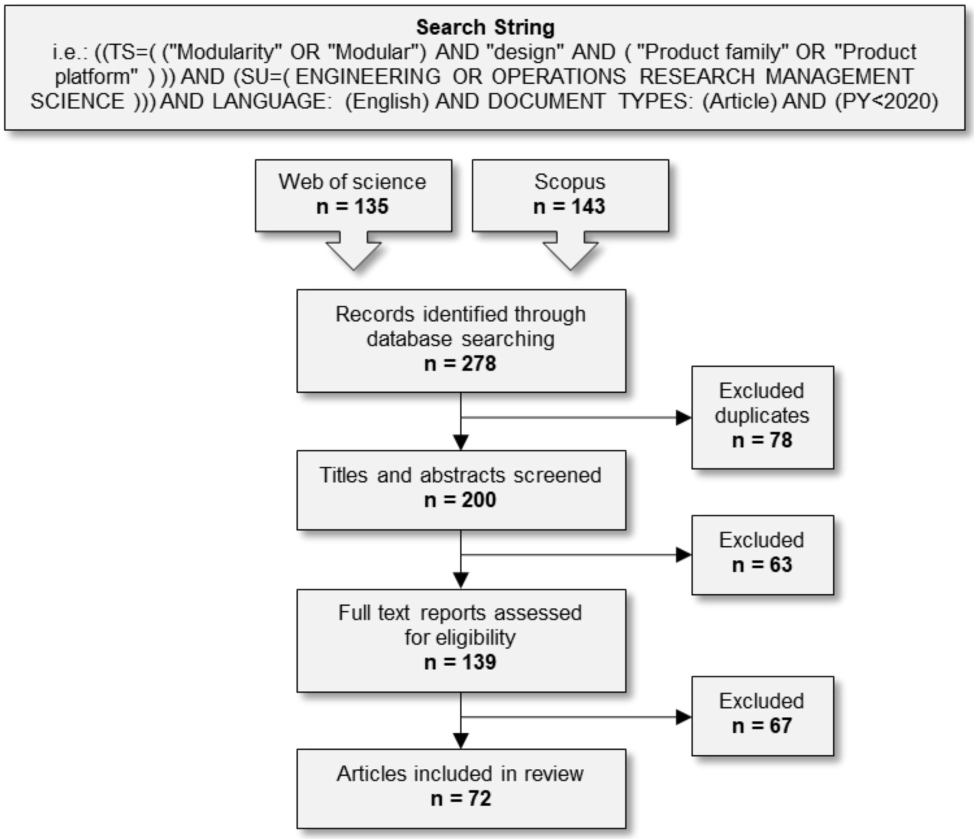


Figure 11. Flowchart of search results.

Table 10. Excluding statistics.

No. of exclusions	Percentage	Excluding criteria
78	37.9%	Duplicated studies
35	17.0%	Absence of methods or techniques addressing modularity in design
18	8.7%	Manufacturing and production for product families
17	8.3%	Design support systems
13	6.3%	Supply chain issues of product families
11	5.3%	Literature review on PBPF and modularity
7	3.4%	Fundamental issues on PBPF and modularity
6	2.9%	Theoretical development and synthesis on PBPF and modularity
5	2.4%	Very specific application not liable to generalization
4	1.9%	Paper not found
3	1.5%	Limited applicability to scale-based product family design
2	1.0%	Customer co-design
2	1.0%	Out of context
2	1.0%	Civil construction
1	0.5%	Aesthetics in product design
1	0.5%	Service design
1	0.5%	Software development
206	100.0%	Total

Table 11. List of primary studies included in the review.

Id	Title	Authors and year
R_1	A methodology of developing product family architecture for mass customization	(Jiao and Tseng, 1999a)
R_2	A metric for evaluating design commonality in product families	(Kota, Sethuraman and Miller, 2000)
R_3	Architecture of product family: Fundamentals and methodology	(Du, Jiao and Tseng, 2001)
R_4	Modular product architecture	(Dahmus, Gonzalez-Zugasti and Otto, 2001)
R_5	Design for variety: developing standardized and modularized product platform architectures	(Martin and Ishii, 2002)
R_6	Managing modularity of product architectures: Toward an integrated theory	(Mikkola and Gassmann, 2003)
R_7	Modular product family design: Agent-based pareto-optimization and quality loss function-based post-optimal analysis	(Rai and Allada, 2003)
R_8	An integrated method for designing modular products	(Asan, Polat and Serdar, 2004)
R_9	Robust modular product family design using a modified Taguchi method	(Jiang and Allada, 2005)
R_{10}	Managing modularity in product family design with functional modeling	(Zhang, Tor and Britton, 2006)
R_{11}	A comprehensive metric for evaluating component commonality in a product family	(Thevenot and Simpson, 2007)
R_{12}	A multi-criteria assessment tool for screening preliminary product platform concepts	(Otto and Hölttä-Otto, 2007)
R_{13}	An index-based method to manage the tradeoff between diversity and commonality during product family design	(Thevenot *et al.*, 2007)
R_{14}	Improving an existing product family based on commonality/diversity, modularity, and cost	(Alizon, Shooter and Simpson, 2007)
R_{15}	On the module identification for product family development	(Meng, Jiang and Huang, 2007)
R_{16}	A customer needs motivated conceptual design methodology for product portfolio planning	(Stone *et al.*, 2008)
R_{17}	A product platform concept development method	(Park *et al.*, 2008)
R_{18}	A cooperative coevolutionary algorithm for design of platform-based mass customized products	(Li, Huang and Newman, 2008)
R_{19}	Optimal product portfolio formulation by merging predictive data mining with multilevel optimization	(Tucker and Kim, 2008)
R_{20}	Optimal platform investment for product family design	(Zacharias and Yassine, 2008)
R_{21}	Multiobjective evolutionary optimization for adaptive product family design	(Li and Huang, 2009)
R_{22}	Integration of rough set and neural network ensemble to predict the configuration performance of a modular product family	(Zhu *et al.*, 2010)
R_{23}	Modularity analysis and commonality design: A framework for the top-down platform and product family design	(Liu, Wong and Lee, 2010)
R_{24}	Developing modular product family using GeMoCURE within an SME	(Yan and Stewart, 2010)
R_{25}	Flexible optimization decision for product design agility with embedded real options	(Dong, Shao and Xiong, 2011)
R_{26}	Optimal component sharing in a product family by simultaneous consideration of minimum description length and impact metric	(Arciniegas and Kim, 2011)
R_{27}	New dependency model and biological analogy for integrating product design for variety with market requirements	(ElMaraghy and AlGeddawy, 2012)
R_{28}	Product platform flexibility planning by hybrid real options analysis	(Jiao, 2012)
R_{29}	Optimizing families of reconfigurable aircraft for multiple missions	(Pate, Patterson and German, 2012)
R_{30}	Modular product development through platform-based design and DFMA	(Emmatty and Sarmah, 2012)
R_{31}	From user requirements to commonality specifications: An integrated approach to product family design	(Simpson *et al.*, 2012)
R_{32}	An ISM, DEI, and ANP based approach for product family development	(Hsiao *et al.*, 2013)

(continued)

Table 11. (continued)

Id	Title	Authors and year
R_{33}	Reactive design methodology for product family platforms, modularity and parts integration	(AlGeddawy and ElMaraghy, 2013)
R_{34}	Modular design of product families for quality and cost	(Agard and Bassetto, 2013)
R_{35}	An integrated method for flexible platform modular architecture design	(Li *et al.*, 2013)
R_{36}	Reduction of product platform complexity by vectorial Euclidean algorithm	(Navarrete *et al.*, 2013)
R_{37}	A structured approach to platform-driven product planning	(Sahin-Sariisik *et al.*, 2014)
R_{38}	A module generation algorithm for product architecture based on component interactions and strategic drivers	(Borjesson and Hoelttae-Otto, 2014)
R_{39}	A methodology to define a reconfigurable system architecture for a compact heat exchanger assembly machine	(Mesa *et al.*, 2014)
R_{40}	A modular method of developing an eco-product family considering the reusability and recyclability of customer products	(Yang, Yu and Jiang, 2014)
R_{41}	Joint optimization of product family configuration and scaling design by Stackelberg game	(Du, Jiao and Chen, 2014)
R_{42}	Predicting configuration performance of modular product family using principal component analysis and support vector machine	(Meng *et al.*, 2014)
R_{43}	A modular product multi-platform configuration model	(Hanafy and Elmaraghy, 2015)
R_{44}	A network methodology for structure-oriented modular product platform planning	(Fan *et al.*, 2015)
R_{45}	Incorporating quality function deployment with modularity for the end-of-life of a product family	(Yu *et al.*, 2015)
R_{46}	Modular deployment using TRM and function analysis	(Scalice *et al.*, 2015)
R_{47}	Module family design for modular product	(Adhitama and Rosenstiel, 2015)
R_{48}	A multi-principle module identification method for product platform design	(Wei *et al.*, 2015)
R_{49}	New modular product-platform-planning approach to design macroscale reconfigurable unmanned aerial vehicles	(Chowdhury *et al.*, 2016)
R_{50}	Analysis of architectural complexity for product family and platform	(Kim *et al.*, 2016)
R_{51}	A systematic adaptable platform architecture design methodology for early product development	(Li *et al.*, 2016)
R_{52}	A new methodology to cluster derivative product modules: an application	(Aydin and Ulutas, 2016)
R_{53}	Hierarchical game joint optimization for product family-driven modular design	(Ma *et al.*, 2016)
R_{54}	Brownfield Process: A method for modular product family development aiming for product configuration	(Pakkanen, Juuti and Lehtonen, 2016)
R_{55}	An integrated approach to product family redesign using commonality and variety metrics	(Jung and Simpson, 2016)
R_{56}	Design of adaptable product platform for heavy-duty gantry milling machines based on sensitivity design structure matrix	(Cheng *et al.*, 2017)
R_{57}	Modular platform optimization in conceptual vehicle body design via modified graph-based decomposition algorithm and cost-based priority method	(Hou *et al.*, 2017)
R_{58}	Development of sustainable platform for modular product family: a case study	(Shamsuzzoha and Helo, 2017)
R_{59}	An integrated framework for product line design for modular products: product attribute and functionality-driven perspective	(Goswami, Daultani and Tiwari, 2017)
R_{60}	Development of product platforms: Theory and methodology	(Johannesson *et al.*, 2017)
R_{61}	Cost effects of modular product family structures: Methods and quantification of impacts to support decision making	(Ripperda and Krause, 2017)

(continued)

Table 11. (continued)

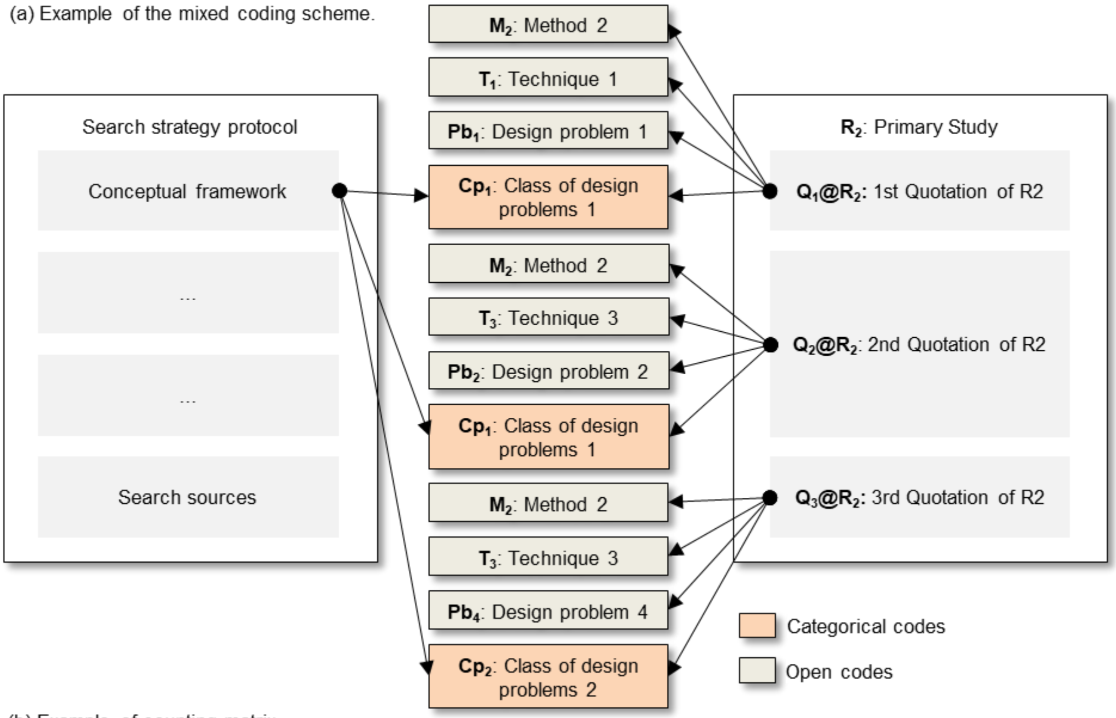
Id	Title	Authors and year
R_{62}	Product-family shared-component selection based on the consistency constraint function	(Hou *et al.*, 2018)
R_{63}	Coordinated optimization of low-carbon product family and its manufacturing process design by a bilevel game-theoretic model	(Xiao *et al.*, 2018)
R_{64}	A method for coupling analysis of association modules in product family design	(Cheng, Xiao and Wang, 2018)
R_{65}	An optimization model for low carbon oriented modular product platform planning (MP3)	(Wang *et al.*, 2018)
R_{66}	A fuzzy method for propagating functional architecture constraints to physical architecture	(Bonjour *et al.*, 2009)
R_{67}	Methodology for reconfigurable fixture architecture design	(Bejlegaard *et al.*, 2018)
R_{68}	Modular product platforming with supply chain postponement decisions by leader-follower interactive optimization	(Xiong, Du and Jiao, 2018)
R_{69}	Product family platform selection using a Pareto front of maximum commonality and strategic modularity	(Baylis, Zhang and McAdams, 2018)
R_{70}	Value analysis for customizable modular product platforms: theory and case study	(Colombo *et al.*, 2019)
R_{71}	Deciding on the total number of product architectures	(Askhøj *et al.*, 2019)
R_{72}	Module-based machinery design: a method to support the design of modular machine families for reconfigurable manufacturing systems	(Gauss, Lacerda and Sellitto, 2019)

The next step was to perform a content analysis (Bardin, 1993; Mayring, 2014). With this regard, the primary studies included in the review configured the context units from which the registration units, i.e., text quotations, have been encoded (Bardin, 1993). Coupled with that, a mixed coding scheme, compound by categorical and open codes were established as shown in Appendix A (Table A2) (Oliver and Sutcliffe, 2012). While the categorical codes were defined a priori, the open codes emerged during the analytical reading of the primary studies (Dresch, Lacerda and Antunes Jr, 2015). One example of categorical codes were the classes of design problems prevalent in the literature, which resulted from the conceptual framework formulation (Barnett-Page and Thomas, 2009). The methods, techniques, and design problems, in turn, are examples of open codes that were only possible to identify during the analysis of the context units in depth. Figure 12(a) illustrates the mixed coding scheme.

Besides the coding scheme, the following counting principles integrated the coding system adopted in this research: occurrence, co-occurrence, and frequency. The occurrence relates to the presence of code in a context unit (Bardin, 1993). For example, in Figure 12(a), the code T_3 is assigned to quotations Q_2 and Q_3 of the primary study

R_2. Although it appears twice in the context unit, its occurrence is accounted only once as illustrates the binary integer 1 intersecting the row T_3 and column R_2 in Figure 12(b). Regarding the co-occurrence, it consists of the simultaneous presence of two or more codes in a context unit (Bardin, 1993). Using the same example of code T_3 in Figure 12(a) it is possible the visualize that T_3 simultaneously appears with codes, M_2, Pb_2, Pb_4, Cp_1 and Cp_2, in the primary study R_2. It is also possible to deduct it from Figure 12(b), assuming that each primary study relates to a unique method.

(a) Example of the mixed coding scheme.



(b) Example of counting matrix.

			Primary studies						Design problems						Classes of design problems		
			R_1	R_2	R_3	R_4	R_5	R_6	Pb_1	Pb_2	Pb_3	Pb_4	Pb_5	Pb_6	Cp_1	Cp_2	Cp_3
Artifacts	Methods	M_1	1						1						1		
		M_2		1					1	1		1			1	1	
		M_3			1				1		1				1	1	
		M_4				1				1					1		
		M_5					1						1				1
		M_6						1						1			1
	Techniques	T_1	1	1					1						1		
		T_2			1				1		1				1	1	
		T_3		1		1				1		1			1	1	
		T_4					1						1				1
		T_5						1						1			1

Figure 12. (a) Example of the mixed coding scheme; (b) Example of counting matrix.

The frequency, in turn, can be determined as the number of times each code occurs in a context unit (Mayring, 2014). However, in the present research, this measure only makes sense when generalized for the corpus of analysis, which consists of a set of primary studies included in the SLR, i.e., $CA = \{R_i\}$. For example, in Figure 12(a), the method M_2 is assigned to three quotations (Q_1, Q_2 and Q_3), thus its frequency within the primary study R_2 would be $f_{M2} = 3$, irrelevant information if each primary study relates to only one method. Similarly, a problem Pb_i can be assigned to four quotations (Q_i, Q_{i+1}, Q_{i+2} and Q_{i+3}) within a given primary study R_i to present four complementary techniques (T_i, T_{i+1}, T_{i+2} and T_{i+3}) used together to solve it. It does not mean that Pb_i is more important than Pb_{i+1} that appears only once in the context unit, neither that M_i tackles four times the same problem Pb_i. But if we expand this counting principle to the corpus of analysis, useful information might emerge. Let's suppose the six primary studies present in Figure 12(b) compound a corpus of analysis; then it is possible to infer the class of design problems Cp_1 is tackled by 66,7% of the existing methods. Moreover, it is reasonable to attest that the most robust method is M_2, because it accomplishes two classes of problems (Cp_1 and Cp_2) and three problems (Pb_1, Pb_2 and Pb_4). In this sense, we restated the definition of frequency as being the number of times each code occurs in a corpus of analysis.

After defining the coding system, the next task was to encode and understand the raw data. This process was assisted by the qualitative data analysis software Atlas Ti (*ATLAS.ti 8 Windows | ATLAS.ti*, 2019), and one important issue here was to link the design problem to its potential class of design problems defined a priori. Besides that, it was also needed to identify the sequence the design problems occur along the course of the methods analyzed. This procedure of establishing the causal relationship between problems was performed through a syntopic reading (Adler and van Doren, 1972), and based on the reasoning of effect-cause-effect retrieved from Theory of Constraints thinking process (Cox and Schleier, 2010). Figure 13 gives an example of a code hierarchy resulted from this process, wherein seven design problems, organized in sequence, compound the second class of design problems for product families.

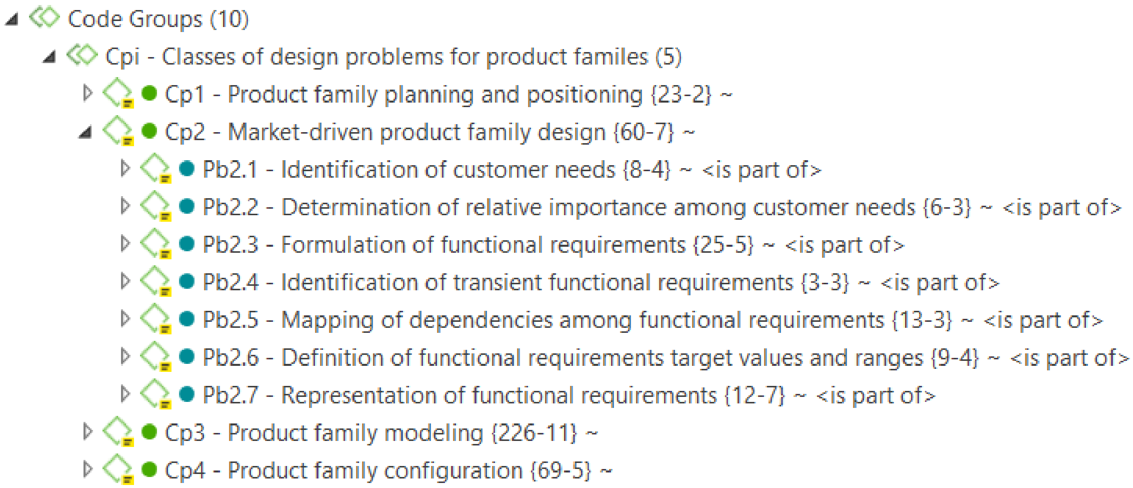


Figure 13. Example of codes hierarchy in software Atlas Ti.

The goal of meta-synthesis is to produce a new and integrative interpretation of findings that is more substantive than those resulting from individual investigations (Jensen and Allen, 1996; Finfgeld, 2003). In this sense, the final product is usually expressed in graphical form to permit mapping the nature and variety of concepts studied, identifying associations between different subjects, and providing explanations for the results from the various primary studies (Dresch, Lacerda and Antunes Jr, 2015).

With that intention, the next issue was to formulate a functional model that connects the functionalities of all methods identified in the corpus of analysis. The Functional Basis, a design language that describes the overall function as a set of simpler subfunctions while showing their connectivity (Stone and Wood, 2000), was used to aid in this task. In this context, the first thing done was to convert a design problem (Pb_i) into a sub-function (S_i) that corresponds to an action intended to solve it. For example, in Figure 13, the design problem $Pb_{2.1}$ consists of the *identification of* *customer needs*, thus an action to solve it would be *identify the costumer needs.* This transformation was performed by simply converting a substantive into a verb in an imperative form. After, the sub-functions were connected by the information flows on which they operate. The execution order of each sub-function followed the sequence resulting from the enconding task ilustrated in Figure 13. Later, the clustering of design problems into classes was checked by the Dominant flow, Branching flow, and Conversion-transmission flow huristics (Stone, Wood and Crawford, 2000). Then, the clustering solution was depicted in a design structure matrix (Browning, 2001) and had its quality assessed by the Modularity Index (MI) (Jung and Simpson, 2017). Details on the heuristics, design structure matrix, and MI are given in its references. According to this checking, those problems not belonging to a previously assigned class were relocated, and the coding system, along with the encoding process of raw data, was updated. Figure 14 gives an example of a generic functional model, wherein each sub-function (S_i) corresponds to an action intended to solve a particular design problem (Pb_i). The dashed lines around the sub-functions represent the classes of design problems (Cp_i), defined here as a set of design problems which share common characteristics, either practical or theoretical, and contain useful artifacts for their solution, i.e. $Cp_i = \{Pb_i, T_i, M_i\}$ (Dresch, Lacerda and Antunes Jr, 2015).

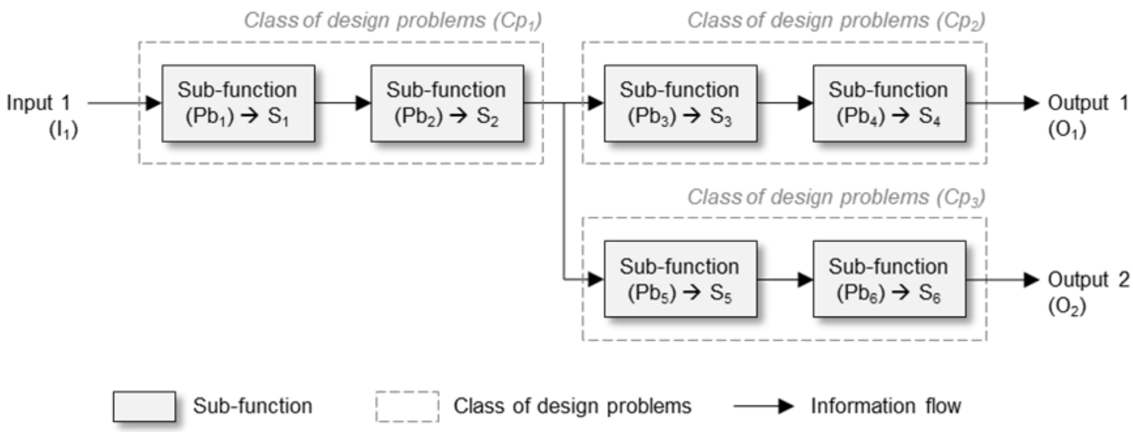


Figure 14. Example of a generic functional model.

After checking the clustering solution, the structured classes of design problems (Cp_i) were established based on the co-occurrence among the design problems (Pb_i), techniques (T_i), methods (M_i), evaluation approaches (Et_i), products classification (Pt_i), and primary studies (R_i) as exemplified in Table 12. Besides that, to support the subsequent association analysis, part of the content of Table 12 was converted into a binary incidence matrix as illustrated by Table 13.

Table 12. Example of structured classes of problems.

Classes of problems	Problems	Artifacts		Evaluation approach	Products classification	Primary studies
		Techniques	Methods			
Cp_1	Pb_1	T_1	M_1	ET_1	PT_1	R_1
			M_2	ET_2		R_2
		T_2	M_3	ET_1		R_3
	Pb_2	T_3	M_2	ET_2		R_2
			M_4	ET_3	PT_2	R_4
Cp_2	Pb_3	T_2	M_3	ET_1	PT_1	R_3
	Pb_4	T_3	M_2	ET_2		R_2
Cp_3	Pb_5	T_4	M_5	ET_4	PT_3	R_5
	Pb_6	T_5	M_6		PT_4	R_6

Table 13. Example of incidence matrix.

		Problems						Classes of problems		
		Pb_1	Pb_2	Pb_3	Pb_4	Pb_5	Pb_6	Cp_1	Cp_2	Cp_3
Primary studies	R_1	1						1		
	R_2	1	1		1			1	1	
	R_3	1		1				1	1	
	R_4		1					1		
	R_5					1				1
	R_6						1			1
Frequency		3	2	1	1	1	1	4	2	2
Rel. Freq.		50%	33.3%	16.7%	16.7%	16.7%	16.7%	66.7%	33.3%	33.3%

The association analysis is useful for discovering relationships hidden in large data sets (Zhang and Zhang, 2002). The uncovered relationships can be represented in the form of association rules or sets of frequent items, i.e.$\{Cp_1\} \rightarrow \{Cp_2\}$. This rule was extracted from the data set shown in Table 13 and suggested that a relationship exists between Cp_1 and Cp_2. In the context of this work, let $C = \{c_1, c_2, \dots c_i\}$ be the set of all codes retrieved from the content analysis and $R = \{r_1, r_2, \dots r_i\}$ be the set of all relationships between primary studies and codes. Each relationship r_i contains a subset of codes assigned from C. In association analysis, a collection of zero or more items is termed an itemset. If an itemset contains k items, it is called a k-itemset. For instance, $\{Cp_1, Cp_2\}$ is an example of a 2-itemset. The null (or empty) set is an itemset that does not contain any items. The relationship width is defined here as the number of items present in a relationship. A relationship r_i is said to contain an itemset X if X is a subset of r_i. For example, another relationship shown in Table 13 contains the itemset $\{Cp_1\}$ but not $\{Cp_1, Cp_3\}$. An essential property of an itemset is its support count, which refers to the number of relationships that contain a particular itemset (Tan *et al.*, 2019). Mathematically, the support count, $\sigma(X)$, for an itemset X can be stated as follows:

$$\sigma(X) = |\{r_i | X \subseteq r_i, r_i \in R\}| \quad 2$$

where the symbol $|.|$ note the number of elements in a set. In the data set shown in Table 13, the support count for $\{Cp_1, Cp_2\}$ is equal to two because there are only two relationships that contain Cp_1 and Cp_2 concurrently. An association rule is an implication expression of the form $X \rightarrow Y$, where X and Y are disjoint itemsets, i.e., $X \cap Y = \emptyset$. The strength of an association rule can be measured in terms of its support, confidence, and lift (Zhang and Zhang, 2002; Gkoulalas-Divanis and Verykios, 2010; Tan *et al.*, 2019). Support determines how often a rule applies to a given data set, while confidence determines how frequently items in Y appear in relationships that contain X. Lift, in turn, computes the ratio between the rule's confidence and the support of the itemset in the rule consequent. The formal definitions of these metrics are:

$$\text{Support, } s(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{N} \quad 3$$

$$\text{Confidence, } c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)} \quad 4$$

$$\text{Lift, } l(X \rightarrow Y) = \frac{c(X \rightarrow Y)}{s(Y)} \tag{5}$$

A lattice structure can be used to enumerate the list of all possible itemsets. In general, a data set that contains k items can potentially generate up to $2^k - 1$ frequent itemset, excluding the null set (Tan *et al.*, 2019). Because k can be very large in many practical applications, the search space of itemsets that need to be explored is exponentially large. To reduce the search space during frequent itemset generation, this research adopted the Apriori, which consists of an association rule mining algorithm that uses support-based pruning to systematically control the exponential growth of candidate itemsets (Zhang and Zhang, 2002). Its execution on this work was performed in software R (Team, 2019). Details on the frequent itemset generation through the Apriori algorithm is provided in (Tan *et al.*, 2019).

In summary, this section provided methodological procedures for searching, selecting, and analyzing the content of primary studies. Besides that, with the purpose of synthesizing the findings, a construction heuristic was proposed to build and assess functional models and classes of design problems. Additionally, a technique to identify the association rules among the classes of design problems was presented. The results of this process are presented in the next section.

3.3 Meta-Synthesis

The presentation of meta-synthesis is divided into two parts. The first part introduces the functional model that connects the design methods for MBPFs identified in this research. In this sense, the model not only presents the sub-functions intended to solve the design problems but also indicates the causal relationship among them along with its respective input and output flows. The second part, in turn, shows the structured classes of design problems that complement the functional model by cataloging the techniques meant to execute each sub-function of the model. Moreover, it presents the definitions of methods and techniques, how they have been tested, for what product they have been developed, and the most common association rules concerning the classes and its design problems.

3.3.1 Functional Model of MBPF Design

From the corpus of analysis, 72 methods were identified. Although these methods do not tackle the same design problems, at a higher level of abstraction, they somehow share the same objective of designing product families through the use of modularity. In this sense, following the reasoning of Stone and Wood (2000), we derived the overall function of this set of methods as being the: *"Design module-based product families."* The overall function expresses the solution-neutral relationship between inputs and outputs (Pahl *et al.*, 2007). With this respect, seven inputs and one output flows were identified in this review, as shown in Figure 15(a). For each input flow, a chain of sub-functions was established based on 25 design problems that emerged during the content analysis, as shown in Table A2. Then, the function chains were aggregated in a functional model, and the sub-functions were grouped into classes of problems as presented in Figure 15(b). In this model, the continuous and dashed arrows respectively indicate the information and feedback flows among the sub-functions, while the dashed rectangles represent the classes of design problems.

The functional model was compound by 25 sub-functions subdivided into 4 classes of design problems. The first class of design problems identified was the product family planning and positioning, which deals with market objectives along with technology developments guided by corporate strategies (Ulrich and Eppinger, 2012). Within this class, the model starts by strategically planning the MBPF in the sub-

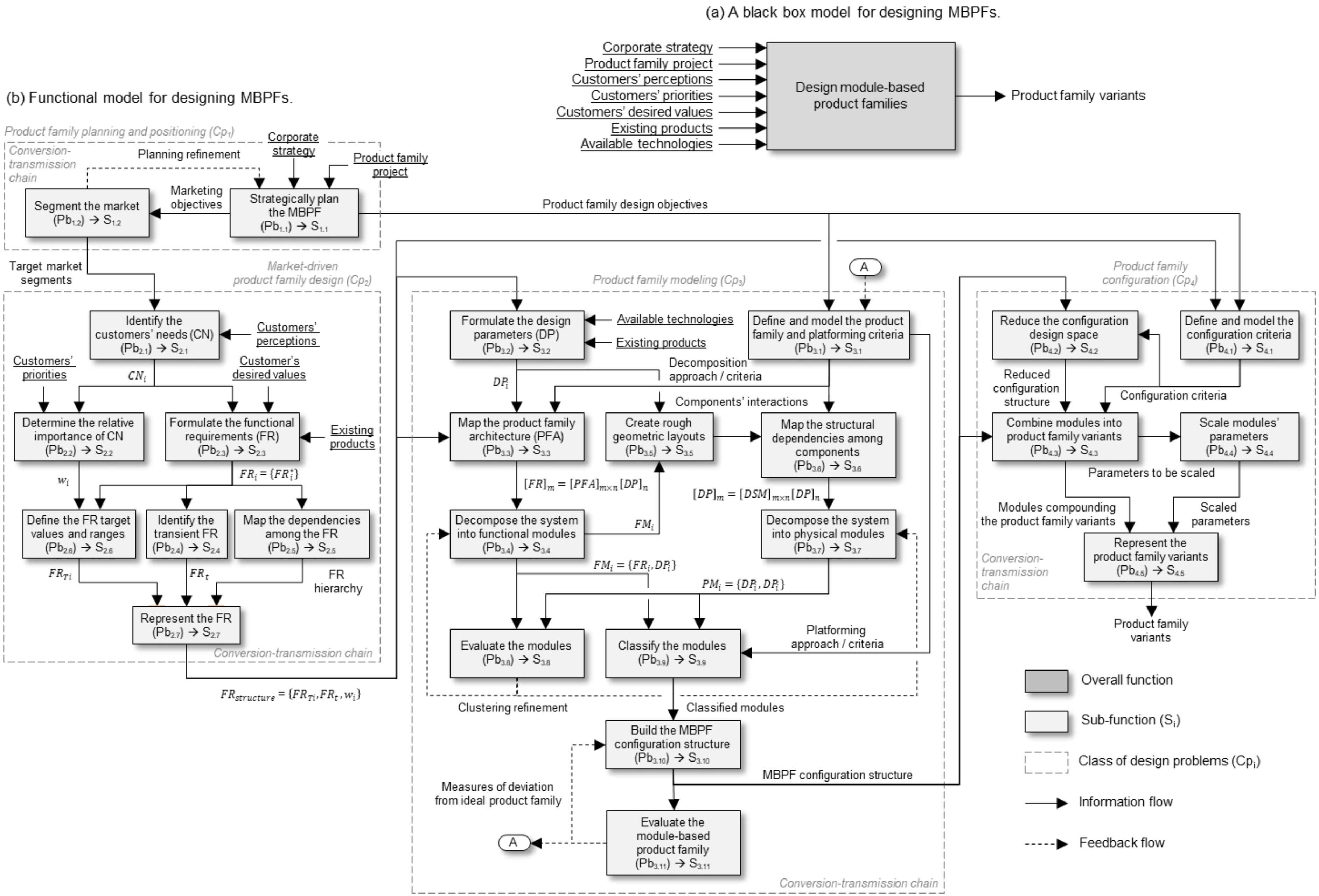


Figure 15. (a) A black box model for designing MBPFs; (b) Functional model for designing MBPFs.

function $S_{1.1}$, wherein strategic axes are incorporated into product family design (Jiao and Tseng, 1999a). Some issues covered by this sub-function include the mapping of future product plans (Martin and Ishii, 2002; Jiang and Allada, 2005), the optimal initial investment in the platform (Zacharias and Yassine, 2008), the relationship between business environment and product architecture (Otto and Hölttä-Otto, 2007; Pakkanen, Juuti and Lehtonen, 2016), the drivers influencing the product modularity and its relative importance (Sahin-Sariisik *et al.*, 2014; Scalice *et al.*, 2015), and the appropriateness of product modularity (Asan, Polat and Serdar, 2004; Shamsuzzoha and Helo, 2017). Those market-related objectives serve as an input flow for segmenting the market in the sub-function $S_{1.2}$. In this step, the market is decomposed into several segments taking into account the industry type, customer consumption levels, regional characteristics, among other factors (Fan *et al.*, 2015). At this stage, clustering procedures (Tucker and Kim, 2008; Hsiao *et al.*, 2013; Colombo *et al.*, 2019) are usually employed for characterizing different customer groups (Jiao and Tseng, 1999a), that along with reverse engineering and benchmarking of existing solutions support the identification of promising product plans and platform leveraging strategies (Thevenot and Simpson, 2007; Simpson *et al.*, 2012). The outputs here are not only target market segments (ElMaraghy and AlGeddawy, 2012; Sahin-Sariisik *et al.*, 2014) but also the planning refinement feedback, as shown in Figure 15(b) (Adhitama and Rosenstiel, 2015; Li *et al.*, 2016).

The second class of design problems found was the market-driven product family design, which handles the transition of customer needs (CN) to functional requirements (FR) (Simpson *et al.*, 2014). The first step here, in sub-function $S_{2.1}$, is to identify CNs by deriving meaning through interpretations of customers' perceptions about the existing products (Cheng *et al.*, 2017). This process is usually assisted by qualitative and quantitative techniques on data collection (Stone *et al.*, 2008; Emmatty and Sarmah, 2012; Colombo *et al.*, 2019), analysis (Tucker and Kim, 2008; Hsiao *et al.*, 2013), and synthesis (Pakkanen, Juuti and Lehtonen, 2016). Coupled with that, arises the necessity to determine the relative importance (w) of each need (Asan, Polat and Serdar, 2004). For that reason, the sub-function $S_{2.2}$ explores the priority level of each desired attribute in a product (Jiao and Tseng, 1999a; Stone *et al.*, 2008) and determines which is the most influent on customer decision making (Du, Jiao and Chen, 2014; Wei *et al.*, 2015). This market-related information must be then translated into engineering specifications; in other words, it should be converted into FRs (Jung and Simpson, 2016; Johannesson *et al.*, 2017). This task of formulating the FRs from the CNs is performed by sub-function $S_{2.3}$ (Cheng *et al.*, 2017), that uses two strategies to that end (Meng, Jiang and Huang, 2007). The first is the inductive functional modelling that describes a product in terms of elementary functions required to achieve its overall function or purpose (Zhang, Tor and Britton, 2006; Stone *et al.*, 2008; Gauss, Lacerda and Sellitto, 2019). The second is deductively elicit the FRs and then map them with CNs through incidence matrices (Yu *et al.*, 2015; Pakkanen, Juuti and Lehtonen, 2016). In both cases, the FRs might derive not only from CNs but also from existing product offerings (Jiao and Tseng, 1999a). While FRs are generic to all members within the same customer group, many functional requirement instances (FR^*) could result from different desired values for a particular FR, i.e. $FR_i = \{FR_i^*\}$ (Jiao and Tseng, 1999a; Martin and Ishii, 2002). In this sense, the sub-function $S_{2.6}$ arranges similar FRs^* into clusters (FR_T) that are characterized by a target value (TV) and a variation range (VR), i.e. $FR_{Ti} = [TV_i, VR_i]$ (Park *et al.*, 2008; Zacharias and Yassine, 2008; Mesa *et al.*, 2014; Bejlegaard *et al.*, 2018; Gauss, Lacerda and Sellitto, 2019). Among a finite set of FRs, there are some prone to change in the future market (Jiang and Allada, 2005).

These transient FRs (FR_t) are identified and have its changing probability estimated in the sub-function $S_{2.4}$ (Bejlegaard *et al.*, 2018; Wang *et al.*, 2018). Another critical factor that affects the level of granularity of product architecture is FR hierarchy (Dahmus, Gonzalez-Zugasti and Otto, 2001; Simpson *et al.*, 2014). This issue is tackled by the sub-function $S_{2.5}$ through the mapping of dependencies among the FRs (Alizon, Shooter and Simpson, 2007; Bonjour *et al.*, 2009; Yan and Stewart, 2010). Finally, through the sub-function $S_{2.7,}$ the functional view of a product family is performed from an abstract level to individual instances (Jiao and Tseng, 1999a; Kota, Sethuraman and Miller, 2000; Yang, Yu and Jiang, 2014; Gauss, Lacerda and Sellitto, 2019).

The third class of design problems identified was the product family modeling, which comprehends the definition of modules, platforms, and the product family configuration structure in terms of design parameters (DP) and FRs (Jiao, Simpson and Siddique, 2007; Simpson *et al.*, 2014). In this class, the model starts by defining and modeling the product family and platforming criteria in sub-function $S_{3.1}$. From the design objectives coming from the sub-function $S_{1.1}$, decisions on what decomposition strategy to follow (Stone *et al.*, 2008; Liu, Wong and Lee, 2010; Emmatty and Sarmah, 2012; Yang, Yu and Jiang, 2014), what criteria to use for clustering the modules (Arciniegas and Kim, 2011; Hsiao *et al.*, 2013; Yu *et al.*, 2015; Ma *et al.*, 2016; Hou *et al.*, 2017; Cheng, Xiao and Wang, 2018), and what approach to adopt for identifying the platform are made at this stage (Fan *et al.*, 2015; Hou *et al.*, 2017, 2018). In terms of decomposition, there are two prevalent strategies, functional and physical decomposition. The first derives from the mapping relationships between two domains, i.e., $FR \rightarrow DP$ (Suh, 2001), while the second comes from the mapping relationships within the same domain, i.e., $DP \rightarrow DP$ (Bonjour *et al.*, 2009). These strategies are not mutually exclusive; on the contrary; they are complementary in some cases (Asan, Polat and Serdar, 2004; Borjesson and Hoelttae-Otto, 2014; Wei *et al.*, 2015). Independently of what strategy is used, there are common criteria employed for clustering the modules or even to identify the platforms. These criteria, along with its frequencies, are presented in Table 14.

Table 14. Modular platforming criteria of sub-function $S_{3.2}$.

Criteria	Frequency	Rel. Freq.	Primary study (Id.)
Interaction or coupling	9	25.0%	R5, R17, R26, R31, R32, R44, R53, R64, R72.
Redesign effort	6	16.7%	R5, R15, R26, R30, R31, R55.
Variety	6	16.7%	R16, R17, R23, R31, R45, R56.
Cost	5	13.9%	R15, R23, R53, R57, R61.
Commonality	5	13.9%	R7, R30, R32, R44, R55.
Environmental	2	5.6%	R40, R65.
Quality	2	5.6%	R57, R62.
Utility	1	2.8%	R30.
Total	36	100.0%	

Regarding the approaches adopted for identifying platforms, there are also two strategies (Stone *et al.*, 2008). One that considers the variability of FRs^* (Park *et al.*, 2008; Yu *et al.*, 2015), and the other that takes into account the level of redesign effort taken across generations (Simpson *et al.*, 2012; Jung and Simpson, 2016). In both, those modules related to the low variability of FRs^* or low redesign effort across generations are defined as common (platforms), while those related to the high variability of FRs^* or high redesign effort across generations are set as differentiate ones (Liu, Wong and Lee, 2010). Given the FRs previously defined, the process continues by formulating the DPs in sub-function $S_{3.2}$ (Jiao and Tseng, 1999a). The DP consists of the physical effect with the ability to fulfill one or more FRs (Gauss, Lacerda and Sellitto, 2019), and its formulation is usually based on the available technologies and the existing products (Pakkanen, Juuti and Lehtonen, 2016; Cheng *et al.*, 2017; Johannesson *et al.*, 2017). The DPs along with the FRs configure the two constituents of the product family architecture (PFA) (Jiao and Tseng, 1999a; Arciniegas and Kim, 2011; Askhøj *et al.*, 2019), that in context of MBPF design, can be defined as decoupled interfaces and the “one-to-one” mapping between FRs and DPs (Navarrete *et al.*, 2013; Borjesson and Hoelttae-Otto, 2014). This mapping process is performed in the sub-function $S_{3.3}$, and is usually depicted by a design matrix, i.e. $[FR]_m = [PFA]_{m\times n}[DP]_n$, where the nonblank entry $a_{ij} \in [PFA]_{m\times n}$ indicates a relationship between FR_i and DP_i (Wei *et al.*, 2015; Wang *et al.*, 2018). Then, in the sub-function $S_{3.4}$, the PFA is decomposed into different functional modules (FM), i.e. $FM_i = \{FR_i, DP_i\}$ (Jiao and Tseng, 1999a). In 80.0% of methods that tackle this sub-function, the functional decomposition is assisted by heuristics (Stone *et al.*, 2008; Liu, Wong and Lee, 2010; Li *et al.*, 2016; Gauss, Lacerda and Sellitto, 2019), while the 20.0% reaming adopt meta-heuristics related to optimization problems (Borjesson and Hoelttae-Otto, 2014; Yang, Yu and Jiang, 2014; Wei *et al.*, 2015). The FMs must physically match the working structure (Pahl *et al.*, 2007). In this sense, the rough geometric layout is elaborated in sub-function $S_{3.5}$ to identify the interactions among physical components (Pakkanen, Juuti and Lehtonen, 2016; Gauss, Lacerda and Sellitto, 2019). Without this, it would be difficult to determine how subsystems, subassemblies, or parts are coupled (Martin and Ishii, 2002). In such cases where the functional decomposition is by-passed, the DPs serve as an input flow; otherwise, the FMs must be considered (Li *et al.*, 2016). The sub-function $S_{3.6}$ models these structural dependencies among components (Yu *et al.*, 2015; Kim *et al.*, 2016; Baylis, Zhang and McAdams, 2018), and similarly to the sub-function $S_{3.3}$, this process is usually depicted by a design structure matrix (DSM), i.e. $[DP]_m = [DSM]_{m\times n}[DP]_n$. However, the nonblank entry $a_{ij} \in [DSM]_{m\times n}$ indicated here, represent a relationship between DP_i and DP_i (Arciniegas and Kim, 2011; Dong, Shao and Xiong, 2011; AlGeddawy and ElMaraghy, 2013). Later, in the sub-function $S_{3.7}$, the DSM is decomposed into physical modules (PM), i.e. $DM_i = \{DP_i, DP_i\}$ (Bonjour *et al.*, 2009). The aim here is to achieve a structure where units are highly interconnected in themselves, but largely independent of other units (Asan, Polat and

Serdar, 2004). In 47.8% of methods that encompass this sub-function, the physical decomposition is performed by meta-heuristics related to optimization problems (Meng, Jiang and Huang, 2007; Agard and Bassetto, 2013; Aydin and Ulutas, 2016), 39.2% adopt heuristics (Agard and Bassetto, 2013; Yu *et al.*, 2015; Baylis, Zhang and McAdams, 2018), and 13.0% do not explicit the approach used (Hsiao *et al.*, 2013; Johannesson *et al.*, 2017; Cheng, Xiao and Wang, 2018). Among those methods assisted by meta-heuristics in functional and physical decomposition, 71.4% make use of evolutionary techniques (Wei *et al.*, 2015; Hou *et al.*, 2017; Wang *et al.*, 2018). It was also noted, in 8.8% of methods that encompass the sub-functions $S_{3.4}$ and $S_{3.7}$, the functional and physical decomposition being performed concurrently (Asan, Polat and Serdar, 2004; Borjesson and Hoelttae-Otto, 2014; Wei *et al.*, 2015). Among the FMs and PMs generated through the decomposition heuristics, only 20.0% are evaluated in the sub-function $S_{3.8}$, while those derived from the meta-heuristics account for less than 14.3%. Independently of what strategy was used to generate the modules, there are common criteria for evaluating them at this stage, as shown in Table 15. The expected output of this step is the clustering refinement feedback to sub-functions $S_{3.4}$ and $S_{3.7}$.

Table 15. Modules' evaluation criteria of sub-function $S_{3.8}$.

Criteria	Frequency	Rel. Freq.	Primary study (Id.)
Modularity	4	20.0%	R12, R35, R37, R72.
Utility	3	15.0%	R1, R12, R72.
Cost	3	15.0%	R1, R12, R72.
Interaction or coupling	2	10.0%	R12, R24.
Lead time	2	10.0%	R12, R30.
Commonality	1	5.0%	R12.
Redesign effort	1	5.0%	R12.
Quality	1	5.0%	R12.
Serviceability	1	5.0%	R12.
Environmental	1	5.0%	R12.
Strategic	1	5.0%	R12.
Total	20	100.0%	

Based on the platform identification approach/criteria coming from the sub-function $S_{3.1}$, the classification of FMs or PMs is then established at sub-function $S_{3.9}$. The generic modules in MBPF are usually classified into platforms and differentiating modules (Hou *et al.*, 2017). The platforms are the ones with less external and internal variety and are suitable for sharing in series instances (Li *et al.*, 2016), while differentiate modules are those with more significant external and internal varieties and are apt to be instantiated in family variants (Fan *et al.*, 2015). The platforms and differentiating modules can assume two different topologies (Du, Jiao and Tseng, 2001). The first is the compound module, which is composed of several sub-modules (Li, Huang and Newman, 2008). The second, is the primitive module, the one that cannot be further decomposed (Li and Huang, 2009). A primitive module, in turn, can be of three types: (i) instance, (ii) scalable, and (iii) empty (Li, Huang and Newman, 2008). An instance module means that all the parameters of the modules have been given. A scalable module means that some of its parameters can be stretched or shrunk, either continuously within a range or discretely in a finite domain. An empty module denotes that the function is optional for the end product, not applying to the platforms. With various modules classified, a configuration structure needs to be established for end product configuration (Rai and Allada, 2003). This process is performed in sub-function $S_{3.10}$ to describe how various product variants are derived from the combination of modules and the interconnections across different levels of assembly (Jiao and Tseng, 1999a). To describe these hierarchical relationships in an MBPF, the concept of Generic Bill-of-Material (GBOM) is usually adopted as a generic data structure (Du, Jiao and Tseng, 2001). A GBOM is defined as an AND/OR tree structure composed (Yan and Stewart, 2010). An AND module is introduced in the GBOM to represent a compound module, wherein all of its sub-modules must be selected and included in a generated product variant. AND modules altogether reflect the commonality of the product family. An OR module is usually an abstract or virtual concept, which means that only one of its sub-modules can be selected and included in the variant. OR modules embody the modularity of a product family. Scalable and instance modules are primitive modules and therefore appear as leaf nodes in the GBOM tree. An empty module is also defined in the GBOM to denote that the function of OR module is optional for the end product. It can only be the sub-module of an OR module as a leaf node in the GBOM tree (Li, Huang and Newman, 2008; Li and Huang, 2009). The output of this stage is the MBPF configuration structure; wherein all modules are supposed to add value to the product. Otherwise, they should not be included in the family (Baldwin and Clark, 2000). In this sense, the sub-function $S_{3.11}$ evaluates the product family as a whole and generates measures of deviation from the ideal, which will serve as feedback improvements (Otto and Hölttä-Otto, 2007). The criteria used for assessing the product families at this stage are presented in Table 16.

Table 16. Product family evaluation criteria of sub-function $S_{3.11}$.

Criteria	Frequency	Rel. Freq.	Primary study (Id.)
Modularity	3	20.0%	R6, R8, R12.
Commonality	3	20.0%	R2, R11, R13.
Utility	2	13.3%	R12, R14.
Variety	2	13.3%	R13, R14.
Interaction or coupling	2	13.3%	R8, R50.
Cost	2	13.3%	R12, R72.
Profit	1	6.7%	R54.
Total	15	100.0%	

The last class of design problems found was the product family configuration, which deals with structural configuration problem wherein the modules formulating the variants are optimally selected from the given MBPF structure (Simpson *et al.*, 2014). In this class, the model starts by defining and modeling the configuration criteria in sub-function $S_{4.1}$, wherein decisions on what criteria to use for modelling the combinatorial and parametric problem are made (Li, Huang and Newman, 2008; Li and Huang, 2009; Colombo *et al.*, 2019). The most common criteria used in this sub-function are presented in Table 17

Table 17. Configuration criteria of sub-function $S_{4.1}$.

Criteria	Frequency	Rel. Freq.	Primary study (Id.)
Cost	9	23.1%	R9, R19, R28, R41, R49, R61, R63, R68, R70.
Demand	7	17.9%	R9, R25, R28, R41, R59, R63, R68.
Profit	5	12.8%	R9, R19, R25, R28, R68.
Price	5	12.8%	R9, R19, R25, R28, R68.
Utility	5	12.8%	R28, R41, R63, R68, R70.
Commonality	3	7.7%	R21, R23, R69.
Quality	2	5.1%	R7, R9.
Modularity	1	2.6%	R69.
Interaction or coupling	1	2.6%	R49.
Environmental	1	2.6%	R63.
Total	39	100.0%	

There are some cases where the configuration space is quite huge (Zhu *et al.*, 2010). Therefore, it is economically unviable for most companies to test and measure all the derivatives one by one (Meng *et al.*, 2014). So generally a limited amount of variants are chosen for performance experiments, and the measurement results are saved into the database as the representative performances for a family of products (Zhu *et al.*, 2010; Meng *et al.*, 2014). In this sense, the sub-function $S_{4.2}$ not only reduces the configuration space but also analyses the historical modularized product configurations to set up prediction models through data mining and regression analysis (Zhu *et al.*, 2010). In such cases where the reduction of configuration space is not required, the original MBPF configuration should be used as an input flow for combining the modules into product family variants. This task is performed by the sub-functions $S_{4.3}$ and $S_{4.4}$. At sub-function $S_{4.3}$ the right combination of modules to formulate the variants is found (Xiong, Du and Jiao, 2018). Then, if there are scalable modules within these variants, the parameters of the modules are determined in sub-function $S_{4.4}$ (Xiao *et al.*, 2018; Colombo *et al.*, 2019). This process of combining and scaling modules is assisted by meta-heuristics in 81.0% of the methods analyzed (Adhitama and Rosenstiel, 2015; Chowdhury *et al.*, 2016). Coupled with that, 58.8% of it makes use of evolutionary techniques (Xiao *et al.*, 2018; Xiong, Du and Jiao, 2018). Then, the structure of the product family variants derived from the combination process is represented in the sub-function $S_{4.5}$ closing the functional model (Du, Jiao and Tseng, 2001; Li, Huang and Newman, 2008; Li and Huang, 2009).

3.3.2 Structured Classes of Problems

With the functional model defined, the next issue was to check the quality of the classes of design problems defined during the encoding process. First, the heuristics proposed by Stone, Wood, and Crawford (2000) were employed to compare if the clustering solution resulting from this process differs from the original proposition. As a result, the four classes of design problems (Cp_1, Cp_2, Cp_3, Cp_4) were identified through the conversion-transmission heuristic, and the only difference found in relation to the original proposition was that the sub-function $S_{1.2}$ (segment the market) should integrate the second class (Cp_2) instead of the first (Cp_1). Then, the interactions among the classes of design problems were depicted in a design structure matrix to calculate the MI as shown in Figure 16. The MI assesses the quality of a clustering solution, ranging from 0 to 1, by capturing the degrees of connection strengths within each independent class and between different classes (MI_1), the density of connections within each class and between classes (MI_2), and the proximity of interactions to the diagonal of the design matrix (MI_3) (Jung and Simpson, 2017). With MI_1, MI_2, and MI_3 considered equally important, i.e. $(w_1, w_2, w_3) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$, the MI was calculated, and the quality of the clustering solution was found to be equals to 0.64. Although the conversion-transmission heuristic indicated the $S_{1.2}$ as being part of Cp_2, as suggests the grey area in Figure 16, it was purposely reattached to the Cp_1 (original proposition) in order to achieve better MI results. Subsequently, the MI was recalculated, and the quality of the clustering solution improved to 0.72. The rectangles outlined in black in Figure 16 represent the final clustering solution.

	$S_{1.1}$	$S_{1.2}$	$S_{2.1}$	$S_{2.2}$	$S_{2.3}$	$S_{2.4}$	$S_{2.5}$	$S_{2.6}$	$S_{2.7}$	$S_{3.1}$	$S_{3.2}$	$S_{3.3}$	$S_{3.4}$	$S_{3.5}$	$S_{3.6}$	$S_{3.7}$	$S_{3.8}$	$S_{3.9}$	$S_{3.10}$	$S_{3.11}$	$S_{4.1}$	$S_{4.2}$	$S_{4.3}$	$S_{4.4}$	$S_{4.5}$	
$S_{1.1}$	x	1	Cp_1																							
$S_{1.2}$	1	x								Cp_2																
$S_{2.1}$		1	x																							
$S_{2.2}$			1	x																						
$S_{2.3}$			1		x																					
$S_{2.4}$					1	x																				
$S_{2.5}$					1		x																			
$S_{2.6}$				1	1			x																		
$S_{2.7}$						1	1	1	x												Cp_3					
$S_{3.1}$										x										1						
$S_{3.2}$	1								1		x															
$S_{3.3}$									1	1	1	x														
$S_{3.4}$												1	x				1									
$S_{3.5}$											1		1	x												
$S_{3.6}$										1				1	x											
$S_{3.7}$															1	x	1									
$S_{3.8}$													1			1	x									
$S_{3.9}$										1			1			1		x								
$S_{3.10}$																		1	x	1						
$S_{3.11}$																			1	x						Cp_4
$S_{4.1}$	1								1												x					
$S_{4.2}$																			1		1	x				
$S_{4.3}$																			1		1	1	x			
$S_{4.4}$																							1	x		
$S_{4.5}$																							1	1	x	

Figure 16. Interactions among the classes of design problems.

The quality of the clustering solution was considered satisfactory, and the structured classes of problems were then formalized, as shown in Appendix A (Table A3). From this table, it was possible to derive that 75.0% of methods have been evaluated in depth by observational techniques such as case studies. In 26.4% of the studies, the methods have been assessed descriptively by constructing detailed scenarios to demonstrate the artifacts' utility, and in 2.8% of situations, they have been tested experimentally through the use of simulation. The sum of frequencies is greater than 100% because of the methods M_{12}, M_{18}, and M_{21} have been evaluated by more than one approach. Regarding the type of products these methods have been developed for, 47.2% of them have been used to design consumers' durables, 27.8% to intermediate goods, 22.2% to capital goods, and 5.6% to military and defense goods. The sum of frequencies here is greater than 100% due to the M_{12} has been used to design more than one type of product. Still, from Table A3, it is possible to derive which technique, among those 193 identified here, is intended to solve each design problem. The definition of each method and technique is presented in Appendix A (Table A2).

Later, Table A3 was converted into a binary matrix (Table A4), wherein only primary studies (R_i), design problems (Pb_i), and classes of design problems (Cp) were considered. In Table A4, the primary studies were rearranged to facilitate the understanding of association rules. Using the Apriori algorithm in software R, with the parameters set to $s(X \rightarrow Y) \geq 0{,}1$, $c(X \rightarrow Y) \geq 0{,}1$, and the relationship width ≥ 2, 13 association rules were identified as shown in Figure 17.

```
     lhs            rhs   support   confidence lift      count
[1]  {CP2}       => {CP3} 0.4166667 0.8823529  1.0953347 30
[2]  {CP3}       => {CP2} 0.4166667 0.5172414  1.0953347 30
[3]  {CP1}       => {CP3} 0.2222222 0.8000000  0.9931034 16
[4]  {CP3}       => {CP1} 0.2222222 0.2758621  0.9931034 16
[5]  {CP4}       => {CP3} 0.1944444 0.5384615  0.6684350 14
[6]  {CP3}       => {CP4} 0.1944444 0.2413793  0.6684350 14
[7]  {CP1}       => {CP2} 0.1666667 0.6000000  1.2705882 12
[8]  {CP2}       => {CP1} 0.1666667 0.3529412  1.2705882 12
[9]  {CP1,CP2}   => {CP3} 0.1388889 0.8333333  1.0344828 10
[10] {CP1,CP3}   => {CP2} 0.1388889 0.6250000  1.3235294 10
[11] {CP2,CP3}   => {CP1} 0.1388889 0.3333333  1.2000000 10
[12] {CP4}       => {CP2} 0.1111111 0.3076923  0.6515837  8
[13] {CP2}       => {CP4} 0.1111111 0.2352941  0.6515837  8
```

Figure 17. Association rules derived from the Apriori algorithm executed in software R.

Rules 1 and 2 indicate that the classes of design problems Cp_2 and Cp_3 co-occur in 41.7% of the primary studies. Still, from these rules, it is possible to infer that Cp_3 is more likely to appear (88.2%) when Cp_2 is present, but the opposite does not happen with the same level of confidence (51.7%). The same reasoning can be applied to the rules from 7 to 11 but at lower support levels. Regarding the rules 3 and 4, the lift values are close to one, which implies no association between the classes Cp_1 and Cp_3 and vice-versa. However, if the Cp_2 is considered together, as posed by rule 9, the lift value increases, and the confidence reaches the level of 83.3%. Rules 5, 6, 12, and 13, in turn, do not follow the same pattern. The lift value lower than one means that item Cp_4 is unlikely to co-occur with Cp_3, or with Cp_2.

In this section, it was presented the results of the systematic literature review, wherein 72 methods, 25 design problems, 25 sub-functions, 193 techniques, 3 evaluation approaches, and 4 types of products, were found. These methods were connected through its corresponding sub-functions in the form of a functional model. While the remaining instances were organized into structured classes of design problems, which had its most common association rules identified. The discussion of the results is presented next.

3.4 Discussion of the Results

The functional model elaborated in this research represents the common underlying structure of MBPF design methods developed over the past 20 years. In this sense, the model not only presents the sub-functions (main steps of the methods) intended to solve the design problems but also indicates the causal relationship (execution order) among them along with its respective input and output flows. The structured classes of design problems, in turn, complements the functional model by cataloging the techniques meant to execute each sub-function of the model. Moreover, it presents the definitions of methods and techniques, how they have been tested, and for what product they have been developed. These entities together serve as a meta-method for organizing the research in the field of MBPF design as well as a roadmap for implementing the MBPFs in industry.

From these entities, it was possible to derive some patterns. The first thing noted from the association rule 1, was that, in 41.7% of the methods, the Cp_3 has 88.2% of probability to exist when Cp_2 is present. This rule consists of the most influential association found here and indicates a consistent presence of market considerations within the product family modeling. However, looking at Table A4 and considering the principle that if an itemset is frequent, then all of its subsets must also be frequent (Tan *et al.*, 2019), it is possible to observe that in only 6.9% of the methods, the FRs derive from the CNs. In other words, among those 30 methods that consider Cp_2 and Cp_3 concurrently, only 5 methods formulate the FRs in sub-function $S_{2.3}$ from the CNs identified in sub-function $S_{2.1}$. What indicates that the FR are formulated by other means, such as reverse engineering and benchmarking of existing solutions (Jiao and Tseng, 1999a; Thevenot and Simpson, 2007; Simpson *et al.*, 2012). These findings comply with previous research indications of lacking methods on customer modeling and integration (Jiao, Simpson and Siddique, 2007; Kumar, Chen and Simpson, 2009).

Another interesting pattern identified in Cp_3, is that only 4.2% of methods perform the functional and physical decomposition concurrently (Asan, Polat and Serdar, 2004; Borjesson and Hoelttae-Otto, 2014; Wei *et al.*, 2015). The functional decomposition is coupled with the concept of modular product architectures (Jiao and Tseng, 1999a, 2000; Jiao, Simpson and Siddique, 2007), which allows each functional element of the product to be changed independently by only changing the corresponding component (Meyer and Lehnerd, 1997). The physical decomposition, in turn, relates to the level of coupling or interaction among components in a product (Arciniegas and Kim, 2011; Dong, Shao and Xiong, 2011; AlGeddawy and ElMaraghy, 2013), not necessarily taking into account its respective functionalities. In our point of view, the FM must physically match the working structure (Pahl *et al.*, 2007), therefore we believe these two decomposition strategies should be handled somehow together, otherwise, this process might result in integral architectures instead of modular ones. Besides that, although it is slightly tackled by the methods M_1 and M_{72} (Jiao and Tseng, 1999a; Gauss, Lacerda and Sellitto, 2019), in sub-functions $S_{3.4}$ and $S_{3.8}$, the existing methods seem to lack a sub-function to specify the parameters of primitive modules (instance or scalable) based on the FR target values and ranges. Still, from the Cp_3, it was also observed that among the FMs and PMs generated through the methods that adopt decomposition heuristics, only 20.0% were evaluated in the sub-function $S_{3.8}$, what indicates the open-loop nature of these methods. On the other hand, among the FMs and PMs decomposed through the methods that adopt meta-heuristics, 14.3% were evaluated in the sub-function $S_{3.8}$, what does not make sense since they already use evaluation criteria during the optimization.

From the rules 5 and 6, based on the lift values lower than one, it is possible to infer that Cp_4 is unlikely to co-occur with Cp_3. Although they appear together in 19.4% of the methods, the relationship between these classes occur, in 57.1% of the cases, by the utilization of the product family structure, retrieved from the subfunction $S_{3.10}$, as an input flow for combining the modules into product family variants in sub-function $S_{4.3}$. This relationship takes the assumption that the modules' set already exists, being deeply sensitive to the ability of the existing modules in accomplishing the customer desired attributes (Zhu *et al.*, 2010; Jiao, 2012; Yifei *et al.*, 2015). To put it differently, if the module set is compound by low utility modules, no matter how robust the configuration procedure is, only low utility variants will be instantiated.

This process of mixing, matching, and scaling modules to generate product family variants is assisted by meta-heuristics in 81.0% of the methods analyzed (Adhitama and Rosenstiel, 2015; Chowdhury *et al.*, 2016). In this sense, broader business indicators such as demand, cost, price, and profit are usually employed at this stage to optimally solve the combinatorial and parametric problem (Li, Huang and Newman, 2008; Li and Huang, 2009). However, nor a threshold to evaluate if the variants instantiated satisfy the minimum requirements desired in a product, neither feedbacks flows to the sub-functions $S_{1.1}$ or $S_{3.1}$ leading to new modules' developments were found. Coupled with that, it is not explicit in the works examined here, the product family configuration (Cp_4) supporting or even playing the role of product line planning, an issue that has been traditionally dealt with in the management science and marketing literature (Jiao, Simpson and Siddique, 2007).

Finally, although it did not appear in the most frequent association rules, it was noted from Table A4, that M_9 (Jiang and Allada, 2005), accounting for 1.4% of the methods, is the unique method that integrates the four classes of design problems identified in this research. Linked to that, rule 9 indicates that 13.9% of the methods consider the three first fist classes of design problems concurrently. These results reinforce the previous research indication of lacking integrated design methods in this field (Simpson *et al.*, 2012; Otto *et al.*, 2016).

3.5 Conclusions

Over the years, active work in developing methods to design MBPFs has been done. However, many of them have been created, and consequently exist, in isolation from one other. As a result, the adoption of these methods in industry and academy alike is inhibited by the seemingly broad array of material without a coherent organizing structure. To bridge this gap, this paper performed a systematic literature review, wherein 72 methods (1999-2019) to design MBPFs and their respective instances have been connected in the form of a functional model and structured classes of design problems. These entities together serve as a meta-method for organizing the research in the field of MBPF design as well as a roadmap for implementing MBPFs in the industry. The main contributions of this work include: (i) constructing a functional model that connects the design methods for MBPFs; (ii) suggesting structured classes of design problems that complement the functional model by cataloging the techniques meant to execute each sub-function of the model; (iii) proposing a construction heuristic to build and assess functional models and classes of design problems.

Regarding the limitations of the present study, three were identified. The first lies in the fact that some useful literature might have been omitted since no relook at the references of those studies included in the review has been performed. The second is that association rules are sensitive to the classes of design problems. In this sense, the adoption of optimization techniques, using MI as an objective function, for the classes' formation might enhance the robustness of future research. The third is that the study did not catalog the tools (software) to operationalize the identified techniques.

Our particular interest in this research regarded the module-based product family design, one of the approaches used to design product families. Therefore, the integrative connection among the existing methods and its respective shortcomings presented here accounted for a fraction of the theoretical framework on product family design. What makes us believe that future works synthesizing and integrating the extant methods to design product families, independent of the type of product architecture they have, could bring broader integrative interpretations, than the ones presented in this study.

4 ARTICLE 2 - MARKET-DRIVEN PRODUCT FAMILY DESIGN: SYSTEMATIC LITERATURE REVIEW AND META-SYNTHESIS [2]

[2] Article to be submitted to the Journal of Research in Engineering Design (RED).

Abstract: Customers' needs continually evolve and shift over time, and the result is an increasing demand for product variety and newer versions of products. Believing the product variety can help manufacturing companies to increase sales and profits, many companies have been attempting to provide more product variants without sacrificing production efficiency. One way to manage this trade-off is through the product family design, a field of study wherein marketing, engineering, and economic aspects are highly interdependent. To understand how the interconnected relationships among these three domains take place into the product family design, this paper presents a systematic literature review and meta-synthesis of 21 articles (1998-2019) published in peer-reviewed journals. The research findings are synthesized in the form of a functional model and structured classes of design problems, wherein the existing methods to design market-driven product families and their corresponding instances are connected. Besides that, the main contribution of this work includes the identification of four design problems and sub-functions not reported by previous works regarding the synthesis of methods for designing module-based product families.

Keywords: product family design, market-driven product family design, systematic literature review, meta-synthesis.

4.1 Introduction

Customers' needs continually evolve and shift over time, and their demand for product variety and newer versions of products has increased rapidly in recent decades (Simpson *et al.*, 2014). Based on the belief the product variety can help manufacturing companies to increase sales and profits (Wei *et al.*, 2015; Zhu, Li and Feng, 2017), many companies have been attempting to provide more product variants without sacrificing production efficiency (Jiao, Simpson and Siddique, 2007).

One way to manage this conflict is through the design of product families (Simpson *et al.*, 2014). Generally, a product family refers to a set of products derived from a standard product platform to satisfy various market applications (Meyer and Lehnerd, 1997). Platforms, in turn, are intellectual and material assets shared across a product family to minimize manufacturing complexity (Erens and Verhulst, 1997). In this context, the prominent approach to product family design is through the development of module-based product families, wherein product family members are instantiated by mixing and matching functional modules from the platform (Ulrich, 1995; Du, Jiao and Tseng, 2001). An alternative approach, considered as a subset of the former (Fujita and Yoshida, 2004), is through the development of a scale-based product family, which consists of scaling one or more variables to change the platform specifications while common parameters remain constant (Simpson, 2004).

The product family design is challenging for many aspects, and addressing its front-end issues is a complex activity (Colombo *et al.*, 2019). In general, the front-end issues are subdivided into four prevalent classes of design problems: (i) product family positioning, (ii) market-driven product family design, (iii) product family modeling, and (iv) product family configuration (Gauss, Lacerda and Miguel, 2020) (Article 1). The first two classes account for the marketing-related issues, which include customer involvement, product portfolio design, product family positioning, and transition or mapping from customer needs to functional requirements (Simpson *et al.*, 2014). While the last two classes are grounded on engineering-related issues, which include product family configuration, product architecture, design of families and platforms, leveraging commonality and modularity, and optimization of the family and platform design (Simpson *et al.*, 2014). A recent study, concerning 72 methods for designing module-based product families, has shown that 1.4% of methods address the four classes of design problems concurrently. Among those methods (41.7%) considering marketing-related issues in its formulation, less than 7% derive the desired attributes in a product straight from the customers. Still from this study, it is seen that only 15.3% of methods account for enterprise-level indicators in product family configuration (Gauss, Lacerda and Miguel, 2020) (Article 1). Findings that comply with previous research indications of lacking methods integrating marketing, engineering, and economic aspects into product family design (Jiao, Simpson and Siddique, 2007; Kumar, Chen and Simpson, 2009; Colombo *et al.*, 2019).

The problem is the marketing and engineering variables are often highly interdependent in product family design. Moreover, the coupled relationships between them imply that any change in one variable can potentially influence the outputs of the other(s), with both affecting the economic benefits of an enterprise (Chen, Hoyle and Wassenaar, 2013). Therefore, in the design of optimal or near-optimal product families, marketing, engineering, and economic requirements often cannot be pursued separately or even sequentially (Luo, 2011).

Based on the aforementioned, and given the fact the module-based approach accounts for a fraction of the theoretical framework on product family design, the following research questions arose: (i) which methods address marketing-related issues into product family design, independent of the type of product architecture? (ii) which methods encompass broader business indicators into product family design? (iii) what kind of design problems do these methods account for? (iv) for which type of products have these methods been developed? (v) how has the performance of these methods been assessed? (vi) what are the main steps of these methods? (vii) what is the execution order of these steps? (viii) which techniques are used to execute each step of these methods? (ix) is there a common underlying structure among these methods?

This paper aims at answering these questions through a systematic literature review in addition to a meta-synthesis of 21 articles that integrate marketing-related issues and broader business indicators into product family design. These articles have been published between 1998 and 2019 in international journals that include research on engineering, production, and operations management. Besides the building of a functional model for designing market-driven product families along with its respective structured classes of design problems, the main contribution of this work includes the identification of four design problems and sub-functions not reported by previous works regarding the synthesis of methods for designing module-based product families.

The remainder of this paper is structured as follows. Section 4.2 contains the research approach. Section 4.3 shows the functional model and the structured classes of design problems resulting from the literature mapping and analysis, followed by Section 4.4 which critically analyses the research findings. Finally, Section 4.5 presents the research contributions and limitations along with its future directions.

4.2 Systematic Literature Review

The product family design, as well as other engineering disciplines, is typically concerned with construction problems related to not yet existing entities (van Aken and Romme, 2009; Vaishnavi, Kuechler and Petter, 2017). This conception complies with the goals of research performed under the design science paradigm, which seeks to produce knowledge to solve real problems or to design something that does not yet exist (Simon, 1996; van Aken, 2005). While design science is the epistemological basis, design science research is the method that operationalizes research in this field (Lacerda *et al.*, 2013). Given the conceptual coupling between product family design and design science, this paper adopts the nine-step systematic literature review adapted to design science research by Gauss, Lacerda, and Miguel (2020) (Article 1), as shown in Figure

18.

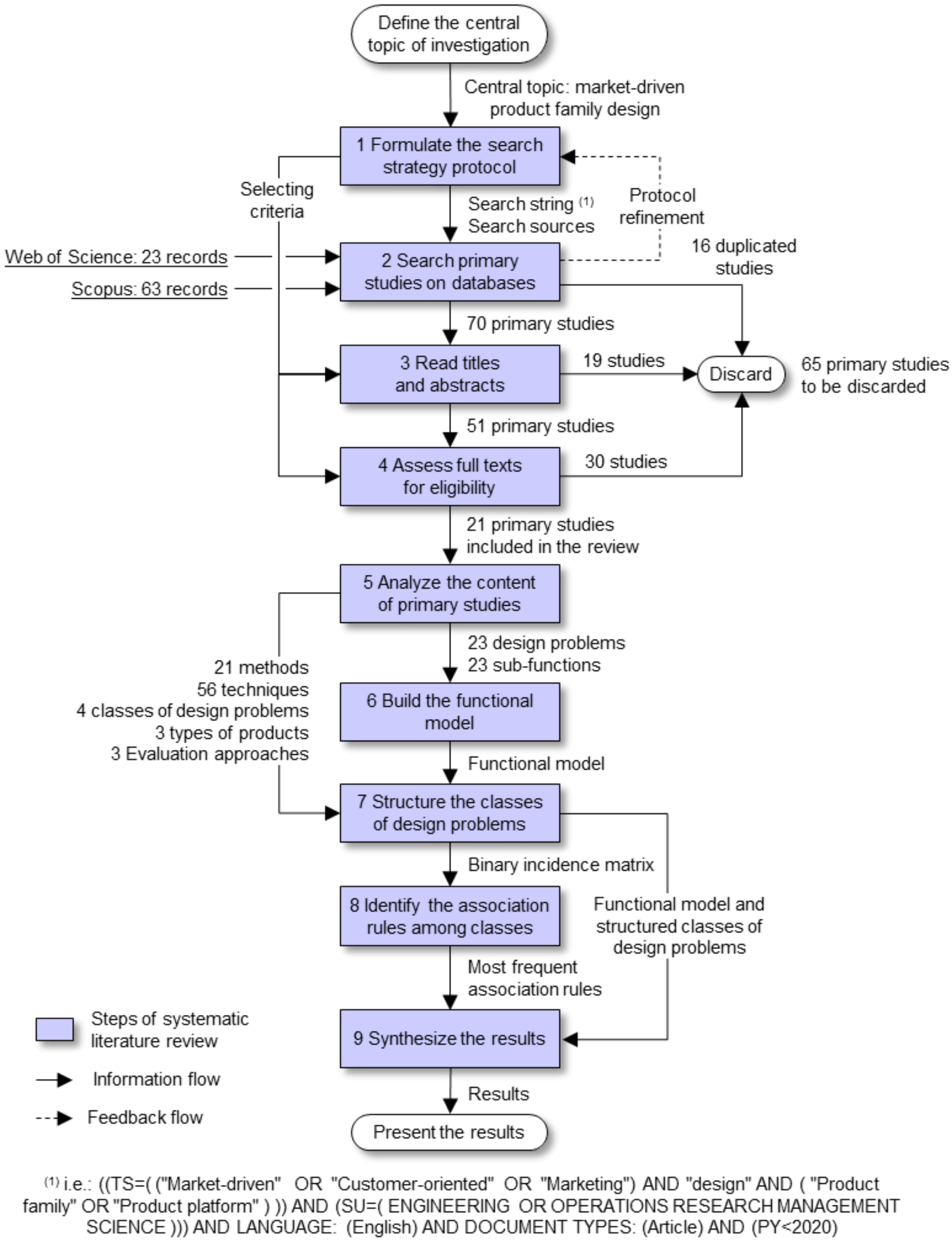


Figure 18. Systematic literature review workflow.

The process started by defining the market-driven product family design as the central topic of this research. Then, in step 1, to clarify the review question and to limit its scope, a conceptual framework was developed based on the fundamental references regarding (i) product family design, (ii) product line planning, and (iii) product development process. As a result, the following review questions and the search string presented in Figure 18 were formulated.

- Which methods address marketing-related issues into product family design, independent of the type of product architecture?
- Which methods encompass broader business indicators into product family design?
- What kind of design problems do these methods account for?
- For which type of products have these methods been developed?
- How has the performance of these methods been assessed?
- What are the main steps of these methods?
- What is the execution order of these steps?
- Which techniques are used to execute each step of these methods?
- Is there a common underlying structure among these methods?

The search was conducted in Web of Science and Scopus since both provide quick access to the principal citation databases worldwide and have smart tools to track, analyze, and visualize research (Morandi and Camargo, 2015; *Content - How Scopus Works - Scopus - | Elsevier solutions*, 2017; Clarivate Analytics, 2019). Besides that, to ensure the quality of the primary studies, only articles published in peer-reviewed international journals have been considered. Therefore, the English language was used as an inclusion criterion. Concerning the period and subject area, the articles published up to 2020 that encompassed the research in engineering, production, and operations management were consulted. Appendix B (Table B1) shows additional selecting criteria in the search strategy protocol.

With the research strategy defined and based on a search limited to the article title, abstract, and keywords, 86 primary studies were found in step 2. Then they were checked for duplicates, followed by an inspection of the titles and abstracts in step 3 (Brunton, Stansfield and Thomas, 2012). After, 51 potentially relevant studies were analyzed in-depth in step 4, and 21 in compliance with the research scope were selected for review, as shown in Figure 18. Table 18 presents the excluding statistics, while Table 19 gives the list of 21 primary studies included in the review.

Table 18. Excluding statistics.

No. of exclusions	Percentage	Excluding criteria
23	35.4%	Absence of methods or techniques addressing market considerations and broader business indicators into product family design
16	24.6%	Duplicated studies
5	7.7%	Manufacturing and production for product families
5	7.7%	Supply chain issues of product families
3	4.6%	Design support systems
3	4.6%	Theoretical development and synthesis of product family design
3	4.6%	Literature review on product family design
2	3.1%	Service design
2	3.1%	Paper not found
1	1.5%	Fundamental issues on product family design
1	1.5%	Out of context
1	1.5%	Software development
61	100.0%	Total

Table 19. List of primary studies included in the review.

Id	Title	Authors and year
R_1	Product line development with customer interaction	(Márkus and Váncza, 1998)
R_2	Design for variety: Developing standardized and modularized product platform architectures	(Martin and Ishii, 2002)
R_3	Effective product family design using physical programming	(Messac, Martinez and Simpson, 2002)
R_4	Product platform design to improve commonality in custom products	(Farrell and Simpson, 2003)
R_5	Prescribing the content and timing of product upgrades	(Wilhelm, Damodaran and Li, 2003)
R_6	An integrated method for designing modular products	(Asan, Polat and Serdar, 2004)
R_7	A structural component-based approach for designing product family	(Hsiao and Liu, 2005)
R_8	Understanding customer satisfaction in product customization	(Du, Jiao and Tseng, 2006)
R_9	Mapping product design specification for mass customization	(Krishnapillai and Zeid, 2006)
R_{10}	Designing a family of development-intensive products	(Krishnan and Zhu, 2006)
R_{11}	Managing modularity in product family design with functional modeling	(Zhang, Tor and Britton, 2006)
R_{12}	Market segmentation for product family positioning based on fuzzy clustering	(Zhang, Jiao and Ma, 2007)
R_{13}	A method to improve platform leveraging in a market segmentation grid for an existing product line	(Farrell and Simpson, 2008)
R_{14}	Optimal product portfolio formulation by merging predictive data mining with multilevel optimization	(Tucker and Kim, 2008)
R_{15}	Optimal platform investment for product family design	(Zacharias and Yassine, 2008)
R_{16}	Integration of marketing research techniques into house of quality and product family design	(Kazemzadeh *et al.*, 2009)
R_{17}	Evolutionary product line design balancing customer needs and product commonality	(Chen, Jiao and Tseng, 2009)
R_{18}	A market-driven approach to product family design	(Kumar, Chen and Simpson, 2009)
R_{19}	Research on customer-oriented optimal configuration of product scheme based on Pareto genetic algorithm	(Yifei *et al.*, 2015)
R_{20}	Product family architecture design with predictive, data-driven product family design method	(Ma and Kim, 2016)
R_{21}	Coordinated optimisation of platform-driven product line planning by bilevel programming	(Miao *et al.*, 2017)

The next step was to perform a content analysis in step 5 (Bardin, 1993; Mayring, 2014). With this regard, the primary studies included in the review configured the context units from which the registration units, i.e., text quotations, have been encoded (Bardin, 1993). Coupled with that a coding system composed of a mixed coding scheme and counting principles was established. The mixed coding scheme, compound by categorical and open codes is shown in Appendix B (Table B2) (Oliver and Sutcliffe, 2012; Dresch, Lacerda and Antunes Jr, 2015). Regarding the counting principles, the ones adopted in this research were the occurrence, co-occurrence, and frequency. The occurrence relates to the presence of code in a context unit, while co-occurrence consists of the simultaneous presence of two or more codes in a context unit (Bardin, 1993). The frequency, in turn, can be determined as the number of times each code occurs in the corpus of analysis, in other words, the set of primary studies included in the review (Gauss, Lacerda and Miguel, 2020) (Article 1).

Still in step 5, after defining the coding system, the next task was to encode and understand the raw data. This process was assisted by the qualitative data analysis software Atlas Ti (*ATLAS.ti 8 Windows | ATLAS.ti*, 2019), wherein 21 methods, 23 design problems, 23 sub-functions, 56 techniques, 3 evaluation approaches, 3 types of products, and 4 classes of design problems were found. One important issue here was to link the design problem to its potential class of design problems. Besides that, it was also needed to identify the sequence the design problems occur along the course of the methods analyzed. This procedure of establishing the causal relationship between problems was performed through a syntopic reading (Adler and van Doren, 1972), and based on the reasoning of effect-cause-effect retrieved from Theory of Constraints thinking process (Cox and Schleier, 2010). Figure 19 gives an example of a code hierarchy resulted from this process, wherein six design problems, organized in sequence, compound the second class of design problems for product families.

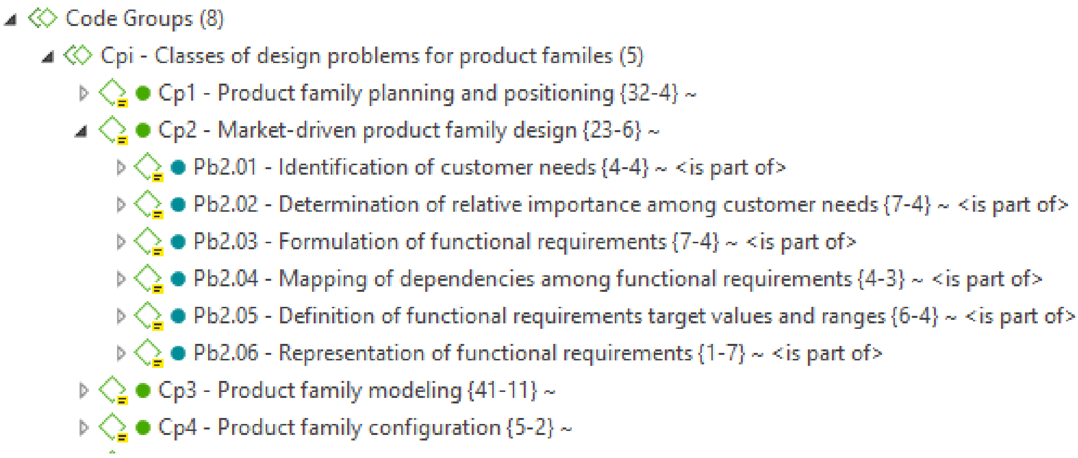


Figure 19. Example of codes hierarchy in software Atlas Ti.

The goal of meta-synthesis is to produce a new and integrative interpretation of findings that is more substantive than those resulting from individual investigations (Jensen and Allen, 1996; Finfgeld, 2003). In this sense, the final product is usually expressed in graphical form to permit mapping the nature and variety of concepts studied, identifying associations between different subjects, and providing explanations for the results from the various primary studies (Dresch, Lacerda and Antunes Jr, 2015).

With that intention, and following the heuristic proposed by Gauss, Lacerda, and Miguel (2020) (Article 1), the next issue was to formulate a functional model that connects the functionalities of all methods identified in step 6. In this context, the first thing to do was to convert a design problem (Pb_i) into a sub-function (S_i) corresponding to an action intended to solve it. After, the sub-functions were connected by the information flows on which they operate. The execution order of each sub-function followed the sequence resulting from the enconding task ilustrated in Figure 19. Later, the clustering of design problems into classes, defined during the encoding process, was depicted in a design structure matrix (Browning, 2001) and had its quality assessed by the Modularity Index (MI) (Jung and Simpson, 2017). Details on the design structure matrix and MI are given in its references. According to this checking, those problems not belonging to a previously assigned class were relocated, and the coding system, along with the encoding process of raw data, was updated. Figure 20 gives an example of a generic functional model, wherein each sub-function (S_i) corresponds to an action intended to solve a particular design problem (Pb_i). The dashed lines around the sub-functions represent a class of design problems (Cp_i), defined here as a set of design problems which share common characteristics, and contain useful artifacts for their solution, i.e. $Cp_i = \{Pb_i, T_i, M_i\}$ (Dresch, Lacerda and Antunes Jr, 2015).

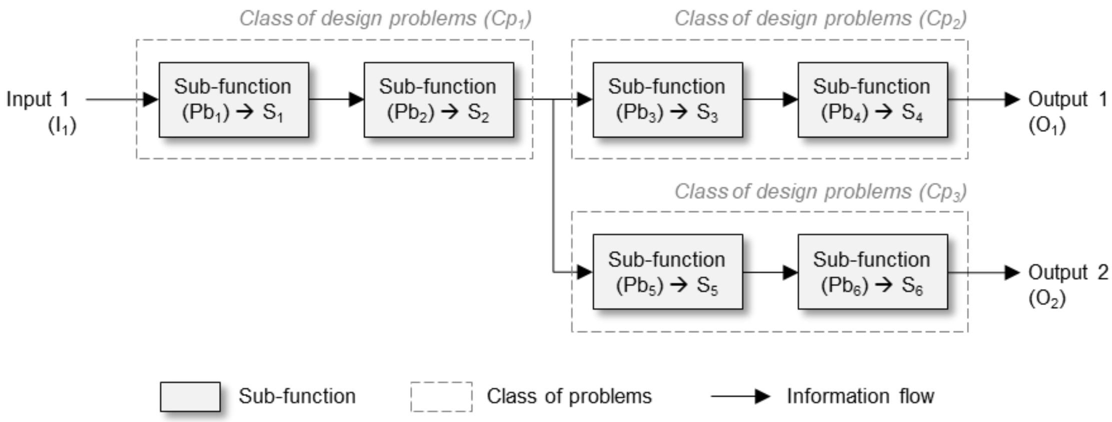


Figure 20. Example of a generic functional model (Gauss, Lacerda and Miguel, 2020).

After checking the clustering solution, in step 7, the structured classes of design problems (Cp_i) were established based on the co-occurrence among the design problems (Pb_i), techniques (T_i), methods (M_i), evaluation approaches (Et_i), products classification (Pt_i), and primary studies (R_i) as shown in Table B3. Besides that, to support the subsequent association analysis, part of the content of Table B3 was converted into a binary incidence matrix, as presented by Table B4.

The association analysis, performed in step 8, is useful for discovering relationships hidden in large data sets (Zhang and Zhang, 2002). The uncovered relationships can be expressed in terms of association rules or sets of frequent items, i.e.$\{Cp_1\} \rightarrow \{Cp_2\}$. This rule suggestes that a relationship exists between Cp_1 and Cp_2. In the context of this work, let $C = \{c_1, c_2, \dots c_i\}$ be the set of all codes retrieved from the content analysis and $R = \{r_1, r_2, \dots r_i\}$ be the set of all relationships between primary studies and codes. Each relationship r_i contains a subset of codes assigned from C. In association analysis, a collection of zero or more items is termed an itemset. If an itemset contains k items, it is called a k-itemset. For instance, $\{Cp_1, Cp_2\}$ is an example of a 2-itemset. The null (or empty) set is an itemset that does not contain any items. The relationship width is defined here as the number of items present in a relationship. A relationship r_i is said to contain an itemset X if X is a subset of r_i. An essential property of an itemset is its support count, which refers to the number of relationships that contain a particular itemset (Tan *et al.*, 2019). Mathematically, the support count, $\sigma(X)$, for an itemset X can be stated as follows:

$$\sigma(X) = |\{r_i | X \subseteq r_i, r_i \in R\}| \qquad 6$$

where the symbol $|.|$ note the number of elements in a set. An association rule is an implication expression of the form $X \rightarrow Y$, where X and Y are disjoint itemsets, i.e., $X \cap Y = \emptyset$. The strength of an association rule can be measured in terms of its support, confidence, and lift (Zhang and Zhang, 2002; Gkoulalas-Divanis and Verykios, 2010; Tan *et al.*, 2019). Support determines how often a rule applies to a given data set, while confidence determines how frequently items in Y appear in relationships that contain X. Lift, in turn, computes the ratio between the rule's confidence and the support of the itemset in the rule consequent. The formal definitions of these metrics are:

$$\text{Support, } s(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{N} \qquad 7$$

Confidence, $c(X \rightarrow Y) = \frac{\sigma(X \cup Y)}{\sigma(X)}$ 8

Lift, $l(X \rightarrow Y) = \frac{c(X \rightarrow Y)}{s(Y)}$ 9

A lattice structure can be used to enumerate the list of all possible itemsets. However, in many practical applications, the search space of itemsets is exponentially large (Tan *et al.*, 2019). To reduce the search space during frequent itemset generation, this research adopted the Apriori, which consists of an association rule mining algorithm that uses support-based pruning to systematically control the exponential growth of candidate itemsets (Zhang and Zhang, 2002). Its execution on this work was performed in software R (Team, 2019). Details on the frequent itemset generation through the Apriori algorithm is provided in (Tan *et al.*, 2019).

In summary, this section provided methodological procedures for searching, selecting, and analyzing the content of primary studies. Besides that, with the purpose of synthesizing the findings, a construction heuristic to build and assess functional models and classes of design problems was shown. Additionally, a technique to identify the association rules among the classes of design problems was presented. The results of this process, accounting for step 9, are synthesized and presented in the next section.

4.3 Meta-Synthesis

The presentation of meta-synthesis is divided into two parts. The first part introduces the functional model that connects the market-driven product family design methods identified in this research. In this sense, the model not only presents the sub-functions intended to solve the design problems but also indicates the causal relationship among them along with its respective input and output flows. The second part, in turn, shows the structured classes of design problems that complement the functional model by cataloging the techniques meant to execute each sub-function of the model. Moreover, it presents the definitions of methods and techniques, how they have been tested, for what product they have been developed for, and the most common association rules concerning the classes and its design problems.

4.3.1 Functional Model of Market-Driven Product Family Design

From the corpus of analysis, 21 methods were identified. Although these methods do not tackle the same design problems, they somehow share the same objective of designing product families from a market-driven perspective. In this sense, following the reasoning of Stone and Wood (2000), we derived the overall function of this set of methods as being the: *"Design market-driven product families."* The overall function expresses the solution-neutral relationship between inputs and outputs (Pahl *et al.*, 2007). With this respect, seven inputs and one output flows were identified, as shown in Figure 21(a). For each input flow, a chain of sub-functions was established based on design problems that emerged during the content analysis. Then, the function chains were aggregated in a functional model, and the sub-functions were grouped into classes of problems, as presented in Figure 21(b). In this model, the continuous and dashed arrows respectively indicate the information and feedback flows among the sub-functions, while the dashed rectangles represent the classes of design problems.

The functional model was compound by 23 sub-functions subdivided into 4 classes of design problems. The first class of design problems identified was the product family planning and positioning, which deals with market objectives and technology developments guided by corporate strategies (Ulrich and Eppinger, 2012). Within this class, the model starts by strategically planning the product family in the sub-function $S_{1.1}$, wherein strategic axes are incorporated into product family design (Jiao and Tseng, 1999a). Some issues covered by this sub-function include the mapping of future

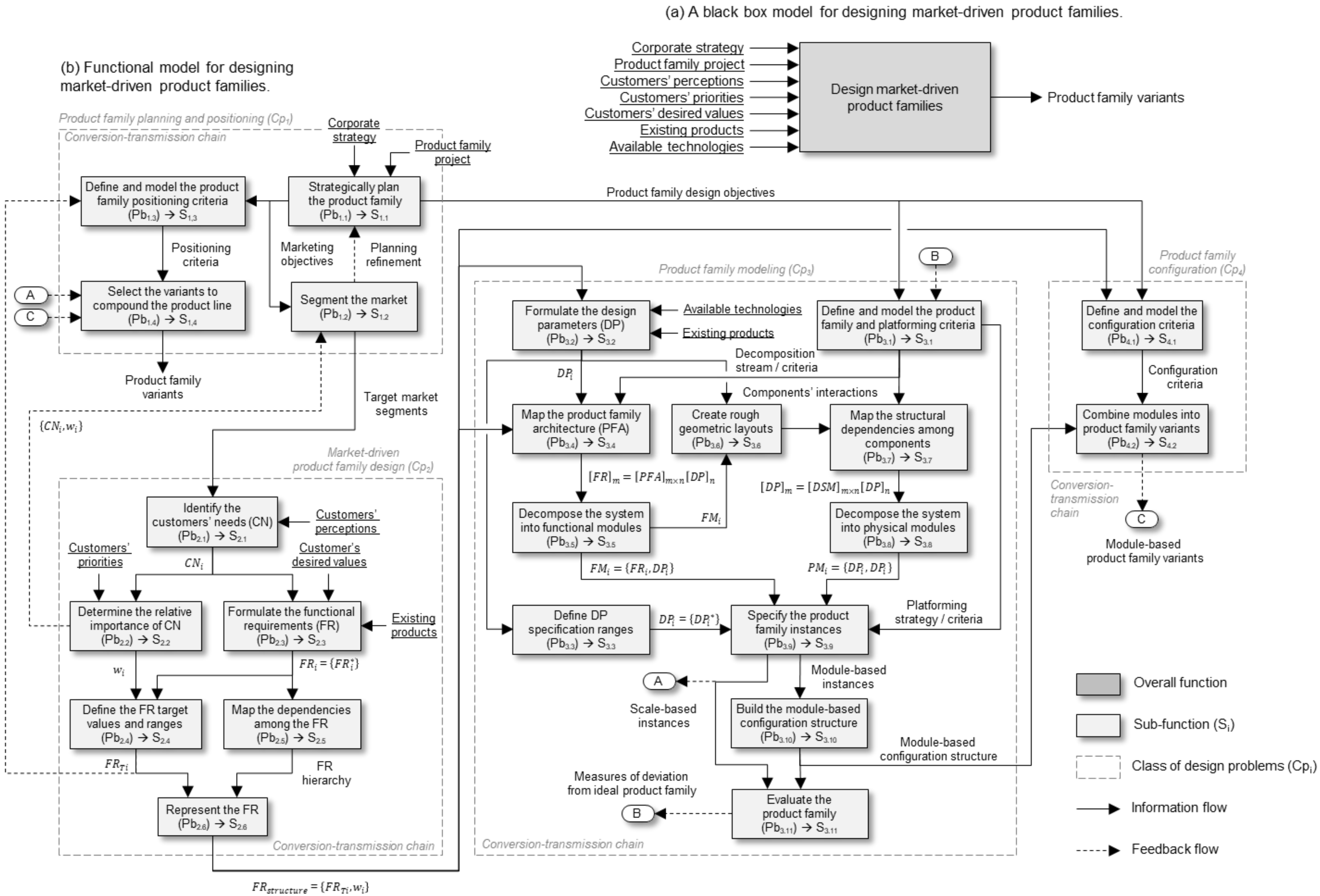


Figure 21. (a) A black box model for designing market-driven product families; (b) Functional model for designing market-driven product families.

product plans (Martin and Ishii, 2002; Hsiao and Liu, 2005), the optimal initial investment in the platform (Zacharias and Yassine, 2008), the platform leveraging strategies (Messac, Martinez and Simpson, 2002; Farrell and Simpson, 2003), the prescription of content and timing for products' upgrades (Wilhelm, Damodaran and Li, 2003), and the appropriateness of product modularity (Asan, Polat and Serdar, 2004). The market-related objectives resulting from this first step serve as an input flow for segmenting the market in 77.8% of the methods addressing the sub-function $S_{1.2}$. The 22.2% remaining, derive the market segments from the customers' needs and its relative importance, as indicates the feedback flow coming from the sub-function $S_{2.2}$. In this step ($S_{1.2}$), the market is decomposed into several segments taking into account the industry type, customer consumption levels, regional characteristics, among other factors (Kazemzadeh *et al.*, 2009). For that reason, the clustering procedures (Zhang, Jiao and Ma, 2007) are usually employed here for characterizing different customer groups (Márkus and Váncza, 1998; Hsiao and Liu, 2005). Besides that, reverse engineering and benchmarking of existing solutions support the identification of promising product plans and platform leveraging strategies (Messac, Martinez and Simpson, 2002; Farrell and Simpson, 2003, 2008). The outputs here are not only target market segments (Tucker and Kim, 2008) but also the planning refinement feedback (Kumar, Chen and Simpson, 2009), as shown in Figure 21(b). Apart from the market-related objectives, the functional requirements target values and ranges compound the input flows for defining and modeling the product family positioning criteria in sub-function $S_{1.3}$. The reasoning here is to provide specific variants for each segment and identify opportunities for adjusting products to attract more customers (Márkus and Váncza, 1998; Krishnan and Zhu, 2006). To that end, the product family positioning is usually modeled as an optimization problem based on customer preferences (Farrell and

Simpson, 2008; Chen, Jiao and Tseng, 2009; Kumar, Chen and Simpson, 2009; Ma and Kim, 2016; Miao *et al.*, 2017), where the objective is to maximize profit, the share of choices, or sales (Jiao, Simpson and Siddique, 2007). Among those methods that tackle the sub-function $S_{1.3}$, the most common criteria used to model such a problem are presented in Table 20.

Table 20. Product family positioning criteria of sub-function $S_{1.3}$.

Criteria	Frequency	Rel. Freq.	Primary study (Id.)
Profit	6	19.4%	R1, R5, R10, R18, R20, R21.
Price	6	19.4%	R1, R5, R10, R18, R20, R21.
Demand	6	19.4%	R5, R10, R17, R18, R20, R21.
Cost	6	19.4%	R1, R5, R10, R18, R20, R21.
Utility	5	16.1%	R1, R13, R18, R20, R21.
Quality	1	3.2%	R10.
Commonality	1	3.2%	R17.
Total	31	100.0%	

After defining and modeling the positioning criteria, the selection of variants to make up each product line is carried out in sub-function $S_{1.4}$. In 83.3% of methods encompassing this sub-function, the selection of product variants is performed through the use of meta-heuristics related to optimization problems (Farrell and Simpson, 2008; Chen, Jiao and Tseng, 2009; Kumar, Chen and Simpson, 2009; Ma and Kim, 2016; Miao *et al.*, 2017), while the 16.7% remaining adopts heuristics (Márkus and Váncza, 1998). Besides that, 57.1% of the variants compounding the product lines derived from the scale-based approach (Farrell and Simpson, 2008; Kumar, Chen and Simpson, 2009; Ma and Kim, 2016), and 42.9% came from the module-based approach (Márkus and Váncza, 1998; Chen, Jiao and Tseng, 2009; Miao *et al.*, 2017). As a result, the expected output of this stage is the minimum possible number of variants that cover the maximum possible customer preferences for a specific segment (Simpson *et al.*, 2014).

The second class of design problems found was the market-driven product family design, which handles the transition of customer needs (CN) to functional requirements (FR) (Simpson *et al.*, 2014). The first step here, in sub-function $S_{2.1}$, is to identify CN

by deriving meaning through interpretations of customers' perceptions about the existing products (Asan, Polat and Serdar, 2004). This process is usually assisted by qualitative and quantitative techniques on data collection, analysis, and synthesis (Du, Jiao and Tseng, 2006; Tucker and Kim, 2008; Kazemzadeh *et al.*, 2009). Coupled with that, arises the necessity to determine the relative importance (w) of each need (Asan, Polat and Serdar, 2004; Hsiao and Liu, 2005). For that reason, the sub-function $S_{2.2}$ explores the priority level of each desired attribute in a product and determines which is the most influent on customer decision making (Du, Jiao and Tseng, 2006; Kazemzadeh *et al.*, 2009). This market-related information must be then translated into engineering specifications; in other words, it should be converted into FR (Hsiao and Liu, 2005). This task of formulating the FRs from the CNs is performed by sub-function $S_{2.3}$, that uses two strategies for that end (Asan, Polat and Serdar, 2004). The first is the inductive functional modelling that describes a product in terms of elementary functions required to achieve its overall function or purpose (Zhang, Tor and Britton, 2006; Zacharias and Yassine, 2008). The second is deductively elicit the FRs and then map them with CNs through incidence matrices (Martin and Ishii, 2002; Kazemzadeh *et al.*, 2009). In both cases, the FRs might derive not only from CNs but also from existing product offerings (Du, Jiao and Tseng, 2006). While FRs are generic to all members within the same customer group, many functional requirement instances (FR^{*}) could result from different desired values for a particular FR, i.e. $FR_i = \{FR_i^{*}\}$ (Martin and Ishii, 2002; Krishnapillai and Zeid, 2006; Ma and Kim, 2016). In this sense, the sub-function $S_{2.4}$ arranges similar FRs^{*} into clusters (FR_T) that are characterized by a target value (TV) and a variation range (VR), i.e. $FR_{Ti} = [TV_i, VR_i]$ (Messac, Martinez and Simpson, 2002; Zacharias and Yassine, 2008). Although this procedure is usually performed within a market segment already defined (Farrell and Simpson, 2003), some authors advocate its adoption to identify the market segments in sub-function $S_{1.2}$ (Zhang, Jiao and Ma, 2007; Kazemzadeh *et al.*, 2009). Another critical factor that affects the level of granularity of product architecture is FR hierarchy (Moon, Park and Simpson, 2014). This issue is tackled by the sub-function $S_{2.5}$ through the mapping of dependencies among the FR (Asan, Polat and Serdar, 2004; Zhang, Tor and Britton, 2006; Zacharias and Yassine, 2008; Kazemzadeh *et al.*, 2009). Finally, through the sub-function $S_{2.6}$, the functional view of a product family is performed from an abstract level to individual instances (Asan, Polat and Serdar, 2004).

The third class of design problems identified was the product family modeling, which comprehends the definition of product family instances in terms of design parameters (DP) and FRs (Jiao, Simpson and Siddique, 2007; Simpson *et al.*, 2014). In this class, the model starts by defining and modeling the product family and platforming criteria in sub-function $S_{3.1}$. From the design objectives coming from the sub-function $S_{1.1}$, decisions on what design approach to follow (Messac, Martinez and Simpson, 2002; Miao *et al.*, 2017), what criteria to use for specifying the instances (Farrell and Simpson, 2003; Du, Jiao and Tseng, 2006; Kumar, Chen and Simpson, 2009), and what strategy to adopt for identifying the platforms are made at this stage (Martin and Ishii, 2002; Krishnapillai and Zeid, 2006). In terms of product family design, there are two prevalent approaches, the module-based and scale-based product family design (Messac, Martinez and Simpson, 2002). In the module-based approach, the product family members are instantiated by mixing and matching modules from the platform (Ulrich, 1995). In the scale-based approach, one or more scaling variables are used to change the platform specifications (Simpson, 2004). Within the former, two decomposition streams arise, the functional and physical decomposition. The functional decomposition derives from the mapping relationships between two domains, i.e., $FR \rightarrow DP$ (Suh, 2001), while the physical decomposition comes from the mapping relationships within the same domain, i.e., $DP \rightarrow DP$ (Krishnapillai and Zeid, 2006). Independently of what design approach is used, there are common criteria employed for specifying the instances or even identifying the platforms. These criteria, along with its frequencies, are presented in Table 21.

Table 21. Product family and platforming criteria of sub-function $S_{3.1}$.

Criteria	Frequency	Rel. Freq.	Primary study (Id.)
Utility	4	36.4%	R3, R4, R8, R18.
Cost	3	27.3%	R8, R18, R21.
Interaction or coupling	2	18.2%	R2, R9.
Redesign effort	1	9.1%	R2.
Variety	1	9.1%	R9.
Total	11	100.0%	

Regarding the strategies adopted for identifying platforms, there are also two. One that considers the variability of FRs^* (Krishnapillai and Zeid, 2006), and the other that takes into account the level of redesign effort taken across generations (Martin and Ishii, 2002). In both, those instances related to the low variability of FRs^* or low redesign effort across generations are defined as platforms. Given the FR previously defined, the process continues by formulating the DPs in sub-function $S_{3.2}$ (Jiao and Tseng, 1999a). The DP consists of the physical effect with the ability to fulfill one or more FRs (Gauss, Lacerda and Sellitto, 2019), and its formulation is usually based on the available technologies and the existing products (Messac, Martinez and Simpson, 2002; Farrell and Simpson, 2003; Du, Jiao and Tseng, 2006). In some cases, there might be necessary a DP to assume different specification values (DP^*) to accomplish its correspondent FR_T, i.e. $DP_i = \{DP_i^*\}$. This task of defining the DP_i^* is executed in sub-function $S_{3.3}$ (Messac, Martinez and Simpson, 2002; Ma and Kim, 2016). The DPs along with the FRs configure the two constituents of the PFA (Jiao and Tseng, 1999a), that in context of module-based product family design, can be defined as decoupled interfaces and the "one-to-one" mapping between FRs and DPs (Ulrich, 1995). This mapping process is performed in the sub-function $S_{3.4}$, and is usually depicted by a design matrix, i.e. $[FR]_m = [PFA]_{m \times n}[DP]_n$, where the nonblank entry $a_{ij} \in [PFA]_{m \times n}$ indicates a relationship between FR_i and DP_i (Hsiao and Liu, 2005; Krishnapillai and Zeid, 2006; Zhang, Tor and Britton, 2006). Then, in the sub-function $S_{3.5}$, the PFA is decomposed into different functional modules (FM), i.e. $, FM_i = \{FR_i, DP_i\}$ (Wilhelm, Damodaran and Li, 2003; Zhang, Tor and Britton, 2006; Zacharias and Yassine, 2008). The FMs must physically match the working structure (Pahl *et al.*, 2007). In this sense, the rough geometric layout is elaborated in sub-function $S_{3.6}$ to identify the interactions among physical components (Asan, Polat and Serdar, 2004). Without this, it would be difficult to determine how subsystems, subassemblies, or parts are coupled (Martin and Ishii, 2002). In such cases where the functional decomposition is bypassed, the DPs serve as an input flow for $S_{3.6}$; otherwise, the FMs must be considered. The sub-function $S_{3.7}$ models these structural dependencies among components (Martin and Ishii, 2002), and similarly to the sub-function $S_{3.4}$, this process is usually depicted by a design structure matrix (DSM), i.e. $[DP]_m = [DSM]_{m \times n}[DP]_n$. However, the nonblank entry $a_{ij} \in [DSM]_{m \times n}$ indicated here, represent a relationship between DP_i and DP_i (Hsiao and Liu, 2005; Krishnapillai and Zeid, 2006). Later, in the sub-function $S_{3.8}$, the DSM is decomposed into physical modules (PM), i.e. $PM_i = \{DP_i, DP_i\}$ (Asan, Polat and Serdar, 2004; Hsiao and Liu, 2005; Krishnapillai and Zeid, 2006). The aim here is to achieve a structure where units are highly interconnected in themselves, but largely independent of other units (Asan, Polat and Serdar, 2004). In 60.0% of methods addressing the sub-functions $S_{3.5}$ and $S_{3.8}$, the functional and physical decomposition are performed by heuristics (Asan, Polat and Serdar, 2004; Zhang, Tor and Britton, 2006; Zacharias and Yassine, 2008). The 40.0% remaining do not explicit the technique used (Hsiao and Liu, 2005; Krishnapillai and Zeid, 2006). In all cases, both decomposition streams are executed independently of one another. Based on the input flows coming from the sub-functions $S_{3.1}$, $S_{3.3}$, $S_{3.5}$ and $S_{3.8}$, the product family instances are specified in the sub-function $S_{3.9}$. Depending on the design approach used, scaled-based or module-based instances might arise (Messac, Martinez and Simpson, 2002). When the scale-based approach is employed, the sub-functions from $S_{3.4}$ to $S_{3.8}$ are by-passed, and the resulting instances come to integrate the searching space of sub-function $S_{1.4}$ (Ma and Kim, 2016). In such cases, the specification of variants is performed by meta-heuristics related to parametric optimization problems in 75.0% of methods (Messac, Martinez and Simpson, 2002; Farrell and Simpson, 2003; Kumar, Chen and Simpson, 2009). The module-based instances derived from the sub-functions $S_{3.5}$ and $S_{3.8}$, in turn, go straight to the sub-function $S_{3.10}$, where the structure for end product configuration is established (Du, Jiao and Tseng, 2006; Miao *et al.*, 2017). The output of this stage is the module-based configuration structure; wherein all modules are supposed to add value to the product (Hsiao and Liu, 2005). Otherwise, they should not be included in the family (Baldwin and Clark, 2000). In this sense, the sub-function $S_{3.11}$ evaluates the product family as a whole (modular or scalable) and generates measures of deviation from the ideal that will serve as feedback improvements (Asan, Polat and Serdar, 2004; Kazemzadeh *et al.*, 2009). The criteria used for assessing the product families at this stage are presented in Table 22.

Table 22. Product family evaluation criteria of sub-function $S_{3.11}$.

Criteria	Frequency	Rel. Freq.	Primary study (Id.)
Modularity	1	20.0%	R6.
Commonality	1	20.0%	R16.
Utility	1	20.0%	R16.
Interaction or coupling	1	20.0%	R6.
Cost	1	20.0%	R16.
Total	5	100.0%	

The last class of design problems found was the product family configuration, which deals with structural configuration problem, wherein the modules compounding the variants are optimally selected (Simpson et al., 2014). In this class, the model starts by defining and modeling the configuration criteria in sub-function $S_{4.1}$ (Tucker and Kim, 2008). At this stage, decisions on what criteria to use for modeling the combinatorial problem are made (Yifei et al., 2015), and the most common criteria identified are presented in Table 23.

Table 23. Configuration criteria of sub-function $S_{4.1}$.

Criteria	Frequency	Rel. Freq.	Primary study (Id.)
Profit	1	25.0%	R14.
Price	1	25.0%	R14.
Cost	1	25.0%	R19.
Utility	1	25.0%	R19.
Total	4	100.0%	

The configuration structure and configuration criteria serve as input flows for combining the modules into product family variants at sub-function $S_{4.2}$. This process of combining modules is assisted by meta-heuristics in 100.0% of the methods analyzed (Tucker and Kim, 2008; Yifei *et al.*, 2015), and similar to the scale-based instances, the module-based variants resulting from this process, come to integrate the searching space of sub-function $S_{1.4}$.

4.3.2 Structured Classes of Problems

With the functional model defined, the next issue was to check the quality of the classes of design problems defined during the encoding process. To that end, the interactions among the classes of design problems were depicted in a design structure matrix to calculate the MI, as shown in Figure 22.

	$S_{1.1}$	$S_{1.2}$	$S_{1.3}$	$S_{1.4}$	$S_{2.1}$	$S_{2.2}$	$S_{2.3}$	$S_{2.4}$	$S_{2.5}$	$S_{2.6}$	$S_{3.1}$	$S_{3.2}$	$S_{3.3}$	$S_{3.4}$	$S_{3.5}$	$S_{3.6}$	$S_{3.7}$	$S_{3.8}$	$S_{3.9}$	$S_{3.10}$	$S_{3.11}$	$S_{4.1}$	$S_{4.2}$
$S_{1.1}$	x																						
$S_{1.2}$	1	x				1																	
$S_{1.3}$	1		x					1															
$S_{1.4}$			1	x															1				1
$S_{2.1}$		1			x																		
$S_{2.2}$					1	x																	
$S_{2.3}$					1		x																
$S_{2.4}$						1	1	x															
$S_{2.5}$							1		x														
$S_{2.6}$								1	1	x													
$S_{3.1}$	1										x										1		
$S_{3.2}$										1		x											
$S_{3.3}$												1	x										
$S_{3.4}$										1	1	1		x									
$S_{3.5}$														1	x								
$S_{3.6}$												1			1	x							
$S_{3.7}$											1					1	x						
$S_{3.8}$																	1	x					
$S_{3.9}$											1		1		1			1	x				
$S_{3.10}$																			1	x			
$S_{3.11}$																				1	x		
$S_{4.1}$	1									1												x	
$S_{4.2}$																				1		1	x

Figure 22. Interactions among the classes of design problems.

The MI assesses the quality of a clustering solution, ranging from 0 to 1, by capturing the degrees of connection strengths within each independent class and between different classes, the density of connections within each class and between classes, and the proximity of interactions to the diagonal of the design matrix (Jung and Simpson, 2017). With the weighting factors set to $(w_1, w_2, w_3) = \left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right)$, the MI was calculated and the quality of the clustering solution was found to be equals to 0.69. With MI value considered satisfactory, the structured classes of problems were then formalized, as shown in Appendix B (Table B3). From this table, it was possible to derive that 59.1% of methods have been evaluated in depth by observational techniques such as case studies. In 36.4% of the studies, the methods have been assessed descriptively by constructing detailed scenarios to demonstrate the artifacts' utility, and in 4.6% of situations, they have been tested experimentally using simulation. The sum of frequencies is higher than 100.0% because of the method M_5 has been evaluated by more than one approach. Regarding the type of products these methods have been developed for, 42.9% of them have been used to design consumers' durables, 42.9% to intermediate goods, and 14.2% to capital goods. Still, from Table B3, it is possible to derive which technique, among those 56 identified here, is intended to solve each design problem. The definition/function of each method and technique is presented in Appendix B (Table B2).

Later, Table B3 was converted into a binary matrix represented by Table B4, wherein only primary studies (R_i), design problems (Pb_i), and classes of design problems (Cp) were considered. In Table B4, the primary studies were rearranged to facilitate the understanding of the association rules. Using the Apriori algorithm in software R, with the parameters set to $s(X \rightarrow Y) \geq 0{,}1$, $c(X \rightarrow Y) \geq 0{,}1$, and the relationship width ≥ 2, 9 association rules were identified, as shown in Figure 23.

```
    lhs           rhs    support   confidence lift      count
[1] {CP2}      => {CP3} 0.5238095 0.9166667  1.4807692 11
[2] {CP3}      => {CP2} 0.5238095 0.8461538  1.4807692 11
[3] {CP3}      => {CP1} 0.4761905 0.7692308  0.9502262 10
[4] {CP1}      => {CP3} 0.4761905 0.5882353  0.9502262 10
[5] {CP2}      => {CP1} 0.4285714 0.7500000  0.9264706  9
[6] {CP1}      => {CP2} 0.4285714 0.5294118  0.9264706  9
[7] {CP2,CP3}  => {CP1} 0.3809524 0.7272727  0.8983957  8
[8] {CP1,CP2}  => {CP3} 0.3809524 0.8888889  1.4358974  8
[9] {CP1,CP3}  => {CP2} 0.3809524 0.8000000  1.4000000  8
```

Figure 23. Association rules derived from the Apriori algorithm executed in software R.

Rules 1 and 2 indicate that the classes of design problems Cp_2 and Cp_3 co-occur in 52.4% of the primary studies. Still, from these rules, it is possible to infer that Cp_3 is more likely to appear (91.7%) when Cp_2 is present, but the opposite does not happen with the same level of confidence (84.6%). The same reasoning can be applied to the rules 8 and 9 but at lower support levels. Regarding the rules from 3 to 7, the lift values lower than one indicate the classes of design problems at the left-hand side (lhs) are unlikely to co-occur with the ones at the right-hand side (rhs).

In this section, it was presented the results of the systematic literature review, wherein 21 methods, 23 design problems, 23 sub-functions, 56 techniques, 3 evaluation approaches, and 3 types of products, were found. These methods were connected through its corresponding sub-functions in the form of a functional model. While the remaining instances were organized into structured classes of design problems, which had its most common association rules identified. The discussion of the results is presented next.

4.4 Discussion of the Results

The functional model elaborated in this research represents the common underlying structure of the market-driven product family design methods developed over the last 21 years. In this sense, the model not only presents the sub-functions (main steps of the methods) intended to solve the design problems but also indicates the causal relationship (execution order) among them along with its respective input and output flows. The structured classes of design problems, in turn, complements the functional model by cataloging the techniques meant to execute each sub-function of the model. Moreover, it presents the definitions of methods and techniques, how they have been tested, and for what product they have been developed. These entities together organize the knowledge of market-driven product family design as well as produce a new and integrative interpretation of this particular fraction of the field of product family design.

From these entities, it was possible to derive some patterns. In Cp_1, it was noted the market segmentation ($S_{1.2}$) being performed from two different perspectives. The first perspective, adopted in 77.8% of the methods addressing the sub-function $S_{1.2}$, consists of deductively segmenting the market from the objectives coming from the sub-function $S_{1.1}$ (Márkus and Váncza, 1998; Messac, Martinez and Simpson, 2002; Farrell and Simpson, 2003, 2008; Hsiao and Liu, 2005; Tucker and Kim, 2008; Kumar, Chen and Simpson, 2009). The second perspective, employed in 22.2% of methods, inductively derive the market segments from the customers' needs and its relative importance coming from the sub-function $S_{2.2}$ (Zhang, Jiao and Ma, 2007; Kazemzadeh et al., 2009). In this sense, a trade-off between quality and cost in market segmentation arises; where the data collection on customers' preferences, being performed before the segmentation, might generate more precise market segments, and the opposite, might reduce the searching space, resulting in time and costs savings.

Still from Cp_1, the selection of variants to make up each product line is carried out in sub-function $S_{1.4}$, wherein 57.1% of the variants compounding the product lines derived from the scale-based approach (Farrell and Simpson, 2008; Kumar, Chen and Simpson, 2009; Ma and Kim, 2016), and 42.9% came from the module-based approach (Márkus and Váncza, 1998; Chen, Jiao and Tseng, 2009; Miao et al., 2017). Regarding the last approach, it is implicit in the works of Tucker and Tim (Tucker and Kim, 2008) and Yifei et al. (2015), the product family configuration (Cp_4) supporting or even overlapping the role of sub-functions $S_{1.3}$ and $S_{1.4}$ within the product family planning and positioning (Cp_1), two sub-functions not reported by the work of Gauss, Lacerda, and Miguel (2020) (Article 1). This reasoning can also be strengthened by the convergence among the broader business criteria (cost, demand, price, and profit) adopted by 9 methods (42.9%) for selecting or combining the variants in sub-functions $S_{1.4}$ and $S_{4.2}$ (Márkus and Váncza, 1998; Wilhelm, Damodaran and Li, 2003; Krishnan and Zhu, 2006; Tucker and Kim, 2008; Chen, Jiao and Tseng, 2009; Kumar, Chen and Simpson, 2009; Yifei et al., 2015; Ma and Kim, 2016; Miao et al., 2017).

Another thing noted from the association rule 1, was that, in 52.4% of the methods, the Cp_3 has 91.7% of probability to exist when Cp_2 is present. This rule consists of the most influential association found here and indicates a consistent presence of market considerations within the product family modeling. However, looking at Table B4 and considering the principle that if an itemset is frequent, then all of its subsets must also be frequent (Tan *et al.*, 2019), it is possible to observe that in only 14.3% of the methods, the FRs derive from the CNs. In other words, among those 11 methods that consider Cp_2 and Cp_3 concurrently, only 3 methods formulate the FR in sub-function $S_{2.3}$ from the CN identified in sub-function $S_{2.1}$. What indicates that the FRs are formulated by other means, such as reverse engineering and benchmarking of existing solutions (Jiao and Tseng, 1999a; Thevenot and Simpson, 2007; Simpson *et al.*, 2012). Besides that, in Cp_3, two sub-functions, derived from the scale-based approach, might complement the functional model established in the work of Gauss, Lacerda, and Miguel (2020) (Article 1) on module-based product family design, they are: $S_{3.3}$ and $S_{3.9}$. In the first, different specification values of DP are defined to accomplish its correspondent FR_T (Messac, Martinez and Simpson, 2002; Ma and Kim, 2016), while in the second, the parameters of product family instances are specified (Messac, Martinez and Simpson, 2002; Farrell and Simpson, 2003; Kumar, Chen and Simpson, 2009). Both can play an essential role in defining the parameters of primitive modules (instance or scalable), an issue not well addressed in module-based product family design methods.

Finally, although no method covering the four classes of design problems identified in this research was found, rule 8 indicates that 38.1% of the methods consider the three first classes concurrently. These results are consistently higher than those obtained in our previous research on module-based product family design (Gauss, Lacerda and Miguel, 2020), but those 61.9% methods remaining, reinforce other research indications of lacking integrated methods in this field of product family design (Simpson *et al.*, 2012; Otto *et al.*, 2016).

4.5 Conclusions

Customers' needs continually evolve and shift over time, and the result is an increasing demand for product variety and newer versions of products. Believing the product variety can help manufacturing companies to increase sales and profits, many companies have been attempting to provide more product variants without sacrificing production efficiency. One way to manage this trade-off is through the product family design, a field of study wherein marketing, engineering, and economic aspects are often highly interdependent. To understand how the interconnected relationships among these three domains take place into the product family design, this paper presented a systematic literature review, wherein 21 methods (1998-2019) to design market-driven product families have been connected in the form of a functional model and structured classes of design problems. These entities together organize the knowledge of market-driven product family design as well as produce a new and integrative interpretation of this particular fraction of the field of product family design. Besides that, the main contribution of this work includes the identification of four sub-functions ($S_{1.3}$, $S_{1.4}$, $S_{3.3}$, and $S_{3.9}$) not reported by the work of Gauss, Lacerda, and Miguel (2020) (Article 1) regarding the module-based product family design.

The main limitation of this work lies in the fact that some useful literature might have been omitted since no relook at the references of those studies included in the review has been performed. In terms of future work, we reinforce the proposition of Gauss, Lacerda, and Miguel (2020) to synthesize and integrate the extant methods to design product families. However, we suggest doing so by a longitudinal perspective capable of not only identifying the connections among the existing methods but also able to deduct its future patterns.

5 ARTICLE 3 - MARKET-DRIVEN MODULARITY: A DESIGN METHOD DEVELOPED UNDER DESIGN SCIENCE PARADIGM [3]

[3] Article to be submitted to the Journal of Operations Management (JOOM).

Abstract: This paper uses design science research (DSR) to integrate marketing, engineering, and economic aspects into a single approach to conceptually design lucrative product families. In this sense, the traditional stages of DSR methodologies are decomposed into 32 steps to provide practical guidance on the artifacts' design and evaluation. By following these steps, a field problem gives rise to a method, entitled Market-Driven Modularity (MDM), which is validated through a series of practical applications and experts' judgments. The main contributions of this research include: (i) The systematic integration of four classes of design problems prevalent in literature into a single method to conceptually design lucrative product families. (ii) The proposition of an open architecture of techniques to execute each step of the method in contexts from low to high data availability. (iii) The introduction of Functional to Physical Decomposition, an approach to deal with functional and physical modularity in product family architectures. (iv) The presentation of practical guidance on the artifact's design and evaluation. (v) The usage of a quantitative approach to measure the pragmatic validity and practical relevance. Finally, (vi) the MDM itself as the first method to design modular product families, developed under the design science paradigm.

Keywords: design science research; product family design; modularity.

5.1 Introduction

The ever-increasing diversity of customers' needs has been pushing companies to provide more product variants without sacrificing production efficiency (Jiao, Simpson and Siddique, 2007). In industry and academy alike, the negative impact of product variety on operational performance has been traditionally addressed by two complementary approaches: the product line planning and product family design (Miao *et al.*, 2017). The product line planning consists of optimally selecting the group of products to be marketed to one specific market (Kahn, 2012), while the product family design consists of designing a set of products sharing common elements yet target different market segments (Simpson *et al.*, 2014).

Although numerous product line planning methods in management science and marketing literature deal with the selection problem using various objectives derived from profit, few of them explicitly address product design details not directly perceived by customers (Jiao, Simpson and Siddique, 2007). These approaches normally assume that any combination of product attributes can somehow be attained by design engineers post hoc (Michalek *et al.*, 2011). In contrast, most existing product family design approaches are targeted at identifying an optimal commonality decision in order to minimize cost while meeting pre-specified performance tiers (Kumar, Chen and Simpson, 2009). As a result, these engineering approaches do not sufficiently examine broader business indicators such as demand and profit (Michalek *et al.*, 2011).

The problem is the marketing and engineering variables are often highly interdependent in product family design. Moreover, the coupled relationships between them imply that any change in one variable can potentially influence the outputs of the other(s), with both affecting the economic benefits of an enterprise (Chen, Hoyle and Wassenaar, 2013). Therefore, in the design of optimal or near-optimal product families, marketing, engineering, and economic requirements often cannot be pursued separately or even sequentially (Luo, 2011). But how to integrate marketing, engineering, and economic aspects into a single approach to design lucrative product families? Which characteristics this approach should have to present satisfactory results? Is there an already-developed method that accomplishes these characteristics?

Given the conceptual coupling between product family design and design science, this work used design science research (DSR) to answer these questions. In this sense, the traditional stages of DSR methodologies have been decomposed into 32 steps to provide practical guidance on the artifacts' design and evaluation. By following these steps, a field problem gave rise to a method, entitled Market-Driven Modularity (MDM), which has been validated through a series of practical applications and experts' judgments.

The main contributions of this research include: (i) The systematic integration of four classes of design problems prevalent the literature into a single method to conceptually design lucrative product families. (ii) The proposition of an open architecture of techniques to execute each step of the method in contexts from low to high data availability. (iii) The introduction of Functional to Physical Decomposition, an approach to deal with functional and physical modularity in product family architectures. (iv) The presentation of practical guidance on the artifact's design and evaluation. (v) The usage of a quantitative approach to measure the pragmatic validity and practical relevance. Finally, (vi) the MDM itself as the first method to design modular product families, developed under the design science paradigm.

The remainder of this paper is structured as follows. Section 5.2 synthesizes the related studies on product family design and design science research. Section 5.3 contains the methodological procedures used to develop and validate the MDM. Section 5.4 describes the MDM in detail. Section 5.5 presents the results of multiple evaluation cycles as well as the MDM's construction and contingence heuristics. Section 5.6 critically analyses the research findings. Finally, Section 5.7 presents the research contributions and limitations along with its future directions.

5.2 Literature Review

The product family design is an effective strategy to provide variety at reduced costs (Simpson *et al.*, 2014). Generally speaking, a product family refers to a set of products derived from a standard product platform to satisfy various market applications (Meyer and Lehnerd, 1997). Platforms, in turn, are intellectual and material assets shared across a family of products, to minimize manufacturing complexity (Erens and Verhulst, 1997). In this context, the prominent approach to product family design is through the development of module-based product families, wherein product family members are instantiated by mixing and matching functional modules from the platform (Ulrich, 1995; Du, Jiao and Tseng, 2001). An alternative approach, considered as a subset of the former (Fujita and Yoshida, 2004), is through the development of a scale-based product family, which consists of scaling one or more variables to change the platform specifications while common parameters remain constant (Simpson, 2004).

The product family design is challenging for many aspects. It involves selecting business strategies, considering multiple marketing issues, engineering customer needs, studying customer behavior and choice-related issues, as well as carefully considering engineering aspects of design, such as manufacturability, technological aspects, and design support issues (Simpson *et al.*, 2014). In general, these problems can be grouped into four prevalent classes: (i) Product family positioning, which aims at maximizing customers' preferences with the lowest number of variants. (ii) Market-driven product family design, that deals with the transition of customers' needs to functional requirements. (iii) Product family modeling, which comprehends the definition of modules and platforms. Finally, (iv) product family configuration, wherein the modules compounding the variants are optimally selected (Jiao, Simpson and Siddique, 2007).

Over the years, active work in developing methods to design product families has been done (Borjesson and Hoelttae-Otto, 2014; Otto *et al.*, 2016). Among those methods related to this study, the one encompassing four classes of design problems is the work of Jiang and Allada (2005). However, this method assumes the modules' set already exists, being deeply sensitive to the ability of extant modules in accomplishing the customer desired attributes. Besides that, the product family configuration is used to configure one variant at a time instead of building an optimal or near-optimal product family structure. In like manner, other methods only entail the three first classes of design problems (Jiao and Tseng, 1999a; Asan, Polat and Serdar, 2004; Hsiao and Liu, 2005; Kazemzadeh *et al.*, 2009; Hsiao *et al.*, 2013; Sahin-Sariisik *et al.*, 2014; Ma and Kim, 2016; Pakkanen, Juuti and Lehtonen, 2016). But the main limitation of them lies in the inability to optimally or near-optimally combine the designed modules into product family variants or even selecting the most adequate ones to compose the product family structure.

There is another group of methods, encompassing the product family modeling, which focuses on modules identification (Thevenot *et al.*, 2007; Arciniegas and Kim, 2011; Agard and Bassetto, 2013; AlGeddawy and ElMaraghy, 2013; Li *et al.*, 2013; Borjesson and Hoelttae-Otto, 2014; Aydin and Ulutas, 2016; Ma *et al.*, 2016; Hou *et al.*, 2017, 2018; Miao *et al.*, 2017). Within this group, a few methods, if any, perform the functional and physical decomposition concurrently. Besides that, these approaches occasionally measure the quality of the clustering solution, indicating in this way its open-loop nature. Still from this group, some approaches combine the product family positioning with product family modeling (ElMaraghy and AlGeddawy, 2012; Simpson *et al.*, 2012; Fan *et al.*, 2015; Miao *et al.*, 2017), while others combine the market-driven product family reasoning with the product family modeling (Dahmus, Gonzalez-Zugasti and Otto, 2001; Zhang, Tor and Britton, 2006; Du, Jiao and Tseng, 2006; Krishnapillai and Zeid, 2006; Meng, Jiang and Huang, 2007; Park *et al.*, 2008; Stone *et al.*, 2008; Bonjour *et al.*, 2009; Yan and Stewart, 2010; Emmatty and Sarmah, 2012; Yang, Yu and Jiang, 2014; Wei *et al.*, 2015; Jung and Simpson, 2016; Cheng *et al.*, 2017; Bejlegaard *et al.*, 2018; Wang *et al.*, 2018; Gauss, Lacerda and Sellitto, 2019). In both, less than a quarter, derive the customer desired attributes straight from themselves.

The last group of methods focuses on the product family configuration. More specifically in the process of mixing, matching, and scaling modules to generate product family variants (Tucker and Kim, 2008; Jiao, 2012; Pate, Patterson and German, 2012; Hanafy and Elmaraghy, 2015; Goswami, Daultani and Tiwari, 2017; Xiao *et al.*, 2018). In this group, the major part, solve the combinatorial and parametric problem through meta-heuristics and some use enterprise-level indicators to compound the objective function. Some methods also consider the product family design and configuration being performed together (Rai and Allada, 2003; Li, Huang and Newman, 2008; Li and Huang, 2009; Dong, Shao and Xiong, 2011; Chowdhury *et al.*, 2016; Baylis, Zhang and McAdams, 2018; Colombo *et al.*, 2019). However, they assume the modules' set already exists, and use the configuration process to generate product family variants instead of building product family structures. Additionally, nor a threshold to evaluate if the variants instantiated satisfy the desired attributes in a product, neither feedbacks leading to new modules' developments are found. Moreover, it is not explicit in these works, the product family configuration supporting or even playing the role of product line planning, an issue that has been traditionally dealt with in the management science and marketing literature (Jiao, Simpson and Siddique, 2007).

The product family design, as well as other engineering disciplines, is typically concerned with construction problems related to not yet existing entities (van Aken and

Romme, 2009; Vaishnavi, Kuechler and Petter, 2017). This conception complies with the goals of research performed under the design science paradigm, which seeks to produce knowledge to solve real problems or to design something that does not yet exist (Simon, 1996; van Aken, 2005). While design science is the epistemological basis, design science research (DSR) is the method that operationalizes research in this context (Lacerda *et al.*, 2013). DSR, unlike other research methods, produces knowledge in the form of prescription or design, wherein prescription supports the problem-solving and design aids the artifact's development (Dresch, Lacerda and Antunes Jr, 2015). Although there exists a conceptual coupling between product family design and design science, to the best of our knowledge, besides the works of Koh, Caldwell, and Clarkson (2013) and Andre and Elgh (2018), no other study has been conducted by DSR in this field. However, these studies lack practical guidance on artifact's design and evaluation. Issues not derived from the research on product family design, or any other field, but from the DSR methodologies that only approach the research conduction from higher abstraction levels.

Different methods for conducting research based on design science exist in the literature (Bunge, 1980; Nunamaker, Chen and Purdin, 1990; Takeda *et al.*, 1990; Eekels and Roozenburg, 1991; Walls, Wyidmeyer and Sawy, 1992; Cole, 2005; Gregor and Jones, 2007; Peffers *et al.*, 2007; Baskerville, Pries-Heje and Venable, 2009; Alturki, Gable and Bandara, 2011; van Aken, Berends and van der Bij, 2012; Dresch, Lacerda and Antunes Jr, 2015). Despite the differences in methods' steps, they share the same outcome, which is the well-tested, well-understood, and well-documented innovative generic design that has been field-tested to establish pragmatic validity (van Aken, Chandrasekaran and Halman, 2016). According to Kvale and Brinkman in Van Burg (2011), the pragmatic validity has to do with "the extent to which the research creates guidelines that generate the desired outcomes when those guidelines are applied". However, the extant DSR literature does not provide sufficient instruction on the artifact's design (Gacenga *et al.*, 2012). Moreover, there is little or no guidance on how or why one can or should choose among the different paradigms or methods to achieve DSR evaluation goals (Venable, Pries-Heje and Baskerville, 2016; Coetzee, 2019; Gassel, Reymen and Maas, 2019).

In summary, from a product family design perspective, there is a lack of integrated approaches modeling the customers' preferences and using it to design and configure gainful product family structures. From a design science research perspective, there is a lack of practical guidance on how to design and validate artifacts. This paper aims at bridging these gaps by developing an integrated method to conceptually design modular product families that balance the fulfillment of market needs and the resulting profitability to achieve them while providing practical guidance on the artifact's design and evaluation.

5.3 Research Design

This study followed the DSR methodology proposed by Dresch, Lacerda, and Antunes (2015). The 12 stages originally conceived by them were decomposed into 32 steps for better guiding the process of artifact's design and evaluation, as shown in Figure 24. In this context, the process started by identifying the missing link between marketing and engineering domains into product family design, as the research problem, in step 1.1. An issue that emerged from our experience in designing modular product families oriented to cost-reduction.

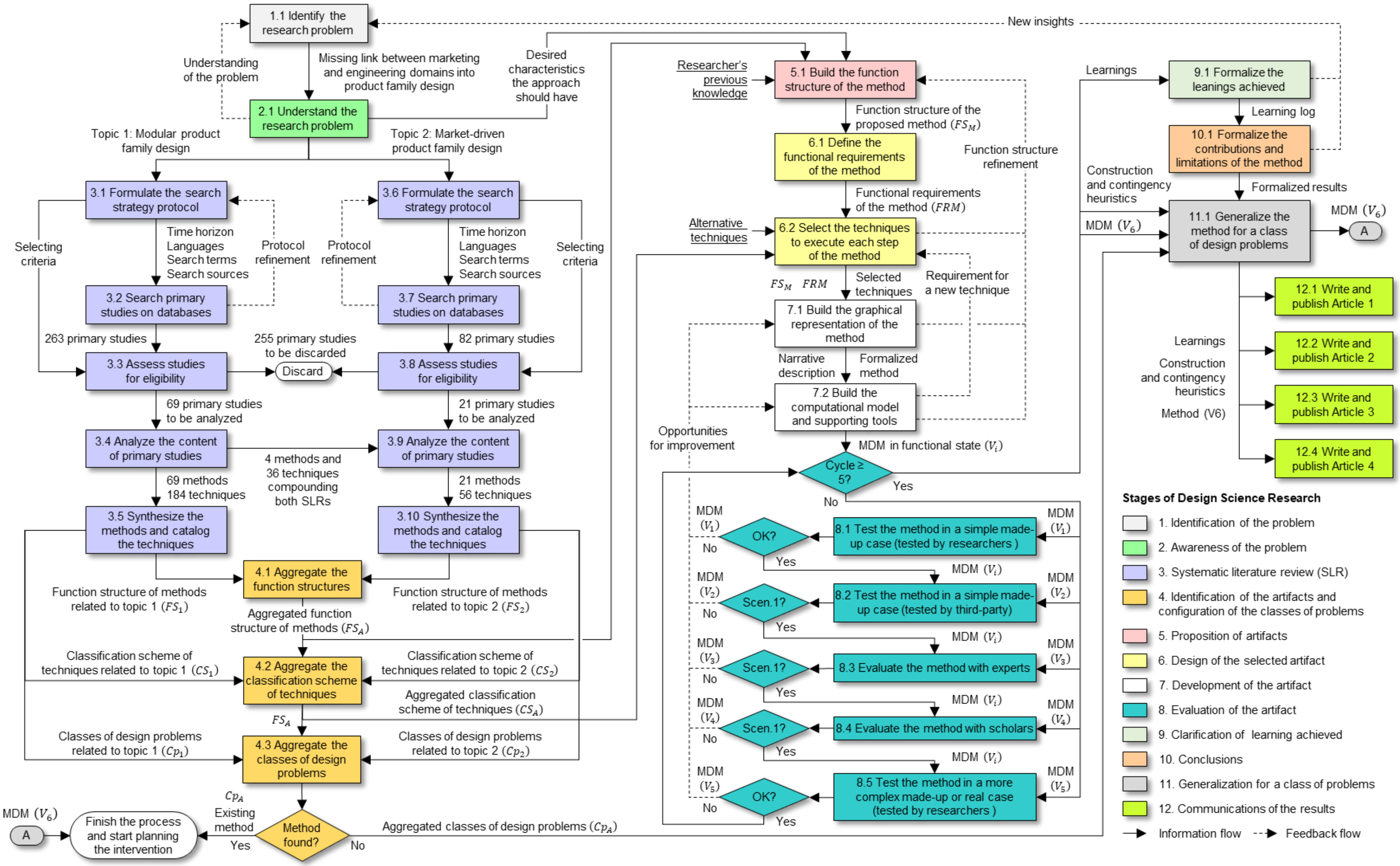


Figure 24. Research strategy.

In step 2.1, the systemic structure presented in Figure 25 was built to help us understand the problem (Senge, 1990). It made clear the coupling relationships between marketing and engineering variables into product family design, and how they affect broader business indicators such as demand, price, cost, and profit (Kumar, Chen and Simpson, 2009; Luo, 2011; Michalek *et al.*, 2011; Chen, Hoyle and Wassenaar, 2013).

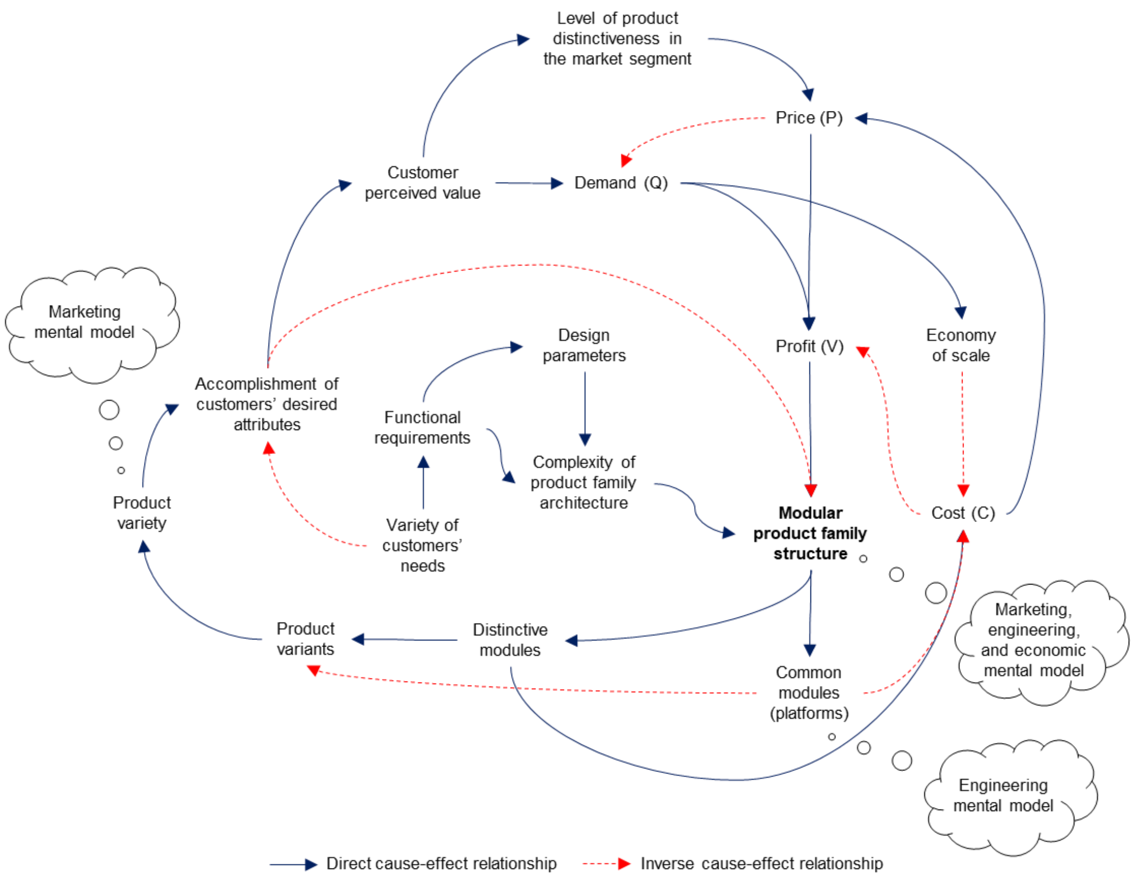


Figure 25. Systemic structure.

From an engineering mental model, we were using modularity as a mechanism to reduce the complexity of product family structure and consequently costs. However, this reinforcing feedback of increasing commonality, to reduce cost and increase profit, is balanced by another feedback leading to product variety, which is typically linked to a marketing mental model. In this system, the modular product family structure was identified as the high-leverage point, in other words, the variable that a small change in it might lead to significant system improvement (Senge, 1990). However, it would only be possible to achieve substantial changes if the mental model behind the building of the product family structure considered marketing and engineering variables concurrently. Moreover, if the goal of an enterprise is to make money now and in the future (Goldratt and Cox, 1984), this mental model should also incorporate an economic perspective. Therefore, we started seeing modularity from marketing, engineering, and economic perspectives together, wherein the product variety and cost are moderated by profit.

From this broader perspective, the following questions arose: (i) How to integrate marketing, engineering, and economic aspects into a single approach to design lucrative product families? (ii) Which characteristics this approach should have to present satisfactory results? (iii) Is there an already-developed method that accomplishes these characteristics?

Based on previous knowledge and supported by literature, we started by enumerating the main characteristics the approach should have:

- It should model the customers' preferences and use it to design and configure a product family structure that balances the fulfillment of market needs and the resulting profitability to achieve them (Chen, Hoyle and Wassenaar, 2013);
- It should entail the initial phases of the product development process since the cost of change grows throughout the product life-cycle, and the decisions made at the early design stages account for more than 80% of the product's committed costs (Rozenfeld et al., 2006; Mascle and Zhao, 2008; Charter and Tischner, 2017; Xiao et al., 2018);
- It should be used to design different types of products, such as consumer durables, intermediate, and capital goods;
- It should handle the design process in contexts of low and high data availability.

Then, two topics were defined for searching the existent methods. The first was the “modular product family design” since modularity is the core of many product family design approaches (Simpson, Siddique and Jiao, 2006; Kong *et al.*, 2009; Otto *et al.*, 2016). The second topic, in turn, derived from the following reasoning: If modular approaches do not tackle the problem, which another do? Therefore, the “market-driven product family design” was stated.

Later, one systematic literature review (SLR) was performed for each topic (Morandi and Camargo, 2015). In this stage, the process started by formulating the search strategy protocols presented in Table A1 and Table B1. The protocols gave rise to both search strings of Figure 26, from which 364 records were reduced to 89 studies after the eligibility assessment. The excluding criteria leading to this data reduction are shown in Table 24. Then the content of each selected study was analyzed in-depth (Bardin, 1993). As a result, 89 methods (M), 29 design problems (Pb), 29 sub-functions (S), 211 techniques (T), 3 evaluation approaches (Et), 4 types of products (Pt), and 4 classes of design problems (Cp) were identified.

These findings were synthesized in steps 3.5 and 3.10, and then aggregated in the form of function structure of the methods (FS_A) (Stone and Wood, 2000), classification scheme of techniques (CS_{A}) (Pahl *et al.*, 2007), and classes of design problems (Cp_{A}) (Dresch, Lacerda and Antunes Jr, 2015) in stage 4. A process that followed the heuristic proposed by Gauss, Lacerda, and Miguel (2020) (Article 1). At the end of this exploratory phase, no method accomplishing all desired characteristics was found. Therefore, in step 5.1, a new proposition was done according to shows Figure 27. Wherein the sub-functions filled in yellow derived from the function structure of methods related to topic 1 (FS_1), the sub-functions filled in blue derived from the function structure of methods related to topic 2 (FS_2), and the sub-functions filled in

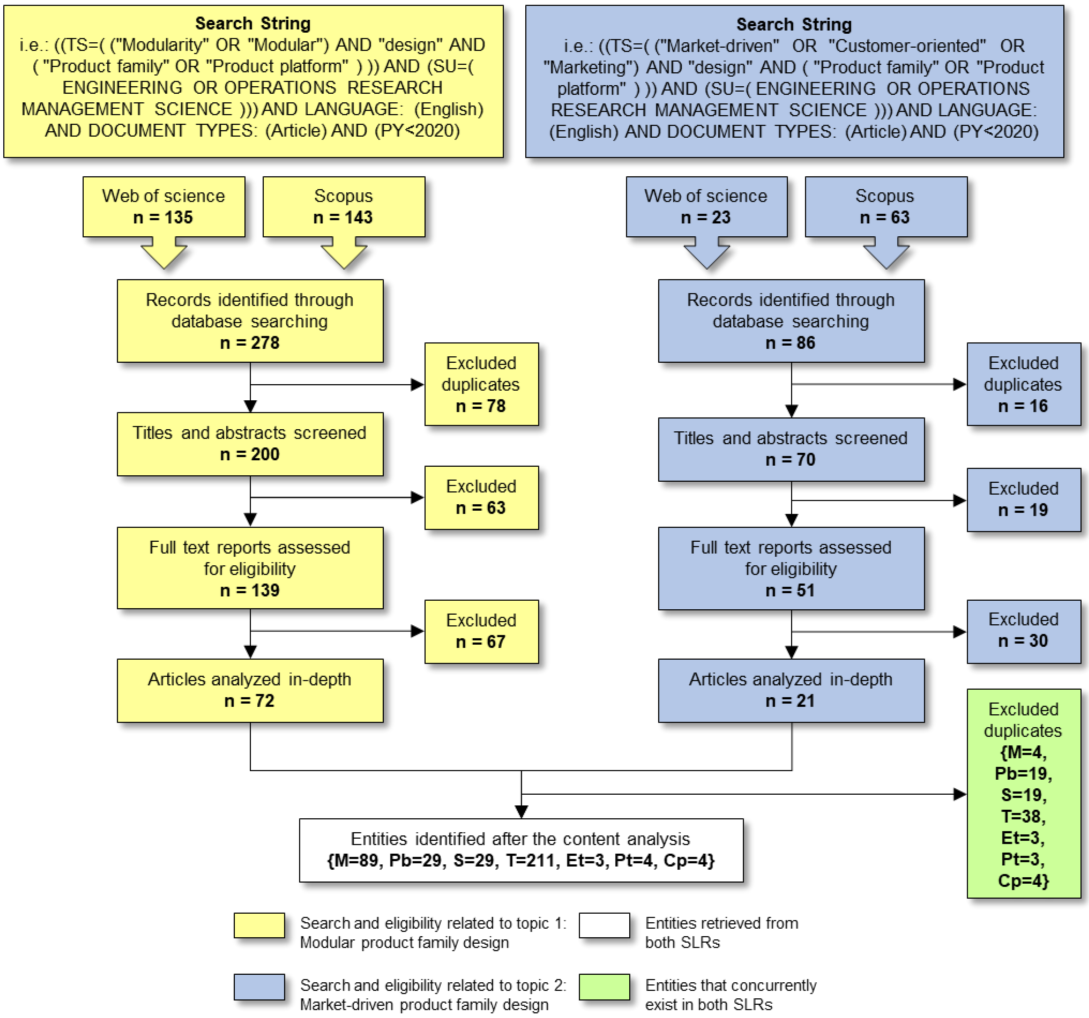


Figure 26. Results of search and eligibility for both *SLR*s.

Table 24. Excluding statistics of both *SLR*s.

Exclusions	Percentage	Excluding criteria
98	35.6%	Duplicated studies
35	12.7%	Absence of methods or techniques addressing modularity in design
23	8.4%	Absence of methods or techniques addressing marketing issues into product family design
23	8.4%	Manufacturing and production for product families
20	7.3%	Design support systems
18	6.5%	Supply chain issues of product families
14	5.1%	Literature review on product family design and modularity
9	3.3%	Theoretical development and synthesis on product family design and modularity
8	2.9%	Fundamental issues on product family design and modularity
6	2.2%	Paper not found
5	1.8%	Very specific application not liable to generalization
3	1.1%	Service design
3	1.1%	Limited applicability to scale-based product family design
3	1.1%	Out of context
2	0.7%	Customer co-design
2	0.7%	Software development
2	0.7%	Civil construction
1	0.4%	Aesthetics in product design
275	100,0%	Total

green derived from both, FS_1 and FS_2. The sub-functions filled in grey, in turn, were abductively added based on the desired characteristics that emerged during the awareness of the problem.

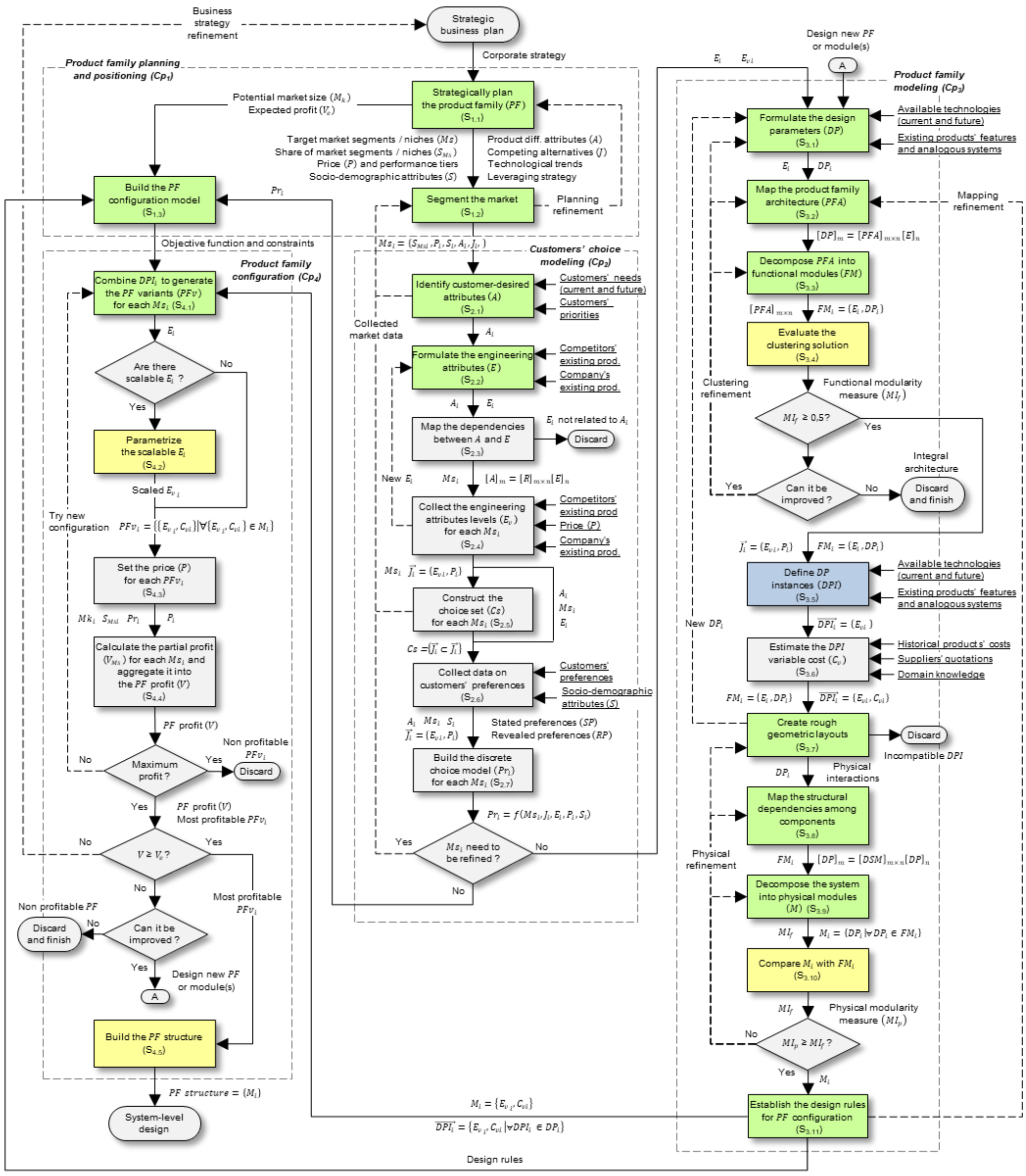


Figure 27. Function structure of the proposed method (FS_M), version 6.

With the function structure of the proposed method (FS_M) defined, the next issue was to select the most suitable techniques to execute its corresponding sub-functions. This process took place at stage 6 and started by formulating the list of functional requirements (FRM) that each sub-function of the method should fulfill to contribute to the problem-solving. Table 25 presents the FRM defined at step 6.1.

Table 25. Functional requirements of the method (FRM), version 6.

Classes of design problems (Cp_i)	Sub-functions (S_i)	Inputs	Outputs	Functional Requirements	Techniques
Cp_1 - Product family planning and positioning	$S_{1.1}$ - Strategically plan the product family (PF)	▪ Corporate strategy ▪ Planning refinement	▪ Potential market size (M_k) ▪ Expected profit (V_e) ▪ Potential market segments/niches (Ms) ▪ Share of market segments/niches (S_{Ms}) ▪ Price (P) and performance tiers ▪ Socio-demographic attributes (S) ▪ Product diff. attributes (A) ▪ Competing alternatives (J) ▪ Technological trends ▪ Leveraging strategy	▪ Estimate the potential market size (M_k) and the expected profit (V_e) ▪ Define the product family positioning and its leveraging strategy ▪ Define the technological trends for product family development ▪ Define which type of project must be developed	▪ Delphi (Dalkey, 1969) ▪ Three-point estimate (Premachandra, 2001) ▪ Domain knowledge (Jiao and Tseng, 1999a) ▪ Survey (Forza, 2002) ▪ Descriptive statistics (Montgomery and Runger, 2011) ▪ Market segmentation grid (Meyer and Lehnerd, 1997) ▪ Technology roadmap (Phaal and Muller, 2009) ▪ Aggregate project plan (Wheelwright and Clark, 1992)
	$S_{1.2}$ - Segment the market	▪ Potential mkt. seg. / niches (Ms) ▪ Share of mkt. seg. / niches (S_{Ms}) ▪ Price (P) and performance tiers ▪ Socio-demographic attributes (S) ▪ Product diff. attributes (A) ▪ Competing alternatives (J) ▪ Technological trends ▪ Leveraging strategy ▪ Collected market data	▪ Market segments (Ms_i) ▪ Planning refinement	▪ Refine the segmentation defined apriori in terms of the number of segments, competing alternatives, share, price and performance tiers ▪ Synthesize the corporate strategy into objective measures for product family development	▪ Delphi (Dalkey, 1969) ▪ Market segmentation grid (Meyer and Lehnerd, 1997) ▪ Latent class analysis (Chen, Hoyle and Wassenaar, 2013) ▪ Requirements list (Pahl *et al.*, 2007)
	$S_{1.3}$ - Build the PF configuration model	▪ Potential market size (M_k) ▪ Expected profit (V_e) ▪ Choice probability model (Pr_e) ▪ Design rules	▪ Objective function and Constraints	▪ Aggregate the customer's choice probability, the design rules, the set of design parameter instances, and the enterprise-level indicators into a single model for selecting and parameterizing the physical modules to compound the PF structure.	▪ Mathematical modeling (Hilier and Lieberman, 2015)
Cp_2 - Customers' choice modeling	$S_{2.1}$ - Identify customer-desired attributes (A)	▪ $Ms_i = \{S_{Msi}, P_i, S_i, A_i, J_i\}$ ▪ Customers' needs (current and future) ▪ Customers' priorities	▪ Customer-desired attributes (A_i)	▪ Identify features and financial attributes that customers (will) consider when purchasing the product	▪ Qualitative interviews (Malhotra and Birks, 2007) ▪ Direct observation (Kazemzadeh *et al.*, 2009) ▪ Focus group (Malhotra and Birks, 2007) ▪ Survey (Forza, 2002) ▪ Descriptive statistics (Montgomery and Runger, 2011) ▪ Content analysis (Bardin, 1993)
	$S_{2.2}$ - Formulate the engineering attributes (E)	▪ Customer-desired attributes (A_i) ▪ Competitors' existing products ▪ Company's existing products ▪ New E_i	▪ Engineering attributes (E_i)	▪ Transform the customer-desired attributes into quantifiable product properties to be used in the engineering product development process	▪ Analysis of existing technical systems (Pahl *et al.*, 2007). ▪ Benchmarking (Thevenot and Simpson, 2007) ▪ Reverse engineering (Thevenot and Simpson, 2007)
	$S_{2.3}$ - Map the dependencies between A and E	▪ Customer-desired attributes (A_i) ▪ Engineering attributes (E_i)	▪ $[A]_m = [R]_{m \times n}[E]_n$	▪ Map the qualitative customer-desired attributes into quantitative engineering attributes to support the further construction of the choice models	▪ Design matrix (Suh, 2001)

(continued)

Table 25. (continued).

Classes of design problems (Cp_i)	Sub-functions (S_i)	Inputs	Outputs	Functional Requirements	Techniques
Cp_2 - Customers' choice modeling	$S_{2.4}$ - Collect the engineering attributes levels (E_v) for each Ms_i	▪ Market segments (Ms_i) ▪ $[A]_m = [R]_{m \times n}[E]_n$ ▪ Competitors' existing products ▪ Company's existing products ▪ Price (P)	▪ $\vec{J_i} = \{E_{vi}, P_i\}$ ▪ New E_i	▪ Collect different E values (E_{vi}), and price (P_i), such that a set of E_{vi} and P_i belong to a competing alternative (J_i), i.e. $\vec{J_i} = \{E_{vi}, P_i\}$	▪ Analysis of existing technical systems (Pahl *et al.*, 2007) ▪ Benchmarking, Reverse engineering (Thevenot and Simpson, 2007)
	$S_{2.5}$ - Construct the choice set (Cs) for each Ms_i	▪ Market segments (Ms_i) ▪ $\vec{J_i} = \{E_{vi}, P_i\}$	▪ $Cs = \{\vec{J_i} \subset \vec{J_i}\}$ ▪ Collected market data	▪ Reduce the number of choice alternatives to be evaluated without losing the statistical significance	▪ Fractional factorial design (Montgomery and Runger, 2011)
	$S_{2.6}$ - Collect data on customers' preferences	▪ Customer-desired attributes (A_i) ▪ Engineering attributes (E_i) ▪ Market segments (Ms_i) ▪ $Cs = \{\vec{J_i} \subset \vec{J_i}\}$ ▪ Customers' preferences ▪ Socio-demographic attributes (S)	▪ Stated preferences (SP) ▪ Revealed preferences (RP)	▪ Collect data on customers' preferences by asking them to compare the engineering attributes of a product against each other, or to pick an alternative from a choice set	▪ Qualitative interviews (Malhotra and Birks, 2007) ▪ Focus group (Malhotra and Birks, 2007) ▪ Survey (Forza, 2002)
	$S_{2.7}$ - Build the discrete choice model (Pr_i) for each Ms_i	▪ Customer-desired attributes (A_i) ▪ Market segments (Ms_i) ▪ $\vec{J_i} = \{E_{vi}, P_i\}$ ▪ Socio-demographic attributes (S) ▪ Stated preferences (SP) ▪ Revealed preferences (RP)	▪ $Pr_i = f(Ms_i, J_i, E_i, P_i, S_i)$	▪ Estimate the weights/coefficients of the engineering attributes, and then calculate the utility function for all alternative of each target market segment ▪ Estimate the choice probability of each alternative within its market segment	▪ Analytic hierarchy process (Alonso and Lamata, 2006; Saaty, 2008) ▪ Nested logit (Chen, Hoyle and Wassenaar, 2013) ▪ Maximum likelihood estimation (Chen, Hoyle and Wassenaar, 2013) ▪ Data scaling (Chen, Hoyle and Wassenaar, 2013)
Cp_3 - Product family modeling	$S_{3.1}$ - Formulate the design parameters (DP)	▪ Engineering attributes (E_i) ▪ Available technologies (current and future) ▪ Existing products' features and analogous systems ▪ Clustering refinement ▪ New DP_i	▪ Design parameters (DP_i)	▪ Define the logical entity with the ability to fulfill one or more E_i.	▪ Domain knowledge (Jiao and Tseng, 1999a) ▪ Classification scheme (Pahl *et al.*, 2007)
	$S_{3.2}$ - Map the product family architecture (PFA)	▪ Engineering attributes (E_i) ▪ Design parameters (DP_i) ▪ Clustering refinement ▪ Mapping refinement	▪ $[DP]_m = [PFA]_{m \times n}[E]_n$	▪ Map the logical coupling between the E_i and DP_i, i.e. $[E]_m = [PFA]_{m \times n}[DP]_n$.	▪ Design matrix (Suh, 2001)
	$S_{3.3}$ - Decompose PFA into functional modules (FM)	▪ $[DP]_m = [PFA]_{m \times n}[E]_n$ ▪ Clustering refinement	▪ Functional modules (FM_i) ▪ Integral architectures	▪ Decompose the PFA into functional modules (FM), i.e. $FM_i = \{E_i, DP_i\}$.	▪ Rank order clustering (King, 1980) ▪ Cluster identification algorithm (Kusiak and Chow, 1987)
	$S_{3.4}$ - Evaluate the clustering solution	▪ $[DP]_m = [PFA]_{m \times n}[E]_n$ ▪ $FM_i = \{E_i, DP_i\}$	▪ Functional modularity measure (MI_f)	▪ Capture the strength and density of connections within each independent FM and between different FMs, i.e. $MI_f \geq 0{,}5$.	▪ Modularity index (Jung and Simpson, 2017)
	$S_{3.5}$ - Define DP instances (DPI)	▪ $FM_i = \{E_i, DP_i\}$ ▪ $\vec{J_i} = \{E_{vi}, P_i\}$ ▪ Available technologies (current and future) ▪ Existing products' features and analogous systems	▪ $\overrightarrow{DPI_i} = \{E_{vi}\}$	▪ Define different instances (DPI) for a particular DP along with its respective E_v i.e. $\overrightarrow{DPI_i} = \{E_{vi}\}$	▪ Classification scheme (Pahl *et al.*, 2007) ▪ Analysis of existing technical systems (Pahl *et al.*, 2007) ▪ Benchmarking (Thevenot and Simpson, 2007) ▪ Reverse engineering (Thevenot and Simpson, 2007)

(continued)

Table 25. (continued).

Classes of design problems (Cp_i)	Sub-functions (S_i)	Inputs	Outputs	Functional Requirements	Techniques
Cp_3 - Product family modeling	$S_{3.6}$ - Estimate the DPI variable cost (C_v)	▪ $\overrightarrow{DPI_i} = \{E_{vi}\}$ ▪ Historical product s' costs ▪ Suppliers' quotations ▪ Domain knowledge	▪ $\overrightarrow{DPI_i} = \{E_{vi}, C_{vi}\}$	▪ Estimate the variable cost (C_v) for each DPI based on its cost-related design features (CDF)	▪ Pragmatic approach to product costing (Jiao and Tseng, 1999b) ▪ Request for quotation (Gümüş, 2014) ▪ Three-point estimate (Premachandra, 2001)
	$S_{3.7}$ - Create rough geometric layouts	▪ $FM_i = \{E_i, DP_i\}$ ▪ $\overrightarrow{DPI_i} = \{E_{vi}, C_{vi}\}$ ▪ Physical refinement	▪ Physical interactions ▪ New DP_i ▪ Incompatible DPI	▪ Identify the physical interactions between DPs	▪ Sketching e rendering (Koos Eissen *et al.*, 2007)
	$S_{3.8}$ - Map the structural dependencies among components	▪ Design parameters (DP_i) ▪ Physical interactions ▪ Physical refinement	$[DP]_m = [DSM]_{m \times n}[DP]_n$	▪ Map the physical coupling between DPs, i.e. $[DP]_m = [DSM]_{m \times n}[DP]_n$.	▪ Design structure matrix (Browning, 2001)
	$S_{3.9}$ - Decompose the system into physical modules (M)	▪ $[DP]_m = [DSM]_{m \times n}[DP]_n$ ▪ $FM_i = \{E_i, DP_i\}$ ▪ Physical refinement	▪ Physical modules (M_i)	▪ Decompose the DSM into physical modules (M), i.e. $M_i = \{DP_i\}$.	▪ Functional to physical decomposition (Authors)
	$S_{3.10}$ - Compare M_i with FM_i	▪ Functional modularity measure (MI_f)	▪ $MI_p \geq MI_f$	▪ Compare if $MI_p \geq MI_f$	▪ Modularity index (Jung and Simpson, 2017)
	$S_{3.11}$ - Establish the design rules for PF configuration	▪ $M_i = \{E_{v_i}, C_{vi}\}$	▪ Design rules ▪ Mapping refinement ▪ $M_i = \{E_{v_i}, C_{vi}\}$ ▪ $\overrightarrow{DPI_i} = \{E_{v_i}, C_{vi} \mid \forall DPI_i \in DP_i\}$	▪ Define design rules for product family configuration	▪ Generic bill-of-material (Li, Huang and Newman, 2008) ▪ Mathematical modeling (Hilier and Lieberman, 2015)
Cp_4 - Product family configuration	$S_{4.1}$ - Combine DPI_i to generate the PF variants (PFv) for each Ms_i	▪ Objective function and constraints ▪ $M_i = \{E_{v_i}, C_{vi}\}$ ▪ $\overrightarrow{DPI_i} = \{E_{v_i}, C_{vi} \mid \forall DPI_i \in DP_i\}$	▪ $PFv_i = \left\{\{E_{v_i}, C_{vi}\} \middle\| \forall \{E_{v_i}, C_{vi}\} \in M_i\right\}$ ▪ Scalable engineering attributes (E_i)	▪ Combine the design parameter instances to generate PF variants for each target market segment	▪ Design heuristic - Substitute way of achieving functions (Daly et al., 2012) ▪ Genetic algorithm (Meng, Jiang and Huang, 2007)
	$S_{4.2}$ - Parametrize the scalable E_i	▪ Scalable engineering attributes (E_i)	▪ Scaled E_{v_i}	▪ Set the values for the scalable engineering attributes	▪ Genetic algorithm (Meng, Jiang and Huang, 2007)
	$S_{4.3}$ - Set the price (P) for each PFv variant	▪ $PFv_i = \left\{\{E_{v_i}, C_{vi}\} \middle\| \forall \{E_{v_i}, C_{vi}\} \in M_i\right\}$	▪ Price (P)	▪ Set the price for each product family variant	▪ Trial-and-error (Rui, Cuervo-Cazurra and Annique Un, 2016) ▪ Genetic algorithm (Meng, Jiang and Huang, 2007)
	$S_{4.4}$ - Calculate the partial profit (V_{Ms}) for each Ms_i and aggregate it into the PF profit (V)	▪ Potential market size (M_k) ▪ Share of mkt. seg. / niches (S_{Ms}) ▪ Choice probability (Pr_i) ▪ Price (P)	▪ PF profit (V)	▪ Calculate the partial profit for each target market segment, and then aggregate it into a measure that represents the product family profitability	▪ Mathematical modeling (Hilier and Lieberman, 2015)
	$S_{4.5}$ - Build the PF structure	▪ Most profitable PFv_i	▪ $PF\ structure = \{M_i\}$	▪ Build the product family structure with physical modules compound by design parameter instances retrieved from the most profitable variants of each segment	▪ Generic bill-of-material (Li, Huang and Newman, 2008)

Then, in step 6.2, 211 techniques (T_i) integrating the aggregated classification scheme (CS_A) along with 15 alternative techniques (AT_i) derived from the researcher's previous knowledge have been assessed. The selection procedure employed here was the Elimination and Preference (Pahl *et al.*, 2007), wherein all unsuitable techniques are eliminated, and among the remaining ones, those that are patently better than the rest are given preference. Based on functional requirements along with 5 other selecting criteria presented in Figure 28, 38 techniques were selected, as presented in Table 25.

Created by:	Date:	**SELECTION CHART**	Pg.
Gauss	*07/02/2019*	*Market-Driven Modularity (MDM)*	1

Technique (T) evaluated by SELECTION CRITERIA
+ Yes
- No
? Lack of information
! Check functional requirements

DECISION
+ Pursue technique
- Eliminate technique
? Collect information
! Check functional requirements for changes

A: Does it fulfill the demands of functional requirements?
B: Is it compatible with the neighbor techniques?
C: Is it technically feasible?
D: Does it encompass different classes of products (consumer, intermediate and capital goods)?
E: Are there available tools to perform the technique?
F: Is it preferred by the researcher?
G:

Enter the sub-function and its executing technique [Si,Ti]	A	B	C	D	E	F	G	Remarks (Indications, Reasons)	DECISION
S1.1, T3	-							*Does not translate and synthesise the corporate strategy into objective measures.*	-
S1.1, T4	-							*It is adopted for developing a business competitive strategy (out of boundaries of MDM)*	-
S1.1, T26	+	+	+	+	+	-		*Can be used to categorize and select which project comply with MDM scope*	+
S1.1, T27	-							*Does not translate and synthesise the corporate strategy into objective measures.*	-
S1.1, T38	-							*Does not translate and synthesise the corporate strategy into objective measures.*	-
S1.1, T44	-							*Does not translate and synthesise the corporate strategy into objective measures.*	-
S1.1, T45	+	+	+	+	+	+		*Articulates product family leveraging strategies in a multidimensional space.*	+
S1.1, T58	-							*Does not translate and synthesise the corporate strategy into objective measures.*	-
S1.1, T76	-							*Does not translate and synthesise the corporate strategy into objective measures.*	-
...	...	...	...	...	...	...	...	...	...

Figure 28. Selection chart of techniques.

At stage 7, the internal environment of the proposed method, entitled Market-Driven Modularity (MDM), was defined (Simon, 1996). This process started by building the graphical representation and the narrative description of MDM in step 7.1. Besides that, the mathematical model to operationalize the MDM was implemented in step 7.2. At the end of this stage, the first version of the method (V_1) was ready to be tested. Details on MDM will be provided in the next sections.

So far, the decisions made accounted for the MDM's development. Although Figure 27 and Table 25 presented the MDM in its final version (V_6), it only reached that format after five evaluation/refinement cycles, as shown in Figure 24. The first cycle (step 8.1) was performed by the researchers themselves in a simple made-up case, wherein a family of modular axes was conceptually designed for seven market segments. The data used in the design process came from the website of eight competitors in this market. Figure 29 illustrates some modules retrieved from this process, and the opportunities for improvement resulting from it were implemented giving rise to the second version of the method (V_2).

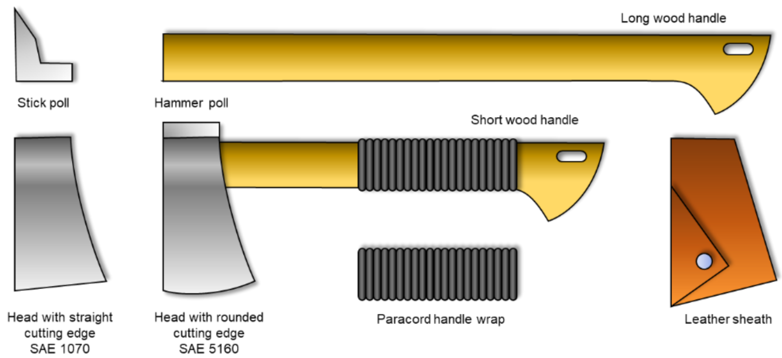


Figure 29. Family of modular axes.

In the second cycle (step 8.2), the MDM was tested in simple made-up cases conducted by Graduate Students of Advanced Manufacturing. The MDM V_2 was taught for 8 students along 12 hours, and then used to conceptually design families of tool trolleys and roof cargo boxes. The students were selected for convenience, and its characterization can be found in Table C2. At the end of the design process, the participants' opinions were captured through a questionnaire composed of closed and open questions, as presented in Appendix D (Malhotra and Birks, 2007). The top terms, constructs, and dimensions associated with the questionnaire are given in Table C3. After data collection, then the Median ($\tilde{x}$) (Montgomery and Runger, 2011) and the

Free-Marginal Multirater Kappa $\left(k_{free}\right)$ (Randolph, 2005) was calculated for each closed question. For the open questions, in turn, the Content Analysis was used to derive the moderating variables (MV_i) and its respective frequencies (f) (Bardin, 1993). In this context, the $\tilde{x}$ measures the amplitude of agreement, k_{free} measures the level of agreement among the respondents, and f measures how often the moderating variables (MV_i), supposed to reduce the amplitude of agreement, occur. Table 26 presents four possible scenarios and its respective actions resulting from the participants' opinions. At the end of this second cycle, the MDM was updated to version 3 (V_3).

Table 26. Scenarios and actions resulting from participants' opinions.

Id.	Conditions	Action
1	$k_{free} \geq 0.41$ and $\tilde{x} = 3$	No changes in the method are required, and the process should go-ahead to the next step.
2	$k_{free} \geq 0.41$ and $\tilde{x} < 3$	Changes in the method are required, and the process should go back to step 7.1 or 7.2.
3	$k_{free} < 0.41$ and $\tilde{x} = 3$	Changes in the method are required, and the process should go back to step 7.1 or 7.2.
4	$k_{free} < 0.41$ and $\tilde{x} < 3$	Changes in the method are required, and the process should go back to step 7.1 or 7.2.

In the third cycle (step 8.3), the MDM V_3 was presented to 10 experts in product development instead of being tested by them. The presentation was conducted personally or remotely, along one hour of duration, with the audio being recorded when permitted. In this cycle, the audio records served as an additional source to derive MV_i and f through the Content Analysis (Bardin, 1993). The experts were selected through snowball sampling, wherein only individuals having a bachelor's degree and more than five years of experience in product development have been included (Floyd and Fowler, 2014). The characterization of the participants can also be found in Table C2. At the end of this cycle, the opportunities for improvement led us to update the method to its version 4 (V_4).

The fourth cycle (step 8.4) followed the same reasoning of the third but with slight differences. Here, a 15 min video presenting the MDM V_4 was recorded. The video along with supplementary material was sent to 9 scholars worldwide. Among them, 5 responded to the questionnaire, and 4 gave their impressions by e-mail. In this cycle, the e-mails served as an additional source to derive MV_i and f through the Content Analysis (Bardin, 1993). The scholars were selected by judgment and snowball sampling, wherein those individuals conducting research related to product development were included. The characterization of the participants is found in Table C2. At the end of this cycle, the MDM was updated to version 5 (V_5).

The last evaluation cycle (step 8.5) was performed by the researchers themselves in a more complex made-up case, wherein a family of collaborative robotic palletizers was conceptually designed for six market niches (Popple, 2009). The MDM (V_5) application was supported by two experts, and the data used in the design process came from projects quoted by two machine manufacturers, as well as from the website of three leading competitors in this market. Figure 30 illustrates some modules derived from this process, and the shortcomings resulting from it were used to update the method to its final (V_6).

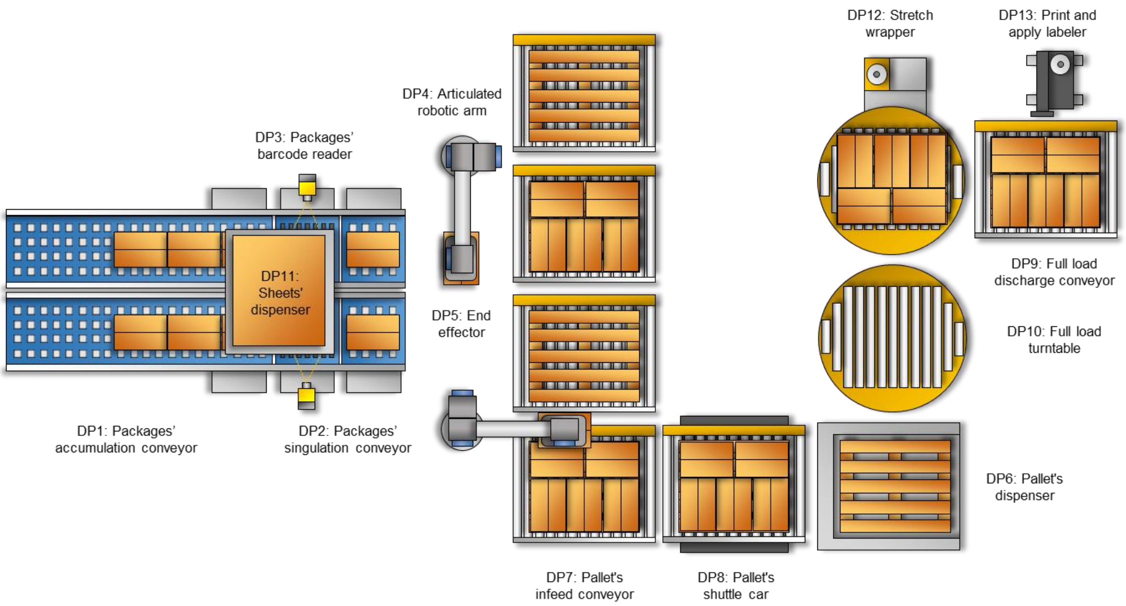


Figure 30. Family of collaborative robotic palletizers.

The outputs of stage 8 are the duly evaluated method ($MDM\ V_6$) along with its construction and contingency heuristics (Dresch, Lacerda and Antunes Jr, 2015). The learnings achieved during the MDM's design and evaluation were identified in step 9.1 and then cataloged in Table C1 (Cole, 2005; van Aken, Berends and van der Bij, 2012). In step 10.1, the research contributions, limitations as well its future directions were formalized (Vaishnavi, Kuechler and Petter, 2017). While in step 11.1, the MDM, together with its construction and contingency heuristics, were generalized for a particular class of design problems (Venable, 2006; Gregor, 2009). The outcomes of stages from 8 to 11 will be presented in the two next sections.

Finally, at stage 12, the knowledge generated from the research process was compiled into four articles to be shared with scholars and practitioners. The first two articles encompassed the procedures and results retrieved from the two problem-related topics investigated in the systematic literature review stage. The third article (this one) covered the MDM's design and evaluation from a design science perspective. The last article entailed the MDM in its functional state applied to a complex made-up case.

5.4 Proposed Method: Market-Driven Modularity

The Market-Driven Modularity (MDM) consists of an integrated method to conceptually design modular product families that balance the fulfillment of market needs and the resulting profitability to achieve them. To prevent the development of non-profitable product families, the MDM uses the discrete choice modeling for quantifying the customers' preferences, modularity as a mechanism to provide product variety, the product family as a strategy to manage the trade-off between the variety and cost, and profit as a moderating variable to balance the level of accomplishment of the customers' needs.

Concerning its external environment, the MDM is intended to be adopted in the early design stages of the product development process of small, midsize, and large companies that produce consumer durables, intermediate, and capital goods. Besides that, the MDM has been developed to redesign the existing families from a modular point of view as well as to design new modules, new families, and new generations of families, in contexts from low to high data availability.

Regarding the internal functional environment, it is composed of 26 steps, arranged in 4 classes of design problems, that can be performed by, at least, 38 techniques. The reasoning behind the method is to define the target market segments, model the customers' choice probabilities for each of them, and then define a modular product family architecture, corresponding to all segments. With the product family architecture defined, the design parameter instances are generated and combined into a finite set of variants for each segment. Then, after setting the price, the demand is estimated, and the resulting profit of each variant is calculated. The most profitable variants have their gain aggregated into the product family profit, and the design parameter instances compounding them are selected to integrate the physical modules of the product family structure. If the product family profit matches the expected profit, the process is finished. Otherwise, the process should be restarted until the product family reaches the desired gain or until it is discarded. The two expected MDM outcomes are: (i) the modular product family structure that better balances the fulfillment of market needs and the resulting profitability to achieve them, (ii) and the decision on investing or not in the product family design.

From a more detailed perspective, the MDM method starts by converting the corporate strategy into objective measures for product family design, as shown in Figure 31. This process takes place at the first class of design problems, named here as Product

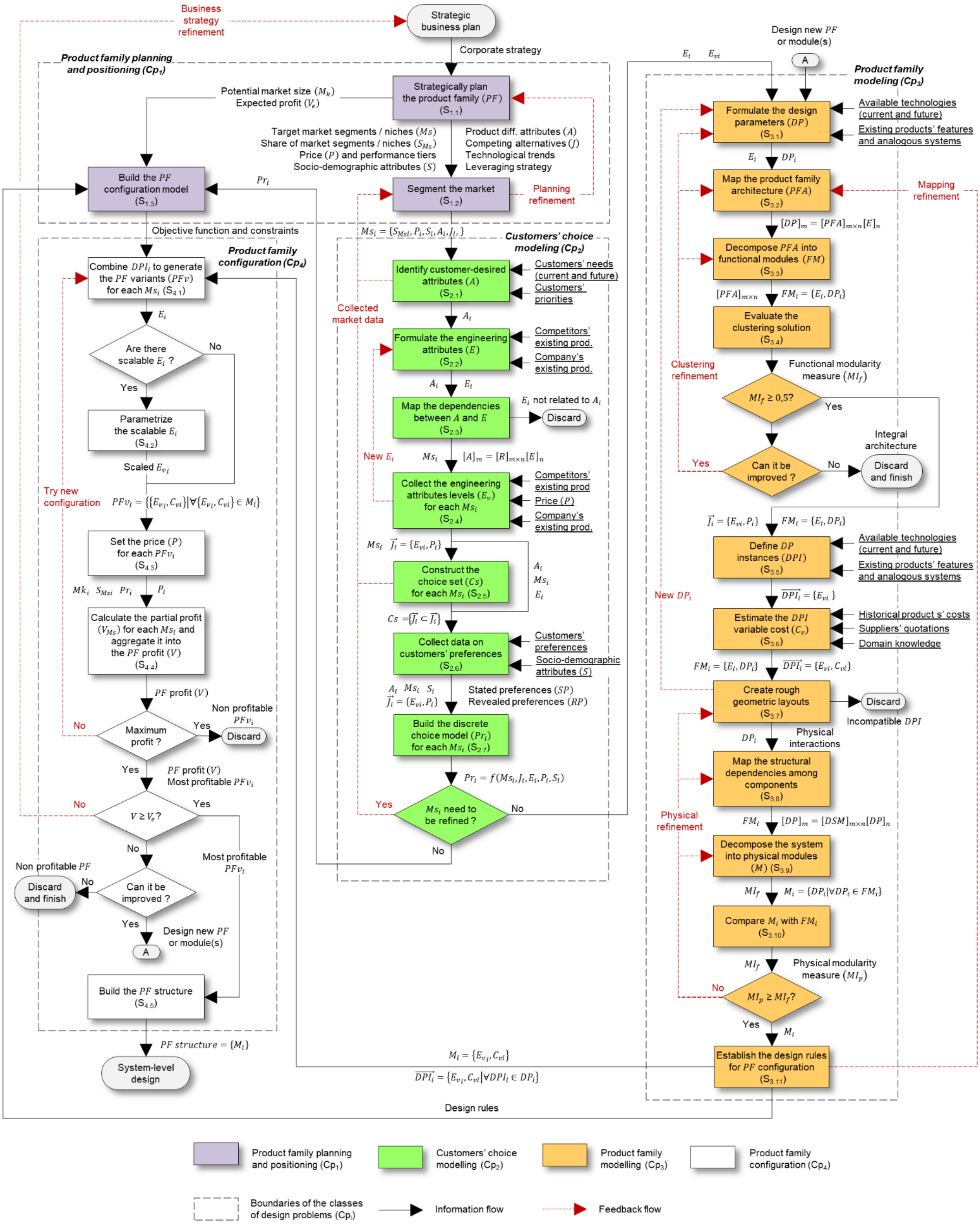


Figure 31. Internal Functional Environment of MDM.

Family Planning and Positioning (Cp_1). Within this class, at step $S_{1.1}$, the potential market size (M_k) and the expected profit (V_e) are estimated. Besides that, the target market segments (Ms), the technological trends and the product family leveraging strategy are also established. In the next step $(S_{1.2})$, the market segmentation is refined, and the resulting specifications serve as an input flow for identifying the customer desired attributes (A) at stage $S_{2.1}$, or, as feedback for improving the strategic product family planning at stage $S_{1.1}$.

The identification of customers' desired attributes consists of the first step $(S_{2.1})$ of the second class of design problems entitled here as the Customers' Choice Modeling (Cp_2). These attributes might derive from current or future needs and despite its nature, they need to be converted into a language that engineers use to develop products. In some cases, the data gathered here $(S_{2.1})$ might be useful for refining the market segmentation at the previous step. The translation from customers' to engineering attributes (E) is performed at the stage $S_{2.2}$, and the relationship (R) between them is mapped in step $S_{2.3}$, i.e. $[A]_m = [R]_{m\times n}[E]_n$. Those customer-related engineering attributes should go forward to step $S_{2.4}$; otherwise, they should be discarded. In general, the engineering attributes might assume different levels within and across segments; for that reason, at stage $S_{2.4}$, a set of competing alternatives (J) for each segment is captured. Each competing alternative consists of a vector compound by engineering attribute values (E_v) and price (P), i.e. $\vec{J_i} = \{E_{vi}, P_i\}$. For those market pull product families, the competing alternatives usually derive from competitors or the company's existing products, while, for those technology push product families, the alternatives are deducted based on the product family planning and positioning. In both situations, the life-cycle of the competing alternative should be assessed before deciding if it is going to integrate the choice set. At this stage $(S_{2.4})$, new engineering attributes might emerge; in such cases, they serve as feedback for the step $S_{2.2}$. The next step is to define the set of alternatives by which the customers will state their preferences within each segment. Sometimes, the number of alternatives might be particularly high, difficulting in this way, the preference statement. When it happens, the choice set (Cs) must be reduced without losing the statistical significance in step $S_{2.5}$, i.e. $Cs = \{\vec{J_\iota} \subset \vec{J_\iota}\}$. In situations where the data reduction is not required, the step $S_{2.5}$ should be by-passed. With the choice set defined, the data on customers' preferences are collected in stage $S_{2.6}$. In contexts of low data availability, the key customers, or experts in the field, are asked to compare the customer desired attributes against each other for each target market segment. In contexts of high data availability, in turn, the customers are requested to pick an alternative from a choice set, emulating in this way, the real purchasing decisions within each segment. At stage $S_{2.7}$, depending on the technique used, the engineering attributes' coefficients (β), or weights (w), are estimated based on customers' stated/revealed preferences. Then, the utility (W) and the choice probability (Pr) of each alternative comprising the same target segment are modeled. If any deviation on market segmentation is found during the customers' choice modeling, the process should restart until the marginal difference become insignificant.

With the customer's choice modeled, the next issue is to define the product family architecture, decompose it into functional/physical modules, and then generate the design parameter instances that can potentially compose the product family structure. This process is performed in the third class of problems named here as Product Family Modeling (Cp_3). Within this class, at stage $S_{3.1}$, the process starts by formulating those logical entities with the ability to accomplish one or more engineering attributes. These logical entities are named here as design parameters (DP), and their formulation derives not only from the available technology and existing product features but also from future technology trends and analogy with other systems. Once defined, the design parameters are mapped to engineering attributes, giving rise to the product family architecture (PFA) in stage $S_{3.2}$, i.e. $[DP]_m = [PFA]_{m \times n}[E]_n$. The product family architecture defined here comprises all target segments together, and it should be designed to meet the functional independence axiom (Suh, 1998). Then, at stage $S_{3.3}$, the product family architecture is decomposed into functional modules (FM) and have its clustering solution evaluated at stage $S_{3.4}$, i.e. $FM_i = \{E_i, DP_i\}$. If the clustering solution accomplishes the desired level of functional modularity, i.e. $M_f \geq 0{,}5$, the process should go forward. Otherwise, the clustering refinement should be performed until it reaches the expected value or until an integral architecture is found, i.e. $M_f < 0{,}5$. In the last situation , the process should be finished as indicated by Figure 31. With the functional modules defined, the next issue is to specify the engineering attributes values (E_v) resulting from different physical characteristics that a design parameter might assume. This task of defining the design parameter instances (DPI) is performed at the stage $S_{3.5}$, i.e. $\overrightarrow{DPI_i} = \{E_{vi}\}$. Couple with that arises the trade-off between the engineering attribute levels and the costs to achieve it. For that reason, the variable cost (C_v) of each design parameter instance is estimated at stage $S_{3.6}$, i.e. $\overrightarrow{DPI_i} = \{E_{vi}, C_v\}$. The variable cost is assumed here to overcome the limitations of traditional cost accountability in dealing with product mix-related decisions (Cox and Schleier, 2010). After that, to identify the physical interactions within and across functional modules, rough geometric layouts are created in step $S_{3.7}$. In this step, just as incompatible design parameter instances may be discarded, new design parameters might emerge giving rise to feedback from here $(S_{3.7})$ to the stage $S_{3.1}$. The physical interactions resulting from this step, serve as an input flow for mapping the structural dependencies among design parameters in step $S_{3.8}$. Then, at stage $S_{3.9}$, the functional decomposition is transferred to the physical decomposition, and the relationship between these two modularity indices is evaluated at stage $S_{3.10}$. The reasoning here is that the physical modularity must not prevent the functional modularity as suggests Figure 32.

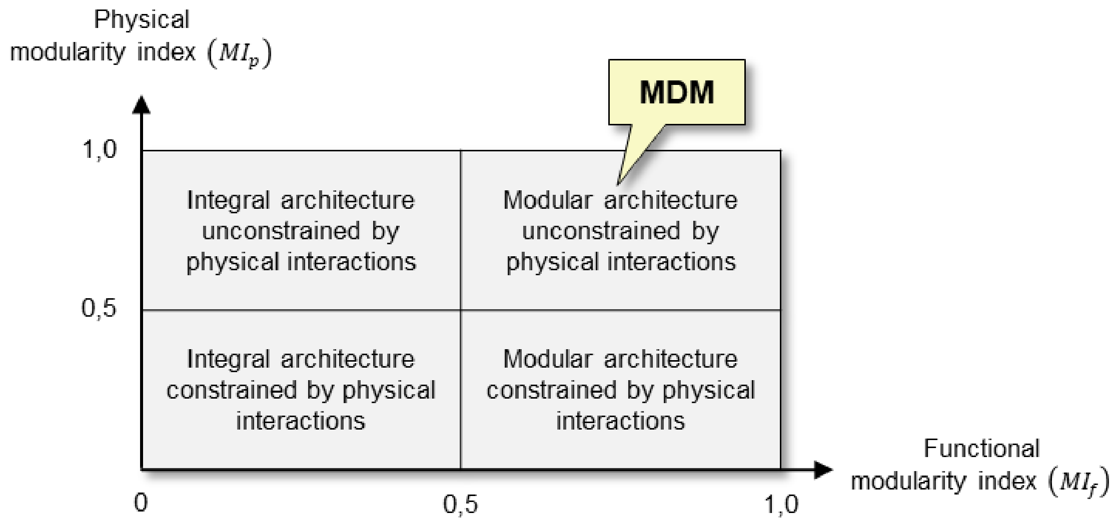


Figure 32. Relationship between functional (M_f) and physical (M_p) modularity indices.

If a modular architecture unconstrained by physical interactions is reached, i.e. $(MI_p \geq MI_f \mid MI_f \geq 0{,}5)$, the process goes forward. Otherwise, the iterative refinement on physical modularity should be performed until it reaches the desired value. Finally, the design rules for product family configuration are defined at the stage $S_{3.11}$. It is also seen that when setting the configuration rules, modifications to the product family architecture may arise as indicated by the feedback to step $S_{3.2}$.

Back to the first class (Cp_1), at stage $S_{1.3}$, the issue is to aggregate the customer's choice probability, the design rules, the set of design parameter instances, and the enterprise-level indicators (demand, price, and profit) into a single model for combining, selecting and parameterizing the design parameter instance to compound the modular product family structure. This configuration process takes place in the fourth class of design problems, named here as Product Family Configuration (Cp_4). At this class, the process starts at the stage $S_{4.1}$, where the design parameter instances are combined into family variants for each target market segment. If some design parameter instances contain scalable engineering attributes, its values are adjusted in the step $S_{4.2}$. With the product variant configured and parametrized, the price (P) is set at stage $S_{4.3}$, and its demand (Q) and partial profit (V_{Ms}) are calculated in the next step. The $S_{4.4}$ not only calculates the partial profit of each variant in its respective segment but also aggregates it into the product family profit (V). The steps from $S_{4.1}$ to $S_{4.4}$ of this fourth class are performed repeatedly until it reaches the optimal or near-optimal profitability. At the end of this process, the non-profitable variants are discarded, and the resulting product family profitability is compared to the expected gain (V_e). If it is considered satisfactory, i.e. $V \geq V_e$, those most profitable variants are computed, and the design parameter instances compounding them are selected to integrate physical modules of the product family structure at the stage $S_{4.5}$. Otherwise, the process should be restarted until the product family reaches the desired value or until it is discarded. The output of the proposed method is the modular product family structure that better balance the trade-off between the fulfillment of market needs and the resulting profitability to meet them. Therefore, this is the structure that should be developed in the subsequent design stages of the product development process, not covered in this research.

Depending on the company's maturity and the context of data availability, different techniques might be adopted to execute each step of the method. In this sense, Table 27 complements the MDM functional structure by presenting an open architecture of techniques. The reasoning behind Table 27 is that, for each key activity compounding a method's step, there are one or more techniques capable of executing it. Those techniques placed at the left-hand side of the column Techniques are more suitable for contexts of low data availability. While those positioned at the right-hand side are more suitable for contexts of high data availability. Those techniques placed in the middle, in turn, can be used for both scenarios.

Table 27. MDM's open architecture of techniques.

Classes of design problems (Cp_i)	Steps of the method (S_i)	Key activities	(low data availability) Techniques	(high data availability)
Cp_1 - Product family planning and positioning	$S_{1.1}$ - Strategically plan the product family (PF)	Estimate the potential market size (M_k) and the expected profit (V_e).	Delphi (Dalkey, 1969), Three-point estimate (Premachandra, 2001), Domain knowledge (Jiao and Tseng, 1999a).	Survey (Forza, 2002), Descriptive statistics (Montgomery and Runger, 2011).
		Define the product family positioning and its leveraging strategy.	Market segmentation grid (Meyer and Lehnerd, 1997).	
		Define the technological trends for product family development.	Technology roadmap (Phaal and Muller, 2009).	
		Define which type of project must be developed.	Aggregate project plan (Wheelwright and Clark, 1992).	
	$S_{1.2}$ - Segment the market	Refine the segmentation defined apriori in terms of the number of segments, competing alternatives, share, price and performance tiers.	Delphi (Dalkey, 1969), Market segmentation grid (Meyer and Lehnerd, 1997).	Latent class analysis (Chen, Hoyle and Wassenaar, 2013).
		Synthesize the corporate strategy into objective measures for product family development.	Requirements list (Pahl *et al.*, 2007).	
	$S_{1.3}$ - Build the PF configuration model	Aggregate the customer's choice probability, the design rules, the set of design parameter instances, and the enterprise-level indicators into a single model for selecting and parameterizing the physical modules to compound the PF structure.	Mathematical modeling (Hilier and Lieberman, 2015).	
Cp_2 - Customers' choice modeling	$S_{2.1}$ - Identify customer-desired attributes (A)	Identify features and financial attributes that customers (will) consider when purchasing the product.	Qualitative interviews (Malhotra and Birks, 2007); Direct observation (Kazemzadeh *et al.*, 2009); Focus group (Malhotra and Birks, 2007).	Survey (Forza, 2002), Descriptive statistics (Montgomery and Runger, 2011).
			Content analysis (Bardin, 1993).	
	$S_{2.2}$ - Formulate the engineering attributes (E)	Transform the customer-desired attributes into quantifiable product properties to be used in the engineering product development process.	Analysis of existing technical systems (Pahl *et al.*, 2007). Benchmarking, Reverse engineering (Thevenot and Simpson, 2007).	
	$S_{2.3}$ - Map the dependencies between A and E	Map the qualitative customer-desired attributes into quantitative engineering attributes to support the further construction of the choice models.	Design matrix (Suh, 2001).	
	$S_{2.4}$ - Collect the engineering attributes levels (E_v) for each Ms_i	Collect different E values (E_{vi}), and price (P_i), such that a set of E_{vi} and P_i belong to a competing alternative (J_i), i.e. $\vec{J_i} = \{E_{vi}, P_i\}$.	Analysis of existing technical systems (Pahl *et al.*, 2007). Benchmarking, Reverse engineering (Thevenot and Simpson, 2007).	
	$S_{2.5}$ - Construct the choice set (Cs) for each Ms_i	Reduce the number of choice alternatives to be evaluated without losing the statistical significance.		Fractional factorial design (Montgomery and Runger, 2011).
	$S_{2.6}$ - Collect data on customers' preferences	Collect data on customers' preferences by asking them to compare the engineering attributes of a product against each other, or to pick an alternative from a choice set.	Qualitative interviews (Malhotra and Birks, 2007); Focus group (Malhotra and Birks, 2007).	Survey (Forza, 2002).
	$S_{2.7}$ - Build the discrete choice model (Pr_i) for each Ms_i	Estimate the weights/coefficients of the engineering attributes, and then calculate the utility function for all alternative of each target market segment.	Analytic hierarchy process (Alonso and Lamata, 2006; Saaty, 2008).	Nested logit, Maximum likelihood estimation (Chen, Hoyle and Wassenaar, 2013).
		Estimate the choice probability of each alternative within its market segment.	Analytic hierarchy process (Saaty, 2008); Data scaling (Chen, Hoyle and Wassenaar, 2013).	Nested Logit (Chen, Hoyle and Wassenaar, 2013).

(continued)

Table 27. (Continued)

<table>
<tr><th>Classes of design problems (Cp_i)</th><th>Steps of the method (S_i)</th><th>Key activities</th><th>(low data availability) Techniques</th><th>Techniques (high data availability)</th></tr>
<tr><td>Cp_3 - Product family modeling</td><td>$S_{3.1}$ - Formulate the design parameters (DP)</td><td>Define the logical entity with the ability to fulfill one or more E_i.</td><td colspan="2">Domain knowledge (Jiao and Tseng, 1999a),
Classification scheme (Pahl et al., 2007).</td></tr>
<tr><td></td><td>$S_{3.2}$ - Map the product family architecture (PFA)</td><td>Map the logical coupling between the E_i and DP_i, i.e. $[E]_m = [PFA]_{m \times n}[DP]_n$.</td><td colspan="2">Design matrix (Suh, 2001).</td></tr>
<tr><td></td><td>$S_{3.3}$ - Decompose PFA into functional modules (FM)</td><td>Decompose the PFA into functional modules (FM), i.e. $FM_i = \{E_i, DP_i\}$.</td><td colspan="2">Rank order clustering (King, 1980),
Cluster identification algorithm (Kusiak and Chow, 1987).</td></tr>
<tr><td></td><td>$S_{3.4}$ - Evaluate the clustering solution</td><td>Capture the strength and density of connections within each independent FM and between different FMs, i.e. $MI_f \geq 0{,}5$.</td><td colspan="2">Modularity index (Jung and Simpson, 2017).</td></tr>
<tr><td></td><td>$S_{3.5}$ - Define DP instances (DPI)</td><td>Define different instances (DPI) for a particular DP along with its respective E_v i.e. $\overrightarrow{DPI_i} = \{E_{vi}\}$</td><td colspan="2">Classification scheme, Analysis of existing technical systems (Pahl et al., 2007).
Benchmarking, Reverse engineering (Thevenot and Simpson, 2007).</td></tr>
<tr><td></td><td>$S_{3.6}$ - Estimate the DPI variable cost (C_v)</td><td>Estimate the variable cost (C_v) for each DPI based on its cost-related design features (CDF).</td><td colspan="2">Pragmatic approach to product costing (Jiao and Tseng, 1999b),
Request for quotation (Gümüş, 2014),
Three-point estimate (Premachandra, 2001).</td></tr>
<tr><td></td><td>$S_{3.7}$ - Create rough geometric layouts</td><td>Identify the physical interactions between DPs.</td><td colspan="2">Sketching e rendering (Koos Eissen et al., 2007).</td></tr>
<tr><td></td><td>$S_{3.8}$ - Map the structural dependencies among components</td><td>Map the physical coupling between DPs, i.e. $[DP]_m = [DSM]_{m \times n}[DP]_n$.</td><td colspan="2">Design structure matrix (Browning, 2001).</td></tr>
<tr><td></td><td>$S_{3.9}$ - Decompose the system into physical modules (M)</td><td>Decompose the DSM into physical modules (M), i.e. $M_i = \{DP_i\}$.</td><td colspan="2">Functional to physical decomposition (Authors).</td></tr>
<tr><td></td><td>$S_{3.10}$ - Compare M_i with FM_i</td><td>Compare if $MI_p \geq MI_f$.</td><td colspan="2">Modularity index (Jung and Simpson, 2017).</td></tr>
<tr><td></td><td>$S_{3.11}$ - Establish the design rules for PF configuration</td><td>Define design rules for product family configuration.</td><td colspan="2">Generic bill-of-material (Li, Huang and Newman, 2008),
Mathematical modeling (Hilier and Lieberman, 2015).</td></tr>
<tr><td>Cp_4 – Product Family Configuration</td><td>$S_{4.1}$ - Combine DPI_i to generate the PF variants (PFv) for each Ms_i</td><td>Combine the design parameter instances to generate PF variants for each target market segment.</td><td>Design heuristic - Substitute way of achieving functions (Daly et al., 2012).</td><td>Genetic algorithm (Meng, Jiang and Huang, 2007).</td></tr>
<tr><td></td><td>$S_{4.2}$ - Parametrize the scalable E_i</td><td>Set the values for the scalable engineering attributes.</td><td>Design heuristic - Scale up or down (Daly et al., 2012).</td><td>Genetic algorithm (Meng, Jiang and Huang, 2007).</td></tr>
<tr><td></td><td>$S_{4.3}$ - Set the price (P) for each PFv variant</td><td>Set the price for each product family variant.</td><td>Trial-and-error (Rui, Cuervo-Cazurra and Annique Un, 2016).</td><td>Genetic algorithm (Meng, Jiang and Huang, 2007).</td></tr>
<tr><td></td><td>$S_{4.4}$ - Calculate the partial profit (V_{Ms}) for each Ms_i and aggregate it into the PF profit (V)</td><td>Calculate the partial profit for each target market segment, and then aggregate it into a measure that represents the product family profitability.</td><td colspan="2">Mathematical modeling
(Hilier and Lieberman, 2015).</td></tr>
<tr><td></td><td>$S_{4.5}$ - Build the PF structure</td><td>Build the product family structure with physical modules compound by design parameter instances retrieved from the most profitable variants of each segment.</td><td colspan="2">Generic bill-of-material (Li, Huang and Newman, 2008).</td></tr>
</table>

5.5 Results

This section presents the results of the MDM evaluation cycles, divided into four sub-sections: (i) results of evaluation cycle 1, (ii) results of evaluation cycles 2, 3, and 4, (iii) results of evaluation cycle 5, and (iv) construction and contingency heuristics.

5.5.1 Results of Evaluation Cycle 1

In the first cycle, the MDM was used to conceptually design a family of modular axes for seven market segments (Ms). The total market size (M_k), in terms of axes a year, and the share of each market segment (S_{Ms}) was estimated according to Table 28. After that, the customers' choice probabilities (Pr) and the modular product family architecture were modeled. Coupled with that, the design parameter instances (DPI) along with its respective engineering attribute values (E_v) and variables cost (C_v) were defined, i.e. $\overrightarrow{DPI_i} = \{E_{vi}, C_v\}$. As a result, 5 functional modules (FM) and 23 design parameter instances gave rise to design space with 576 potential product family variants, as shown in Figure 34 (a). Then, the MDM's configuration model (Figure 33) was used to select the most profitable variants, one for each target market niche, as presented in Table 28. The last row of this table shows the results of the product family 1 $(PF1)$ covering all segments together, wherein the maximum profit found was $V = 2.08\text{x}10^6\ [USD/year]$.

Later, the design parameter instances compounding the most profitable variants were selected to integrate the final structure of product family 1. The solution compound by 5 physical modules (M), 13 design parameter instances, and capable of generating up to 96 variants is shown in Figure 34 (b). However, no threshold indicating if it is worth it to invest in the product family design was considered in the MDM V_1. This learning, as well as the others achieved in this first cycle of evaluation, are given in Table C1.

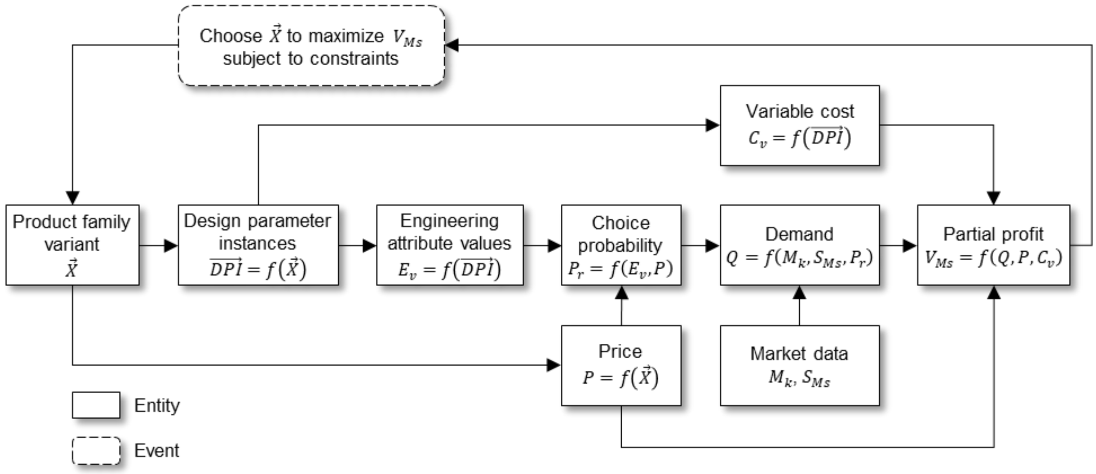


Figure 33. MDM's configuration model.

Table 28. Results of the configuration process of product family 1.

Ms	Product family variant $(\vec{X})$	M_k	S_{Ms}	Pr	Q	C_v [USD/SKU]	P [USD/SKU]	V_{Ms} [USD/year]
Ms1	[DPI1.5, DPI2.2, DPI3.6, DPI4.1, DPI5.1]	200,000	28.0%	14.5%	8,110	3.95	54.13	406,937.29
Ms2	[DPI1.5, DPI2.1, DPI3.7, DPI4.1, DPI5.2]	200,000	13.0%	21.7%	5,638	3.36	75.67	407,639.69
Ms3	[DPI1.5, DPI2.1, DPI3.7, DPI4.1, DPI5.2]	200,000	23.0%	22.6%	10,375	3.36	62.00	608,368.62
Ms4	[DPI1.7, DPI2.1, DPI3.7, DPI4.1, DPI5.2]	200,000	3.0%	17.9%	1,073	3.51	85.00	87,454.72
Ms5	[DPI1.6, DPI2.1, DPI3.4, DPI4.1, DPI5.2]	200,000	23.0%	13.9%	6,399	7.33	77.50	449,011.06
Ms6	[DPI1.6, DPI2.1, DPI3.5, DPI4.3, DPI5.2]	200,000	2.0%	30.1%	1,205	6.56	65.00	70,426.49
Ms7	[DPI1.6, DPI2.2, DPI3.4, DPI4.1, DPI5.2]	200,000	8.0%	10.6%	1,694	7.43	39.00	53,486.66
PF1	[DPI1.5, DPI1.6, DPI1.7, DPI2.1, DPI2.2, DPI3.4, DPI3.5, DPI3.6, DPI3.7, DPI4.1, DPI4.3, DPI5.1, DPI5.2]	200,000	100.0%	17.2%	34,495	157,087.58	2,240,412.10	2,083,324.52

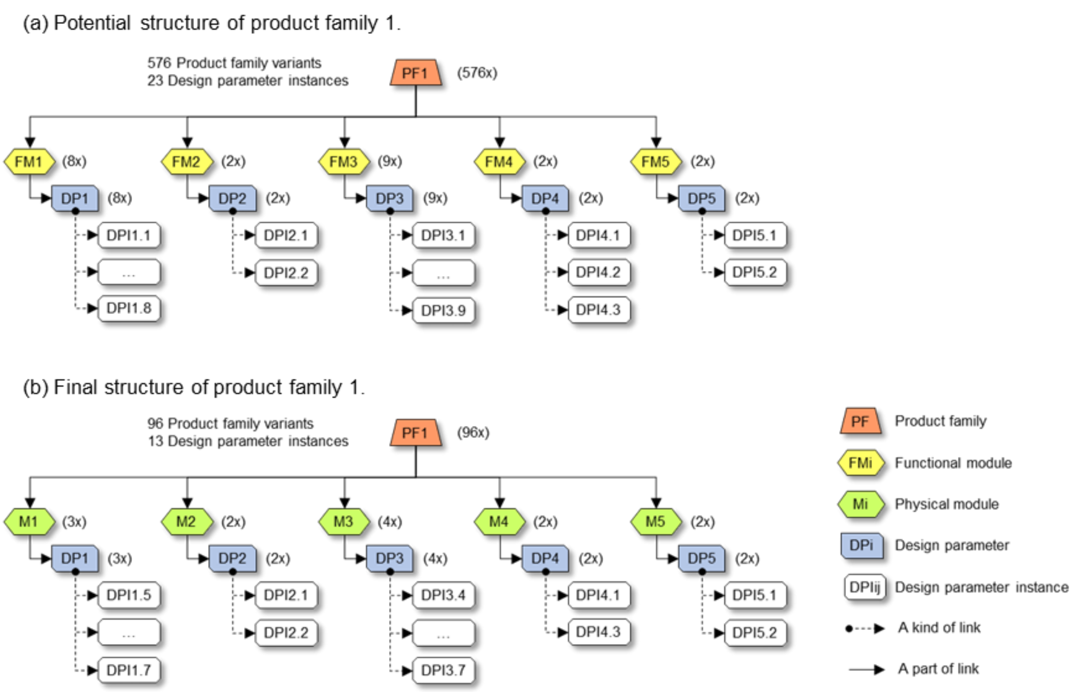


Figure 34. (a) Potential structure of product family 1; (b) Final structure of product family 1.

5.5.2 Results of Evaluation Cycles 2, 3, and 4

The results of evaluation cycles 2, 3, and 4 are summarized in Table 29. In this table, the hierarchy between the top terms, constructs, dimensions, and moderating variables is depicted, while its respective definitions are given in Table C3. The underlined numbers presented in Table 29 point out those values below the acceptable threshold adopted in this research, i.e. $k_{free} < 0.41$ or $\tilde{x} < 3$. Besides that, the characters within parentheses indicate to which question the dimension is related to.

In general, the *pragmatic validity* and *practical relevance* reached the highest amplitude of agreement $(\tilde{x} = 3)$. Regarding the level of agreement among ratters, these top terms ranged from moderate $\left(0.41 < k_{free} < 0.6\right)$ to substantial agreement $\left(0.61 < k_{free} < 0.8\right)$ in cycles 2 and 3, and from slight $\left(0.01 < k_{free} < 0.2\right)$ to fair $\left(0.21 < k_{free} < 0.4\right)$ agreement in cycle 4 (Landis and Koch, 1977). Concerning the constructs, the *internal environment* appeared to be the most robust one, since it fell within scenario 1 ($k_{free} \geq 0.41$ and $\tilde{x} = 3$) in the three cycles of evaluation. One construct that did not originally integrate the questionnaire was the *artifact's evaluation*, which emerged from the content analysis of those responses given by e-mail. Despite the satisfying results of the top terms and constructs in cycles 2 and 3, were the dimensions' results, and its respective moderating variables (MV_i) that led us to refine the MDM from version 2 to 5. The learnings achieved during these evaluation cycles, and how they influenced the MDM updates are given in Table C1.

Table 29. Results of the evaluation cycles 2, 3 and 4.

Top term / Construct / Dimension / Moderating variable	Cycle 2 Students (n=8)			Cycle 3 Experts (n=10)			Cycle 4 Scholars (n=5/9)		
	x̃	k_{free}	f	x̃	k_{free}	f	x̃	k_{free}	f
1.0 Pragmatic validity	**3**	**0.66**	**-**	**3**	**0.55**	**-**	**3**	**0.16**	**-**
1.1 External environment	*3*	*0.75*	-	*3*	*0.67*	-	*3*	*0.28*	-
1.1.1 (Q07) Company size	3	0.63	-	3	0.30	-	2	-0.05	-
MV.02 Cultural barriers	-	-	0	-	-	1	-	-	0
MV.09 Method's complexity	-	-	0	-	-	0	-	-	1
MV.11 Organizational immaturity	-	-	0	-	-	2	-	-	2
MV.28 Complex products	-	-	0	-	-	0	-	-	1
MV.29 Low scale orders	-	-	0	-	-	0	-	-	1
MV.35 Divergent key performance indicators (KPI)	-	-	0	-	-	1	-	-	0
1.1.2 (Q09) Production strategy	3	0.63	-	3	1.00	-	2	0.10	-
MV.29 Low scale orders	-	-	0	-	-	0	-	-	3
1.1.3 (Q11) Product development phases	3	1.00	-	3	0.70	-	3	0.10	-
MV.02 Cultural barriers	-	-	0	-	-	1	-	-	0
MV.09 Method's complexity	-	-	0	-	-	1	-	-	1
MV.10 Lack of a design supporting system	-	-	1	-	-	2	-	-	0
MV.11 Organizational immaturity	-	-	1	-	-	0	-	-	1
1.1.4 (Q13) Product type	3	0.63	-	3	1.00	-	3	0.10	-
MV.01 Aesthetics requirements	-	-	1	-	-	1	-	-	0
MV.03 Manufacturing under the customer's drawings	-	-	1	-	-	0	-	-	0
MV.07 Convenience goods	-	-	1	-	-	0	-	-	0
MV.28 Complex products	-	-	0	-	-	0	-	-	1
1.1.5 (Q15) Single market segment	3	1.00	-	3	0.30	-	3	1.00	-
MV.04 Low heterogeneity	-	-	1	-	-	0	-	-	0
1.1.6 (Q17) Multiple market segments	3	0.63	-	3	0.70	-	3	0.40	-
MV.01 Aesthetics requirements	-	-	1	-	-	0	-	-	0
MV.11 Organizational immaturity	-	-	0	-	-	0	-	-	1
MV.28 Complex products	-	-	0	-	-	0	-	-	1
1.2 Internal environment	*3*	*0.59*	-	*3*	*0.46*	-	*3*	*0.59*	-
1.2.1 (Q19) Steps' sufficiency	3	1.00	-	3	0.17	-	2	1.00	-
MV.11 Organizational immaturity	-	-	0	-	-	0	-	-	1
MV.12 Redundant steps	-	-	0	-	-	1	-	-	0
MV.28 Complex products	-	-	1	-	-	0	-	-	1
1.2.2 (Q21) Steps' execution order	3	0.63	-	3	0.47	-	3	0.63	-
MV.09 Method's complexity	-	-	1	-	-	0	-	-	1
1.2.3 (Q23) Adequacy of feedback flows	3	1.00	-	3	0.47	-	3	1.00	-
MV.14 Customers' satisfaction feedback	-	-	0	-	-	1	-	-	0
1.2.4 (Q25) Applicability of techniques	3	0.20	-	3	0.47	-	3	0.20	-
MV.06 Other existing techniques	-	-	1	-	-	0	-	-	0
MV.15 bottom-up techniques	-	-	0	-	-	1	-	-	0
MV.28 Complex products	-	-	0	-	-	0	-	-	1
1.2.5 (Q27) Suitability of qualitative techniques	3	0.63	-	3	0.70	-	3	0.63	-
MV.05 Uncertainty of estimated data	-	-	4	-	-	0	-	-	0
1.2.6 (Q29) Suitability of quantitative techniques	3	0.36	-	3	0.47	-	2	0.36	-
MV.30 Qualitative techniques	-	-	0	-	-	0	-	-	2
1.2.7 (Q31) Applicability of existing tools	3	0.30	-	3	0.47	-	3	0.30	-
MV.11 Organizational immaturity	-	-	0	-	-	2	-	-	1
1.2.8 (Q33) Missing steps	-	-	-	-	-	-	-	-	-
MV.08 Willingness to modularization	-	-	1	-	-	0	-	-	0
MV.16 Future customers' needs	-	-	0	-	-	1	-	-	0
MV.17 Life-cycle of competing alternatives	-	-	0	-	-	1	-	-	0
MV.18 Modularization as function of production volume	-	-	0	-	-	1	-	-	0
MV.19 Regulatory standards	-	-	0	-	-	1	-	-	0
MV.20 Management of change	-	-	0	-	-	1	-	-	0
MV.21 Strategic pricing definition	-	-	0	-	-	1	-	-	0
MV.22 Configuration management	-	-	0	-	-	1	-	-	0
MV.27 Technological trends	-	-	0	-	-	1	-	-	0
MV.31 Modularity maturity level	-	-	0	-	-	0	-	-	1
1.3 Artifacts' evaluation	-	-	-	-	-	-	-	-	-
1.3.1 (NA) Artifact's evaluation	-	-	-	-	-	-	-	-	-
MV.33 Practical application	-	-	0	-	-	0	-	-	3

(continued)

Table 29. (continued).

Top term / Construct / Dimension / Moderating variable	Cycle 2 Students (n=8)			Cycle 3 Experts (n=10)			Cycle 4 Scholars (n=5/9)		
	x̃	k_{free}	f	x̃	k_{free}	f	x̃	k_{free}	f
2.0 Practical relevance	**3**	**0.70**	**-**	**3**	**0.58**	**-**	**3**	**0.25**	**-**
2.1 General utility	*3*	*0.70*	-	*3*	*0.58*	-	*3*	*0.25*	-
2.1.1 (Q34) Customers' choice modeling	3	0.36	-	3	0.30	-	3	0.40	-
MV.05 Uncertainty of estimated data	-	-	1	-	-	0	-	-	0
MV.13 Future customers' needs	-	-	0	-	-	2	-	-	1
2.1.2 (Q36) Market-driven variants	3	1.00	-	3	1.00	-	3	0.40	-
2.1.3 (Q38) Balance of market needs and profitability	3	1.00	-	3	0.30	-	3	0.40	-
MV.05 Uncertainty of estimated data	-	-	0	-	-	1	-	-	0
MV.13 Future customers' needs	-	-	0	-	-	0	-	-	1
MV.23 Fixed costs	-	-	0	-	-	1	-	-	0
MV.26 Uncertainty of cost estimation	-	-	0	-	-	0	-	-	1
MV.32 Uncertainty of market size estimation	-	-	0	-	-	0	-	-	1
2.1.4 (Q40) Product family economic potential	3	0.20	-	3	0.20	-	3	0.10	-
MV.05 Uncertainty of estimated data	-	-	2	-	-	2	-	-	1
MV.13 Future customers' needs	-	-	0	-	-	0	-	-	1
MV.17 Life-cycle of competing alternatives	-	-	0	-	-	1	-	-	0
MV.23 Fixed costs	-	-	0	-	-	1	-	-	0
MV.24 Price of competing alternatives	-	-	0	-	-	1	-	-	0
MV.26 Uncertainty of cost estimation	-	-	0	-	-	1	-	-	0
MV.32 Uncertainty of market size estimation	-	-	0	-	-	0	-	-	1
2.1.5 (Q42) Trade-off between variety and cost	3	1.00	-	3	0.70	-	2	-0.20	-
2.1.6 (Q44) Utility	3	0.63	-	3	1.00	-	3	0.40	-
MV.09 Method's complexity	-	-	2	-	-	0	-	-	2
MV.25 Clarify the design strategy	-	-	0	-	-	1	-	-	0
MV.34 Static scenarios under a deterministic perspective	-	-	0	-	-	0	-	-	1

5.5.3 Results of Evaluation Cycle 5

In cycle 5, the MDM was used to conceptually design a family of collaborative robotic palletizers. First, collaborative robotics was identified as a technological trend in the palletizer market. Then, future opportunities related to it guided the definition of six target market niches (Ms) and its respective parameters. The total market size (M_k), in terms of palletizing positions a year, and the share of each market niche (S_{Ms}) was estimated according to Table 30. At this stage, an expected profit $V_e \geq 2\text{x}10^6\ [USD/year]$ was defined as the threshold that would justify the investment in the product family design. After that, the customer's choice probabilities (Pr_i) and the modular product family architecture were modeled. Coupled with that, the design parameter instances (DPI) along with its respective engineering attribute values (E_v) and variables cost (C_v) were defined, i.e. $\overrightarrow{DPI_\iota} = \{E_{vi}, C_v\}$. As a result, 9 functional modules (FM) and 110 design parameter instances gave rise to a design space overcoming millions of potential product family variants, as shown in Figure 35 (a). Then, the MDM's configuration model (Figure 33) was used to select the most profitable variants, one for each target market niche, as presented in Table 30. The last row of this table shows the results of the product family 2 ($PF2$) covering all niches together. The maximum profit found, $V = 11.9\text{x}10^6\ [USD/year]$, was higher than the expected profit $V_e = 2\text{x}10^6\ [USD/year]$, indicating in this way that it would be worth it to invest in the product family design.

Table 30. Results of the configuration process of product family 2.

Ms	Product family variant ($\vec{X}$)	M_k	S_{Ms}	Pr	Q	C_v [USD/SKU]	P [USD/SKU]	V_{Ms} [USD/year]
Ms1.1	[DPI1.5, DPI2.2, DPI5.2, DPI4.7, DPI11.1, DPI7.3]	773	10.0%	13.9%	11	36,650.50	157,500.00	1,329,344.50
Ms1.2	[DPI1.5, DPI2.2, DPI5.18, DPI4.7, DPI11.1, DP12.1, DPI7.3]	773	7.0%	17.9%	9	44,850.50	157,500.00	1,126,495.00
Ms1.3	[DPI1.5, DPI2.2, DPI5.38, DPI4.21, DPI11.1, DPI7.3]	773	3.0%	17.4%	4	46,255.50	157,500.00	444,978.00
Ms2.1	[DPI1.5, DPI2.2, DPI5.32, DPI4.20, DPI11.1, DPI12.4, DPI7.1, DPI8.4, DPI9.1, DPI10.1, DPI6.3, DP13.1, DP3.1]	773	38.0%	19.2%	56	170,480.50	245,000.00	4,173,092.00
Ms2.2	[DPI1.5, DPI2.2, DPI5.32, DPI4.20, DPI11.1, DPI12.4, DPI7.1, DPI8.1, DPI9.1, DPI10.1, DPI6.3, DP13.1, DP3.1]	773	26.0%	19.2%	39	170,293.00	245,000.00	2,913,573.00
Ms2.3	[DPI1.5, DPI2.2, DPI5.39, DPI4.20, DPI11.1, DPI12.4, DPI7.1, DPI8.1, DPI9.1, DPI10.1, DPI6.3, DPI13.1, DPI3.1]	773	11.0%	25.2%	21	167,793.00	260,000.00	1,936,347.00
PF2	[DPI1.5, DPI2.2, DPI5.2, DPI5.18, DPI5.32, DPI5.38, DPI5.39, DPI4.7, DPI4.20, DPI 4.21, DPI11.1, DPI12.1, DPI12.4, DPI7.1, DPI7.3, DPI8.1, DPI8.4, DPI9.1, DPI10.1, DPI6.3, DPI13.1, DPI3.1]	773	95.0%	18.2%	141	20,748,670.50	32,672,500.00	11,923,829.50

With the product family profit considered satisfactory, the design parameter instances compounding the most profitable variants were selected to integrate the final structure of family 2. The solution compound by 9 physical modules (M), 22 design parameter instances, and capable of generating up to 120 variants is shown in Figure 35 (b). According to the MDM proposition, this is the product family structure that better balances the fulfillment of market needs and the resulting profitability to achieve them. Therefore, is the one that should be developed in the subsequent design stages of the product development process.

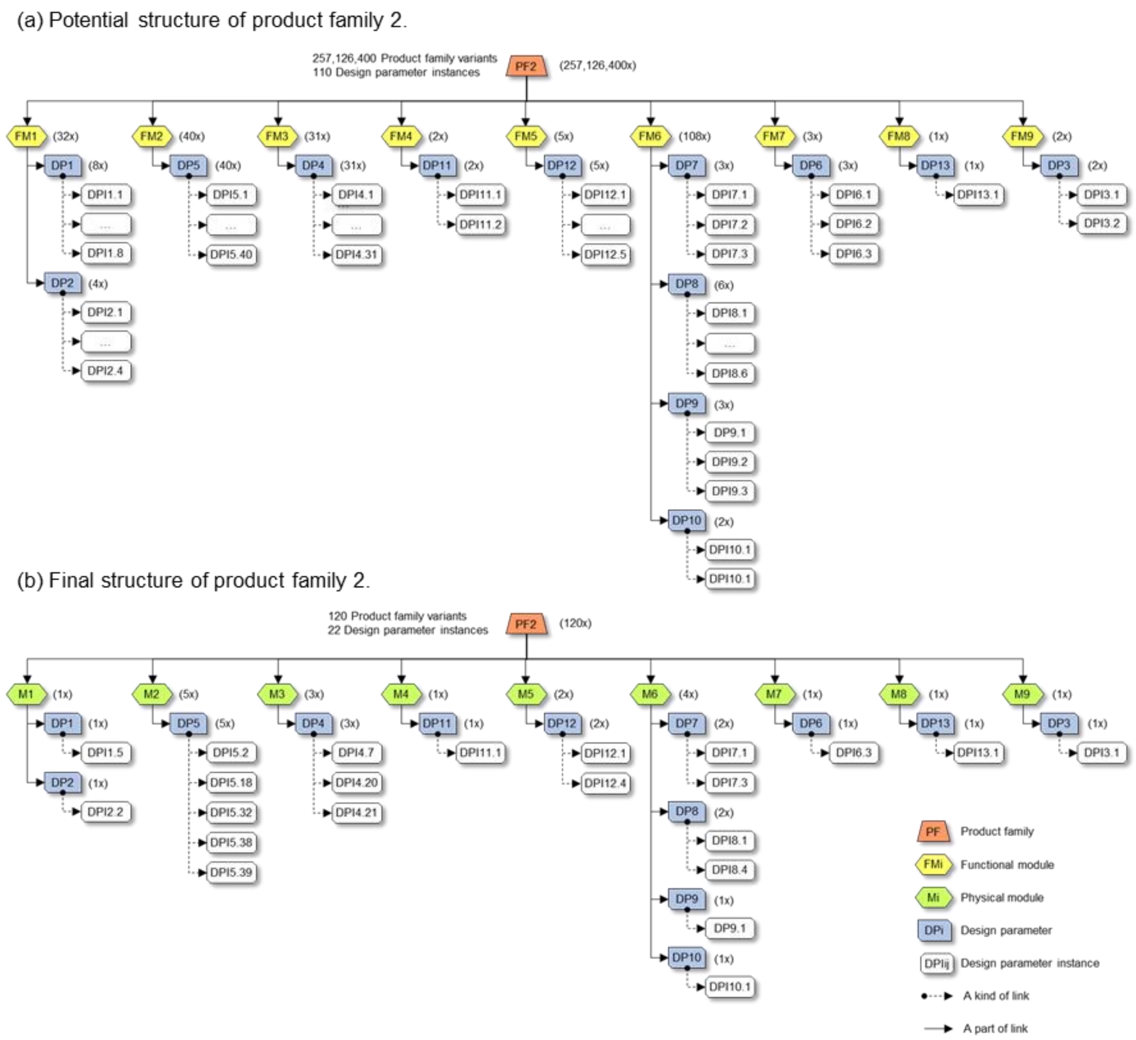


Figure 35. (a) Potential structure of product family 2; (b) Final structure of product family 2.

To assess the limits of MDM's configuration model, a sensitivity analysis was performed guided by the following question: What are the values of the influencing variables (Var) that would keep the MDM outcomes the same? To answer this question, 8 scenarios were evaluated in comparison to the one obtained in the last row of Table 30 (Scenario 0), as shown in Table 31. The strategy adopted here was to change one influencing variable at a time while keeping the others constant.

Table 31. Sensitive analysis.

Scenario	Var	ΔVar	M_k	S_{Ms}	Pr	Q	C_v [USD]	P [USD]	V [USD/year]	ΔV
0 - Highest price (base)	P	-	773	95%	18.2%	141	20,748,670.50	32,672,500.00	11,923,829.50	-
1 - Minimum price	P	-14.3%	773	95%	19.4%	150	21,674,605.00	29,607,857.14	7,933,252.14	-33.5%
2 - Minimum market size	M_k	-10.2%	694	95%	18.2%	128	18,726,009.00	29,632,500.00	10,906,491.00	-8.5%
3 - Different share of market niches	S_{Ms}	random	773	95%	16.6%	127	13,898,271.00	26,177,500.00	12,279,229.00	3.0%
4 - Variation in the DPI's variable cost	C_v	±97.0%	773	95%	18.2%	141	17,661,010.43	32,672,500.00	15,011,489.57	25.9%
5 - Variation in the DPI's engineering attribute values	E_v	±6.0%	773	95%	18.3%	142	20,919,151.00	32,917,500.00	11,998,349.00	0.6%
6 - Variation in the engineering attribute weights	w	±40.0%	773	95%	18.1%	140	20,578,377.50	32,427,500.00	11,849,122.50	-0.6%
7 - Variation in the engineering attribute values of competing alternatives	E_v	±13.0%	773	95%	18.2%	141	20,748,858.00	32,672,500.00	11,923,642.00	0.0%
8 - Addition of one more competing alternative within each market niche	J	+20.0%	773	95%	15.1%	116	17,141,865.50	26,925,000.00	9,783,134.50	-18.0%

The results indicated that the product family structure is more sensitive than the decision on investment in the product family design. In other words, it was the product family structure that limited the increase of ΔVar. Also, the variable that influenced it the most was the engineering attribute value (Ev), since a small change on it (±6.0%), would originate different structural solutions. The decision on investment in the product family design, in turn, is most influenced by the price (P) followed by the variable cost (Cv). It can be seen by its respective variations in profit (-33.5% and +25.9%) presented in Scenarios 1 and 4. These results indicate, that even from a deterministic perspective, the outcomes of the MDM are reasonably stable. The learnings achieved during this evaluation cycle, and how they influenced the MDM updates are given in Table C1. At the end of these five cycles, the MDM was considered ready to be used on a large scale.

5.5.4 Construction and Contingency Heuristics

The construction heuristic consists of the set of design rules leading to the proper functioning of the artifact's internal environment. The contingency heuristic, in turn, defines the artifact's limits and its using conditions regarding the external environment (Dresch, Lacerda and Antunes Jr, 2015). To synthesize both, the CIMO-logic was used, as shown in Table 32. The reasoning behind CIMO is: For this problem-in-Context, it is useful to use this

Intervention, which will produce through these Mechanisms this Outcome (Denyer, Tranfield and Van Aken, 2008). In this sense, the contingency heuristic has to do with the Context, while the construction heuristic relates to the Intervention and Mechanism. Coupled with that, two more variables were considered, the Evidence and Limitations.

Table 32. Market-Driven Modularity boundaries.

	Artifact: Market-Driven Modularity (MDM)	Evidence
Context	Design lucrative product families regarding the following dimensions:	
	D1 Company size (see LM1):	
	D1.1 Small;	Cycles 2 and 5
	D1.2 Midsize;	Cycles 1 and 2
	D1.3 Large.	Cycles 3 and 4
	D2 Production strategy (see LM2):	
	D2.1 Make-to-stock (MTS);	Cycles 1, 2 ,3 and 4
	D2.2 Assemble-to-order (ATO);	Cycles 2 ,3 and 4
	D2.3 Make-to-order (MTO).	Cycles 2, 3, 4 and 5
	D3 Product development phase:	
	D3.1 Planning;	Cycles 1, 2, 3 and 5
	D3.2 Conceptual design;	Cycles 1, 2, 3 and 5
	D3.3 System-level design.	Cycles 1, 2, 3 and 5
	D4 Product type (see LM3 and LM4):	
	D4.1 Consumer durables;	Cycles 1, 2, and 3
	D4.2 Intermediate goods;	Cycles 2, and 3
	D4.3 Capital goods.	Cycles 2, 3, and 5
	D5 Market amplitude (see LM5):	
	D5.1 Single segment;	Cycles 2 and 4
	D5.2 Multiple segments.	Cycles 1, 2, 3 and 5
	D6 Design strategy:	
	D6.1 Redesign of existing families;	Cycle 2
	D6.2 Design of new families;	Cycles 1 and 2
	D6.3 Design of new modules;	Cycle 5
	D6.4 Design of new generations of families.	Cycle 5
	D7 Data availability (see LM6):	
	D7.1 Low data availability;	Cycles 1, 2 and 5
	D8 Data behavior:	
	D8.1 Deterministic.	Cycles 1, 2 and 5
	D9 Scenario behavior:	
	D8.2 Single and static.	Cycles 1, 2 and 5
Intervention	I1 Strategically plan the product family positioning:	
	I1.1 Define the technological trends for a specific target market;	Cycle 5
	I1.2 Identify future product opportunities related to technological trends;	Cycle 5
	I1.3 Estimate the potential market size;	Cycles 1 and 5
	I1.4 Segment the market and estimate its respective shares;	Cycles 1 and 5
	I1.5 Stablish the product family leveraging strategy;	Cycles 1 and 5
	I1.6 Define the expected profit that justifies the investment in the product family.	Cycles 5
	I2 Model the customers' choice:	
	I2.1 Define the customer-related engineering attributes;	Cycles 1 and 5
	I2.2 Select the potential competing alternatives for each segment;	Cycles 1 and 5
	I2.3 Identify the engineering attribute values and price for each alternative;	Cycles 1 and 5
	I2.4 Capture customer preferences;	Cycles 1 and 5
	I2.5 Model the customer's choice for each target market segment.	Cycles 1 and 5

(continued)

Table 32. (continued).

	Artifact: Market-Driven Modularity (MDM)	Evidence
Intervention	I3 Model the product family:	
	I3.1 Formulate the design parameters;	Cycles 1 and 5
	I3.2 Define the modular product family architecture covering all segments;	Cycles 1 and 5
	I3.3 Specify the deign parameter instances in terms of engineering attribute values and variable cost;	Cycles 1 and 5
	I3.4 Stablish the design rules for product family configuration.	Cycles 1 and 5
	I4 Configure the product family structure:	
	I4.1 Define the number of variants to be configured to each segment;	Cycles 1 and 5
	I4.2 Configure the variants and set the price;	Cycles 1 and 5
	I4.3 Calculate the demand and profit for each variant;	Cycles 1 and 5
	I4.4 Select the most profitable variants and aggregate its partial profits into the product family profit;	Cycles 1 and 5
	I4.5 Compare the product family profit with the expected profit;	Cycle 5
	I4.6 Build the final product family structure with the design parameter instances compounding the most profitable variants of each segment.	Cycles 1 and 5
Mechanism	M1 The MDM's internal environment (26 steps arranged in 4 classes of design problems, that can be performed by, at least, 38 techniques).	Cycles 1, 2, 3, 4, and 5
Outcomes	O1 Decision on investing or not in the product family design;	Cycle 5
	O2 The product family structure that better balance the fulfillment of market needs and the resulting profitability to achieve them.	Cycles 1 and 5
Limitations	LM1 The MDM is more suitable to be adopted by large companies due to the required knowledge base and organizational structure;	Cycles 2 and 3
	LM2 The MDM is not adequate for those companies, where the orders are not typically repeated on a large scale, i.e. ETO;	Cycle 4
	LM3 The potential inability of modularity in providing aesthetics variety, an attribute deeply required in consumer durables. This issue led us to think that MDM is more suitable for designing "low dependent product parts", an expression used in the automotive industry to refer to those parts not related to the product style;	Cycle 3
	LM4 The MDM is more suitable to design products of any complexity but at high granularity levels;	Cycles 4 and 5
	LM5 The low heterogeneity of a single market segment might reduce the effect of MDM;	Cycle 3
	LM6 Although the MDM has been designed to be used in contexts of high data availability, it has not been empirically tested in this situation so far. Therefore its suitability for this context is theoretically grounded on previous works such as (Kumar, Chen and Simpson, 2009; Chen, Hoyle and Wassenaar, 2013).	Previous studies

5.6 Discussion of the Results

In evaluation cycles, 2, 3, and 4 the opinions of students, experts, and scholars were captured to measure the pragmatic validity and practical relevance. In this context, it was noted that, although the amplitude of agreement reached the highest value $(\tilde{x} = 3)$ in the three cycles, the level of agreement among raters (k_{free}) decreases from cycle 2 to 4, as shown in Figure 36. We believe that two potential reasons might explain this behavior. The first has to do with the ability to critically analyze a subject, which increases as more is known about it. The second relates to the time employed to understand a situation, which implies a reduction of understanding as less contact you have with it. In cycles 2, 3, and 4, the knowledge about the product family design increased along the cycles, and the time exposure to understand the proposed method, decreased from 12 hours in cycle 2, passing to 1 hour in cycle 3, achieving 15 min. in cycle 4.

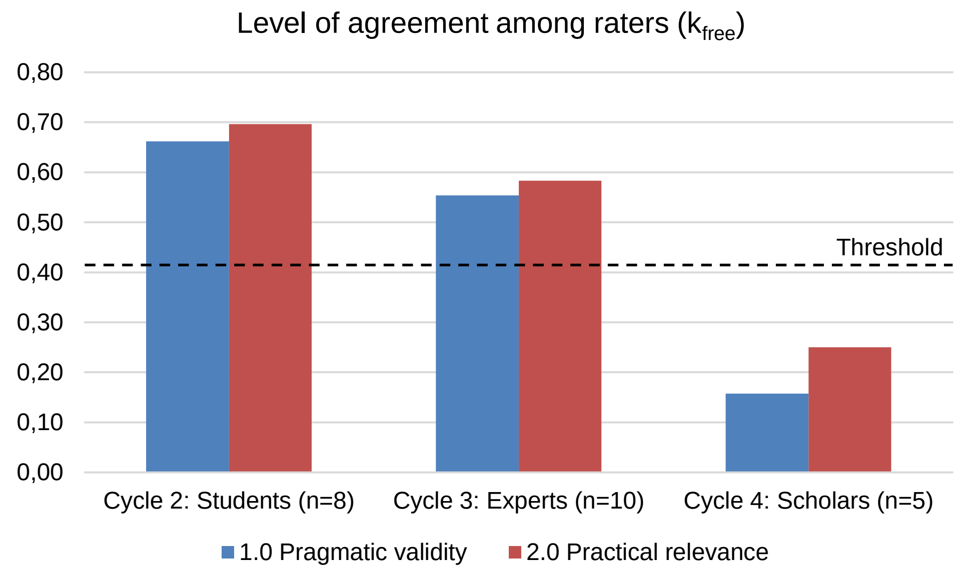


Figure 36. Comparison of the level of agreement among raters (k_{free}).

We do not know which of these two reasons determined this pattern, but we advise, for future research, to keep the same procedure and time exposure for, at least, the experts and scholars. Another interesting issue, brought by three scholars who evaluated the MDM was that, in their opinion, the empirical application of the artifact, either in made up or real cases, is more substantial than the experts' judgment (see learning 24 of Table C1). However, depending on the boundary conditions of the artifact being developed, it might be unfeasible to test it in all situations required. For that reason, this research combined both, the empirical application along with experts' judgment to produce valuable additional information on pragmatic validity and practical relevance (van Aken, Chandrasekaran and Halman, 2016). Even with that strategy, there was a situation (see LM6 Table 32) in which the MDM could not be tested, having its applicability supported by previous research. Finally, this work performed 5 cycles until it reaches a satisfactory solution, however, it may vary depending on the research scope. Thus, for future research, we suggest changing the number of cycles, in the conditional decision at the end of stage 8, to the question mark *Saturation reached?*, referring to the instance of the law of diminishing returns (Eisenhardt, 1989).

5.7 Conclusions

This paper used design science research (DSR) to integrate marketing, engineering, and economic domains into a single approach to design lucrative product families. In this sense, the traditional stages of DSR methodologies were decomposed into 32 steps to provide practical guidance on the artifacts' design and evaluation. By following these steps, a field problem gave rise to a method, entitled Market-Driven Modularity (MDM), which was validated through a series of practical applications and experts' judgments. The main contributions of this research include: (i) The systematic integration of four classes of design problems prevalent in literature into a single method to conceptually design lucrative product families. (ii) The proposition of an open architecture of techniques to execute each step of the method in contexts from low to high data availability. (iii) The introduction of Functional to Physical Decomposition, an approach to deal with functional and physical modularity in product family architectures. (iv) The presentation of practical guidance on the artifact's design and evaluation. (v) The usage of a quantitative approach to measure the pragmatic validity and practical relevance. Finally, (vi) the MDM itself as the first method to design modular product families, developed under the design science paradigm. Regarding the limitations, the first one lies in the different procedures and time exposure adopted in evaluation cycles 2, 3 and 4, as described in the last section. The second has to do with the fact of not having tested the MDM in the context of high data availability. The third relates to the fact of only testing the MDM in made-up cases and not in real ones. Although we believe, these limitations did not affect the quality of results obtained, they are issues to overcome in future studies. In terms of future research directions, we identified opportunities in three fields of study: product family design, modularity, and DSR. In product family design, an issue to investigate is how to design lucrative product families for multiple dynamic scenarios under uncertainty. Concerning modularity, the sufficiency of the current definition of modular architectures when the intensity of relationships is considered, and the potential limitation of modularity in dealing with aesthetics variety, configure two topics of study. Finally, regarding DSR, the levels of artifacts' evaluation which lead to satisfactory results in terms of pragmatic validity and practical relevance must be better understood.

6 ARTICLE 4 - MARKET-DRIVEN MODULARITY: AN INTEGRATED METHOD TO CONCEPTUALLY DESIGN MODULAR PRODUCT FAMILIES [4]

[4] Article to be submitted to the Journal of Intelligent Manufacturing (JIM).

Abstract: This paper introduces the Market-Driven Modularity (MDM), an integrated method to conceptually design modular product families that balance the fulfillment of market needs and the resulting profitability to achieve them. To do that MDM uses discrete choice modeling for quantifying the customers' preferences, modularity as a mechanism to provide product variety, product family as a strategy to manage the trade-off between the variety and cost, and profit as a moderating variable to balance the level of accomplishment of customers' needs. To provide a better understanding of the proposed method, this paper presents an illustrative application of the MDM within the development process of a family of collaborative robotic palletizers for multiple market segments. The results indicate, that even from a deterministic perspective and under a context of low data availability, the two MDM outcomes, a lucrative product family structure, and the decision on investment in the product family design, are reasonably stable.

Keywords: modularity; product family design; choice modeling.

6.1 Introduction

Based on the belief the product variety can positively influence sales and profits, many companies have been attempting to accommodate the ever-increase diversity of customer preferences on its product offerings without sacrificing production efficiency (Zhu, Li and Feng, 2017). In industry and academy alike, this issue has been addressed by two complementary, but still not integrated, approaches: the product line planning and product family design (Miao *et al.*, 2017). The product line planning consists of optimally selecting the group of products to be marketed to one specific market (Kahn, 2012), while the product family design consists of designing a set of products sharing common elements yet target different market segments (Simpson *et al.*, 2014).

Although numerous product line planning methods in management science and marketing literature deal with the selection problem using various objectives derived from profit, few of them explicitly address product design details not directly perceived by customers (Jiao, Simpson and Siddique, 2007). These approaches normally assume that any combination of product attributes can somehow be attained by design engineers post hoc (Michalek *et al.*, 2011). In contrast, most existing product family design approaches are targeted at identifying an optimal commonality decision in order to minimize cost while meeting pre-specified performance tiers (Kumar, Chen and Simpson, 2009). As a consequence, they do not sufficiently examine broader business indicators such as demand and profit (Michalek *et al.*, 2011).

Addressing front-end issues in product family design is a complex activity (Colombo *et al.*, 2019), which, in general, can be subdivided into four prevalent classes of design problems: (i) product family positioning, (ii) market-driven product family design, (iii) product family modeling, and (iv) product family configuration (Gauss, Lacerda and Miguel, 2020). The first two classes account for the marketing-related issues, which include customer involvement, product portfolio design, product family positioning, and transition or mapping from customer needs to functional requirements (Simpson *et al.*, 2014). While the last two classes are grounded on engineering-related issues, which include the product family configuration, product architecture, design of families and platforms, leveraging commonality and modularity, and optimization of the family and platform design (Simpson *et al.*, 2014). A recent study, concerning 72 methods for designing module-based product families, has shown that 1.4% of methods address the four classes of design problems concurrently. Among those methods

(41.7%) considering marketing-related issues in its formulation, less than 7% derive the desired attributes in a product straight from the customers. Still from this study, it is seen that only 15.3% of methods account for enterprise-level indicators in product family configuration (Gauss, Lacerda and Miguel, 2020) (Article 1). Findings that comply with other research indications of lacking methods integrating marketing, engineering, and economic issues into product family design (Jiao, Simpson and Siddique, 2007; Kumar, Chen and Simpson, 2009; Colombo *et al.*, 2019).

The problem is the marketing and engineering variables are often highly interdependent in product family design. Moreover, the coupled relationships between them imply that any change in one variable can potentially influence the outputs of the other(s), with both affecting the economic benefits of an enterprise (Chen, Hoyle and Wassenaar, 2013). Therefore, in the design of optimal or near-optimal product families, marketing, engineering, and economic requirements often cannot be pursued separately or even sequentially (Luo, 2011).

In the light of these previous research indications, this paper introduces the Market-Driven Modularity (MDM), an integrated method to conceptually design modular product families that balance the fulfillment of market needs and the resulting profitability to achieve them. In this sense, the MDM aims to prevent the development of non-profitable product families deriving from the missing link between marketing and engineering domains. Moreover, it intends to overcome this problem by not only considering the interrelationships between customer preferences and engineering feasibility into product family design but also accounting for its respective influence on enterprise-level indicators such as demand, price, and profit. To do that MDM uses discrete choice modeling for quantifying the customers' preferences (Chen, Hoyle and Wassenaar, 2013), modularity as a mechanism to provide product variety (Ulrich and

Tung, 1994), product family as a strategy to manage the trade-off between the variety and cost (Simpson, 2004), and profit as a moderating variable to balance the level of accomplishment of the customers' needs (Kumar, Chen and Simpson, 2009).

The main contributions of this research include the systematic integration of four classes of design problems prevalent in the literature into a single method to conceptually design lucrative product families. The introduction of Functional to Physical Decomposition, an approach to deal with functional and physical modularity in product family architectures, and the presentation of a heuristic to estimate the product's variable costs at early design stages.

The remainder of this paper is structured as follows. Section 6.2 synthesizes the related research on product family design. Section 6.3 describes the MDM external and internal environment. Section 6.4 presents an illustrative application of the MDM within the development process of a family of collaborative robotic palletizers. Section 6.5 critically analyses the MDM outcomes. Finally, the last section provides the research contributions and limitations as well as its future directions.

6.2 Related Work

The product family design is an effective strategy to provide variety at a reduced cost (Simpson *et al.*, 2014). Generally speaking, a product family refers to a set of products derived from a standard product platform to satisfy various market applications (Meyer and Lehnerd, 1997). Platforms, in turn, are intellectual and material assets shared across a family of products, to minimize manufacturing complexity (Erens and Verhulst, 1997). In this context, the prominent approach to product family design is through the development of module-based product families, wherein product family members are instantiated by mixing and matching functional modules from the platform

(Ulrich, 1995; Du, Jiao and Tseng, 2001). An alternative approach, considered as a subset of the former (Fujita and Yoshida, 2004), is through the development of a scale-based product family, which consists of scaling one or more variables to change the platform specifications while common parameters remain constant (Simpson, 2004).

The product family design is challenging for many aspects. It involves selecting business strategies, considering multiple marketing issues, engineering customer needs, studying customer behavior and choice-related issues, as well as carefully considering engineering aspects of design, such as manufacturability, technological aspects, and design support issues (Simpson *et al.*, 2014). In general, these problems can be grouped into four prevalent classes: (i) Product family positioning, which aims at maximizing customers' preferences with the lowest number of variants. (ii) Market-driven product family design, that deals with the transition of customers' needs to functional requirements. (iii) Product family modeling, which comprehends the definition of modules and platforms. Finally, (iv) product family configuration, wherein the modules compounding the variants are optimally selected (Jiao, Simpson and Siddique, 2007).

Over the years, active work in developing methods to design product families has been done (Borjesson and Hoelttae-Otto, 2014; Otto *et al.*, 2016). Among those methods related to this study, the one encompassing four classes of design problems is the work of Jiang and Allada (2005). However, this method assumes the modules' set already exists, being deeply sensitive to the ability of extant modules in accomplishing the customer desired attributes. Besides that, the product family configuration is used to configure one variant at a time instead of building an optimal or near-optimal product family structure. In like manner, other methods only entail the three first classes of design problems (Jiao and Tseng, 1999a; Asan, Polat and Serdar, 2004; Hsiao and Liu, 2005; Kazemzadeh *et al.*, 2009; Hsiao *et al.*, 2013; Sahin-Sariisik *et al.*, 2014; Ma and

Kim, 2016; Pakkanen, Juuti and Lehtonen, 2016). But the main limitation of them lies in the inability to optimally or near-optimally combine the designed modules into product family variants or even selecting the most adequate ones to compose the product family structure.

There is another group of methods, encompassing the product family modeling, which focuses on modules identification (Thevenot *et al.*, 2007; Arciniegas and Kim, 2011; Agard and Bassetto, 2013; AlGeddawy and ElMaraghy, 2013; Li *et al.*, 2013; Borjesson and Hoelttae-Otto, 2014; Aydin and Ulutas, 2016; Ma *et al.*, 2016; Hou *et al.*, 2017, 2018; Miao *et al.*, 2017). Within this group, a few methods, if any, perform the functional and physical decomposition concurrently. Besides that, these approaches occasionally measure the quality of the clustering solution, indicating in this way its open-loop nature. Still from this group, some approaches combine the product family positioning with product family modeling (ElMaraghy and AlGeddawy, 2012; Simpson *et al.*, 2012; Fan *et al.*, 2015; Miao *et al.*, 2017), while others combine the market-driven product family reasoning with the product family modeling (Dahmus, Gonzalez-Zugasti and Otto, 2001; Zhang, Tor and Britton, 2006; Du, Jiao and Tseng, 2006; Krishnapillai and Zeid, 2006; Meng, Jiang and Huang, 2007; Park *et al.*, 2008; Stone *et al.*, 2008; Bonjour *et al.*, 2009; Yan and Stewart, 2010; Emmatty and Sarmah, 2012; Yang, Yu and Jiang, 2014; Wei *et al.*, 2015; Jung and Simpson, 2016; Cheng *et al.*, 2017; Bejlegaard *et al.*, 2018; Wang *et al.*, 2018; Gauss, Lacerda and Sellitto, 2019). In both, less than a quarter, derive the customer desired attributes straight from themselves.

The last group of methods focuses on the product family configuration. More specifically in the process of mixing, matching, and scaling modules to generate product family variants (Tucker and Kim, 2008; Jiao, 2012; Pate, Patterson and German, 2012; Hanafy and Elmaraghy, 2015; Goswami, Daultani and Tiwari, 2017; Xiao *et al.*, 2018).

In this group, the major part, solve the combinatorial and parametric problem through meta-heuristics and some use enterprise-level indicators to compound the objective function. Some methods also consider the product family design and configuration being performed together (Rai and Allada, 2003; Li, Huang and Newman, 2008; Li and Huang, 2009; Dong, Shao and Xiong, 2011; Chowdhury *et al.*, 2016; Baylis, Zhang and McAdams, 2018; Colombo *et al.*, 2019). However, they assume the modules' set already exists, and use the configuration process to generate product family variants instead of building product family structures. Additionally, nor a threshold to evaluate if the variants instantiated satisfy the desired attributes in a product, neither feedbacks leading to new modules' developments are found. Moreover, it is not explicit in these works, the product family configuration supporting or even playing the role of product line planning, an issue that has been traditionally dealt with in the management science and marketing literature (Jiao, Simpson and Siddique, 2007).

In summary, there is a lack of integrated approaches modeling the customers' preferences and using it to design and configure gainful product family structures, the gap this paper aims to overcome.

6.3 Proposed Method: Market-Driven Modularity (MDM)

The MDM consists of an integrated method to conceptually design market-driven product families. Regarding its external environment (Dresch, Lacerda and Antunes Jr, 2015), the MDM is intended to be adopted in the early design stages of the product development process of small, midsize, and large companies that produce consumer (durables), intermediate, and capital goods. Besides that, the MDM has been developed to redesign the existing families from a modular point of view as well as to design new modules, new families, and new generations of families, in contexts from low to high data availability as illustrates the Figure 37.

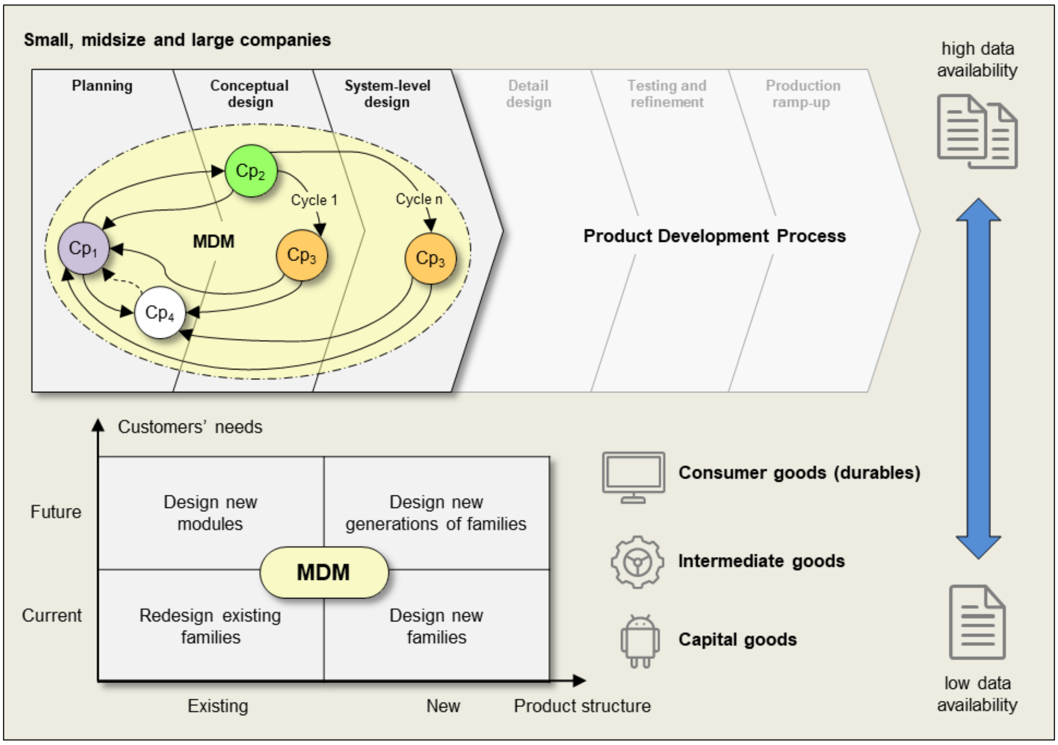


Figure 37. External Environment of Usage of MDM.

Its internal functional environment (Simon, 1996) is composed of 26 steps, arranged in 4 classes of design problems, that can be performed by, at least, 38 techniques. The reasoning behind the method is to define the target market segments, model the customers' choice probabilities for each of them, and then define a modular product family architecture, corresponding to all segments. With the product family architecture defined, the design parameter instances are generated and combined into a finite set of variants for each segment. Then, after setting the price, the demand is estimated, and the resulting profit of each variant is calculated. The most profitable variants have their gain aggregated into the product family profit, and the design parameter instances compounding them are selected to integrate the physical modules of the product family structure. If the product family profit matches the expected profit, the process is finished. Otherwise, the process should be restarted until the product family reaches the desired gain or until it is discarded. The two expected MDM outcomes are:

(i) the modular product family structure that better balances the fulfillment of market needs and the resulting profitability to achieve them, (ii) and the decision on investing or not in the product family design. Figure 38 presents an overview of the MDM proposition, while Figure 39 shows its internal functional environment in depth.

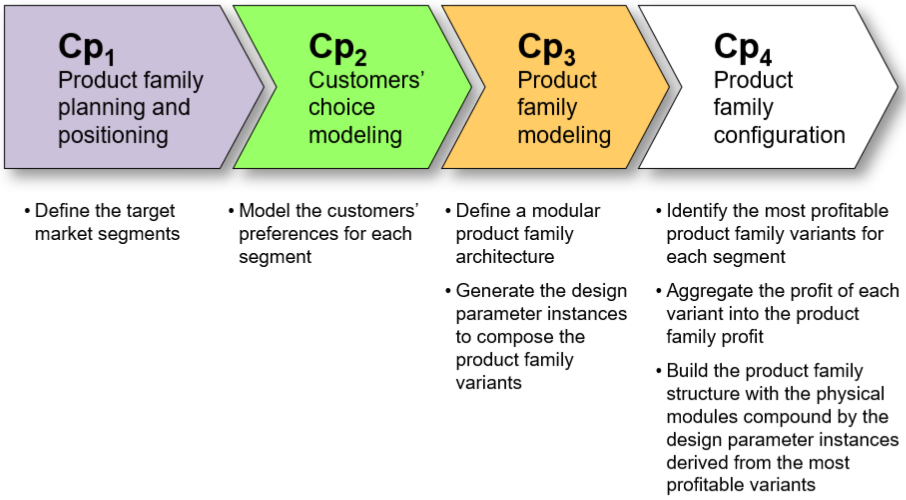


Figure 38. Overview of the MDM proposition.

From a more detailed perspective, the MDM method starts by converting the corporate strategy into objective measures for product family design. This process takes place at the first class of design problems, named here as Product Family Planning and Positioning (Cp_1). Within this class, at step $S_{1.1}$, the potential market size (M_k) and the expected profit (V_e) are estimated. Besides that, the target market segments (Ms), the technological trends and the product family leveraging strategy are also established. In the next step $(S_{1.2})$, the market segmentation is refined, and the resulting specifications serve as an input flow for identifying the customer desired attributes (A) at stage $S_{2.1}$, or, as feedback for improving the strategic product family planning at stage $S_{1.1}$.

The identification of customers' desired attributes consists of the first step $(S_{2.1})$ of the second class of design problems entitled here as the Customers' Choice Modeling (Cp_2). These attributes might derive from current or future needs and despite its nature,

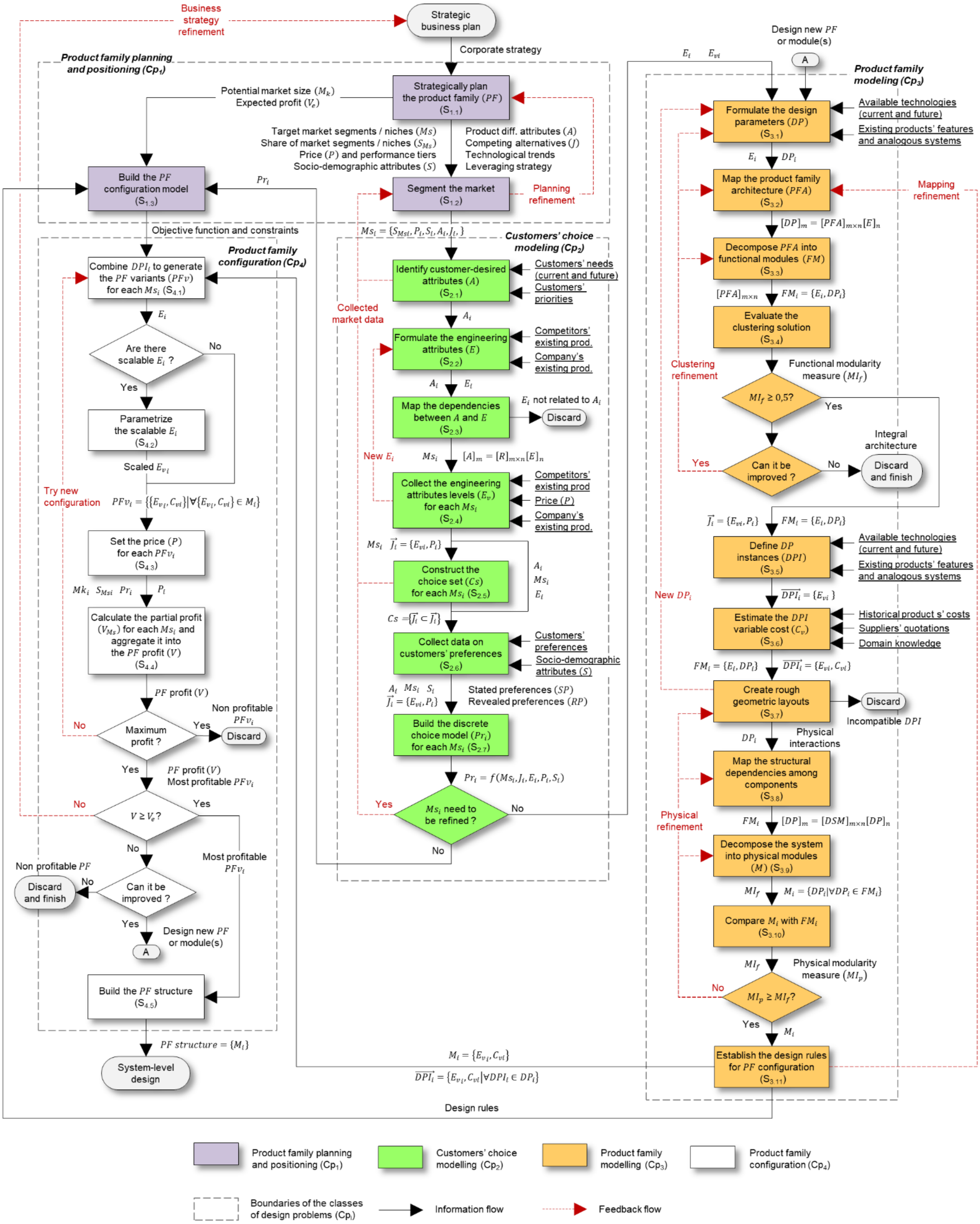


Figure 39. Internal Functional Environment of MDM.

they need to be converted into a language that engineers use to develop products. In some cases, the data gathered here ($S_{2.1}$) might be useful for refining the market segmentation at the previous step. The translation from customers' to engineering attributes (E) is performed at the stage $S_{2.2}$, and the relationship (R) between them is mapped in step $S_{2.3}$, i.e. $[A]_m = [R]_{m\times n}[E]_n$. Those customer-related engineering attributes should go forward to step $S_{2.4}$; otherwise, they should be discarded. In general, the engineering attributes might assume different levels within and across segments; for that reason, at stage $S_{2.4}$, a set of competing alternatives (J) for each segment is captured. Each competing alternative consists of a vector compound by engineering attribute values (E_v) and price (P), i.e. $\vec{J_\iota} = \{E_{vi}, P_i\}$. For those market pull product families, the competing alternatives usually derive from competitors or the company's existing products, while, for those technology push product families, the alternatives are deducted based on the product family planning and positioning. In both situations, the life-cycle of the competing alternative should be assessed before deciding if it is going to integrate the choice set. At this stage ($S_{2.4}$), new engineering attributes might emerge; in such cases, they serve as feedback for the step $S_{2.2}$. The next step is to define the set of alternatives by which the customers will state their preferences within each segment. Sometimes, the number of alternatives might be particularly high, difficulting in this way, the preference statement. When it happens, the choice set (Cs) must be reduced without losing the statistical significance in step $S_{2.5}$, i.e. $Cs = \{\vec{J_\iota} \subset \vec{J_\iota}\}$. In situations where the data reduction is not required, the step $S_{2.5}$ should be by-passed. With the choice set defined, the data on customers' preferences are collected in stage $S_{2.6}$. In contexts of low data availability, the key customers, or experts in the field, are asked to compare the customer desired attributes against each other for each target market segment. In contexts of high data availability, in turn, the customers are requested to pick an alternative from a choice set, emulating in this way, the real purchasing decisions within each segment. At stage $S_{2.7}$, depending on the technique used, the engineering attributes' coefficients (β), or weights (w), are estimated based on customers' stated/revealed preferences. Then, the utility (W) and the choice probability (Pr) of each alternative comprising the same target segment are modeled. If any deviation on market segmentation is found during the customers' choice modeling, the process should restart until the marginal difference become insignificant.

With the customer's choice modeled, the next issue is to define the product family architecture, decompose it into functional/physical modules, and then generate the design parameter instances that can potentially compose the product family structure. This process is performed in the third class of problems named here as Product Family Modeling (Cp_3). Within this class, at stage $S_{3.1}$, the process starts by formulating those logical entities with the ability to accomplish one or more engineering attributes. These logical entities are named here as design parameters (DP), and their formulation derives not only from the available technology and existing product features but also from future technology trends and analogy with other systems. Once defined, the design parameters are mapped to engineering attributes, giving rise to the product family architecture (PFA) in stage $S_{3.2}$, i.e. $[DP]_m = [PFA]_{m\times n}[E]_n$. The product family architecture defined here comprises all target segments together, and it should be designed to meet the functional independence axiom (Suh, 1998). Then, at stage $S_{3.3}$, the product family architecture is decomposed into functional modules (FM) and have its clustering solution evaluated at stage $S_{3.4}$, i.e. $FM_i = \{E_i, DP_i\}$. If the clustering solution accomplishes the desired level of functional modularity, i.e. $M_f \geq 0{,}5$, the process should go forward. Otherwise, the clustering refinement should be performed until it reaches the expected value or until an integral architecture is found, i.e. $M_f < 0{,}5$. In the last situation , the process should be finished as indicated by Figure 39. With the functional modules defined, the next issue is to specify the engineering attributes values (E_v) resulting from different physical characteristics that a design parameter might assume. This task of defining the design parameter instances (DPI) is performed at the stage $S_{3.5}$, i.e. $\overrightarrow{DPI_\iota} = \{E_{vi}\}$. Couple with that arises the trade-off between the engineering attribute levels and the costs to achieve it. For that reason, the variable cost (C_v) of each design parameter instance is estimated at stage $S_{3.6}$, i.e. $\overrightarrow{DPI_\iota} = \{E_{vi}, C_v\}$. The variable cost is assumed here to overcome the limitations of traditional cost accountability in dealing with product mix-related decisions (Cox and Schleier, 2010). After that, to identify the physical interactions within and across functional modules, rough geometric layouts are created in step $S_{3.7}$. In this step, just as incompatible design parameter instances may be discarded, new design parameters might emerge giving rise to feedback from here $(S_{3.7})$ to the stage $S_{3.1}$. The physical interactions resulting from this step, serve as an input flow for mapping the structural dependencies among design parameters in step $S_{3.8}$. Then, at stage $S_{3.9}$, the functional decomposition is transferred to the physical decomposition, and the relationship between these two modularity indices is evaluated at stage $S_{3.10}$. The reasoning here is that the physical modularity must not prevent the functional modularity as suggests Figure 40. If a modular architecture unconstrained by physical interactions is reached, i.e. $\left(MI_p \geq MI_f \,\middle|\, MI_f \geq 0{,}5\right)$, the process goes forward. Otherwise, the iterative refinement on physical modularity should be performed until it reaches the desired value. Finally, the design rules for product family configuration are defined at the stage $S_{3.11}$. It is also seen that when setting the configuration rules, modifications to the product family architecture may arise as indicated by the feedback to step $S_{3.2}$.

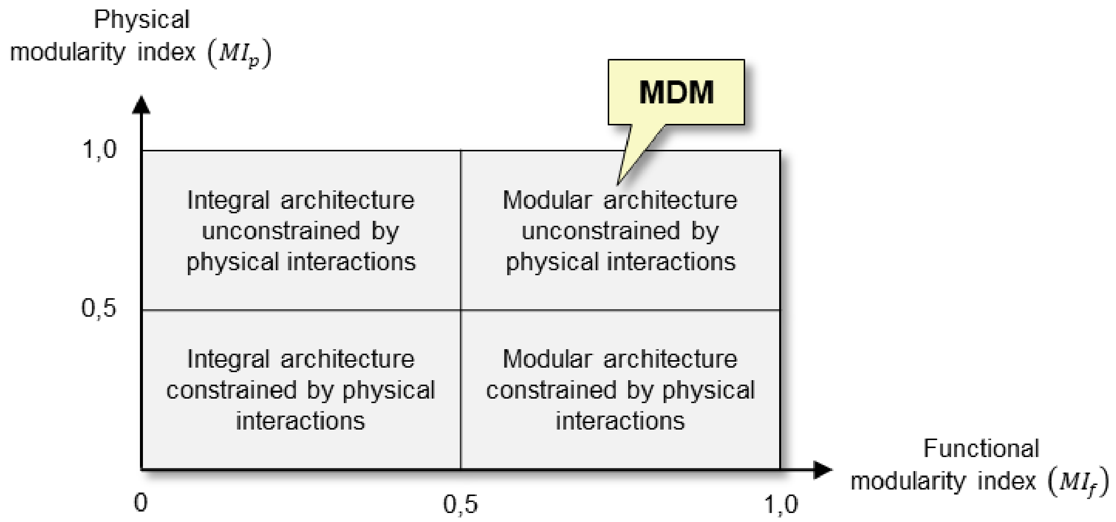


Figure 40. Relationship between functional (M_f) and physical (M_p) modularity indices.

Back to the first class (Cp_1), at stage $S_{1.3}$, the issue is to aggregate the customer's choice probability, the design rules, the set of design parameter instances, and the enterprise-level indicators (demand, price, and profit) into a single model for combining, selecting and parameterizing the design parameter instance to compound the modular product family structure. This configuration process takes place in the fourth class of design problems, named here as Product Family Configuration (Cp_4). At this class, the process starts at the stage $S_{4.1}$, where the design parameter instances are combined into family variants for each target market segment. If some design parameter instances contain scalable engineering attributes, its values are adjusted in the step $S_{4.2}$. With the product variant configured and parametrized, the price (P) is set at stage $S_{4.3}$, and its demand (Q) and partial profit (V_{Ms}) are calculated in the next step. The $S_{4.4}$ not only calculates the partial profit of each variant in its respective segment but also aggregates it into the product family profit (V). The steps from $S_{4.1}$ to $S_{4.4}$ of this fourth class are performed repeatedly until it reaches the optimal or near-optimal profitability. At the end of this process, the non-profitable variants are discarded, and the resulting product family profitability is compared to the expected gain (V_e). If it is considered satisfactory, i.e. $V \geq V_e$, those most profitable variants are computed, and the design parameter instances compounding them are selected to integrate physical modules of the product family structure at the stage $S_{4.5}$. Otherwise, the process should be restarted until the product family reaches the desired value or until it is discarded. The output of the proposed method is the modular product family structure that better balance the trade-off between the fulfillment of market needs and the resulting profitability to meet them. Therefore, this is the structure that should be developed in the subsequent design stages of the product development process, not covered in this research.

So far, it has been presented the MDM steps and how they have been organized. However, depending on the company's maturity and the context of data availability, different techniques might be adopted to execute each step of the method. We mean by low data availability, those scenarios where the data are scarce or the cost to obtain it are particularly elevated. The high data availability consists of the opposite. Table 33 complements the MDM functional structure by suggesting the techniques to be used in each key activity compounding the method's steps. The reasoning behind Table 33 is that those techniques placed at the left-hand side of the column Techniques are more suitable for contexts of low data availability. While those positioned at the right-hand side are more suitable for contexts of high data availability. Those techniques placed in the middle, in turn, can be used for both scenarios. It does not mean that some techniques cannot be used in one or another scenario, or that new techniques cannot be adopted by the method. On the contrary, the proposition here is to guide practitioners towards MDM usage. Although the techniques configure an essential part of the MDM method, only some of them will be briefly covered in the next section to not make this paper too extensive, and those who not, can be accessed by its respective references indicated in Table 33.

Table 33. MDM suggested techniques

Classes of design problems (Cp_i)	Steps of the method (S_i)	Key activities	Techniques (low data availability)	Techniques (high data availability)
Cp₁ - Product family planning and positioning	S$_{1.1}$ - Strategically plan the product family (PF)	Estimate the potential market size (M_k) and the expected profit (V_e).	Delphi (Dalkey, 1969), Three-point estimate (Premachandra, 2001), Domain knowledge (Jiao and Tseng, 1999a).	Survey (Forza, 2002), Descriptive statistics (Montgomery and Runger, 2011).
		Define the product family positioning and its leveraging strategy.	Market segmentation grid (Meyer and Lehnerd, 1997).	
		Define the technological trends for product family development.	Technology roadmap (Phaal and Muller, 2009).	
		Define which type of project must be developed.	Aggregate project plan (Wheelwright and Clark, 1992).	
	S$_{1.2}$ - Segment the market	Refine the segmentation defined apriori in terms of the number of segments, competing alternatives, share, price and performance tiers.	Delphi (Dalkey, 1969), Market segmentation grid (Meyer and Lehnerd, 1997).	Latent class analysis (Chen, Hoyle and Wassenaar, 2013).
		Synthesize the corporate strategy into objective measures for product family development.	Requirements list (Pahl *et al.*, 2007).	
	S$_{1.3}$ - Build the PF configuration model	Aggregate the customer's choice probability, the design rules, the set of design parameter instances, and the enterprise-level indicators into a single model for selecting and parameterizing the physical modules to compound the PF structure.	Mathematical modeling (Hilier and Lieberman, 2015).	
Cp₂ - Customers' choice modeling	S$_{2.1}$ - Identify customer-desired attributes (A)	Identify features and financial attributes that customers (will) consider when purchasing the product.	Qualitative interviews (Malhotra and Birks, 2007); Direct observation (Kazemzadeh *et al.*, 2009); Focus group (Malhotra and Birks, 2007).	Survey (Forza, 2002), Descriptive statistics (Montgomery and Runger, 2011).
			Content analysis (Bardin, 1993).	
	S$_{2.2}$ - Formulate the engineering attributes (E)	Transform the customer-desired attributes into quantifiable product properties to be used in the engineering product development process.	Analysis of existing technical systems (Pahl *et al.*, 2007). Benchmarking, Reverse engineering (Thevenot and Simpson, 2007).	
	S$_{2.3}$ - Map the dependencies between A and E	Map the qualitative customer-desired attributes into quantitative engineering attributes to support the further construction of the choice models.	Design matrix (Suh, 2001).	
	S$_{2.4}$ - Collect the engineering attributes levels (E_v) for each Ms_i	Collect different E values (E_{vi}), and price (P_i), such that a set of E_{vi} and P_i belong to a competing alternative (J_i), i.e. $\vec{J_i} = \{E_{vi}, P_i\}$.	Analysis of existing technical systems (Pahl *et al.*, 2007). Benchmarking, Reverse engineering (Thevenot and Simpson, 2007).	
	S$_{2.5}$ - Construct the choice set (Cs) for each Ms_i	Reduce the number of choice alternatives to be evaluated without losing the statistical significance.		Fractional factorial design (Montgomery and Runger, 2011).
	S$_{2.6}$ - Collect data on customers' preferences	Collect data on customers' preferences by asking them to compare the engineering attributes of a product against each other, or to pick an alternative from a choice set.	Qualitative interviews (Malhotra and Birks, 2007); Focus group (Malhotra and Birks, 2007).	Survey (Forza, 2002).
	S$_{2.7}$ - Build the discrete choice model (Pr_i) for each Ms_i	Estimate the weights/coefficients of the engineering attributes, and then calculate the utility function for all alternative of each target market segment.	Analytic hierarchy process (Alonso and Lamata, 2006; Saaty, 2008).	Nested logit, Maximum likelihood estimation (Chen, Hoyle and Wassenaar, 2013).
		Estimate the choice probability of each alternative within its market segment.	Analytic hierarchy process (Saaty, 2008); Data scaling (Chen, Hoyle and Wassenaar, 2013).	Nested Logit (Chen, Hoyle and Wassenaar, 2013).

(continued)

Table 33. (Continued)

Classes of design problems (Cp_i)	Steps of the method (S_i)	Key activities	Techniques (low data availability)	Techniques (high data availability)
Cp_3 - Product family modeling	$S_{3.1}$ - Formulate the design parameters (DP)	Define the logical entity with the ability to fulfill one or more E_i.	Domain knowledge (Jiao and Tseng, 1999a), Classification scheme (Pahl *et al.*, 2007).	
	$S_{3.2}$ - Map the product family architecture (PFA)	Map the logical coupling between the E_i and DP_i, i.e. $[E]_m = [PFA]_{m \times n}[DP]_n$.	Design matrix (Suh, 2001).	
	$S_{3.3}$ - Decompose PFA into functional modules (FM)	Decompose the PFA into functional modules (FM), i.e. $FM_i = \{E_i, DP_i\}$.	Rank order clustering (King, 1980), Cluster identification algorithm (Kusiak and Chow, 1987).	
	$S_{3.4}$ - Evaluate the clustering solution	Capture the strength and density of connections within each independent FM and between different FMs, i.e. $MI_f \geq 0{,}5$.	Modularity index (Jung and Simpson, 2017).	
	$S_{3.5}$ - Define DP instances (DPI)	Define different instances (DPI) for a particular DP along with its respective E_v i.e. $\overrightarrow{DPI_i} = \{E_{vi}\}$	Classification scheme, Analysis of existing technical systems (Pahl *et al.*, 2007). Benchmarking, Reverse engineering (Thevenot and Simpson, 2007).	
	$S_{3.6}$ - Estimate the DPI variable cost (C_v)	Estimate the variable cost (C_v) for each DPI based on its cost-related design features (CDF).	Pragmatic approach to product costing (Jiao and Tseng, 1999b), Request for quotation (Gümüş, 2014), Three-point estimate (Premachandra, 2001).	
	$S_{3.7}$ - Create rough geometric layouts	Identify the physical interactions between DPs.	Sketching e rendering (Koos Eissen *et al.*, 2007).	
	$S_{3.8}$ - Map the structural dependencies among components	Map the physical coupling between DPs, i.e. $[DP]_m = [DSM]_{m \times n}[DP]_n$.	Design structure matrix (Browning, 2001).	
	$S_{3.9}$ - Decompose the system into physical modules (M)	Decompose the DSM into physical modules (M), i.e. $M_i = \{DP_i\}$.	Functional to physical decomposition (Authors).	
	$S_{3.10}$ - Compare M_i with FM_i	Compare if $MI_p \geq MI_f$.	Modularity index (Jung and Simpson, 2017).	
	$S_{3.11}$ - Establish the design rules for PF configuration	Define design rules for product family configuration.	Generic bill-of-material (Li, Huang and Newman, 2008), Mathematical modeling (Hilier and Lieberman, 2015).	
Cp_4 – Product Family Configuration	$S_{4.1}$ - Combine DPI_i to generate the PF variants (PFv) for each Ms_i	Combine the design parameter instances to generate PF variants for each target market segment.	Design heuristic - Substitute way of achieving functions (Daly *et al.*, 2012).	Genetic algorithm (Meng, Jiang and Huang, 2007).
	$S_{4.2}$ - Parametrize the scalable E_i	Set the values for the scalable engineering attributes.	Design heuristic - Scale up or down (Daly *et al.*, 2012).	Genetic algorithm (Meng, Jiang and Huang, 2007).
	$S_{4.3}$ - Set the price (P) for each PFv variant	Set the price for each product family variant.	Trial-and-error (Rui, Cuervo-Cazurra and Annique Un, 2016).	Genetic algorithm (Meng, Jiang and Huang, 2007).
	$S_{4.4}$ - Calculate the partial profit (V_{Ms}) for each Ms_i and aggregate it into the PF profit (V)	Calculate the partial profit for each target market segment, and then aggregate it into a measure that represents the product family profitability.	Mathematical modeling (Hilier and Lieberman, 2015).	
	$S_{4.5}$ - Build the PF structure	Build the product family structure with physical modules compound by design parameter instances retrieved from the most profitable variants of each segment.	Generic bill-of-material (Li, Huang and Newman, 2008).	

6.4 Illustrative Example

In order to provide a better understanding of the proposed method, this section presents an illustrative application of the MDM within the product development process of a midsize company that produces capital goods. The MDM is employed here to conceptually design a new product family generation in a scenario of low data availability. The context unit is the Brazilian Palletizer Market, and the unit of analysis is a Family of Collaborative Robotic Palletizers. In this sense, palletizing refers to the operation of stacking products onto a pallet for storage or direct distribution, while a palletizer consists of equipment that automates this operation (Popple, 2009). The MDM application has been performed by the corresponding author of this research supported by two experts in the filed. The data used in the design process came from palletizing projects quoted by two Brazilian manufacturers during the last five years, and from the website of three leading competitors in this market.

The palletizer market usually classifies the palletizers by types which include floor level palletizers, high-level palletizers, and robotic palletizers (Gauss, Lacerda and Sellitto, 2019). For long production runs with the same stock-keeping units (SKU) and same-size packages, a conventional palletizer is preferred. However, robotic palletizers are more flexible and reliable for palletizing multiple SKU pallets (Popple, 2009). In the current fast-paced-economy, retailers and distribution centers are shipping fewer pallets with a single SKU and generating more mixed-case pallets with multiple $SKUs$. Thus, robotic palletizers are expected to be used more than conventional ones in the near future (More, 2019). Following this direction, combined with the belief in advanced robotics as a technology enabler for the future of manufacturing (Ghobakhloo, 2018), the MDM started by defining the collaborative robotics as the technological trend for

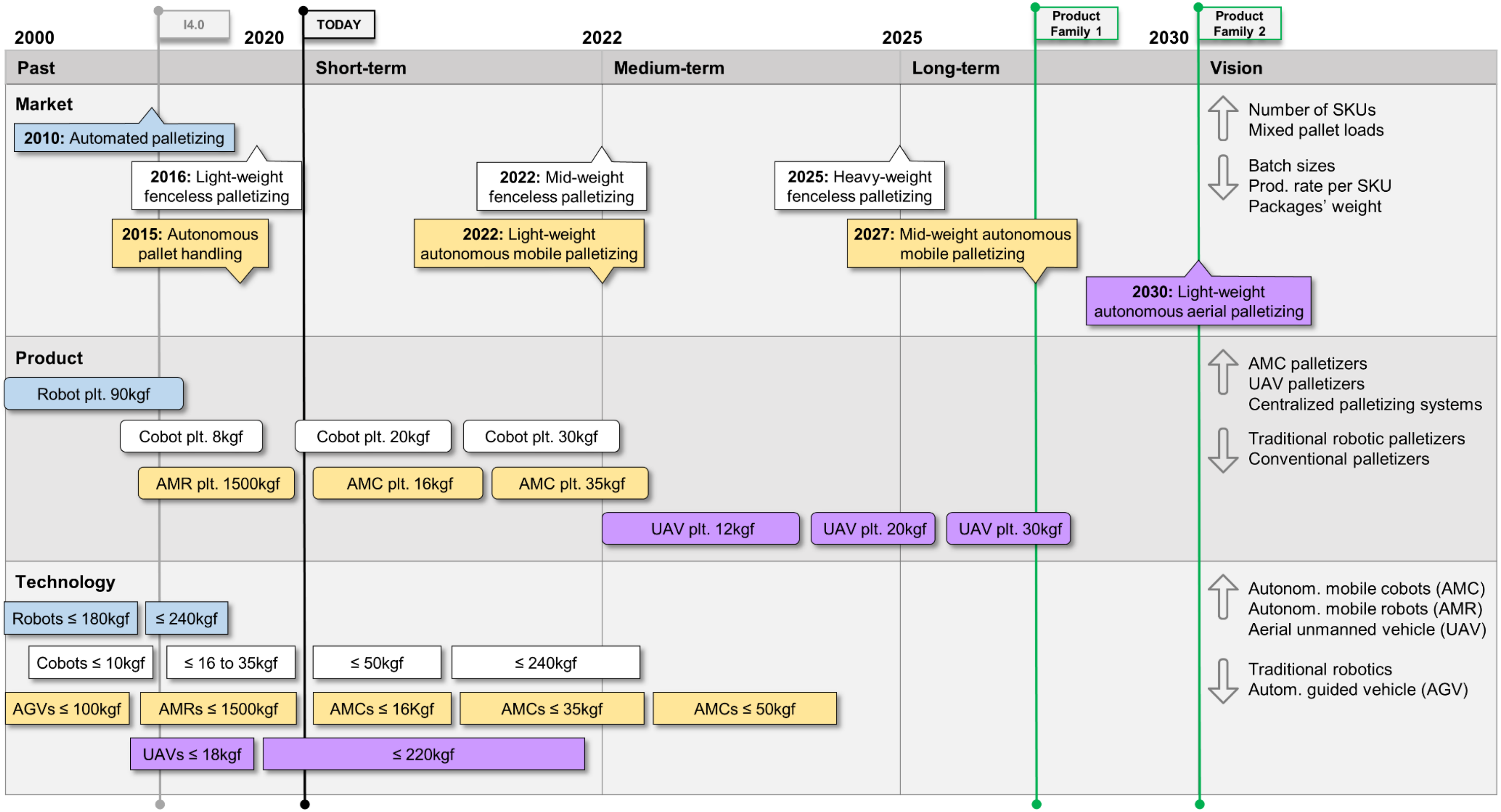


Figure 41. Technology roadmap.

product family development in step $S_{1.1}$. This process was assisted by the Roadmap presented in Figure 41, which depicts the relationship between technology, product, and market layers (Phaal and Muller, 2009). The technology layer shows the transition from traditional to collaborative robotics along with its expected evolution in terms of payload capacity. The product layer, in turn, indicates how this technology progress might support the development of intermediate goods to integrate future manufacturing environments. Finally, the market layer suggests future opportunities in the Palletizer Market resulting from the two bottom layers. Under these circumstances, two potential product families have been identified to accomplish future needs in this market as indicated by Figure 41 and Figure 42. The first one $(PF1)$ consists of a family of autonomous mobile palletizers, while the second one $(PF2)$ is made of a family of autonomous aerial palletizers. Given the fact the $PF2$ can be potentially developed through the $PF1$ platform, this paper will only cover the conceptual design of the product family $PF1$.

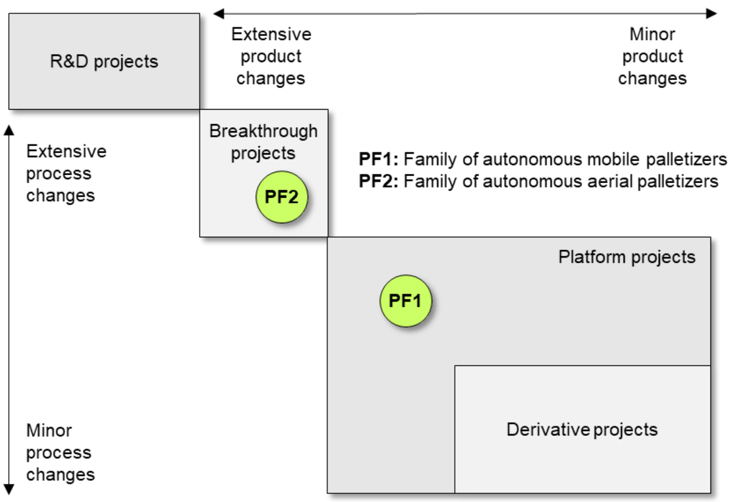


Figure 42. Aggregate project plan (Wheelwright and Clark, 1992).

Another important issue in this first step $(S_{1.1})$ is to estimate the potential market size (M_k). In this illustrative example, such activity has been supported by the available information on the Global Palletizer Market, combined with experts' domain knowledge (Jiao and Tseng, 1999a). According to More (2019), the Global Palletizer Market is projected to reach a value of USD 2.9 billion by 2023. In this context, conforming to the experts' domain knowledge, the Brazilian market historically accounts for not more than 4% of the global demand, and the average price per palletizing position in this market is about USD 150,000.00. Taking into account these values, and assuming no substantial growth in the Brazilian economy to the next years, the market size was estimated to be 773 palletizing positions a year by 2027, i.e, $M_k \approx [(2.9x10^9x0.04)/(150x10^3)] \approx 773$.

Couple with that arises the need to define the target market segments as well as the product family positioning and its respective leveraging strategy. To do that, the MDM adopts the Market Segmentation Grid (MSG) (Meyer and Lehnerd, 1997), but expands its boundaries to a multidimensional perspective as shown in Figure 43. The three-axis compounding the MSG represents the key customers' desired attributes along with its respective performance tiers. The markers distributed along the MSG area consists of a sample of palletizing projects quoted by two Brazilian manufactures in past years. Besides its grid position, the projects can be classified according to its market, and package type. In Figure 43(a), those bars at the left, and the upper left position represent the competing alternatives in this market, while its length indicates how they span the corresponding axis. In Figure 43(b) the areas filled in red, green and blue indicate the boundaries of the product family scope, and the projects dropping within these areas indirectly show the target market segments. Similarly, those bars at the left, and the upper left position of Figure 43(b), indicate the working principle adopted by the product family and the leveraging strategy to span its corresponding axis. In this illustrative example, the working principle adopted was the collaborative robots (cobots), and the leveraging strategy for palletizing multiple $SKUs$ concurrently at different production rates was to use some cobots in parallel. For palletizing different package weights, in turn, the strategy was to scale up the cobots payload capacity.

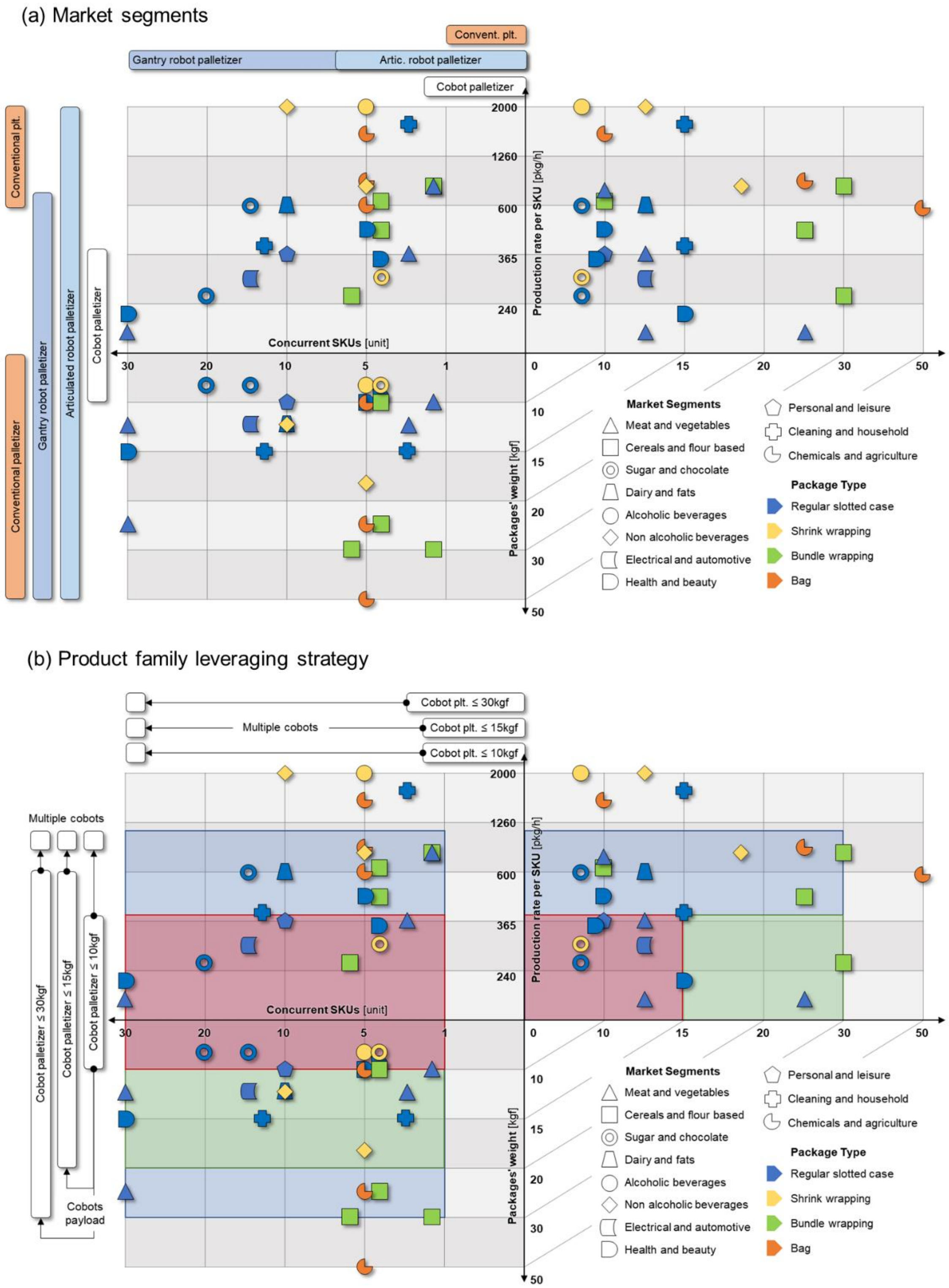


Figure 43. (a) Market segments; (b) Product family leveraging strategy.

Still from the step $S_{1.1}$, it is required to establish the product family expected profit (V_e). In other words, the company needs to define which is the profit threshold that would make itself to invest or not in the product family design. It is important to clarify that the MDM does not aim to evaluate the economic feasibility of a project, instead, it intends to conceptually build a product family structure by balancing the fulfillment of market needs and its resulting profitability, as well as providing its economic potential to the decision-maker. In this sense, for a typical mid-size machine manufacturer, the experts' in the field estimate that a $V_e \geq 2x10^6$ $[USD/year]$ would justify the investment in the product family $PF1$.

With the market size estimated and the product family positioning defined, the next step $(S_{1.2})$ is to understand the potential contribution of each target segment/niche for achieving the expected profit. In this illustrative example, for clarifying the marketing intentions, the process started by redefining the segmentation from a "sector" perspective to a "package" perspective. In this sense, 3 segments concerning the package type, and 3 performance tiers have been established, giving rise to 9 market niches as shown in Figure 44. The areas filled in red, green and blue indicate the clusters encompassing the 6 target market niches to be considered in the product family design. These clusters have been defined qualitatively in Figure 43(b) based on the cobots limitations in terms of payload capacity and production rate. The segments' and performance tiers' proportions, in turn, have been specified based on the projects quoted by two Brazilian manufacturers and assuming no major changes in it for the next seven years according to experts' opinion. Multiplying the proportions of horizontal axes by the vertical axes it is possible to estimate the share of each market niche (S_{Ms}), i.e. $S_{Ms1.1} = 0.5\ x\ 0.2 = 0.1$. After converting the corporate strategy into objective measures for product family design, the information generated at this first class of design problems (Cp_1) is suggested to be compiled into a Requirements' List (Pahl *et al.*, 2007).

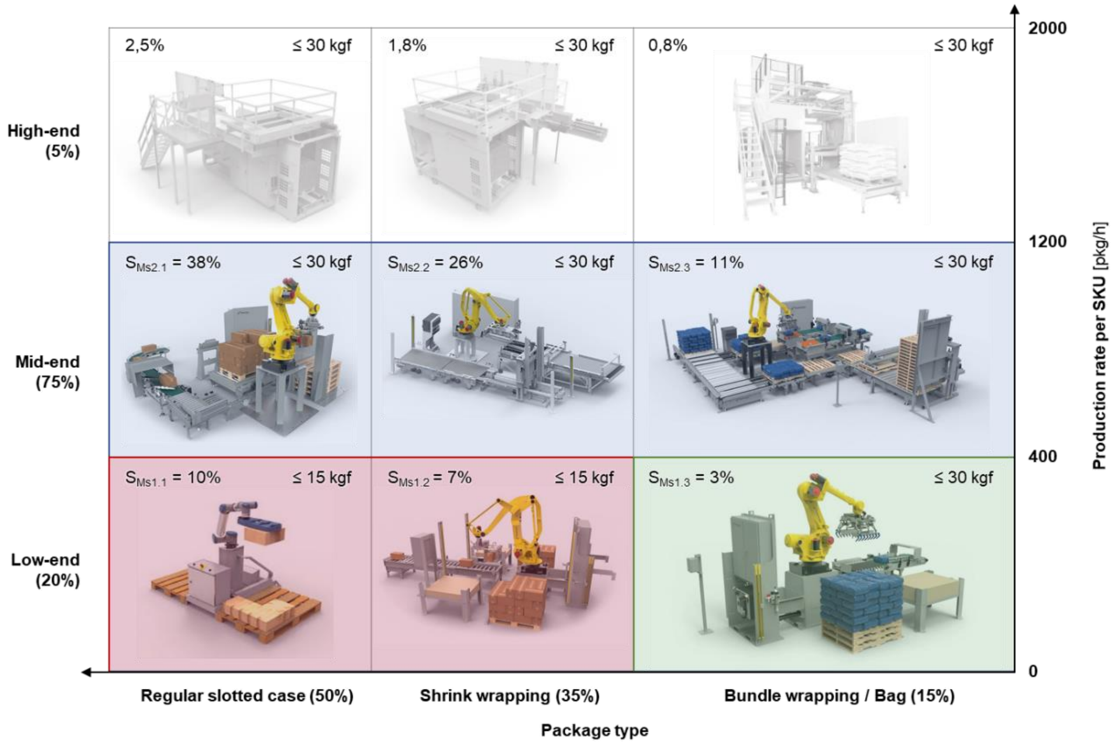


Figure 44. Market segmentation grid (Meyer and Lehnerd, 1997).

The next issue is to model the customer choice probability for each market niche previously defined. This process takes place in the second class of design problems (Cp_2), and starts by identifying the customer desired attributes (A) in step $S_{2.1}$. In the present example, 22 customers desired attributes emerged through the Content Analysis (Bardin, 1993) of 24 project quotations performed by the Brazilian manufacturers which contributed to this research. After identified, these customer desired attributes were checked against the possibility to still exist in seven years from now. With that respect, there was a consensus between the experts that despite its relative importance might change, all of them would still represent the customers' needs in the near future. Finally, each attribute has been associated with its respective market niche as shown in Table 34.

Table 34. Product family attributes deployment.

		Engineering attributes (E_i)									Mkt. segments (Ms_i)		
		+	+	-	-	+	+	...	-	-			
		Production rate per SKU [pkg/h]	Packages' buffering [m]	Package's barcode misreading error [%]	Min. package dimension [mm]	Max. package dimension [mm]	Max. package height [mm]	...	Risk: ISO 13849-1 [Cat]	Price [USD/SKU]	Regular slotted case / Low-end / ≤ 15kgf	...	Bundle wrapping / Bag / Mid-end / ≤ 30kgf
Id.	Customer desired attributes (Ai)	E1	E2	E3	E4	E5	E6	...	E36	P	Ms1.1	...	Ms2.3
A1	Production speed	1									1	…	1
A2	Accumulate packages before entering the palletizer		1								1	…	1
A3	Package's barcode reading			1								…	1
A4	Palletize different sizes of packages				1	1	1				1	…	1
...	...							...			...	…	...
A21	Safety: ISO 13849-1								1		1	…	1
A22	Price									1	1	…	1
Id.	Regular slotted case / Low-end / ≤ 15kgf (Ms1.1)							...					
J1.1-1	Collaborative robotic palletizer A1	360.0	2.0		100.0	600.0	500.0	...	3	57,500.00	1		
J1.1-2	Robot pick and place palletizer B1	600.0	1.5		127.0	609.6	457.2	...	3	125,000.00	1		
J1.1-3	Low infeed palletizer C1	600.0	3.0		150.0	600.0	450.0	...	3	157,500.00	1		
J1.1-4	Robotic palletizer D1	240.0	2.0		100.0	600.0	450.0	...	3	50,833.33	1		
J1.1-5	Robotic palletizer D2	450.0	2.0		100.0	600.0	450.0	...	3	95,937.50	1		
	Relative weights (w)	*0.084*	*0.021*		*0.047*	*0.047*	*0.054*	...	*0.037*	*0.200*			
...	...	...	...	...	...	...	...	...	...	...		...	
Id.	Bundle wrapping / Bag / Mid-end / ≤ 30kgf (Ms2.3)							...					
J2.3-1	Hybrid robotic palletizer B3	900.0	2.0	~~0.0~~	152.4	736.6	381.0	...	3	260,000.00			1
J2.3-2	Low infeed palletizer C2	900.0	3.0	~~0.0~~	150.0	600.0	450.0	...	3	224,500.00			1
J2.3-3	Robotic palletizer D4	720.0	2.0	~~0.0~~	100.0	600.0	300.0	...	3	254,750.00			1
	Relative weights (w)	*0.090*	*0.026*	*0.010*	*0.040*	*0.040*	*0.040*	...	*0.037*	*0.175*			

Right after identifying the customer desired attributes, they need to be translated into quantifiable product properties to be used in the subsequent design stages. In MDM, these properties are called as engineering attributes (E), and its formulation happens in step $S_{2.2}$. In this illustrative example, this process has been performed through the Analysis of Existing Technical Systems (Pahl et al., 2007), combined with Benchmarking (Thevenot and Simpson, 2007). The data source was the product portfolio of five leading competitors in this market, which includes the two manufactures that cooperated with this research. As a result, 36 engineering attributes have been formulated. Then, at the stage $S_{2.3}$, the customer desired attributes (A) have been mapped to engineering attributes (E) by means of a design matrix (Suh, 2001), i.e. $[A]_m = [R]_{m \times n}[E]_n$. The design matrix $[R]_{m \times n}$ includes integer as well as "blank" entries, where nonblank entry $a_{ij} \in [R]_{m\times n}$ indicates a relationship between A and E according to shows Table 34. This table also indicates the desired direction of each engineering attribute, resembling the House of Quality (Chan and Wu, 2002).

In general, the engineering attributes assume different levels within and across segments. For that reason, at stage $S_{2.4}$, a set of competing alternatives (J) for each segment is captured to identify its respective engineering attribute values (E_v) and price (P). This task adopted the same techniques and data source used in the step $S_{2.2}$, and the result was the identification of 10 competing alternatives distributed into 6 market niches, as illustrated in Table 34. In some situations, depending on the market and the product, accessing the price of competing alternatives might be particularly difficult. In such cases, we suggest estimating price thought the multiplication between a typical markup adopted in the market and the product's variable cost (C_v), i.e. $P_i = markup. C_{vi}$, an issue that will be covered later in this section. Compared with price, the engineering attribute levels are a bit easier to access, but they still have their particularities. One example of it, are those competing alternatives integrating the market niche $Ms1.1$ in Table 34. They present blank entries in column $E3$, which means that the engineering attribute $E3$ is related to a customer desired attribute (A) that does not tackle this market niche, as also indicates Figure 45. Another example is those alternatives integrating the market niche $Ms2.3$. They present strikethrough values in column $E3$ of Table 34, which means that this engineering attribute values have not been found during the searching process. That was not the case here, but after capturing the competing alternatives, new engineering attributes might arise, in these situations, the process should go back to the two previous steps.

Since the step $S_{2.5}$ is not required here, it has been bypassed straight to the stage $S_{2.6}$, where the objective is collecting the data on customer preferences. This example simulates the design process of a new product family generation occurring in a context of low data availability. Therefore, the relative importance (w) of each engineering attribute has been deductively defined in conjunction with experts through the use of the Analytic Hierarchy Process (AHP) (Saaty, 2008). This procedure started by structuring a three-level decision hierarchy for each market niche as illustrated in Figure 45. Then, a set of pairwise comparison matrices has been done by each participant, considering the importance of each constituent in seven years from now. Afterward, the geometric mean has been used to obtain the group judgment for each entry of the comparison matrices, as presented in Table 35 (Ossadnik, Schinke and Kaspar, 2016). Each element in an upper hierarchy level (A_i) was then used to compare the elements in the level immediately below (E_i) with respect to it. After that, the matrices acceptance has been evaluated according to the consistency system proposed by Alonso and Lamata (2006) due to its flexibility in dealing with matrices with more than 15 criteria. In all cases, the consistency ratio (CR) appeared to be lower than 0,1, i.e. $CR < 0.1$, indicating an acceptable consistency.

Table 35. Pairwise comparison of customer desired attributes in market niche $MS1.1$.

Reg. slotted case / Low-end / ≤ 15kgf (MS1.1)		A1	A2	A4	A5	A6	A7	A8	A9	A10	A13	A14	A17	A18	A19	A20	A21	A22
Production speed	A1	1.0	5.0	1.0	1.0	1.0	1.0	7.0	5.0	0.3	5.0	5.0	5.0	5.0	5.0	3.0	3.0	0.3
Accumulate pkgs. before entering the plt.	A2		1.0	0.1	0.1	0.2	0.1	3.0	3.0	0.1	0.3	0.3	0.3	3.0	3.0	1.0	0.3	0.1
Palletize different sizes of packages	A4			1.0	3.0	1.0	0.3	7.0	7.0	1.0	5.0	5.0	3.0	9.0	9.0	5.0	3.0	0.3
Palletize different weights of packages	A5				1.0	1.0	0.2	3.0	3.0	0.2	3.0	3.0	1.0	3.0	3.0	3.0	1.0	0.3
Palletize different sizes of pallets	A6					1.0	0.2	3.0	1.0	0.3	1.0	1.0	0.3	1.0	1.0	0.3	0.3	0.1
Palletize different load patterns	A7						1.0	7.0	5.0	1.0	5.0	5.0	3.0	5.0	5.0	5.0	5.0	0.2
Guarantee the load alignment	A8							1.0	0.3	0.1	0.3	0.3	0.3	3.0	0.3	0.2	0.3	0.1
Orientate the pkg's label to external load faces	A9								1.0	0.1	0.3	0.3	0.3	1.0	0.3	0.3	0.3	0.1
Palletize multiple packages concurrently	A10									1.0	5.0	5.0	5.0	7.0	7.0	5.0	5.0	0.3
Place tie/cap sheet between layers	A13										1.0	0.2	3.0	5.0	5.0	3.0	3.0	0.1
Wrap the pallet load with stretched film	A14											1.0	3.0	5.0	5.0	3.0	3.0	0.1
Easiness to configure new patterns	A17												1.0	5.0	5.0	1.0	1.0	0.1
Plug and play installation	A18													1.0	1.0	0.2	0.3	0.1
Reduced footprint	A19														1.0	0.3	0.2	0.1
Reduced downtime	A20															1.0	1.0	0.1
Safety: ISO 13849-1	A21																1.0	0.2
Price	A22																	1.0

Still through the AHP, at stage $S_{2.7}$, the relative weight (w) of each engineering attribute (E) was calculated for its respective market niche, as indicates those lines in italic of Table 34. With the weights defined, the next issue was to model the utility (W)

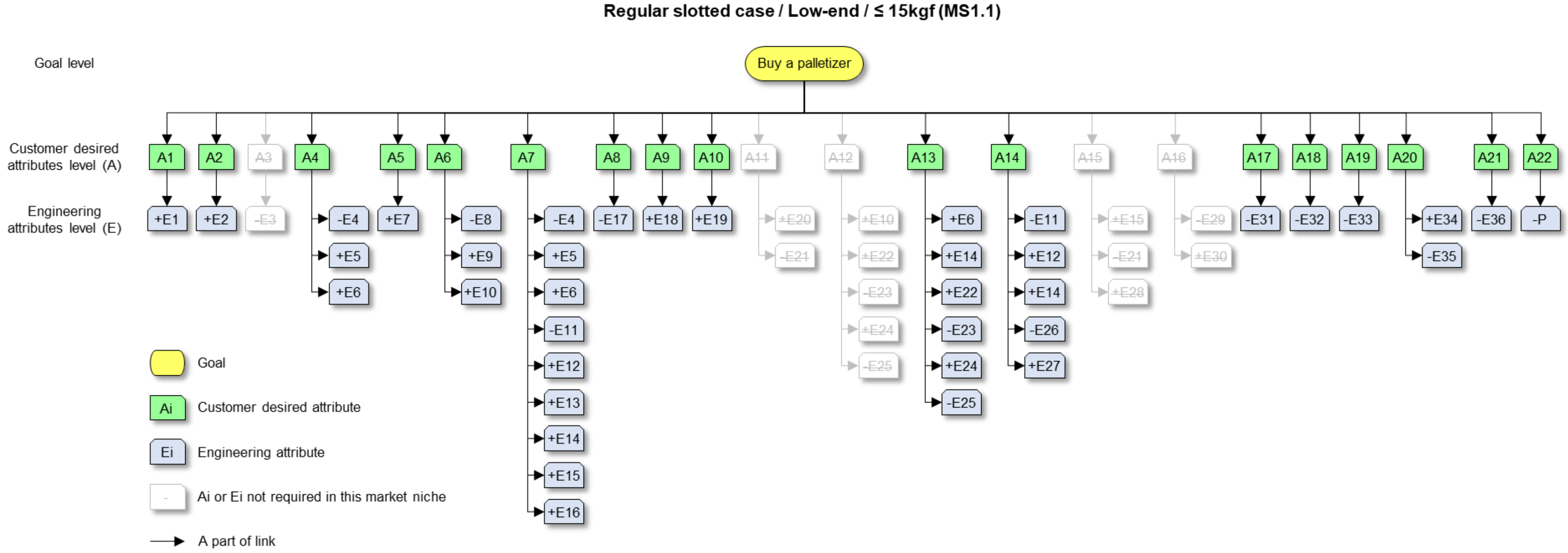


Figure 45. Decision hierarchy of market niche $MS1.1$.

and the choice probability (Pr) of each competing alternative (J) comprising the same market niche. Assuming these variables can be expressed as a linear combination of engineering attributes values (E_v), W and Pr follow the form of Equation 10.

$$W_{ij} = Pr_{ij} = \sum_{i=1}^{n} w_i . \left(\frac{E_{vi}}{\sum_{j=1}^{k} E_{vj}} \right) \quad 10$$

Regarding the Equation 10, some important considerations must be clarified. First, the model exhibits the independence of irrelevant alternatives (IIA) property, which leads to proportional substitution patterns among the alternatives considered (Chen, Hoyle and Wassenaar, 2013). Second, the equality between W and Pr_n is true in cases where the relative weights (w) derive from the AHP. Third, the utility ranges from 0 to 1, i.e. $(0 \leq W \leq 1)$, as well as the choice probability. Fourth, the engineering attributes (E) might have different desired directions $(+\ or\ -)$ as indicated in Table 34. Thus, when E_i assumes a negative direction $(-)$, i.e. "the lower the better", the inverse of its respective engineering attribute value $(1/E_{vi})$ should be considered in Equation 1. Finally, there are some situations where the engineering attribute values are not found, as indicate those strikethrough values in Table 34. In such cases, these E_v should assume the same values of the product variant resulting from the configuration process to be presented later in the fourth class of design problem (Cp_4).

Given all these points, and with no deviation found from the segmentation defined in the step $S_{1.2}$, the next issue is to define the product family architecture, decompose it into functional/physical modules, and then generate the design parameter instances that can potentially compose the product family structure. This process takes place in the third class of problems (Cp_3), and starts by formulating the design parameters (DP) at stage $S_{3.1}$. In this example, the Classification Scheme (Pahl *et al.*, 2007) has been used to help the abductive process of design parameters formulation, as illustrated in Table 36. The task starts by searching and cataloging those working principles (WP) capable of accomplishing one or more engineering attributes (E). Then, the logical entities underlying those working principles sharing the same physical effect are inductively derived. For example, in Table 36, the working principles $WP12$, $WP13$, and $WP14$ differ in morphological aspects, however, they share the same physical effect of gripping packages. In the context of robotic palletizers, the logical entity that executes that function is known as end effector, thus, it has been defined as a design parameter $(DP5)$. Following this pattern, and supported by the experts' domain knowledge, 13 design parameters have been derived from the 31 working principles cataloged in Table 36.

Table 36. Classification scheme.

		Working principles (WPi)						
			...				...	
Dpi	Design parameters (DP)	WP1	...	WP12	WP13	WP14	...	WP31
DP1	Packages' accumulation conveyor	1	...				...	
DP2	Packages' singulation conveyor		...				...	
DP3	Packages' barcode reader		...				...	
DP4	Articulated robotic arm		...				...	
DP5	End effector		...	1	1	1	...	
DP6	Pallet's dispenser		...				...	
DP7	Pallet's infeed conveyor		...				...	
DP8	Pallet's shuttle car		...				...	
DP9	Full load discharge conveyor		...				...	
DP10	Full load turntable		...				...	
DP11	Sheets' dispenser		...				...	
DP12	Stretch wrapper		...				...	
DP13	Print and apply labeler		...				...	1

With design parameters (DP) defined, they need to be mapped to engineering attributes (E) in stage $S_{3.2}$. This task has been performed through a design matrix (Suh, 2001), giving rise to product family architecture (PFA), i.e. $[DP]_m = [PFA]_{m \times n}[E]_n$. Then, by using the Ranking Order Clustering algorithm (King, 1980), the $[PFA]_{m \times n}$ had its rows and columns iteratively rearranged, until both be arranged in order of decreasing value when read as binary words. Right after, at stage $S_{3.3}$, the functional

Table 37. Modular product family architecture.

		Time to configure a new pattern [min]	MTBF [min]	MTTR [min]	Price [US$/SKU]	Installation time [h]	Footprint area [m2]	Safety ISO 13849-1 [Cat]	Production rate per SKU [pkg/h]	Max. package weight [kgf]	Min. package dimension [mm]	Max. package dimension [mm]	Packages' buffering [m]	Max. misalignment per load height [mm/mm]	Min. sheet dimension [mm]	Max. sheet dimension [mm]	Max. package height [mm]	Max. number of pallet patterns [un]	Min. layer dimension [mm]	Max. layer dimension [mm]	Max. number of packages per layer [un]	Max. load height [mm]	Package's rotating angle [°]	No. of SKUs palletized concurrently per palletizer [un]	Sheets' buffering [mm]	Sheet thickness [mm]	Max. load weight [kgf]	Film thickness [mm]	Max. film stretch ratio [%]	Min. pallet dimension [mm]	Max. pallet dimension [mm]	Pallet track height [mm]	Full load discharge buffering [un]	Max. empty pallet height [mm]	Empty pallets' buffering [un]	Min. label dimension [mm]	Max. Label dimension [mm]	Package's barcode misreading error [%]
		E31	E34	E35	P	E32	E33	E36	E1	E7	E4	E5	E2	E17	E23	E24	E6	E16	E11	E12	E13	E14	E18	E19	E22	E25	E15	E26	E27	E8	E9	E21	E28	E10	E20	E29	E30	E3
DP2	Packages' singulation conveyor	1	1	1	1	1	1	1	1	1	1	1	1	1																								
DP1	Packages' accumulation conveyor	1	1	1	1	1	1	1	1	1	1	1	1																									
DP5	End effector	1	1	1	1	1	1	1	1	1	1	1		1	1	1	1																					
DP4	Articulated robotic arm	1	1	1	1	1	1	1	1	1								1	1	1	1	1	1	1														
DP11	Sheets' dispenser	1	1	1	1	1	1	1	1						1	1									1	1												
DP12	Stretch wrapper	1	1	1	1	1	1	1	1									1									1	1	1									
DP9	Full load discharge conveyor	1	1	1	1	1	1	1	1																		1			1	1	1	1					
DP10	Full load turntable	1	1	1	1	1	1	1	1																		1			1	1	1	1					
DP7	Pallet's infeed conveyor	1	1	1	1	1	1	1	1																		1			1	1	1						
DP8	Pallet's shuttle car	1	1	1	1	1	1	1	1																		1			1	1	1						
DP6	Pallet's dispenser	1	1	1	1	1	1	1	1																					1	1	1		1	1			
DP13	Print and apply labeler	1	1	1	1	1	1	1										1																		1	1	
DP3	Packages' barcode reader	1	1	1	1																																	1

modules (FM) have been formatted, and had its clustering solution evaluated by the Functional Modularity Index $\left(MI_f\right)$ at stage $S_{3.4}$ (Jung and Simpson, 2017). This process has been repeatedly performed until the clustering solution reaches the $MI_f = 0.58$, a value higher than the threshold previously defined $\left(MI_f \geq 0.5\right)$. As a result, the product family architecture has been decomposed into 9 functional modules as shown in Table 37, i.e. $FM6 = \{DP7, DP8, DP9, DP10, E8, E9, E21, E28\}$. Although some engineering attributes are influenced by the same design parameters, the final solution consisted of an uncoupled design matrix following the independence axiom of Axiomatic Design (Suh, 1998).

The next issue is to specify the engineering attributes values (E_v) resulting from different physical characteristics that a design parameter (DP) might assume. In MDM, these entities are called design parameter instances (DPI), and its definition is performed at the stage $S_{3.5}$. In this illustrative example, this task has been assisted by the Classification Scheme presented in Table 36, along with the Analysis of Existing Technical Systems (Pahl et al., 2007). Based on available technology and existing product features, a set of instances has been deductively defined for each design parameter (DP) compounding the functional modules (FM), as presented in Table 38. In total, 110 design parameter instances have been specified, and the resulting combination of them, along with its functional modules, overcome millions of product family variants, as illustrated in Figure 46. However, to reduce the manufacturing complexity, the MDM claims that only those variants that better balance the fulfillment of market needs and the resulting profitability to achieve them should integrate the product family structure, otherwise, they should be discarded. In this sense, in addition to the engineering attributes values (E_v), the variable cost (C_v) of a design parameter instance should be computed to allow this further evaluation, i.e. $DPI_i = \{E_{vi}, C_{vi}\}$.

Table 38. Design parameter instances.

FMi	DPIij	Description of design parameter instance	E1	E2	E3	E4	E5	E6	...	E36	Cv
FM1	DPI1.1	Roller top modular belt - 400 x 2000 mm - 30kgf	1	2		150	600		...	2	2,059.38
...	...	...	...	...		...	...		...	...	...
FM1	DPI1.8	Live roller - 600 x 3000 mm - 30kgf	1	3		100	600		...	3	3,002.50
FM1	DPI2.1	Friction top and angled roller modular belt	1	3		150	600		...	2	2,727.50
...	...	...	...	...		...	...		...	...	...
FM1	DPI2.4	Live roller with pneumatic stopper / aligner	1	3		100	600		...	3	3,085.50
FM2	DPI5.1	Vacuum style - surface pad - 400 x 600 mm - 1 x 8 ≤ 8kgf	1			100	600	450	...	2	2,800.00
...	...	...	...			...	...	...	...	...	...
FM2	DPI5.40	Finger style - 400 x 300 mm - 2 x 15 ≤ 30kgf	2			100	600	450	...	3	17,500.00
FM3	DPI4.1	Cobot - payload ≤ 10 kgf - 1 SKU(s)	360						...	3	46,250.00
...	...	...	...						...	...	...
FM3	DPI4.31	Cobot - payload ≤ 50 kgf - AMR - 4 SKU(s)	120						...	3	156,750.00
FM4	DPI11.1	Horizontal sheets' hopper	0,8						...	1	575.00
FM4	DPI11.2	Angled sheets' hopper	0,8						...	1	575.00
FM5	DPI12.1	Self-propelled robots ≤ 250%	1						...	3	16,000.00
...	...	...	...						...	...	...
FM5	DPI12.5	Rotating arm ≤ 400%	1						...	3	50,000.00
FM6	DPI7.1	Chain-driven live roller conveyor (CLRC) - load weight ≤ 2.000 kgf	1						...	3	3,750.00
FM6	DPI7.2	Two / three-strand chain conveyor (TCC) - load weight ≤ 2.000 kgf	1						...	3	4,125.00
FM6	DPI7.3	Manual loading / unloading - load weight ≤ 2.000 kgf	1						...	0	0.00
FM6	DPI8.1	Autonomous mobile robot (AMR) - CLRC - load weight ≤ 1.000 kgf	1						...	3	80,500.00
...	...	...	...						...	...	...
FM6	DPI8.6	Autonomous mobile robot (AMR) - TCC - load weight ≤ 2.000 kgf	1						...	3	149,950.00
FM6	DPI9.1	Chain-driven live roller conveyor (CLRC) - load weight ≤ 2.000 kgf	1						...	3	3,750.00
FM6	DPI9.2	Two / three-strand chain conveyor (TCC) - load weight ≤ 2.000 kgf	1						...	3	4,125.00
FM6	DPI9.3	Manual loading / unloading - load weight ≤ 2.000 kgf	1						...	0	0.00
FM6	DPI10.1	Turntable - CLRC - load weight ≤ 2.000 kgf	1						...	3	7,500.00
FM6	DPI10.2	Turntable - TCC - load weight ≤ 2.000 kgf	1						...	3	8,250.00
FM7	DPI6.1	Stripper style with pallet hopper	1						...	3	6,250.00
FM7	DPI6.2	Lift and separate with pallet hopper	1						...	3	10,250.00
FM7	DPI6.3	Screw style with pallet hopper	1						...	3	8,250.00
FM8	DPI13.1	All-electric automated labeling							...	2	11,250.00
FM9	DPI3.1	Image-based barcode reader (1D and 2D)			0,1				...		3,750.00
FM9	DPI3.2	Laser barcode scanner (1D and 2D)			2				...		2,500.00

The variable cost estimation takes place at the stage $S_{3.6}$, and borrows the concept of the cost-related design feature (CDF) from the Pragmatic Approach to Product Costing (Jiao and Tseng, 1999b). The reasoning here is to identify those design features (CDF_i) from which the variable cost of a design parameter (DP) can be completely determined, and then, to estimate a cost coefficient (θ_i) for each CDF, in order to derive a particular solution from the Equation 11.

$$C_{vi} = \sum_{i=1}^{n} \theta_i . CDF_i \qquad 11$$

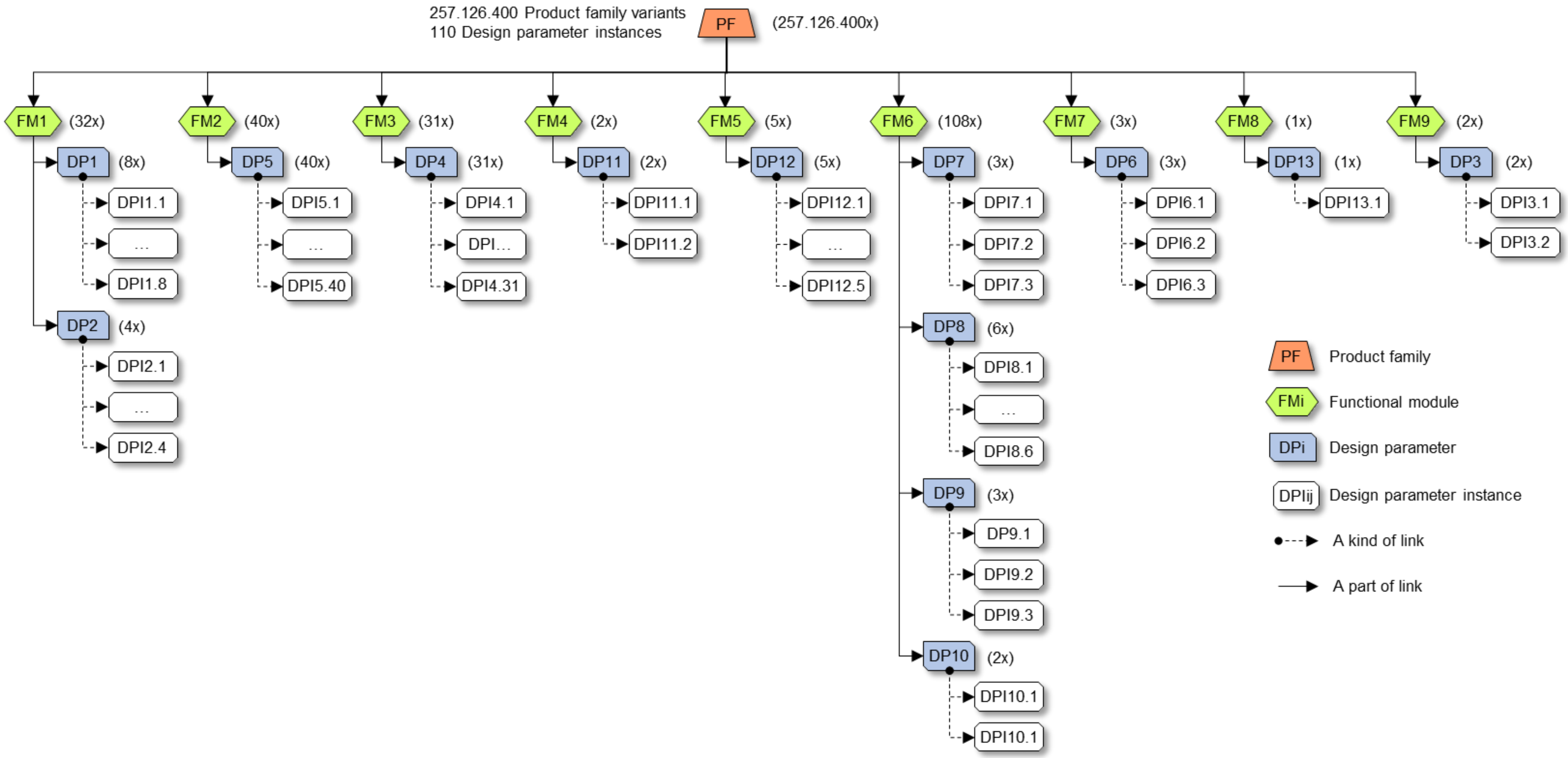


Figure 46. Product family structure with potential design parameter instances.

In this illustrative example, the cost-related design features (CDF) have been identified for every design parameter (DP) compounding the product family structure, as shown in Figure 47. Then, the cost coefficient (θ) of each CDF has been estimated in two complementary ways. When historical data on the products' costing were available, a sample of variable costs related to those features of interest has been averaged. In such cases where the data were not available, some suppliers have been requested for quotation (Gümüş, 2014). As an example, the variable cost of design parameter instance $DPI1.1$ has been estimated as follows. First, the particular solution for $DP1$ has been established according to Equation 12. The cost coefficients comprising this formula derived from the existing products' costing provided by the two manufacturers that contributed to this research. Then, considering $DPI1.1$ composed by a roller top modular belt, with 1,2 m^2 of surface area, and tractioned by one gearmotor, the variable cost has been obtained through Equation 13.

$$DP1_{C_v} = (881.25 \text{ x } CDF1.1.1) + (976.56 \text{ x } CDF1.1.2) + (887.50 \text{ x } CDF1.2) \qquad 12$$

$$DPI1.1_{C_v} = (881.25 \text{ x } 0) + (976.56 \text{ x } 1{,}2) + (887.50 \text{ x } 1) = \ 2{,}059.38\ [USD] \qquad 13$$

After that, to identify the physical interactions within and across functional modules, 12 rough geometric layouts have been created, as illustrated in Figure 48. In this example, no incompatible design parameter instances have been found nor new design parameters have been created. This process took place in step $S_{3.7}$, and the resulting interactions served as an input flow for mapping the structural dependencies among design parameters in step $S_{3.8}$.

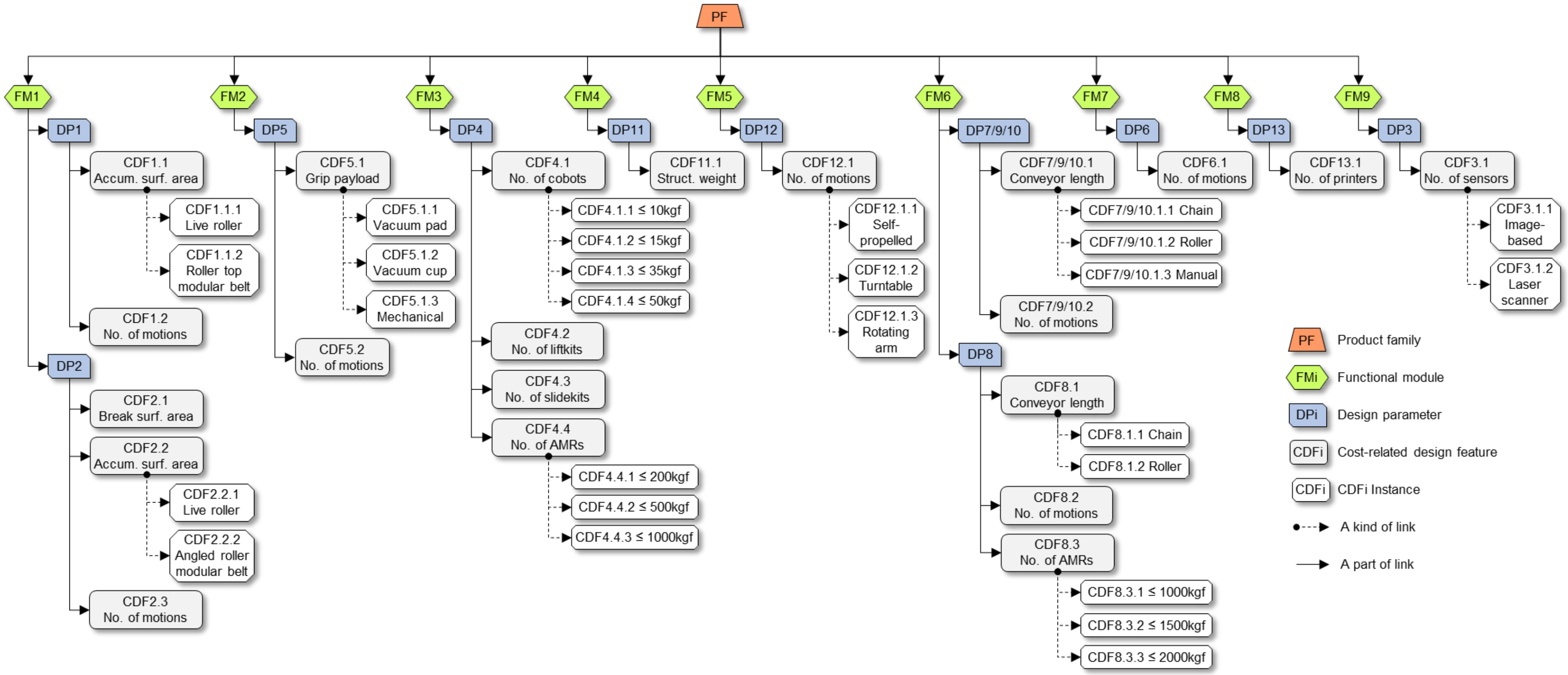


Figure 47. Cost-related design features (CDF).

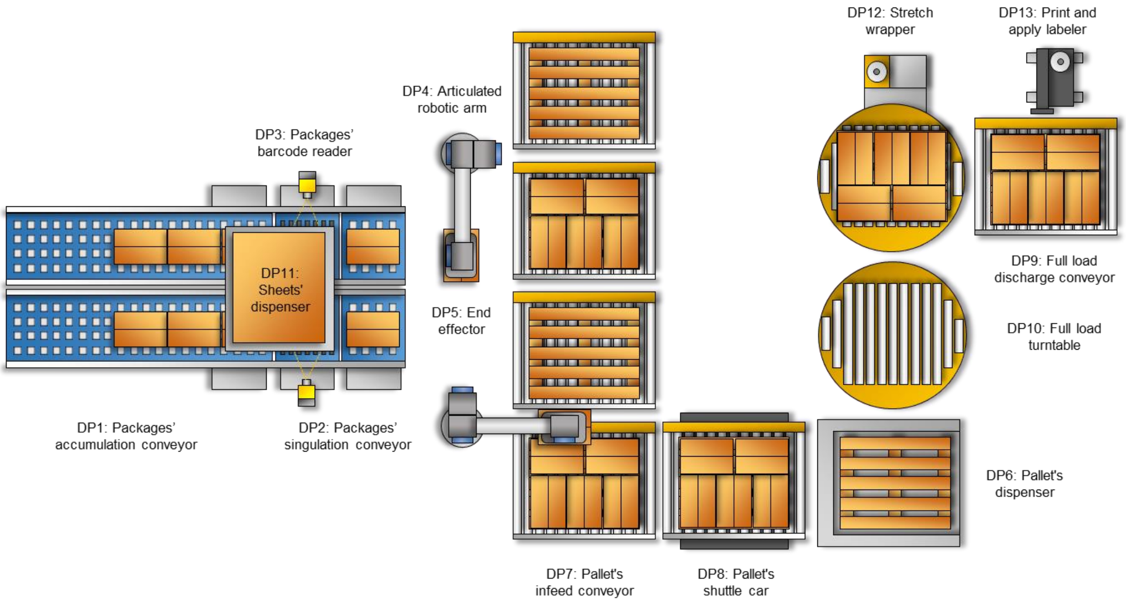


Figure 48. Example of the geometric layout.

Then, at stage $S_{3.9}$, the functional decomposition has been transferred to the physical decomposition. This process consisted of simply forming physical modules (M) as a subset of functional modules (FM) by only considering the design parameters (DP) as constituents, i.e. $M6 \subset FM6 = \{DP7, DP8, DP9, DP10\}$. The result is shown in the Design Structure Matrix (Browning, 2001) presented in Table 39.

Table 39. Physical decomposition.

		DP1	DP2	DP3	DP4	DP5	DP6	DP7	DP8	DP9	DP10	DP11	DP12	DP13
DP1	Packages' accumulation conveyor	1	1									1		
DP2	Packages' singulation conveyor	1	1			1						1		
DP3	Packages' barcode reader			1										
DP4	Articulated robotic arm				1	1								
DP5	End effector		1			1						1		
DP6	Pallet's dispenser						1	1	1					
DP7	Pallet's infeed conveyor						1	1	1	1				
DP8	Pallet's shuttle car						1	1	1	1	1		1	
DP9	Full load discharge conveyor							1	1	1	1		1	1
DP10	Full load turntable								1	1	1		1	
DP11	Sheets' dispenser											1		
DP12	Stretch wrapper								1	1	1		1	
DP13	Print and apply labeler													1

With the physical modules defined, its clustering solution has been evaluated by the Physical Modularity Index (MI_p) at stage $S_{3.10}$ (Jung and Simpson, 2017). As a result, the clustering solution reached the $MI_p = 0.77$, indicating this way the product family has a modular architecture unconstrained by physical interactions.

After defining the modular product family architecture and the potential design parameter instances to integrate it, the last step $(S_{3.8})$ of this third class of design problems is to establish the design rules for product family configuration. These rules consist of a set of mathematical constraints guiding the end product configuration to a feasible solution. Some of these rules include: (i) the payload capacity of the articulated robotic arm should be higher than end effector payload capacity, (ii) the number of packages' accumulation/singulation conveyors should not exceed the number of $SKUs$ being palletized concurrently, (iii) one sheets' dispenser should serve up to 2 $SKUs$ being palletized concurrently, among others. However, these rules are implicit relationships among the entities defined so far $(A, E, E_v, WP, DP, DPI, FM, M)$, which makes it very difficult to manipulate them without creating any "hidden" variable. Therefore, at this stage, the MDM makes use of dummy engineering variables (DE) to facilitate the mathematical formulation of design rules.

Another important issue in this step is to define how the resulting engineering attributes (E), or the total variable cost (C_v) of product family variant (PF_v) is calculated. For example, let's suppose a product family variant having the following configuration: $PF_v = \{DPI1.1, DPI2.1, DPI4.1, DPI5.1, DPI7.3, DPI11.1\}$. If each DPI, has the maximum quantity equals to one, it is possible to calculate its total variable cost (C_v) by summing the corresponding C_v of each DPI chosen. From the values presented in Table 38, the result would be $C_v = 54{,}411.88\ [USD]$. However, if we want to calculate the resulting $E1$ of the same PF_v configuration, the approach should be different. The $E1$ consists of the palletizer production rate measured in terms of packages per hour. This engineering attribute has the design parameter $DP4$ as its basic source. The other design parameters, in turn, work as moderating variables of $DP4$, amplifying or reducing its values. Therefore, the resulting $E1$ should be calculated by multiplying the corresponding E_v of each DPI chosen. From the values presented in Table 38, the result would be $E1 = 288\ [pkg/h]$. In MDM, there are two other approaches for calculating the resulting engineering attributes (E). One that accounts for the maximum E_v of each DPI chosen, and the other that accounts for the minimum.

Back to the first class (Cp_1), at stage $S_{1.3}$, the issue was to aggregate the customer's choice probability, the design rules, the set of design parameter instances, and the enterprise-level indicators (demand, price, and profit) into a single model for combining and selecting the design parameter instances to compound the modular product family structure. In this example, the conceptual model built before the mathematical implementation is shown in Figure 49, and for illustrative simplicity, it has not considered any scalable engineering attributes.

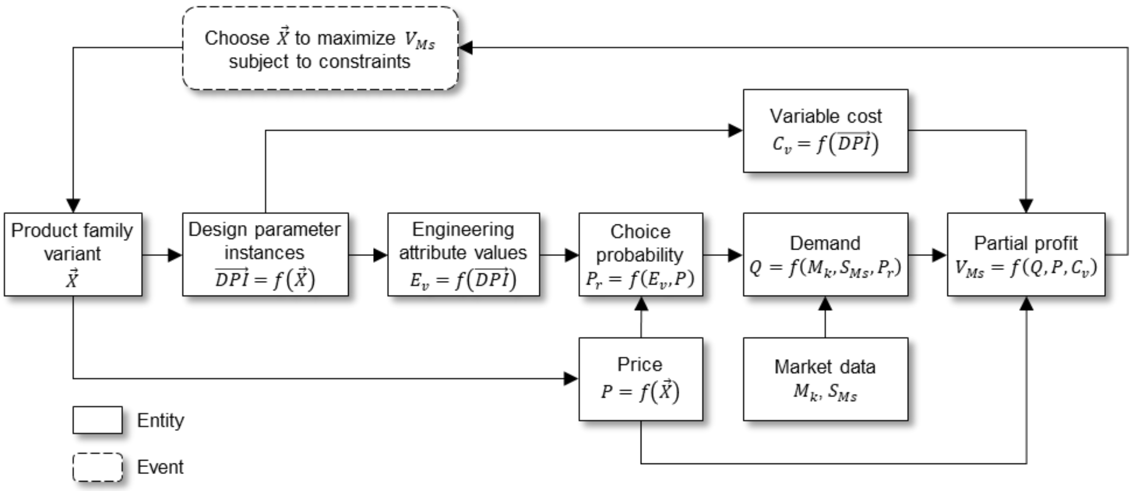


Figure 49. Configuration model.

The reasoning behind this model is to configure a product family variant $(\vec{X})$, characterized by design parameter instances $(\overrightarrow{DPI})$ and price (P), i.e. $\vec{X} = \{\overrightarrow{DPI_\iota}, P\}$, to a specific market niche (M_s), with the purpose of maximizing the partial profit (V_{Ms}), while subject to constraints. One mathematical model has been built for each target market niche (M_s), and the resulting profitability of each niche, the partial profit, was then aggregated into the product family profit (V).

Based on the mathematical model previously defined, the configuration process takes place in the fourth class of design problems (Cp_4). In this example, the steps $S_{4.1}$ and $S_{4.3}$ have been performed through a Genetic Algorithm (GA) (Li, Huang and Newman, 2008) implemented in software R (Team, 2019). Besides that, to relax the computational burden, the experts manually configured a product family variant, for each niche, to serve as a starting point for GA searching as illustrated in Figure 50 (Daly et al., 2012). With a population size of 50 product family variants, crossover probability equals 0.8, mutation probability equivalent to 0.1, and elitism probability set to 0.1, the GA found the results presented in Table 40, after 200 generations.

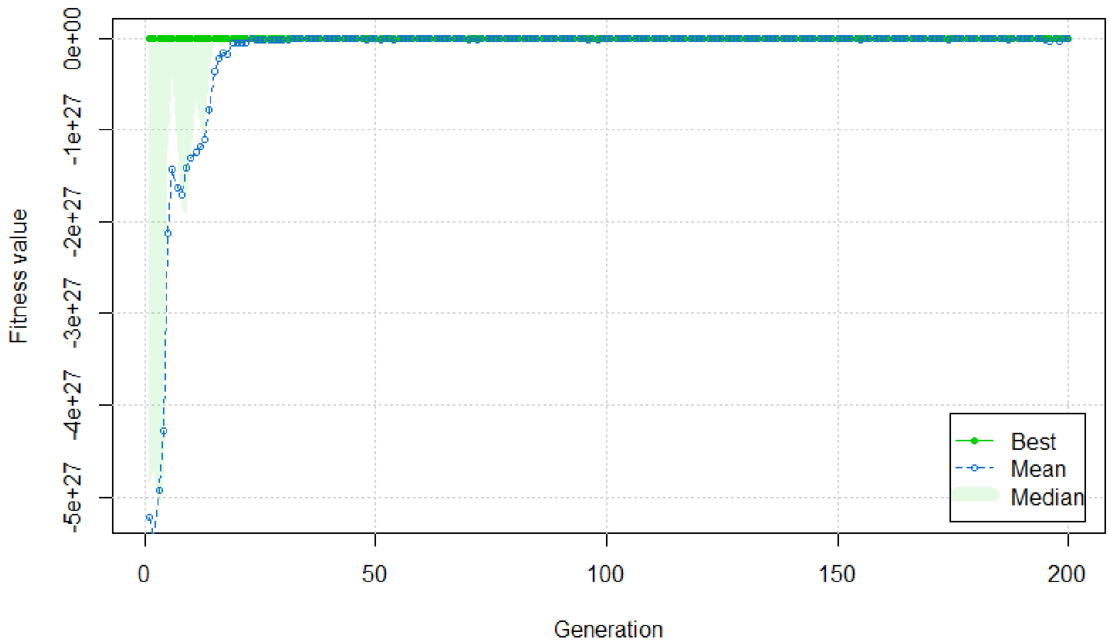


Figure 50. Example of the genetic algorithm evolution process.

Table 40. Results of the configuration process.

Ms	Product family variant $(\vec{X})$	M_k	S_{Ms}	Pr	Q	C_v [USD/SKU]	P [USD/SKU]	V_{Ms} [USD/year]
Ms1.1	[DPI1.5, DPI2.2, DPI5.2, DPI4.7, DPI11.1, DPI7.3]	773	10.0%	13.9%	11	36,650.50	157,500.00	1,329,344.50
Ms1.2	[DPI1.5, DPI2.2, DPI5.18, DPI4.7, DPI11.1, DP12.1, DPI7.3]	773	7.0%	17.9%	9	44,850.50	157,500.00	1,126,495.00
Ms1.3	[DPI1.5, DPI2.2, DPI5.38, DPI4.21, DPI11.1, DPI7.3]	773	3.0%	17.4%	4	46,255.50	157,500.00	444,978.00
Ms2.1	[DPI1.5, DPI2.2, DPI5.32, DPI4.20, DPI11.1, DPI12.4, DPI7.1, DPI8.4, DPI9.1, DPI10.1, DPI6.3, DP13.1, DP3.1]	773	38.0%	19.2%	56	170,480.50	245,000.00	4,173,092.00
Ms2.2	[DPI1.5, DPI2.2, DPI5.32, DPI4.20, DPI11.1, DPI12.4, DPI7.1, DPI8.1, DPI9.1, DPI10.1, DPI6.3, DP13.1, DP3.1]	773	26.0%	19.2%	39	170,293.00	245,000.00	2,913,573.00
Ms2.3	[DPI1.5, DPI2.2, DPI5.39, DPI4.20, DPI11.1, DPI12.4, DPI7.1, DPI8.1, DPI9.1, DPI10.1, DPI6.3, DPI13.1, DPI3.1]	773	11.0%	25.2%	21	167,793.00	260,000.00	1,936,347.00
All (PF1)	*[DPI1.5, DPI2.2, DPI5.2, DPI5.18, DPI5.32, DPI5.38, DPI5.39, DPI4.7, DPI4.20, DPI 4.21, DPI11.1, DPI12.1, DPI12.4, DPI7.1, DPI7.3, DPI8.1, DPI8.4, DPI9.1, DPI10.1, DPI6.3, DPI13.1, DPI3.1]*	*773*	*95.0%*	*18.2%*	*141*	*20,748,670.50*	*32,672,500.00*	*11,923,829.50*

The last row of Table 40 represents the overall results of the product family $PF1$, from which it is possible to obtain the product family profit (V) in step $S_{4.4}$. The maximum profit found, $V = 11.9\text{x}10^6$ $[USD/year]$, is higher than the expected profit $V_e = 2\text{x}10^6$ $[USD/year]$, indicating in this way that it is worth it to invest in the design of the product family. With the product family profit considered satisfactory, i.e. $V \geq V_e$, those most profitable variants have been computed, and the design parameter instances compounding them have been selected to integrate the modular product family structure at the stage $S_{4.5}$. The final product family structure, compound by 9 modules, 22 design parameter instances, and capable of generating up to 120 variants is represented by the Generic Bill-of-Material shown in Figure 51 (Li, Huang and Newman, 2008). According to the MDM proposition, this is the structure that should be developed in the subsequent design stages of the product development process.

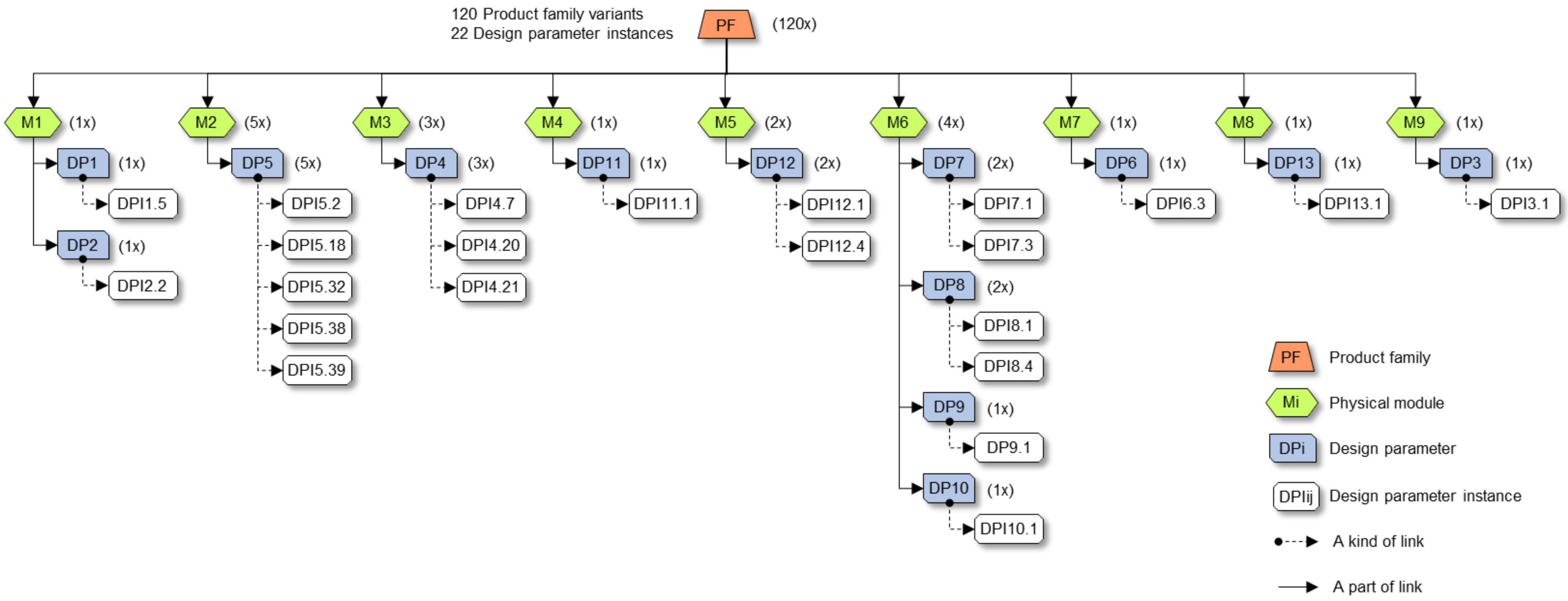


Figure 51. Final product family structure.

6.5 Discussion of the Results

This section discusses the application of MDM in the light of its two expected outcomes: (i) the modular product family structure, (ii) and the decision on investing or not in the product family design. With this respect, the first thing noted was that, although the price (P) has a negative influence on the choice probability (Pr), the model presented the tendency to adjust it as high as possible to maximize the partial profit (V_{Ms}). That is because the marginal increase in partial profit derived from the increment in price is more substantial than the one retrieved from the augmentation in choice probability. To attenuate this behavior, a constraint has been implemented to limit the maximum price as being equals or lower than the price of its most expensive competing alternative in the same market niche (M_s). As a result, the configuration model adjusted the price of each variant presented in Table 40 to its maximum value. In contexts of low data availability, where the weights (w) are estimated through the AHP, this undesired effect tends to aggravate as the number of customers' desired attributes (A) grows. To overcome this limitation, one alternative solution would be to consider the price as a parameter instead of a decision variable of the configuration model. In this sense, the price should be intentionally defined based on the strategy of product family positioning established at the step $S_{1.1}$.

As well as price, other variables might influence the outcomes of the configuration model build trough the MDM method. To better understand the stability of the model, a sensitivity analysis was performed (Hsiao et al., 2013) guided by the following question: What are the values of the influencing variables (Var) that would keep the MDM outcomes the same? To answer this question, 8 scenarios were evaluated in comparison to the one obtained in the last section (Scenario 0). The strategy adopted for the analysis was to change one influencing variable at a time while keeping the others constant. The results of this process are presented in Table 41.

Table 41. Sensitive analysis.

Scenario	Var	ΔVar	M_k	S_{Ms}	Pr	Q	C_v [USD/SKU]	P [USD/SKU]	V [USD/SKU]	ΔV
0 - Highest price (base)	P	-	773	95%	18.2%	141	20,748,670.50	32,672,500.00	11,923,829.50	-
1 - Reduced price	P	-14.3%	773	95%	19.4%	150	21,674,605.00	29,607,857.14	7,933,252.14	-33.5%
2 - Reduced market size	M_k	-10.2%	694	95%	18.2%	128	18,726,009.00	29,632,500.00	10,906,491.00	-8.5%
3 - Different share of market niches	S_{Ms}	random	773	95%	16.6%	127	13,898,271.00	26,177,500.00	12,279,229.00	3.0%
4 - Variation in the DPI's variable cost	C_v	±97.0%	773	95%	18.2%	141	17,661,010.43	32,672,500.00	15,011,489.57	25.9%
5 - Variation in the DPI's engineering attribute values	E_v	±6.0%	773	95%	18.3%	142	20,919,151.00	32,917,500.00	11,998,349.00	0.6%
6 - Variation in the engineering attribute weights	w	±40.0%	773	95%	18.1%	140	20,578,377.50	32,427,500.00	11,849,122.50	-0.6%
7 - Variation in the engineering attribute values of competing alternatives	E_v	±13.0%	773	95%	18.2%	141	20,748,858.00	32,672,500.00	11,923,642.00	0.0%
8 - Addition of one more competing alternative within each market niche	J	+20.0%	773	95%	15.1%	116	17,141,865.50	26,925,000.00	9,783,134.50	-18.0%

In Scenario 1, the price of each variant presented in Table 40 has been reduced in 14.3%, consequently the product family profit (V) dropped 33.5% if compared to Scenario 0. Even with this reduction in profit, the product family structure presented in Figure 51, and the decision to invest in the product family design $(V \geq V_e)$ kept the same. Following the same reasoning, no changes in MDM outcomes have been found by reducing the market size (M_k) in up to 10.2% in Scenario 2. Regarding Scenario 3, the share of each market niche (S_{Ms}) has been randomly altered without changing the total share, i.e. $\sum_{i=Ms1.1}^{Ms2.3} S_{Msi} \leq 0.95$. As a result, an increase of 3.0% in profit and no modifications in MDM outcomes have been perceived. In Scenario 4, a random variation of up to ±97.0% in the variable cost (C_v) of each design parameter instance (DPI) has been found with no implications in MDM outputs as well. That is quite interesting, and the reason for that lies in the fact the variation of each $DPI's$ variable cost is compensated when summed to compound the variable cost of the product family variant (PF_v). The same interpretation holds for the engineering attribute values (E_v) in Scenario 5, but at lower variation rates. In Scenario 6, a random variation of up to ±40.0% in each engineering attribute weight (w) has been percieved with a small change in profit and no changes in MDM outcomes. That is because in AHP the sum of weights is equal to one, i.e. $\sum_{i=1}^{n} w_i = 1$, therefore, the change in one element results in an inverse proportional distribution to the others. Scenario 7, in turn, kept the same results of Scenario 0, with a random variation of up to ±13.0% employed in engineering attribute values of competing alternatives. Finally, in Scenario 8, including up to one competing alternative (J) in each market niche, the total choice probability and profit decrease by 17.0% and 18.0% respectively, but no changes in MDM outcomes are noted. This proportional decrease has to do with the independence of irrelevant alternatives (IIA) property, which consists of an important characteristic of those models where the customer choice probability (Pr) derives from the AHP. This property implies that if a given alternative is changed such that its market share increases, the increased change in market share of the alternative will result in an equal percent decrease in market share for all other alternatives in the choice set. For such cases in which this property is undesirable, the Nested Logit formulations can be used to relax this assumption, as indicated in Table 10 (Chen, Hoyle and Wassenaar, 2013).

In summary, the product family profit of each scenario presented in Table 41 is higher than the expected profit, i.e. $V \geq V_e$. It indicates that the building of the modular product family structure, the first MDM outcome, is more sensitive to the variables analyzed than the decision on investment in the product family design, the second MDM outcome. In other words, it was the product family structure that limited the increase of ΔVar. Regarding the first outcome, the variable that influences it the most is the engineering attribute value (Ev), since a small change on it (±6.0%), might originate a different product family structure. The second outcome, in turn, is most influenced by the price (P) followed by the variable cost (Cv). It can be seen by its respective variations in profit (-33.5% and +25.9%) presented in Scenarios 1 and 4. These results indicate, that even from a deterministic perspective and under a context of low data availability, the outcomes of the MDM are reasonably stable.

Finally, the product family $PF1$ was supposed to be a family of Autonomous Mobile Palletizers, however, due to the high variable cost of those design parameter instances related to the autonomous mobile cobots (AMC) technology, i.e. $\{DPI4.9{:}\,11, DPI4.17{:}\,19, DPI4.23{:}\,25, DPI4.29{:}\,31\}$, the model ended up selecting those related to the cobot technology, i.e, $\{DPI4.7, DPI4.20, DPI4.21\}$. As a result, a family of Collaborative Robotic Palletizers has been conceptually designed. In the same way that the configuration model opted for those cobot-related $DPI's$, in case of the technology becomes less expensive, or even in case of implementing the product family $PF2$, both the AMC or unmanned aerial vehicles (UAV) can give rise to new instances of module $M3$ as illustrated in Figure 52(a) and (b) respectively.

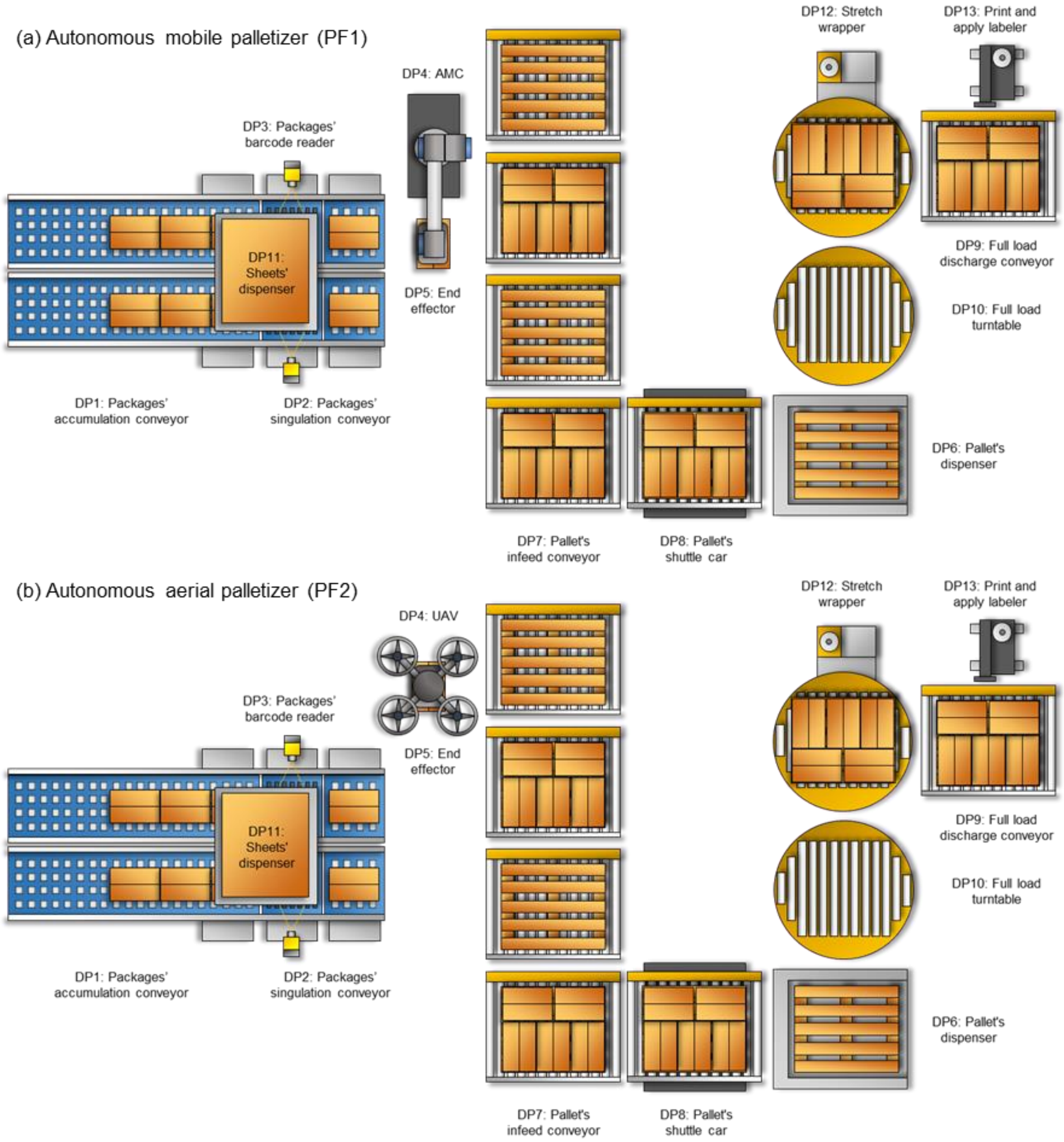


Figure 52. (a) Autonomous mobile palletizer; (b) Autonomous aerial palletizer.

6.6 Conclusions

This paper introduced the Market-Driven Modularity (MDM), an integrated method to conceptually design modular product families that balance the fulfillment of market needs and the resulting profitability to achieve them. To prevent the development of non-profitable product families resulting from the missing-link between marketing and engineering domains, the MDM uses the discrete choice modeling for quantifying the customers' preferences, modularity as a mechanism to provide product variety, the product family as a strategy to manage the trade-off between the variety and cost, and profit as a moderating variable to balance the level of accomplishment of the customers' needs.

In order to provide a better understanding of the proposed method, this paper presented an illustrative application of the MDM within the product development process of a midsize company that produces capital goods. In this sense, the MDM was employed to conceptually design a new product family generation in a scenario of low data availability, where the context unit was the Brazilian Palletizer Market, and the unit of analysis was a Family of Collaborative Robotic Palletizers. The results indicated, that even from a deterministic perspective and under a context of low data availability, the two outcomes of the MDM method, (i) the modular product family structure and (ii) the decision on investment in the product family design, are reasonably stable.

From this application, it was possible to identify the following contributions. (i) The MDM integrated the four classes of design problems (Cp_1, Cp_2, Cp_3, Cp_4) prevalent in the literature into a single method to conceptually design modular product families. Although it has been implicitly done by the work of Jiang and Allada (2005), this explicit and systematic approach proposed by the MDM debuts another class of design problems for those Integrated Methods for Product Family Design. (ii) It is well known that this integration might come together with a certain level of inflexibility, that MDM tries to overcome by mixing and matching the techniques to execute each step of the method according to the context it is inserted. Concerning the Customers' Choice Modelling (Cp_2), the MDM (iii) expands the boundaries of the Market Segmentation Grid (Meyer and Lehnerd, 1997) to a multidimensional perspective, and uses the (iv) Analytic Hierarchy Process (Saaty, 2008) to model the customers' choice probabilities for multiple market segments/niches in contexts of low data availability. With respect to the Product Family Modelling (Cp_3), (v) the MDM uses techniques such as the Classification Scheme and the Analysis of Existing Technical Systems (Pahl *et al.*, 2007) to assist in the abductive process of design parameters formulation. (vi) Moreover, it introduces the Functional to Physical Decomposition, an approach to deal with functional and physical modularity in product family architectures. (vii) Additionally, the MDM presents a heuristic to estimate the variable costs of design parameters at the early design stages. Regarding the Product Family Configuration (Cp_4), different from other methods that adopt it to configure a single variant (Gauss, Lacerda and Miguel, 2020) (Article 1), (viii) the MDM uses this class for building the product family structure. (ix) More than that, when combined with the Product Family Planning and Positioning (Cp_4), it generates not only a lucrative product family structure, but also provides the enterprise-level indicators to support the decision-making on invest or not in the product family design.

We are at the beginning of endeavor towards modularity into product family design, therefore there are some limitations concerning the MDM method, which include: (i) So far, the MDM tackles the design of modular product families from a deterministic perspective, and although it slightly approaches the variation of the influencing variables on its results, it does by only changing one variable at a time. But what would be the results with all variables changing together? (ii) Besides that, the MDM does not consider the profile (risk taker/averse) of the decision-maker into the decision on investing or not in the product family design. (iii) Moreover, the proposed method conceptually designs product families for a single and static scenario. But what would be the product family structure to handle multiple and dynamic scenarios under deep uncertainty? In our point of view, these are not only research limitations but also future research directions on MDM. Besides that, we also encourage the scholars and practitioners to apply the MDM in the design process of consumer and intermediate goods, as well as to compare the results of the same product family design under contexts of low and high data availability.

7 CONCLUSIONS

This work used design science research to integrate marketing, engineering, and economic aspects into a single approach to conceptually design lucrative product families. In this context, the traditional stages of design science research methodologies were decomposed into 32 steps to provide practical guidance on the artifacts' design and evaluation. By following these steps, a field problem gave rise to a method, entitled Market-Driven Modularity (MDM), which was validated through a series of practical applications and experts' judgments.

The main outcomes of this process are synthesized in Figure 53, wherein the design proposition, compound by the MDM (artifact) along with its construction and contingency heuristics, presented major implications in three fields of study: product family design, modularity, and design science research. The implications are of two natures, the contributions, and future research problems. The contributions consist of the MDM outcomes that overcome the current research problems/limitations of a theoretical framework. The future research problems, in turn, are the shortcomings of a theoretical framework revealed by the artifact's design and evaluation.

In this sense, the first contribution of this research lies in the integrative connection among existing methods to design module-based product families. An integration performed in the form of a functional model and structured classes of design problems, with both together serving as a meta-method for organizing the research in the field of module-based product design as well as a roadmap for implementing it in industry. This issue is particularly important given the broad array of methods created over the past years, that consequently exist, in isolation from one other (Borjesson and Hoelttae-Otto, 2014; Otto *et al.*, 2016).

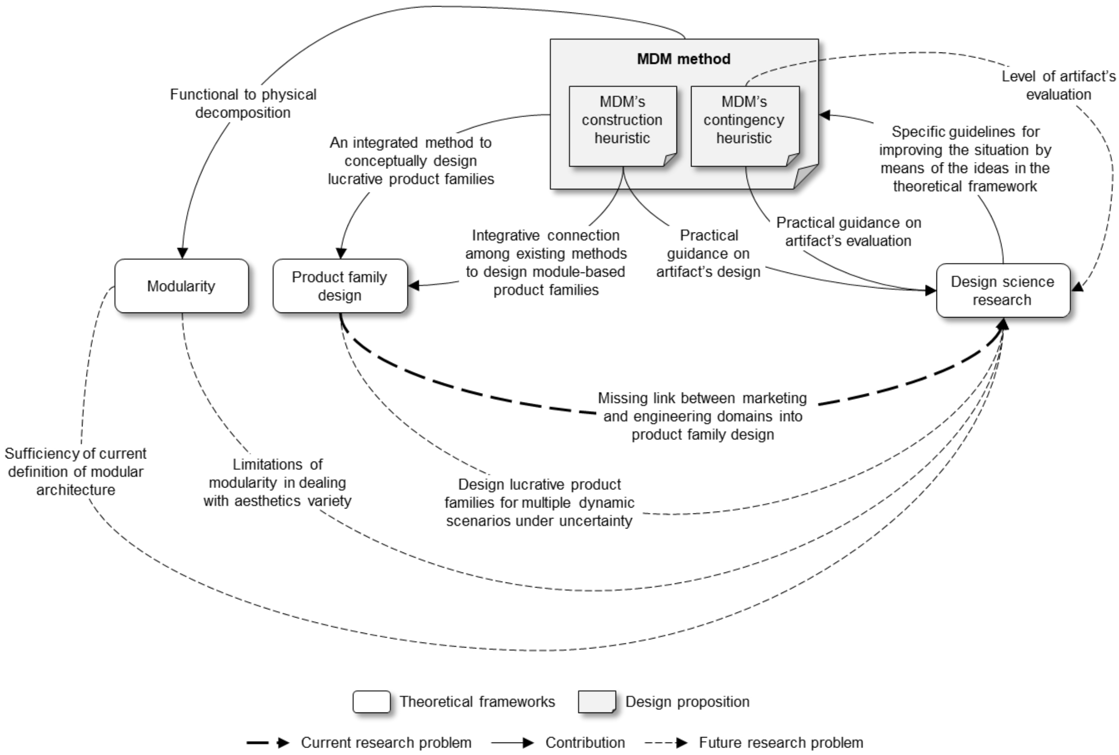


Figure 53. Implications of MDM.

The second contribution, that responds to the research question posed in this work, is the integration of marketing, engineering, and economic aspects into a single method to conceptually design gainful product family structures. This is beneficial for two reasons. First, the demand of potential product family variants can be compared against competing alternatives on the market, so that the economic benefits of a product family design can be assessed before making relevant investments on it, preventing in this way the development of non-profitable product families (Simpson *et al.*, 2014). Then, by highlighting the most valuable combinations, manufacturers can prioritize the modules to be developed at subsequent design stages (Colombo *et al.*, 2019), which directly implies the reduction of design and manufacturing costs as well as in shorter time-to-market (Ulrich and Tung, 1994).

These two contributions enhance the body of knowledge on product family design, that from the MDM's limitations, promotes another cycle of investigation on how to conceptually design lucrative product families for multiple dynamic scenarios under uncertainty. A perspective capable of improving the adaptability of enterprises to meet uncertain markets, customer requirements, technologies, policies, and regulations (Han *et al.*, 2019).

The MDM has also contributed to the theoretical framework of modularity by proposing an approach to deal with the functional and physical decomposition of product family architectures in closed-loop control. This approach prevents the physical interactions constraining the functional modules, favoring in this way the obtention of uncoupled or decoupled designs (Suh, 1998). This way of thinking modularity led us to question the sufficiency of the current definition of modular architectures when the intensity of relationships is considered. This understanding might be beneficial for identifying modular patterns in integral architectures when the decomposition of existing product structures is required. Besides that, another future research direction, that emerged from the MDM's contingency heuristic, was the potential limitation of modularity in dealing with aesthetics variety, a relevant requirement on the design of consumer durables (Fung and Chong, 2007).

Regarding the design science research, two other MDM's contributions have been found. The first was the practical guidance on the abductive process of the artifact's design, an issue slightly tackled by Gacenga et al. (2012) but not well studied and documented since then. The reasoning behind it lies in the fact that it does not matter how robust the artifacts' evaluation is if only low utility artifacts compound the design space. This implies that, at the end of the research process, no artifact or only low utility ones might be validated. In other words, by enhancing the quality of design space, the probability of validating satisfactory artifacts increases, becoming the overall research process more efficient. The second contribution was the practical guidance on the artifact's evaluation, a subject studied in depth by others research (Gill and Hevner, 2013; Venable, Pries-Heje and Baskerville, 2016; Gassel, Reymen and Maas, 2019), but which still lacks detailed procedures on how testing and validating artifacts. Coupled with that arises the need to better define which levels of artifact's evaluation lead to satisfactory results in terms of pragmatic validity and practical relevance.

We are at the beginning of endeavor towards modularity into product family design, therefore there are some limitations concerning the MDM, which include: So far, the MDM tackles the design of modular product families from a deterministic perspective, and although it slightly approaches the variation of the influencing variables on its results, it does by only changing one variable at a time instead of all together. Besides that, the proposed method conceptually designs product families for a single and static scenario, rather than multiple and dynamic ones. Additionally, the MDM does not consider the profile (risk taker/averse) of the decision-maker into the decision on investing or not in the product family design.

REFERENCES

Abbas, O. A. (2008) 'Comparisons Between Data Clustering Algorithms', *The International Arab Journal of Information Technology*, 5(3), pp. 320–325.

Adhitama, Y. and Rosenstiel, W. (2015) 'Module Family Design for Modular Product', *Advances in Intelligent Systems and Computing*, 323, pp. 465–476. doi: 10.1007/978-3-319-11310-4_40 465.

Adler, M. J. and van Doren, C. (1972) *How to read a book*. New York: A Touchstone Book Published by Simon & Schuster.

Agard, B. and Bassetto, S. (2013) 'Modular design of product families for quality and cost', *International Journal of Production Research*, 51(6), pp. 1648–1667. doi: 10.1080/00207543.2012.693963.

van Aken, J. E. (2005) 'Management research as a design science: Articulating the research products of mode 2 knowledge production in management', *British Journal of Management*, 16(1), pp. 19–36. doi: 10.1111/j.1467-8551.2005.00437.x.

van Aken, J. E., Berends, H. and van der Bij, H. (2012) *Problem solving in organizations*. 2nd edn. Cambridge: University Press Cambridge.

van Aken, J. E. and Romme, G. (2009) 'Reinventing the future: Adding design science to the repertoire of organization and management studies', *Organisation Management Journal*, 6(1), pp. 5–12. doi: 10.1057/omj.2009.1.

van Aken, J., Chandrasekaran, A. and Halman, J. (2016) 'Conducting and publishing design science research: Inaugural essay of the design science department of the Journal of Operations Management', *Journal of Operations Management*. Elsevier B.V., 47–48, pp. 1–8. doi: 10.1016/j.jom.2016.06.004.

AlGeddawy, T. and ElMaraghy, H. (2013) 'Reactive design methodology for product family platforms, modularity and parts integration', *CIRP Journal of Manufacturing Science and Technology*, 6(1), pp. 34–43. doi: 10.1016/j.cirpj.2012.08.001.

Alizon, F., Shooter, S. B. and Simpson, T. W. (2007) 'Improving an existing product family based on commonality/diversity, modularity, and cost', *Design studies*, 28(4), pp. 387–409. doi: 10.1016/j.destud.2007.01.002.

Alonso, J. A. and Lamata, M. T. (2006) 'Consistency in the analytic hierarchy process: A new approach', *International Journal of Uncertainty, Fuzziness and Knowlege-Based Systems*, 14(4), pp. 445–459. doi: 10.1142/S0218488506004114.

Alturki, A., Gable, G. G. and Bandara, W. (2011) 'A design science research roadmap', in *DESRIST*. Milwaukee: Springer.

Andersen, A.-L. *et al.* (2017) 'Towards a generic design method for reconfigurable manufacturing systems: Analysis and synthesis of current design methods and evaluation of supportive tools', *Journal of Manufacturing Systems*, 42, pp. 179–195. doi: 10.1016/j.jmsy.2016.11.006.

Andre, S. and Elgh, F. (2018) 'Modeling of transdisciplinary engineering assets using the design platform approach for improved customization ability', *Advanced*

Engineering Informatics. Elsevier, 38, pp. 277–290. doi: 10.1016/j.aei.2018.07.006.

Antunes, J. A. V. *et al.* (2008) *Sistemas de produção: Conceitos e práticas para projetos e gestão da produção enxuta [Production systems: Concepts and practices for lean production projects and management]*. Porto Alegre: Bookman.

Aoussat, A. *et al.* (1995) 'Competitive analysis and product design', in *Proceedings of the 1995 International Conference on Engineering Design,*. Praha, pp. 879–884.

Arciniegas, A. J. R. and Kim, H. M. (2011) 'Optimal component sharing in a product family by simultaneous consideration of minimum description length and impact metric', *Engineering Optimization*, 43(2), pp. 175–192. doi: 10.1080/0305215X.2010.486032.

Asan, U., Polat, S. and Serdar, S. (2004) 'An integrated method for designing modular products', *Journal of Manufacturing Technology Management*, 15(1), pp. 29–49. doi: 10.1108/09576060410512257.

Askhøj, C. *et al.* (2019) 'Deciding on the total number of product architectures', *Concurrent Engineering Research and Applications*, pp. 1–12. doi: 10.1177/1063293X19888968.

ATLAS.ti 8 Windows | ATLAS.ti (2019). Available at: https://atlasti.com/product/v8-windows/ (Accessed: 28 February 2019).

Aungst, S., Barton, R. and Wilson, D. (2003) 'The Virtual Integrated Design Method', *Quality Engineering*. Taylor & Francis, 15(4), pp. 565–579. doi: 10.1081/QEN-120018389.

Aydin, M. and Ulutas, B. H. (2016) 'A new methodology to cluster derivative product modules: an application', *International Journal of Production Research*, 54(23), pp. 7091–7099. doi: 10.1080/00207543.2016.1143133.

Baldwin, C. Y. and Clark, K. B. (2000) *Design Rules: The Power of Modularity*. Cambridge, Massachusetts: MIT Press.

Bardin, L. (1993) *L'analyse de contenu [Content Analysis]*. Paris: Presses Universitaires de France Le Psychologue.

Barnett-Page, E. and Thomas, J. (2009) 'Methods for the synthesis of qualitative research: A critical review', *BMC Medical Research Methodology*, 9(1), pp. 1–11. doi:

10.1186/1471-2288-9-59.

Baskerville, R., Pries-Heje, J. and Venable, J. (2009) ‘Soft design science methodology’, in *DESRIST*. Malvern: ACM.

Baylis, K., Zhang, G. and McAdams, D. A. (2018) ‘Product family platform selection using a Pareto front of maximum commonality and strategic modularity’, *Research in Engineering Design*, 29(4), pp. 547–563. doi: 10.1007/s00163-018-0288-5.

Bejlegaard, M. *et al.* (2018) ‘Methodology for reconfigurable fixture architecture design’, *CIRP Journal of Manufacturing Science and Technology*, 23, pp. 172–186. doi: 10.1016/j.cirpj.2018.05.001.

Bezdek, J. C. (1981) *Pattern Recognition with Fuzzy Objective Function Algorithms*. 1st edn. Edited by M. Nadler. New York, NY: Plenum Press. doi: 10.1007/978-1-4757-0450-1.

Bonjour, E. *et al.* (2009) ‘A Fuzzy Method for Propagating Functional Architecture Constraints to Physical Architecture’, *Journal of Mechanical Design*, 131(6), p. 061002. doi: 10.1115/1.3116253.

Bonvoisin, J. *et al.* (2016) ‘A systematic literature review on modular product design’, *Journal of Engineering Design*, 27(7), pp. 488–514. doi: 10.1080/09544828.2016.1166482.

Borjesson, F. and Hoelttae-Otto, K. (2014) ‘A module generation algorithm for product architecture based on component interactions and strategic drivers’, *Research in Engineering Design*, 25(1), pp. 31–51. doi: 10.1007/s00163-013-0164-2.

Browning, T. R. (2001) ‘Applying the design structure matrix to system decomposition and integration problems: A review and new directions’, *IEEE Transactions on Engineering Management*, 48(3), pp. 292–306. doi: 10.1109/17.946528.

Brunton, G., Stansfield, C. and Thomas, J. (2012) ‘Finding relevant studies’, in *An introduction to systematic reviews*. London: Sage Publications Ltd, pp. 107–134.

Bunge, M. (1980) *Epistemologia*. São Paulo: TA Queiroz Editora Ltda.

Burg, J. Van (2011) ‘Kwaliteitscriteria voor ontwerpgericht wetenschappelijk onderzoek [Quality criteria for design science research]’, in Aken, J. Van and Andriessen, D. D. H. (eds) *Handboek ontwerpgericht wetenschappelijk onderzoek :*

wetenschap met effect. Boom Lemma Uitgevers, pp. 145–164.

Chan, L.-K. and Wu, M.-L. (2002) *Quality function deployment: A literature review*, *European Journal of Operational Research*. doi: 10.1016/S0377-2217(02)00178-9.

Charter, M. and Tischner, U. (2017) *Sustainable Solutions: Developing Products and Services for the Future*. Routledge.

Chen, S. L., Jiao, R. J. and Tseng, M. M. (2009) ‘Evolutionary product line design balancing customer needs and product commonality’, *CIRP Annals - Manufacturing Technology*, 58(1), pp. 123–126. doi: 10.1016/j.cirp.2009.03.014.

Chen, W., Hoyle, C. and Wassenaar, H. J. (2013) *Decision-Based Design: Integrating Consumer Preferences into Engineering Design*. London: Springer-Verlag.

Cheng, Q. *et al.* (2017) ‘Design of adaptable product platform for heavy-duty gantry milling machines based on sensitivity design structure matrix’, *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, 231(24), pp. 4495–4511. doi: 10.1177/0954406216670685.

Cheng, X., Xiao, R. and Wang, H. (2018) ‘A method for coupling analysis of association modules in product family design’, *Journal of Engineering Design*, 29(6), pp. 327–352. doi: 10.1080/09544828.2018.1487531.

Chowdhury, S. *et al.* (2016) ‘New modular product-platform-planning approach to design macroscale reconfigurable unmanned aerial vehicles’, *Journal of Aircraft*, 53(2), pp. 309–322. doi: 10.2514/1.C033262.

Clarivate Analytics (2019) *LibGuides: Web of Science platform: Web of Science: Summary of Coverage*, *Web of Science Platform*. Available at: https://clarivate.libguides.com/webofscienceplatform/coverage (Accessed: 21 February 2019).

Coetzee, R. (2019) ‘Towards designing an artefact evaluation strategy for human factors engineering: A lean implementation model case study’, *South African Journal of Industrial Engineering*, 30(3), pp. 289–303. doi: 10.7166/30-3-2244.

Cole, R. (2005) ‘Being proactive: Where action research meets design research’, in *Proceedings of the Twenty-Sixth International Conference on Information Systems*. Las Vegas.

Colombo, E. F. *et al.* (2019) 'Value analysis for customizable modular product platforms: theory and case study', *Research in Engineering Design*. doi: 10.1007/s00163-019-00326-4.

Content - How Scopus Works - Scopus - | Elsevier solutions (2017). Available at: https://www.elsevier.com/solutions/scopus/how-scopus-works/content (Accessed: 21 February 2019).

Cox, J. F. and Schleier, J. G. (2010) *Theory of Constraints Handbook*. New York: McGraw-Hill. doi: 10.1002/cbdv.200490137.

Dahmus, J. B., Gonzalez-Zugasti, J. P. and Otto, K. N. (2001) 'Modular product architecture', *Design Studies*, 22(5), pp. 409–424. doi: 10.1016/S0142-694X(01)00004-7.

Dalkey, N. (1969) 'An experimental study of group opinion: The Delphi method', *Futures*. IPC Business Press Limited, 1(5), pp. 408–426. doi: 10.1016/S0016-3287(69)80025-X.

Daly, S. R. *et al.* (2012) 'Design Heuristics in Engineering', *Journal of Engineering Education*, 101(4), pp. 601–629.

Denyer, D., Tranfield, D. and Van Aken, J. E. (2008) 'Developing design propositions through research synthesis', *Organ. Stud.*, 29(2), pp. 249–269.

Dong, M., Shao, X. and Xiong, S. (2011) 'Flexible optimization decision for product design agility with embedded real options', in *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, pp. 1431–1446. doi: 10.1177/09544054JEM2216.

Dresch, A., Lacerda, D. P. and Antunes Jr, J. A. V. (2015) *Design Science Research*. Springer. doi: 10.1007/978-3-319-07374-3.

Du, G., Jiao, J. and Chen, M. (2014) 'Joint optimization of product family configuration and scaling design by Stackelberg game', *European Journal of Operational Research*, 232(2), pp. 330–341. doi: 10.1016/j.ejor.2013.07.021.

Du, X., Jiao, J. and Tseng, M. M. (2001) 'Architecture of product family: Fundamentals and methodology', *Concurrent Engineering Research and Applications*, 9(4), pp. 309–325. doi: 10.1177/1063293X0100900407.

Du, X., Jiao, J. and Tseng, M. M. (2006) 'Understanding customer satisfaction in product customization', *International Journal of Advanced Manufacturing Technology*, 31(3–4), pp. 396–406. doi: 10.1007/s00170-005-0177-8.

Eekels, J. and Roozenburg, N. F. M. (1991) 'A methodological comparison of the structures of scientific research and engineering design: their similarities and differences', *Design Studies*, 12(4), pp. 197–203. doi: 10.1016/0142-694X(91)90031-Q.

Eisenhardt, K. M. (1989) 'Building theories from case study research', *Acad. Manag.*, 14, pp. 532–550.

ElMaraghy, H. and AlGeddawy, T. (2012) 'New dependency model and biological analogy for integrating product design for variety with market requirements', *Journal of Engineering Design*, 23(10–11), pp. 719–742. doi: 10.1080/09544828.2012.709607.

Emmatty, F. J. and Sarmah, S. P. (2012) 'Modular product development through platform-based design and DFMA', *Journal of Engineering Design*, 23(9), pp. 696–714. doi: 10.1080/09544828.2011.653330.

Erens, F. and Verhulst, K. (1997) 'Architectures for product families', *Computers in Industry*, 33(2–3), pp. 165–178. doi: 10.1016/S0166-3615(97)00022-5.

Everitt, B. S. *et al.* (2011) *Cluster Analysis*. 5th edn. London, UK: John Wiley & Sons.

Fan, B. *et al.* (2015) 'A network methodology for structure-oriented modular product platform planning', *Journal of Intelligent Manufacturing*, 26(3), pp. 553–570. doi: 10.1007/s10845-013-0815-1.

Farrell, R. S. and Simpson, T. W. (2003) 'Product platform design to improve commonality in custom products', *Journal of Intelligent Manufacturing*, 14(6), pp. 541–556. doi: 10.1023/A:1027306704980.

Farrell, R. S. and Simpson, T. W. (2008) 'A method to improve platform leveraging in a market segmentation grid for an existing product line', *Journal of Mechanical Design, Transactions of the ASME*, 130(3), pp. 1–11. doi: 10.1115/1.2829889.

Ferguson, S. M., Olewnik, A. T. and Cormier, P. (2014) 'A review of mass customization across marketing, engineering and distribution domains toward development of a process framework', *Research in Engineering Design*, 25(1), pp. 11–30. doi: 10.1007/s00163-013-0162-4.

Ferguson, S., Olewnik, A. and Cormier, P. (2011) ‘Exploring marketing to engineering information mapping in mass customization: A presentation of ideas, challenges and resulting questions’, in *Proceedings of the ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*. Washington, DC, USA.

Finfgeld, D. L. (2003) ‘Metasynthesis: The state of the art - So far’, *Qualitative Health Research*, 13(7), pp. 893–904. doi: 10.1177/1049732303253462.

Fixson, S. K. (2007) ‘Modularity and commonality research: Past developments and future opportunities’, *Concurrent Engineering Research and Applications*, 15(2), pp. 85–111. doi: 10.1177/1063293X07078935.

Floyd, J. and Fowler, J. (2014) *Survey Research Methods*. 5th edn, *Sage Publications, Inc*. 5th edn. Boston: Sage Publications, Inc.

Forza, C. (2002) ‘Survey research in operations management: A process-based perspective’, *International Journal of Operations and Production Management*, 22(2), pp. 152–194. doi: 10.1108/01443570210414310.

Frandsen, T. (2017) ‘Evolution of modularity literature: a 25-year bibliometric analysis’, *International Journal of Operations and Production Management*, 37(6), pp. 703–747. doi: 10.1108/IJOPM-06-2015-0366.

Fujita, K. and Yoshida, H. (2004) ‘Product variety optimization simultaneous optimization of module combination and module attributes’, *Concurrent Engineering: Research and Applications*, 12(2), pp. 105–118. doi: 10.1177/1063293X04044758.

Fung, R. Y. K. and Chong, P. Y. (2007) ‘An active styling platform for designing and developing product families’, *Journal of Intelligent Manufacturing*, 18(1), pp. 47–58. doi: 10.1007/s10845-007-0006-z.

Gacenga, F. *et al.* (2012) ‘A proposal and evaluation of a design method in design science research’, *Electronic Journal of Business Research Methods*, 10(2), pp. 89–100.

Gassel, F. van, Reymen, I. and Maas, G. (2019) ‘Validating design principles for creative collaboration’, *International Journal of Design Sciences and Technology*, 23(2), p. 125.

Gauss, L., Lacerda, D. P. and Miguel, P. A. C. (2020) *Module-Based Product Family Design: Systematic Literature Review and Meta-Synthesis*.

Gauss, L., Lacerda, D. P. and Sellitto, M. A. (2019) 'Module-based machinery design: a method to support the design of modular machine families for reconfigurable manufacturing systems', *International Journal of Advanced Manufacturing Technology*. Springer, 102(9), pp. 3911–3936. doi: 10.1007/s00170-019-03358-1.

Gershenson, J. K., Prasad, G. J. and Zhang, Y. (2003) 'Product modularity: definitions and benefits', *Journal of Engineering Design*, 14(3), pp. 295–313. doi: 10.1080/0954482031000091068.

Gershenson, J. K., Prasad, G. J. and Zhang, Y. (2004) 'Product modularity: measures and design methods', *Journal of Engineering Design*, 15(1), pp. 33–51. doi: 10.1080/0954482032000101731.

Ghobakhloo, M. (2018) 'The future of manufacturing industry: a strategic roadmap toward Industry 4.0', *Journal of Manufacturing Technology Management*, 29(6), pp. 910–936. doi: 10.1108/JMTM-02-2018-0057.

Gill, T. G. and Hevner, A. R. (2013) 'A fitness-utility model for design science research', *ACM Transactions on Management Information Systems*, 4(2). doi: 10.1145/2499962.2499963.

Gkoulalas-Divanis, A. and Verykios, C. S. (2010) *Association Rule Hiding for Data Mining*. Springer.

Goldratt, E. M. and Cox, J. (1984) *The Goal*. Croton-on-Hudson, NY: North River Press.

Goswami, M., Daultani, Y. and Tiwari, M. K. (2017) 'An integrated framework for product line design for modular products: product attribute and functionality-driven perspective', *International Journal of Production Research*, 55(13), pp. 3862–3885. doi: 10.1080/00207543.2017.1314039.

Gregor, S. (2009) 'Building theory in the sciences of the artificial', in *DESRIST*, pp. 7–8.

Gregor, S. and Jones, D. (2007) 'The Anatomy of a Design Theory', *Journal of the Association for Information Systems*, 8(5), pp. 312–335. doi: 10.17705/1jais.00129.

Gümüş, M. (2014) 'With or without forecast sharing: Competition and credibility under information asymmetry', *Production and Operations Management*, 23(10), pp. 1732–1747. doi: 10.1111/poms.12192.

Han, X. *et al.* (2019) ‘A systematic literature review of product platform design under uncertainty’, *Journal of Engineering Design*. doi: 10.1080/09544828.2019.1699036.

Hanafy, M. and Elmaraghy, H. (2015) ‘A modular product multi-platform configuration model’, *International Journal of Computer Integrated Manufacturing*, 28(9), pp. 999–1014. doi: 10.1080/0951192X.2014.941407.

Hilier, F. and Lieberman, G. (2015) *Introduction to Operational Research*. 10th edn. New York: McGraw-Hill. doi: 10.2307/2077150.

Hölttä-Otto, K. and De Weck, O. (2007) ‘Degree of modularity in engineering systems and products with technical and business constraints.’, *Concurrent Engineering Research and Applications*, 15(2), pp. 113–126.

Hou, W. *et al.* (2017) ‘Modular platform optimization in conceptual vehicle body design via modified graph-based decomposition algorithm and cost-based priority method’, *Structural and Multidisciplinary Optimization*, 55(6), pp. 2087–2097. doi: 10.1007/s00158-016-1629-5.

Hou, W. *et al.* (2018) ‘Product-family shared-component selection based on the consistency constraint function’, *Proceedings of the Institution of Mechanical Engineers, Part D: Journal of Automobile Engineering*, 232(6), pp. 838–849. doi: 10.1177/0954407017707453.

Hsiao, S.-W. and Liu, E. (2005) ‘A structural component-based approach for designing product family’, *Computers in Industry*, 56(1), pp. 13–28. doi: 10.1016/j.compind.2004.10.002.

Hsiao, S.-W. W. *et al.* (2013) ‘An ISM, DEI, and ANP based approach for product family development’, *Advanced Engineering Informatics*. Elsevier Ltd, 27(1), pp. 131–148. doi: 10.1016/j.aei.2012.10.008.

Jensen, L. A. and Allen, M. N. (1996) ‘Meta-Synthesis of Qualitative Findings’, *Qualitative Health Research*, Vol 6(4), pp. 553–556. doi: 10.1177/016344300022005001.

Jiang, L. and Allada, V. (2005) ‘Robust modular product family design using a modified Taguchi method’, *Journal of Engineering Design*, 16(5), pp. 443–458. doi: 10.1080/09544820500287359.

Jiao, J. (1998) *Design for Mass Customization by Developing Product Family*

Architecture. Hong Kong University of Science and Technology.

Jiao, J. R. (2012) 'Product platform flexibility planning by hybrid real options analysis', *IIE Transactions (Institute of Industrial Engineers)*, 44(6), pp. 431–445. doi: 10.1080/0740817X.2011.609874.

Jiao, J., Simpson, T. W. and Siddique, Z. (2007) 'Product family design and platform-based product development: A state-of-the-art review', *Journal of Intelligent Manufacturing*, 18(1), pp. 5–29. doi: 10.1007/s10845-007-0003-2.

Jiao, J. and Tseng, M. M. (1998) 'Fuzzy ranking for concept evaluation in configuration design for mass customization', *Concurrent Engineering: Research and Applications*, 6(3), pp. 189–206. doi: 10.1177/1063293X9800600303.

Jiao, J. and Tseng, M. M. (1999a) 'A methodology of developing product family architecture for mass customization', *Journal of Intelligent Manufacturing*, 10(1), pp. 3–20. doi: 10.1023/A:1008926428533.

Jiao, J. and Tseng, M. M. (1999b) 'A pragmatic approach to product costing based on standard time estimation', *International Journal of Operations and Production Management*, 19(7), pp. 738–755. doi: 10.1108/01443579910271692.

Jiao, J. and Tseng, M. M. (2000) 'Fundamentals of product family architecture', *Integrated Manufacturing Systems*, 11(7), pp. 469–483. doi: 10.1108/09576060010349776.

Jiao, J. and Zhang, Y. (2005) 'Product portfolio planning with customer-engineering interaction', *IIE Transactions (Institute of Industrial Engineers)*, 37(9), pp. 801–814. doi: 10.1080/07408170590917011.

Johannesson, H. *et al.* (2017) 'Development of product platforms: Theory and methodology', *Concurrent Engineering: Research and Applications*, 25(3), pp. 195–211. doi: 10.1177/1063293X17709866.

Jose, A. and Tollenaere, M. (2005) 'Modular and platform methods for product family design: Literature analysis', *Journal of Intelligent Manufacturing*, 16(3), pp. 371–390. doi: 10.1007/s10845-005-7030-7.

Jung, S. and Simpson, T. W. (2016) 'An integrated approach to product family redesign using commonality and variety metrics', *Research in Engineering Design*, 27(4), pp. 391–412. doi: 10.1007/s00163-016-0224-5.

Jung, S. and Simpson, T. W. (2017) ‘New modularity indices for modularity assessment and clustering of product architecture’, *Journal of Engineering Design*, 28(1), pp. 1–22. doi: 10.1080/09544828.2016.1252835.

Kahn, K. B. (2012) *The PDMA Handbook of New Product Development*. New York, NY: John Wiley & Sons.

Kaplan, R. S. and Anderson, S. R. (2007) *Time-Driven Activity-Based Costing: A Simpler and More Powerful Path to Higher Profits*. Boston, MA: Harvard Business School Press.

Kazemzadeh, R. B. *et al.* (2009) ‘Integration of marketing research techniques into house of quality and product family design’, *International Journal of Advanced Manufacturing Technology*, 41(9–10), pp. 1019–1033. doi: 10.1007/s00170-008-1533-2.

Kim, G. *et al.* (2016) ‘Analysis of Architectural Complexity for Product Family and Platform’, *Journal of Mechanical Design*, 138(7), pp. 1–11. doi: 10.1115/1.4033504.

King, J. R. (1980) ‘Machine-component grouping in production flow analysis: An approach using a rank order clustering algorithm’, *International Journal of Production Research*, 18(2), pp. 213–232. doi: 10.1080/00207548008919662.

Ko, Y. T. and Kuo, P. H. (2010) ‘Modeling concurrent design method for product variety’, *Concurrent Engineering Research and Applications*, 18(3), pp. 207–217. doi: 10.1177/1063293X10379762.

Koh, E. C. Y., Caldwell, N. H. M. and Clarkson, P. J. (2013) ‘A technique to assess the changeability of complex engineering systems’, *Journal of Engineering Design*, 24(7), pp. 477–498. doi: 10.1080/09544828.2013.769207.

Kong, F. B. *et al.* (2009) ‘On modular products development’, *Concurrent Engineering Research and Applications*, 17(4), pp. 291–300. doi: 10.1177/1063293X09353974.

Koos Eissen *et al.* (2007) *Sketching: Drawing techniques for product designers*. 13th edn. BIS Publishers.

Kota, S., Sethuraman, K. and Miller, R. (2000) ‘A Metric for Evaluating Design Commonality in Product Families’, *Journal of Mechanical Design*, 122(4), p. 403. doi: 10.1115/1.1320820.

Krishnan, V. and Zhu, W. (2006) ‘Designing a family of development-intensive products’, *Management Science*, 52(6), pp. 813–825. doi: 10.1287/mnsc.1050.0492.

Krishnapillai, R. and Zeid, A. (2006) ‘Mapping product design specification for mass customization’, *Journal of Intelligent Manufacturing*, 17(1), pp. 29–43. doi: 10.1007/s10845-005-5511-3.

Kumar, D., Chen, W. and Simpson, T. W. (2009) ‘A market-driven approach to product family design’, *International Journal of Production Research*, 47(1), pp. 71–104. doi: 10.1080/00207540701393171.

Kusiak, A. and Chow, W. S. (1987) ‘Efficient solving of the group technology problem’, *Journal of Manufacturing Systems*, 6(2), pp. 117–124. doi: 10.1016/0278-6125(87)90035-5.

Lacerda, D. P. *et al.* (2013) ‘Design science research: método de pesquisa para a engenharia de produção [Design Science Research: a research method to production engineering]’, *Gestão Produção*, 20(4), pp. 741–761. doi: 10.1590/S0104-530X2013005000014.

Landis, J. R. and Koch, G. G. (1977) ‘The measurement of observer agreement for categorical data’, *Biometrics*, 33, pp. 159–174.

Li, L. and Huang, G. Q. (2009) ‘Multiobjective evolutionary optimisation for adaptive product family design’, *International Journal of Computer Integrated Manufacturing*, 22(4), pp. 299–314. doi: 10.1080/09511920802014920.

Li, L., Huang, G. Q. and Newman, S. T. (2008) ‘A cooperative coevolutionary algorithm for design of platform-based mass customized products’, *Journal of Intelligent Manufacturing*, 19(5), pp. 507–519. doi: 10.1007/s10845-008-0137-x.

Li, Z. *et al.* (2013) ‘An integrated method for flexible platform modular architecture design’, *Journal of Engineering Design*, 24(1), pp. 25–44. doi: 10.1080/09544828.2012.668614.

Li, Z. *et al.* (2016) ‘A systematic adaptable platform architecture design methodology for early product development’, *Journal of Engineering Design*, 27(1–3), pp. 93–117. doi: 10.1080/09544828.2015.1112366.

Liu, Z., Wong, Y. S. and Lee, K. S. (2010) ‘Modularity analysis and commonality design: A framework for the top-down platform and product family design’, *International Journal of Production Research*, 48(12), pp. 3657–3680. doi: 10.1080/00207540902902598.

Luo, L. (2011) 'Product Line Design for Consumer Durables: An Integrated Marketing and Engineering Approach', *Journal of Marketing Research*, 48(1), pp. 128–139. doi: 10.1509/jmkr.48.1.128.

Ma, J. and Kim, H. M. (2016) 'Product family architecture design with predictive, data-driven product family design method', *Research in Engineering Design*. Springer London, 27(1), pp. 5–21. doi: 10.1007/s00163-015-0201-4.

Ma, S. *et al.* (2016) 'Hierarchical game joint optimization for product family-driven modular design', *Journal of the Operational Research Society*, 67(12), pp. 1496–1509. doi: 10.1057/jors.2016.32.

Malhotra, N. K. and Birks, D. F. (2007) *Marketing Research: An Applied Approach*. 3rd edn. Prentice Hall Inc. Available at: http://capitadiscovery.co.uk/cardiffmet/items/240307.

March, S. T. and Smith, G. F. (1995) 'Design and natural science research on information technology', *Decision Support Systems*, 15(4), pp. 251–266. doi: 10.1016/0167-9236(94)00041-2.

Márkus, A. A. and Váncza, J. (1998) 'Product line development with customer interaction', *CIRP Annals - Manufacturing Technology*, 47(1), pp. 361–364. doi: 10.1016/s0007-8506(07)62851-0.

Martin, M. V and Ishii, K. (2002) 'Design for variety: Developing standardized and modularized product platform architectures', *Research in Engineering Design*, 13(3), pp. 213–235. doi: 10.1007/S00163-002-0020-2.

Mascle, C. and Zhao, H. P. (2008) 'Integrating environmental consciousness in product/process development based on life-cycle thinking', *International Journal of Production Economics*, 112(1), pp. 5–17. doi: 10.1016/j.ijpe.2006.08.016.

Mayring, P. (2014) 'Qualitative content analysis: theoretical foundation, basic procedures and software solution', *The SAGE Handbook of Qualitative Data Analysis*, pp. 170–183. doi: 10.1111/j.1365-2648.2007.04569.x.

Meng, X., Jiang, Z. and Huang, G. Q. (2007) 'On the module identification for product family development', *International Journal of Advanced Manufacturing Technology*, 35(1–2), pp. 26–40. doi: 10.1007/s00170-006-0712-2.

Meng, Z. *et al.* (2014) ‘Predicting configuration performance of modular product family using principal component analysis and support vector machine’, *Journal of Central South University*, 21(7), pp. 2701–2711. doi: 10.1007/s11771-014-2232-9.

Mesa, J. *et al.* (2014) ‘A methodology to define a reconfigurable system architecture for a compact heat exchanger assembly machine’, *International Journal of Advanced Manufacturing Technology*, 70(9–12), pp. 2199–2210. doi: 10.1007/s00170-013-5451-6.

Messac, A., Martinez, M. P. and Simpson, T. W. (2002) ‘Effective product family design using physical programming’, *Engineering Optimization*, 34(3), pp. 245–261. doi: 10.1080/03052150211746.

Meyer, M. H. and Lehnerd, A. P. (1997) *The Power of Product Platforms: Building Value and Cost Leadership*. Free Press.

Miao, C. *et al.* (2017) ‘Coordinated optimisation of platform-driven product line planning by bilevel programming’, *International Journal of Production Research*, 55(13), pp. 3808–3831. doi: 10.1080/00207543.2017.1294770.

Michalek, J. J. *et al.* (2011) ‘Enhancing marketing with engineering: Optimal product line design for heterogeneous markets’, *International Journal of Research in Marketing*, 28(1), pp. 1–12. doi: 10.1016/j.ijresmar.2010.08.001.

Miguel, P. A. C. (2005) ‘Modularity in product development: a literature review towards a research agenda’, *Product: Management & Development*, 3(2), pp. 165–174.

Mikkola, J. H. and Gassmann, O. (2003) ‘Managing modularity of product architectures: Toward an integrated theory’, *IEEE Transactions on Engineering Management*, 50(2), pp. 204–218. doi: 10.1109/TEM.2003.810826.

Montgomery, D. C. and Runger, G. C. (2011) *Applied Statistics and Probability for Engineers*. 5th edn. John Wiley & Sons, Ltd.

Moon, S. K., Kumara, S. R. T. and Simpson, T. W. (2006) ‘A multi-agent system for modular platform design in a dynamic electronic market environment’, in *Proceedings of the ASME Design Engineering Technical Conference*.

Moon, S. K., Park, K. J. and Simpson, T. W. (2014) ‘Platform design variable identification for a product family using multi-objective particle swarm optimization', *Research in Engineering Design*, 25(2), pp. 95–108. doi: 10.1007/s00163-013-0166-0.

Morandi, M. I. W. M. and Camargo, L. F. R. (2015) 'Systematic Literature Review', in *Design Science Research*. Springer, p. 161.

More, A. (2019) *Palletizer Market 2019 Research by Business Opportunities, Top Manufactures, Industry Growth, Industry Share Report, Size, Regional Analysis and Global Forecast to 2024 | Market Reports World - MarketWatch*. Available at: https://www.marketwatch.com/press-release/palletizer-market-2019-research-by-business-opportunities-top-manufactures-industry-growth-industry-share-report-size-regional-analysis-and-global-forecast-to-2024-market-reports-world-2019-05-31 (Accessed: 7 December 2019).

Navarrete, I. A. *et al.* (2013) 'Reduction of product platform complexity by vectorial Euclidean algorithm', *Journal of Mechanical Science and Technology*, 27(11), pp. 3371–3379. doi: 10.1007/s12206-013-0859-3.

Newcomb, P. J., Bras, B. and Rosen, D. W. (1998) 'Implications of modularity on product design for the life cycle', *Journal of Mechanical Design, Transactions of the ASME*, 120(3), pp. 483–490. doi: 10.1115/1.2829177.

Nunamaker, J. F., Chen, M. and Purdin, T. D. M. (1990) 'Systems development in information systems research', *Journal of Management Information Systems*, 7(3), pp. 89–106. doi: 10.1080/07421222.1990.11517898.

OECD (2008) *OECD Glossary of Statistical Terms*. Paris: OECD Publishing. doi: 10.1787/9789264055087-en.

Oliver, S. and Sutcliffe, K. (2012) 'Describing and analysing studies', in *An introduction to systematic reviews*. London: Sage Publications Ltd, pp. 135–152.

Ossadnik, W., Schinke, S. and Kaspar, R. H. (2016) 'Group Aggregation Techniques for Analytic Hierarchy Process and Analytic Network Process : A Comparative Analysis', *Group Decision and Negotiation*. Springer Netherlands, 25(2), pp. 421–457. doi: 10.1007/s10726-015-9448-4.

Otto, K. *et al.* (2016) 'Global Views on Modular Design Research: Linking Alternative Methods to Support Modular Product Family Concept Development', *Journal of Mechanical Design*, 138(7). doi: 10.1115/1.4033654.

Otto, K. and Hölttä-Otto, K. (2007) ‘A multi-criteria assessment tool for screening preliminary product platform concepts’, *Journal of Intelligent Manufacturing*, 18(1), pp. 59–75. doi: 10.1007/s10845-007-0004-1.

Pahl, G. *et al.* (2007) ‘Engineering design: a systematic approach’, *Springer*, (2), p. 617. doi: 10.1111/dsu.12130.

Pakkanen, J., Juuti, T. and Lehtonen, T. (2016) ‘Brownfield Process: A method for modular product family development aiming for product configuration’, *Design Studies*, 45(B), pp. 210–241. doi: 10.1016/j.destud.2016.04.004.

Park, J. *et al.* (2008) ‘A product platform concept development method’, *Journal of Engineering Design*, 19(6), pp. 515–532. doi: 10.1080/09544820802043583.

Pate, D. J., Patterson, M. D. and German, B. J. (2012) ‘Optimizing Families of Reconfigurable Aircraft for Multiple Missions’, *Journal of Aircraft*, 49(6), pp. 1988–2000. doi: 10.2514/1.C031667.

Peffers, K. *et al.* (2007) ‘A Design Science Research Methodology for Information Systems Research’, *Journal of Management Information Systems*, 24(3), pp. 45–78.

Phaal, R. and Muller, G. (2009) ‘An architectural framework for roadmapping: Towards visual strategy’, *Technological Forecasting and Social Change*. Elsevier Inc., 76(1), pp. 39–49. doi: 10.1016/j.techfore.2008.03.018.

Piran, F. A. S. *et al.* (2016) ‘Modularization strategy: analysis of published articles on production and operations management (1999 to 2013)’, *International Journal of Advanced Manufacturing Technology*, 86(1–4), pp. 507–519. doi: 10.1007/s00170-015-8221-9.

Popple, R. A. (2009) *The science of palletizing: How to choose the right system*. Vancouver: Columbia Machine, Inc.

Premachandra, I. M. (2001) ‘An approximate of the activity duration distribution in PERT’, *Computers and Operations Research*, 28(5), pp. 443–452. doi: 10.1016/S0305-0548(99)00129-X.

Rai, R. and Allada, V. (2003) ‘Modular product family design: agent-based Pareto-optimization and quality loss function-based post-optimal analysis’, *International Journal of Production Research*, 41(17), pp. 4075–4098. doi: 10.1080/0020754031000149248.

Randolph, J. (2005) ‘Free-Marginal Multirater Kappa (multirater Kfree): An Alternative to Fleiss’ Fixed-Marginal Multirater Kappa.’, *Online Submission*.

Ripperda, S. and Krause, D. (2017) ‘Cost Effects of Modular Product Family Structures: Methods and Quantification of Impacts to Support Decision Making’, *Journal of Mechanical Design*, 139(2), pp. 1–12. doi: 10.1115/1.4035430.

Rozenfeld, H. *et al.* (2006) *Gestão de Desenvolvimento de Produtos: Uma referência para a melhoria do processo [New product development: A reference model for process improvements]*. 1st edn, *Editora Saraiva*. 1st edn. Edited by F. A. Bravin et al. New York: Editora Saraiva. doi: 10.1002/cbdv.200490137.

Rui, H., Cuervo-Cazurra, A. and Annique Un, C. (2016) ‘Learning-by-doing in emerging market multinationals: Integration, trial and error, repetition, and extension’, *Journal of World Business*. Elsevier Inc., 51(5), pp. 686–699. doi: 10.1016/j.jwb.2016.07.007.

Saaty, T. L. (2008) ‘Decision making with the analytic hierarchy process’, *International Journal of Services Sciences*, 1(1), p. 83. doi: 10.1504/IJSSCI.2008.017590.

Sahin-Sariisik, A. A. *et al.* (2014) ‘A structured approach to platform-driven product planning’, *Engineering Management Journal*, 26(2), pp. 10–23. doi: 10.1080/10429247.2014.11432007.

Scalice, R. K. *et al.* (2015) ‘Modular deployment using TRM and function analysis’, *Technological Forecasting and Social Change*. Elsevier Inc., 92, pp. 1–11. doi: 10.1016/j.techfore.2014.10.018.

Seifoddini, H. K. (1989) ‘Single linkage versus average linkage clustering in machine cells formation applications’, *Computers and Industrial Engineering*, 16(3), pp. 419–426. doi: 10.1016/0360-8352(89)90160-5.

Selim, H. M., Askin, R. G. and Vakharia, A. J. (1998) ‘Cell formation in group technology: Review, evaluation and directions for future research’, *Computers and Industrial Engineering*, 34(1), pp. 3–20. doi: 10.1016/S0360-8352(97)00147-2.

Senge, P. M. (1990) *The fifth discipline: The art and practice of the learning organization*. New York: Currency Doubleday.

Shamsuzzoha, A. and Helo, P. (2017) ‘Development of sustainable platform for modular product family: a case study’, *Production Planning and Control*, 28(6–8), pp.

512–523. doi: 10.1080/09537287.2017.1309715.

Shiba, S. and Walden, D. (2007) ‘Four Practical Revolutions in Management: Systems for Creating Unique Organizational Capability’. New York, NY: CRC Press, p. 758.

Simon, H. A. (1962) *The sciences of the artificial*. 3rd edn, *Proceedings of the American Philosophical Society*. 3rd edn. Cambridge: MIT Press. doi: 10.1016/S0898-1221(97)82941-0.

Simon, H. A. (1996) *The sciences of the artificial*. 3rd edn. Cambridge: MIT Press.

Simpson, T. W. (2004) ‘Product platform design and customization: Status and promise’, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing: AIEDAM*, 18(1), pp. 3–20. doi: 10.1017/S0890060404040028.

Simpson, T. W. *et al.* (2012) ‘From user requirements to commonality specifications: An integrated approach to product family design’, *Research in Engineering Design*, 23(2), pp. 141–153. doi: 10.1007/s00163-011-0119-4.

Simpson, T. W. *et al.* (2014) *Advances in Product Family and Product Platform Design*. Edited by T. W. Simpson et al. New York, NY: Springer New York. doi: 10.1007/978-1-4614-7937-6.

Simpson, T. W., Siddique, Z. and Jiao, J. (Roger) (2006) *Product Platform and Product Family Design: Methods and Applications*. 1st edn. Springer US. doi: 10.1007/0-387-29197-0.

Slack, N. and Brandon-Jones, A. (2019) *Operations Management*. 9th edn. New York: Pearson Education Limited.

Stone, R. B. *et al.* (2008) ‘A customer needs motivated conceptual design methodology for product portfolio planning’, *Journal of Engineering Design*, 19(6), pp. 489–514. doi: 10.1080/09544820802286711.

Stone, R. B. and Wood, K. L. (2000) ‘Development of a Functional Basis for Design’, *Journal of Mechanical Design*, 122(4), pp. 359–370. doi: 10.1115/1.1289637.

Stone, R. B., Wood, K. L. and Crawford, R. H. (2000) ‘A heuristic method for identifying modules for product architectures’, *Design studies*, 21, pp. 1–47. doi: http://dx.doi.org/10.1016/j.fas.2016.05.142.

Suh, N. P. (1998) ‘Engineering Design Axiomatic Design Theory for Systems’, *Research in Engineering Design*, 10, pp. 189–209. doi: 10.1007/s001639870001.

Suh, N. P. (2001) *Axiomatic Design: Advances and Applications*. New York: Oxford University Press.

Takeda, H. *et al.* (1990) 'Modeling Design Processes', *AI Magazine*, 11(4), pp. 37–48. doi: 10.1609/aimag.v11i4.855.

Tan, P.-N. *et al.* (2019) *Introduction to Data Mining*. 2nd edn. Pearson.

Team, R. C. (2019) 'R: A language and environment for statistical computing'. Vienna, Austria: R Foundation for Statistical Computing. Available at: https://www.r-project.org/.

Thevenot, H. J. *et al.* (2007) 'An index-based method to manage the tradeoff between diversity and commonality during product family design', *Concurrent Engineering Research and Applications*, 15(2), pp. 127–139. doi: 10.1177/1063293X07079318.

Thevenot, H. J. and Simpson, T. W. (2007) 'A comprehensive metric for evaluating component commonality in a product family', *Journal of Engineering Design*, 18(6), pp. 577–598. doi: 10.1080/09544820601020014.

Tucker, C. S. and Kim, H. M. (2008) 'Optimal Product Portfolio Formulation by Merging Predictive Data Mining With Multilevel Optimization', *Journal of Mechanical Design*, 130(4), p. 041103. doi: 10.1115/1.2838336.

Ulrich, K. (1995) 'The role of product architecture in the manufacturing firm', *Research Policy*, 24(3), pp. 419–440. doi: 10.1016/0048-7333(94)00775-3.

Ulrich, K. and Eppinger, S. D. (2012) *Product Design and Development: Fifth Edition, McGraw-Hill*. doi: 10.1016/B978-0-7506-8985-4.00002-4.

Ulrich, K. and Tung, K. (1994) 'Fundamentals of Product Modularity', *Management of Design*, 39, pp. 219–231. doi: 10.1007/978-94-011-1390-8_12.

Vaishnavi, V., Kuechler, B. and Petter, S. (2017) 'Design Science Research in Information Systems', *Design Science Research in Information Systems*, pp. 1–66. doi: 10.1007/978-3-642-29863-9.

Venable, J., Pries-Heje, J. and Baskerville, R. (2016) 'FEDS: A Framework for Evaluation in Design Science Research', *European Journal of Information Systems*, 25(1), pp. 77–89. doi: 10.1057/ejis.2014.36.

Venable, J. R. (2006) ‘The Role of Theory and Theorising in Design Science Research’, in *1st International Conference on Design Science in Information Systems and Technology (DESRIST)*.

Walls, J. G., Wyidmeyer, G. R. and Sawy, O. A. E. (1992) ‘Building an information system design theory for vigilant EIS’, *Information Systems Research*, 3(1), pp. 36–60.

Wang, Q. *et al.* (2018) ‘An Optimization Model for Low Carbon Oriented Modular Product Platform Planning (MP3)’, *International Journal of Precision Engineering and Manufacturing-Green Technology*, 5(1), pp. 121–132. doi: 10.1007/s40684-018-0013-x.

Wei, W. *et al.* (2015) ‘A multi-principle module identification method for product platform design’, *Journal of Zhejiang University-Science A*. A, 16(1), pp. 1–10. doi: 10.1631/jzus.A1400263.

Wheelwright, S. C. and Clark, K. B. (1992) ‘Creating project plans to focus product development.’, *Harvard business review*, 70(2), pp. 70–82. doi: 10.1007/s00464-008-0110-y.

Wheelwright, S. C. and Sasser, W. E. (1989) ‘The New Product Development Map’, *Harvard Business Review*, 67, pp. 112–125.

Van Wie, M. *et al.* (2007) ‘Examination of platform and differentiating elements in product family design’, *Journal of Intelligent Manufacturing*, 18(1), pp. 77–96. doi: 10.1007/s10845-007-0005-0.

Wilhelm, W. E., Damodaran, P. and Li, J. (2003) ‘Prescribing the content and timing of product upgrades’, *IIE Transactions (Institute of Industrial Engineers)*, 35(7), pp. 647–663. doi: 10.1080/07408170304359.

Wouters, M. and Morales, S. (2014) ‘The contemporary art of cost management methods during product development’, *Advances in Management Accounting*, 24, pp. 259–346. doi: 10.1108/S1474-787120140000024008.

Xiao, W. *et al.* (2018) ‘Coordinated optimization of low-carbon product family and its manufacturing process design by a bilevel game-theoretic model’, *Journal of Cleaner Production*, 184, pp. 754–773. doi: 10.1016/j.jclepro.2018.02.240.

Xiong, Y., Du, G. and Jiao, R. J. (2018) ‘Modular product platforming with supply chain postponement decisions by leader-follower interactive optimization’, *International Journal of Production Economics*, 205(September), pp. 272–286. doi:

10.1016/j.ijpe.2018.09.013.

Yan, X.-T. and Stewart, B. (2010) ‘Developing modular product family using GeMoCURE within an SME’, *International Journal of Manufacturing Research*, 5(4), pp. 449–463. doi: 10.1504/IJMR.2010.035813.

Yang, Q., Yu, S. and Jiang, D. (2014) ‘A modular method of developing an eco-product family considering the reusability and recyclability of customer products’, *Journal of Cleaner Production*, 64, pp. 254–265. doi: 10.1016/j.jclepro.2013.07.030.

Yifei, T. *et al.* (2015) ‘Research on customer-oriented optimal configuration of product scheme based on Pareto genetic algorithm’, in *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture*, pp. 148–156. doi: 10.1177/0954405414524742.

Yoshimura, M. and Takeuchi, A. (1994) ‘Concurrent optimization of product design and manufacturing based on information of users’ needs’, *CERA*, 2(1), pp. 33–44.

Yu, S. *et al.* (2015) ‘Incorporating quality function deployment with modularity for the end-of-life of a product family’, *Journal of Cleaner Production*. Elsevier Ltd, 87(C), pp. 423–430. doi: 10.1016/j.jclepro.2014.10.037.

Zacharias, N. A. and Yassine, A. A. (2008) ‘Optimal platform investment for product family design’, *Journal of Intelligent Manufacturing*, 19(2), pp. 131–148. doi: 10.1007/s10845-007-0069-x.

Zhang, C. and Zhang, S. (2002) *Association Rule Mining - Models and Algorithms, Lecture Notes in Artificial Intelligence*. Springer. doi: 10.1007/978-3-642-29066-4{_}11.

Zhang, W. Y., Tor, S. Y. and Britton, G. A. (2006) ‘Managing modularity in product family design with functional modeling’, *International Journal of Advanced Manufacturing Technology*, 30(7–8), pp. 579–588. doi: 10.1007/s00170-005-0112-z.

Zhang, Y., Jiao, J. (Roger) and Ma, Y. (2007) ‘Market segmentation for product family positioning based on fuzzy clustering’, *Journal of Engineering Design*, 18(3), pp. 227–241. doi: 10.1080/09544820600752781.

Zhu, B., Li, Y. and Feng, G. (2017) ‘A fuzzy optimisation method for product variety selection under uncertainty constraints’, *International Journal of Computer Integrated Manufacturing*, 30(6), pp. 606–615. doi: 10.1080/0951192X.2016.1187300.

Zhu, H. *et al.* (2010) ‘Integration of rough set and neural network ensemble to predict the configuration performance of a modular product family’, *International Journal of Production Research*, 48(24), pp. 7371–7393. doi: 10.1080/00207540903349013.

Ziegel, E. R. and Tague, N. R. (1995) *The Quality Toolbox*. 2nd edn, *Technometrics*. 2nd edn. Edited by A. Hytien and P. O’Mara. Willian A. Tony. doi: 10.2307/1269755.

APPENDIX A – ARTICLE 1

Table A1. Protocol for systematic literature review.

1.0	Conceptual framework	1.1	Increased demand for a greater variety of consumer products has forced many companies to rethink their strategies to offer more product variants. For manufacturers, producing a variety of products can satisfy this increasing demand and help companies gain more market share; however, increased variety can lead to higher design and production costs as well as longer lead times for new variants. As a result, a trade-off arises between cost-effectiveness and satisfying diverse customer demand (Simpson et al., 2014).
		1.2	As a general reference, based on data from practical situations, it can be said that the production costs decrease by about 15 to 25% whenever the production scale is duplicated (Antunes et al., 2008).
		1.3	In the same way, it can be stated that there is an increase of 20 to 25% of the cost per unit produced each time the variety of manufactured items is duplicated (Antunes et al., 2008).
		1.4	Leveraging commonality can lead to remarkable cost savings and higher standardization of the product line. On the other hand, variety is desired because more variation results in more customer groups' coverage and satisfying the specific needs of more customers. However, variety of conflicts with commonality (Fixson, 2007; Jiao, Simpson and Siddique, 2007; Simpson et al., 2014).
		1.5	Commonality can reduce manufacturing and design costs by sharing the same component across different products, while concurrently providing the level of product diversity expected within the market space (Simpson et al., 2014). Commonality clusters the components and functions based on similarity or other criteria (Simpson et al., 2014).
		1.6	Modularity descriptions also often encompass a combinatorial element, i.e., modules can be mixed and matched to create new variants of a product (Fixson, 2007). Some modularity descriptions incorporate an assessment of how a product's functionality is provided by the product's components (Fixson, 2007). Modularity decomposes components and functions into independent groups (Simpson et al., 2014). Modularity in design refers to the decomposition of a system into independent modules in such a way that interactions are interdependent within and independent across modules (Miguel, 2005). Modularity has been defined as the relationship between a product`s functional and physical structures such that: (i) There is a one-to-one or many-to-one correspondence between the functional and physical structure; and (ii) unintended interactions between modules are minimized (Jiao and Tseng, 2000).
		1.7	Product architecture can be defined as how the functional elements of a product are arranged into physical units and how these units interact (Ulrich and Eppinger, 2012).
		1.8	A module is a physical or conceptual grouping of components (Jiao and Tseng, 2000).
		1.9	The platform is a set of common components, modules, or parts from which a stream of derivative products can be efficiently developed and launched (Meyer and Lehnerd, 1997)
		1.10	The product family is a set of products that share one or more common "elements" (e.g., components, modules, subsystems, fabrication processes, assembly operations) yet target a variety of different market segments (Simpson et al., 2014). A product family refers to a set of similar products that are derived from a common platform and yet possess specific functionalities to meet particular customer requirements (Meyer and Lehnerd, 1997).
		1.11	Research has found that the trade-off between product variety and cost-effectiveness can be appropriately managed by exploiting product family design (PFD) and platform-based product development, an area that has been widely studied for the past two decades (Simpson et al., 2014).
		1.12	Over the years, there has been active work in developing methods to define these modules for product families. These methods, however, have been developed independently of one another, and it can be daunting to try to compare the methods and understand which approach might be suitable when or how the methods might interlink, if at all (Simpson et al., 2014). The reasons for the lack of practical application of these methods include: (i) The limited knowledge regarding the variety of methods available, and (ii) the difficulty of locating these methods within the company's development process (Bonvoisin et al., 2016).
		1.13	Decisions made at the beginning of the development process account for about 85% of the cost of the final product (Rozenfeld et al., 2006). It has been found that approximately 60% of the committed cost of a product is defined in the conceptual design phase of a PDP (Jiao, 1998). The architecture of the product is established during the concept development and system-level design phases of development (Ulrich and Eppinger, 2012).
2.0	Context	2.1	Engineering, production, and operations management domains.
		2.3	Product Development Process - PDP (Ulrich and Eppinger, 2012).
		2.2	Type of products: consumer goods (durables), intermediate goods, capital goods, and military/defense goods.
3.0	Time horizon	3.1	Product modularity: definitions and benefits (Gershenson, Prasad and Zhang, 2003) - Up to 2003.
		3.2	Product modularity: measures and design methods (Gershenson, Prasad and Zhang, 2004) - Up to 2004.
		3.3	Modular and platform methods for product family design: Literature analysis (Jose and Tollenaere, 2005) - Up to 2005.
		3.4	Modularity in product development: a literature review towards a research agenda (Miguel, 2005) - From 1992 to 2005.
		3.5	Modularity and commonality research: Past developments and future opportunities (Fixson, 2007) - Up to 2007.
		3.6	Product family design and platform-based product development: a state-of-the-art review (Jiao, Simpson and Siddique, 2007) - Up to 2007.
		3.7	Advances in Product Family and Product Platform Design (Simpson *et al.*, 2014) - From 2007 to 2014.
		3.8	Modularisation strategy: analysis of published articles on production and operations management (1999 to 2013) (Piran *et al.*, 2016) - From 1999 to 2013.
		3.9	Evolution of modularity literature: a 25-year bibliometric analysis (Frandsen, 2017) - From 1990 to 2015.
		3.10	This study: up to 2020.

(continued)

Table A1. (continued)

4.0	Theoretical currents	4.1	Module-based product family design
5.0	Language	5.1	English
6.0	Research question	6.1	Which methods address modularity into the design of product families?
		6.2	What kind of design problems do these methods account for?
		6.3	For which kind of products have these methods been developed for?
		6.5	How has the performance of these methods been assessed?
		6.6	What are the main steps of these methods?
		6.7	What is the execution order of these steps?
		6.8	Which techniques are used to execute each step of these methods?
		6.9	Is there a common underlying structure among these methods?
7.0	Review strategy	7.1	Configurative (meta-synthesis).
8.0	Selecting criteria		
8.1	Including criteria	8.1.1	Methods and techniques that address modularity into the design of product families (Following the conceptual framework of this protocol).
		8.1.2	Document type: Articles.
		8.1.3	All-access type.
8.2	Excluding criteria	8.2.1	Scale-based product family design.
		8.2.2	Parametric platforms.
		8.2.3	Integral architecture.
		8.2.4	Modularity in production.
		8.2.5	Modularity in use.
		8.2.6	Organisational modularity.
		8.2.7	Modularity in services.
		8.2.8	Manufacturing and production for product families
		8.2.9	Supply chain issues of product families
		8.2.10	Customer co-design.
		8.2.11	Service design.
		8.2.12	Software development.
		8.2.13	Design support systems.
		8.2.14	Aesthetics in product design.
		8.2.15	Additive manufacturing.
		8.2.16	Literature review.
		8.2.17	Civil construction.
		8.2.18	Document type: Conference paper, review, book, book chapter, and conference review.
		8.2.19	Subjects areas such as computer science, mathematics, decision science, materials science, environmental science economics, econometrics, finance, energy, physics, and astronomy.
9.0	Search terms	9.1	Modularity.
		9.2	Modular.
		9.3	Design.
		9.4	Product family.
		9.5	Product platform.
		9.6	"Modularity" AND "design" AND ("Product family" OR "Product platform")
10.0	Data-bases	10.1	Web of Science.
		10.2	Scopus.

Table A2. Mixed coding scheme.

Id.	Codes	Definition / Function	Type
Cp_i	Classes of design problems for PBPF:		
Cp_1	Product family planning and positioning	Deals with market objectives, along with technology developments guided by corporate strategies (Ulrich and Eppinger, 2012).	Categorical
Cp_2	Market-driven product family design	Deals with the transition of customers' needs to functional requirements (Simpson *et al.*, 2014).	Categorical
Cp_3	Product family modeling	Comprehends the definition of modules, platforms, and the product family configuration structure in terms of design parameters and functional requirements (Simpson *et al.*, 2014).	Categorical
Cp_4	Product family configuration	Deals with structural configuration problem wherein the modules formulating the variant are optimally selected (Simpson *et al.*, 2014).	Categorical
Pb_i	Design problems:		
$Pb_{1.1}$	Strategic product family planning	How to incorporate strategic axes into product family design (Jiao and Tseng, 1999a)?	Open
$Pb_{1.2}$	Market segmentation	How to decompose the market into several segments taking into account the industry type, customer consumption levels, regional characteristics, among other factors (Fan *et al.*, 2015)?	Open
$Pb_{2.1}$	Identification of customer needs	How to derive meaning through interpretations of customers' perceptions about the existing products (Cheng *et al.*, 2017)?	Open
$Pb_{2.2}$	Determination of relative importance among customer needs	How to determine the most influential needs on customer decision making (Du, Jiao and Chen, 2014; Wei *et al.*, 2015)?	Open
$Pb_{2.3}$	Formulation of functional requirements	How to translate the market-centric information into engineering specifications (Jung and Simpson, 2016; Johannesson *et al.*, 2017)?	Open
$Pb_{2.4}$	Identification of transient functional requirements	How to identify the functional requirements there are prone to change in the future market (Jiang and Allada, 2005)?	Open
$Pb_{2.5}$	Mapping of dependencies among functional requirements	How to determine the functional requirements' hierarchy (Alizon, Shooter and Simpson, 2007; Bonjour *et al.*, 2009; Yan and Stewart, 2010)?	Open
$Pb_{2.6}$	Definition of functional requirements target values and ranges	How to arrange similar customers in terms of their desired values (Park *et al.*, 2008; Zacharias and Yassine, 2008; Mesa *et al.*, 2014; Bejlegaard *et al.*, 2018)?	Open
$Pb_{2.7}$	Representation of functional requirements	How to represent the functional view of a product family (Jiao and Tseng, 1999a; Kota, Sethuraman and Miller, 2000; Yang, Yu and Jiang, 2014)?	Open
$Pb_{3.1}$	Definition and modeling of the product family and platforming criteria	What strategy and criteria to use for clustering modules and identifying platforms (Fan *et al.*, 2015; Hou *et al.*, 2017, 2018)?	Open
$Pb_{3.2}$	Formulation of design parameters	How to determine the physical effect with the ability to fulfill one or more functional requirements (Gauss, Lacerda and Sellitto, 2019)?	Open
$Pb_{3.3}$	Mapping of product family architecture	How to map the relationships between functional requirements and design parameters (Navarrete *et al.*, 2013; Borjesson and Hoelttae-Otto, 2014)?	Open
$Pb_{3.4}$	Decomposition of the system into functional modules	How to decompose the product family architecture into design modules (Jiao and Tseng, 1999a)?	Open
$Pb_{3.5}$	Creation of rough geometric layouts	How to identify the interactions among physical components (Pakkanen, Juuti and Lehtonen, 2016)?	Open
$Pb_{3.6}$	Mapping of structural dependencies among components	How to model the structural dependencies among components (Yu *et al.*, 2015; Kim *et al.*, 2016; Baylis, Zhang and McAdams, 2018)?	Open
$Pb_{3.7}$	Decomposition of the system into physical modules	How to decompose a set of structural relationships into physical modules (Bonjour *et al.*, 2009)?	Open
$Pb_{3.8}$	Evaluation of modules	How to evaluate clustering solutions (Jiao and Tseng, 1999a)?	Open
$Pb_{3.9}$	Classification of modules	How to classify modules for product configuration structure (Rai and Allada, 2003)?	Open
$Pb_{3.10}$	Building of the MBPF configuration structure	How to build a hierarchical structure for end product configuration (Li, Huang and Newman, 2008)?	Open
$Pb_{3.11}$	Evaluation of module-based product family design	How to evaluate product families as a whole and generate measures of deviation from the ideal (Otto and Hölttä-Otto, 2007)?	Open
$Pb_{4.1}$	Definition and modeling of configuration criteria	What criteria to use for modeling the combinatorial and parametric problem (Li, Huang and Newman, 2008; Li and Huang, 2009)?	Open
$Pb_{4.2}$	Reduction of configuration design space	How to reduce the configuration design space (Zhu *et al.*, 2010)?	Open
$Pb_{4.3}$	Combination of modules to generate product family variants	How to determine the right combination of modules to formulate the product family variants (Xiong, Du and Jiao, 2018)?	Open

(continued)

Table A2. (continued)

Id.	Codes	Definition / Function	Type
$Pb_{4.4}$	Scaling of modules' parameters	How to determine the parameters of scalable modules compounding the product family variants (Xiao *et al.*, 2018)?	Open
$Pb_{4.5}$	Representation of a product family variants	How to represent the product family variants resulting from the combination process (Du, Jiao and Tseng, 2001; Li, Huang and Newman, 2008; Li and Huang, 2009)?	Open
Ct_i	Criteria:		
Ct_1	Utility	The level of users' satisfaction with a product (Yoshimura and Takeuchi, 1994).	Open
Ct_2	Cost	The amount of expenditure incurred to produce a product (Wouters and Morales, 2014).	Open
Ct_3	Commonality	The sharing of intellectual and material assets across products to minimize manufacturing complexity (Erens and Verhulst, 1997).	Open
Ct_4	Redesign effort	The amount of redesign effort required for future designs of the product (Martin and Ishii, 2002).	Open
Ct_5	Interaction or coupling	The strength of coupling between the components in a product (Martin and Ishii, 2002).	Open
Ct_6	Modularity	The decomposition of a system into independent modules that can be treated as logical units (Newcomb, Bras and Rosen, 1998).	Open
Ct_7	Quality	The state of being free from defects (Chan and Wu, 2002).	Open
Ct_8	Serviceability	The degree to which the servicing of a product can be accomplished (Otto and Hölttä-Otto, 2007).	Open
Ct_9	Environmental	The possible adverse effects resulting from the product life-cycle (Yang, Yu and Jiang, 2014).	Open
Ct_{10}	Lead time	The time between the initiation and completion of a production process (Antunes *et al.*, 2008).	Open
Ct_{11}	Strategic	The relationship between the business environment and product architecture (Otto and Hölttä-Otto, 2007).	Open
Ct_{12}	Demand	The quantity of a product the customer intends to purchase (Antunes *et al.*, 2008).	Open
Ct_{13}	Profit	The economic benefit of a product to an enterprise (Dong, Shao and Xiong, 2011; Chen, Hoyle and Wassenaar, 2013).	Open
Ct_{14}	Price	The amount of money the customer is willing to pay for a product (Cox and Schleier, 2010).	Open
Ct_{15}	Variety	The level of distinctiveness of the product's offering in the marketplace (Chen, Hoyle and Wassenaar, 2013).	Open
P_{ti}	Product classification:		
P_{t1}	Consumer goods (durables)	Consist of durable products that people buy for their use (OECD, 2008).	Categorical
P_{t2}	Intermediate goods	Comprehend those products used in the production of other goods (OECD, 2008).	Categorical
P_{t3}	Capital goods	Consist of machines and equipment used to produce products or provide services (OECD, 2008).	Categorical
P_{t4}	Military and defense goods	Consist of the equipment used in defensive tactics that seek to negate the enemy's offensive tactics (Pate, Patterson and German, 2012).	Categorical
E_{ti}	Evaluation approach:		
E_{t1}	Observational	(i) Case study elements: study the existing or created artifact in-depth in the business environment. (ii) Field study: monitor the use of the artifact in multiple projects (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
E_{t2}	Analytical	(i) Static analysis: examine the structure of the artifact for static qualities. (ii) Architecture analysis: study the fit of the artifact in the technical architecture of the complete technical system. (iii) Optimization: demonstrate the optimal properties inherent to the artifact or demonstrate the limits of the optimization in artifact behavior. (iv) Dynamic analysis: study the artifact during use to evaluate its dynamic qualities (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
E_{t3}	Experimental	(i) Controlled experiment: study the artifact in a controlled environment to determine its qualities. (ii) Simulation: execute the artifact with artificial data (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
E_{t4}	Testing	(i) Functional test (black box): implement the artifact interfaces to discover potential failures and identify defects. (ii) Structural test (white box): perform coverage tests of some metrics for implementing the artifact (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
E_{t5}	Descriptive	(i) Informed argument: use the information of knowledge bases (e.g., relevant research) to construct a convincing argument about the utility of the artifact. (ii) Scenarios: construct detailed scenarios for the artifact to demonstrate its utility (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
M_i	Methods:		
M_1	Product family architecture (PFA)	Characterizes customer needs and subsequently fulfills these needs by configuring and modifying well-established modules and components (Jiao and Tseng, 1999a).	Open
M_2	Product line commonality	Presents an objective measure, called Product Line Commonality (PCI), to capture the level of component commonality in a product family (Kota, Sethuraman and Miller, 2000).	Open
M_3	Generic product structure (GPS)	Characterizes the source of variety based on the hierarchical decomposition of product structures (Du, Jiao and Tseng, 2001).	Open
M_4	Modular product architecture	Decomposes a set of products into shared and individual modules based on functional modeling (Dahmus, Gonzalez-Zugasti and Otto, 2001).	Open

(continued)

Table A2. (continued)

Id.	Codes	Definition / Function	Type
M_5	Design for variety (DFV)	Presents two indices to measure a product's architecture. The first index is the Generational Variety Index (GVI), a measure for redesign effort required for future designs of the product. The second index is the Coupling Index (CI), a measure of the coupling among the product components (Martin and Ishii, 2002).	Open
M_6	Modularisation function	Analyses the degree of modularity of product architecture by taking into consideration the following variables: number of components, the degree of coupling, and the substitutability of new-to-the-firm components (Mikkola and Gassmann, 2003).	Open
M_7	Agent-based modular product family design	Performs a multi-objective optimization using a multi-agent framework to determine the Pareto-design solutions for a given module set, then performs a post-optimization analysis that employees the quality loss function to determine the optimal platform level for a related set of product families and their variants (Rai and Allada, 2003).	Open
M_8	Integrated modular product design	Consists of an integrated method that includes additional tools and stages for complete modular architecture design. The borders of the modular design process are expanded by adding strategic issues, appropriateness to modularity, the degree of modularity and modularity strategies (Asan, Polat and Serdar, 2004).	Open
M_9	Robust modular product family design	Determines the optimal control factors and the suitable periods for designing robust product families by using a modified Taguchi method (Jiang and Allada, 2005).	Open
M_{10}	Functional modeling of modular product family design	Supports the identification of both shared and individual behavioral modules across a family of products for module-based product family design (Zhang, Tor and Britton, 2006).	Open
M_{11}	Comprehensive evaluation of product family commonality	Evaluates the design of a product family on a 0–1 scale based on the components in each product, their size, geometry, material, manufacturing process, assembly, cost, and the allowed diversity in the family (Thevenot and Simpson, 2007).	Open
M_{12}	Multi-criteria assessment for product platforming	Presents a platform concept evaluation tool that is multi-criteria in nature, and scalable to include various alternative criteria as appropriate (Otto and Hölttä-Otto, 2007).	Open
M_{13}	Design for commonality and diversity method (DCDM)	Manages the inherent trade-off between commonality and diversity during all stages of the product family design process (Thevenot *et al.*, 2007).	Open
M_{14}	Improving an existing product family	Identifies sources of improvement to support product family redesign (Alizon, Shooter and Simpson, 2007).	Open
M_{15}	Genetic algorithm-based modules identification	Solves the multi-objective optimization problem of identifying the constituent modules of a product family (Meng, Jiang and Huang, 2007).	Open
M_{16}	Customer-need-motivated conceptual design for product portfolio planning	Outlines platform and differentiating modules during the conceptual design stage of product development and plans a product portfolio before any embodiment design occurs (Stone *et al.*, 2008).	Open
M_{17}	Product platform concept development (PPCDM)	Analyses the variation of technical requirements across the related products and identifies the common modules within the products (Park *et al.*, 2008).	Open
M_{18}	Cooperative coevolutionary algorithm for design of adaptative platform-based products	Identifies an optimal product variant based on the adaptive product platform after giving customer requirements (Li, Huang and Newman, 2008).	Open
M_{19}	Dynamic approach to product architecture optimisation	Determines the optimal product architecture configuration in the multiproduct hierarchy by directly incorporating what customers want in the design and formulation of a family of products (Tucker and Kim, 2008).	Open
M_{20}	Optimal platform investment for product family design	Suggests the optimal initial investment in the platform, the commonality level between variants, and the number of variants to be produced in order to maximize market coverage using both analytical and simulation techniques (Zacharias and Yassine, 2008).	Open
M_{21}	Multiobjective evolutionary optimization for adaptive product family design	Accommodates multi-level commonality in adaptive product family optimization (Li and Huang, 2009).	Open
M_{22}	Configuration performance prediction (CPP)	Integrates rough set and neural network ensemble to predict the configuration performance of a modular product family (Zhu *et al.*, 2010).	Open
M_{23}	Proactive platform-based product family design	Presents a framework to modularise PFA for variety generation and optimization of a family of products with rationalized commonality configuration (Liu, Wong and Lee, 2010).	Open
M_{24}	GeMoCURE	Combines several methods to allow designers to generate design solutions using modular concepts in a systematic manner (Yan and Stewart, 2010).	Open

(continued)

Table A2. (continued)

Id.	Codes	Definition / Function	Type
M_{25}	Evolvable modular platform product design	Presents an evolvable modular platform flexible product decision model, which is based on the modular design concept and considers external environmental factors and customer demand situations (Dong, Shao and Xiong, 2011).	Open
M_{26}	The sharing decision and optimal clustering framework	Identifies candidates for component sharing based on the functional description of each component and its IM score (Arciniegas and Kim, 2011).	Open
M_{27}	Product variants design model (PVDM)	Generates new combinations of parts and components to form modules using hierarchical classification and cladistics (ElMaraghy and AlGeddawy, 2012).	Open
M_{28}	Hybrid valuation of product and project-related flexibility	Determines an optimal flexibility level for product platform planning concerning both technical and financial real options (Jiao, 2012).	Open
M_{29}	Optimal reconfigurable family design	Optimizes the configuration of multi-mission aircraft families with reconfigurable and interchangeable components (Pate, Patterson and German, 2012).	Open
M_{30}	Modular product development through PBD and DFMA	Integrates function-based modular product architecture, platform-based design, design for manufacture, and design for assembly to develop modular products (Emmatty and Sarmah, 2012).	Open
M_{31}	An integrated approach to product family design	Integrates several disparate tools into a framework to translate user needs and requirements into commonality specifications during product family design (Simpson *et al.*, 2012).	Open
M_{32}	An ISM, DEI, and ANP based approach for product family development	Combines modularisation and market segmentation to develop optimal product families (Hsiao *et al.*, 2013).	Open
M_{33}	Reactive products platform design	Automatically redesigns product variants using physical commonality, instead of evaluating alternate solutions provided by designers using commonality indices (AlGeddawy and ElMaraghy, 2013).	Open
M_{34}	Modular design of product families for quality and cost	Optimizes the design of the single-level modules considering the quality and cost simultaneously (Agard and Bassetto, 2013).	Open
M_{35}	Flexible platform modular architecture design	Combines flow analysis, design structure matrix (DSM), and fuzzy clustering in an integrated method for defining modular flexible platforms (Li *et al.*, 2013).	Open
M_{36}	Reduction of product platform complexity	Reduces the product platform complexity based on a matrix representation of technical solutions versus product properties (Navarrete *et al.*, 2013).	Open
M_{37}	Platform-driven product planning	Adapts existing methodologies that tackle different stages of product family development, in order to provide a comprehensive, systematic, and intuitive step-by-step approach to platform development (Sahin-Sariisik *et al.*, 2014).	Open
M_{38}	Hybrid module generation algorithm	Generates modules through an algorithm that balances both module independence and product similarity, allowing product similarity strategy to influence the coupling-driven architecture considerations (Borjesson and Hoelttae-Otto, 2014).	Open
M_{39}	Reconfigurable system architecture	Integrates modular architecture principles, selection algorithms, clustering algorithms, family product features, and functional system analysis in the product design process to obtain modular RMS (Mesa *et al.*, 2014).	Open
M_{40}	Modular eco-product family design	Integrates eco-design with product family design by modularity (Yang, Yu and Jiang, 2014).	Open
M_{41}	Joint optimisation of product family configuration and scaling design	Optimizes product family configuration and scaling design based on a Stackelberg game-theoretic model (Du, Jiao and Chen, 2014).	Open
M_{42}	Predicting configuration performance of modular product family	Predicts the configuration performance of a product family variant through the combination of principal component analysis (PCA) and support vector machine (SVM) (Meng *et al.*, 2014).	Open
M_{43}	Modular product multi-platform (MPMP)	Optimally defines product platforms and families using the relatively new concept of assembly/disassembly (Hanafy and Elmaraghy, 2015).	Open
M_{44}	Structure-oriented modular product platform planning	Applies network theory and network analysis for the planning of modular product platforms (Fan *et al.*, 2015).	Open
M_{45}	Eco-product family design	Incorporates QFD with modularity for the end-of-life of a product family (Yu *et al.*, 2015).	Open
M_{46}	Technology roadmap modular deployment (TRM-MD)	Combines two methods of modular design to transform the products listed on the roadmap into a list of modules, providing a new roadmap based on module releases (Scalice *et al.*, 2015).	Open
M_{47}	Modular product family design	Optimizes module combinations in order to derive final product variants (Adhitama and Rosenstiel, 2015).	Open
M_{48}	Multi-principle module identification method for product platform design	Uses an improved strength Pareto evolutionary algorithm (ISPEA-II) to address the multi-principle modules' identification (Wei *et al.*, 2015).	Open

(continued)

Table A2. (continued)

Id.	Codes	Definition / Function	Type
M_{49}	Comprehensive product platform planning (CP3)	Designs both scale-based and modular product families (Chowdhury *et al.*, 2016).	Open
M_{50}	Analysis of architectural complexity for product family and platform	Assesses the architectural complexity of product platform and variant architectures (Kim *et al.*, 2016).	Open
M_{51}	Systematic adaptable platform architecture design	Proposes an adaptable platform methodology to solve systematic design for hierarchical platform architecture (Li *et al.*, 2016).	Open
M_{52}	A new methodology to cluster derivative product modules	Maximizes the use of common product modules by considering platform-based derivative products and modular product design approaches to minimize the planning complexity in the supply chain, manufacturing, and service for derivative products (Aydin and Ulutas, 2016).	Open
M_{53}	Hierarchical game joint optimisation for product family-driven modular design	Identifies and groups components into independent modules through the combination of TSM and PFM within a coherent framework of hierarchical joint optimization (Ma *et al.*, 2016).	Open
M_{54}	Brownfield process (BfP)	Rationalizes the existing product variety towards a modular product family that enables product configuration (Pakkanen, Juuti and Lehtonen, 2016).	Open
M_{55}	An integrated approach to product family redesign using commonality and variety metrics	Uses multiple product family metrics to establish an effective platform redesign strategy (Jung and Simpson, 2016).	Open
M_{56}	Design of adaptable product platform	Uses axiomatic design and sensitivity design structure matrix for identification of adaptable product platform (Cheng *et al.*, 2017).	Open
M_{57}	Modular platform optimisation	Generates modules based on a modified graph-based decomposition algorithm, then selects the shared modules based on a cost-based priority method (Hou *et al.*, 2017).	Open
M_{58}	Development of sustainable platform for modular product family	Defines the type of platform architecture (modular or integral) as well as provides design guidelines and checklists for the product designers (Shamsuzzoha and Helo, 2017).	Open
M_{59}	An integrated framework for product line design for modular products	Maximizes the product line design by taking into account the functional product attributes, customer demand, and product cost (Goswami, Daultani and Tiwari, 2017).	Open
M_{60}	Development of product platforms	Supports platform development teams in modeling and configuring product families through a holistic object-oriented methodology (Johannesson *et al.*, 2017).	Open
M_{61}	Complexity cost management	Quantifies the cost effects to support concept selection during modular product family design (Ripperda and Krause, 2017).	Open
M_{62}	Product-family shared-component selection based on the consistency constraint function	Selects shared components in modular platforms based on the collaborative optimization consistency constraint function (Hou *et al.*, 2018).	Open
M_{63}	Coordinated optimisation of low-carbon product family and its manufacturing process design	Optimizes the low-carbon product family architecting (L-CPFA) and low-carbon manufacturing process configuration (L-CMPC) (Xiao *et al.*, 2018).	Open
M_{64}	A method for coupling analysis of association modules in product family design	Analyses coupled design for product family based on coupling incidence path and correlation impact degree (Cheng, Xiao and Wang, 2018).	Open
M_{65}	Low carbon oriented modular product platform planning	Determines the modular planning in pursuit of maximizing the low carbon performance of the product with controlling the implementation probability of MP3 (Wang *et al.*, 2018).	Open
M_{66}	Propagation method	Identifies the product architecture by propagating the constraints from the functional domain to the physical domain (Bonjour *et al.*, 2009).	Open
M_{67}	Methodology for reconfigurable fixture architecture design	Designs reconfigurable fixtures through a generic architecture design methodology (Bejlegaard *et al.*, 2018).	Open
M_{68}	Modular product platforming with supply chain postponement decisions	Deals with the collaborative optimisation of modular product planning and supply chain postponement to maximize manufacturers' profits (Xiong, Du and Jiao, 2018).	Open
M_{69}	Product family platform selection using a Pareto front of maximum commonality and strategic modularity	Identifies multiple component sharing options that lie along a Pareto front of maximum commonality and strategic modularity (Baylis, Zhang and McAdams, 2018).	Open

(continued)

Table A2. (continued)

Id.	Codes	Definition / Function	Type
M_{70}	Value analysis for customizable modular product platforms	Ranks alternative platform configurations according to customers' preferences (Colombo *et al.*, 2019).	Open
M_{71}	Deciding the number of architectures	Defines the total number of product architectures in a company developing a new product generation (Askhøj *et al.*, 2019).	Open
M_{72}	Module-based machinery design	Conceptually designs modular machine families for reconfigurable manufacturing systems (Gauss, Lacerda and Sellitto, 2019).	Open
T_i	Techniques:		
T_1	Affinity diagram (KJ Method)	Organizes ideas, problems, and solutions into related groups after a brainstorm (Shiba and Walden, 2007).	Open
T_2	Multi pick-up method (MPM)	Reduces many statements down to a manageable number (Shiba and Walden, 2007).	Open
T_3	Survey	Provides statistical descriptions of people by asking questions, usually of a sample (Forza, 2002).	Open
T_4	Competitive analysis	Identifies and quantifies the relative strengths and weaknesses for developing a competitive strategy (Aoussat *et al.*, 1995).	Open
T_5	Pareto chart	Presents a vertical bar graph in which values are plotted in decreasing order of relative frequency from left to right (Ziegel and Tague, 1995).	Open
T_6	Analytical hierarchy process (AHP)	Compares alternatives through a scale of absolute judgments that represents, how much more, one element dominates another concerning a given attribute (Saaty, 2008).	Open
T_7	Fuzzy clustering means (FCM)	Assigns data points to clusters allowing each data point to belong to multiple clusters with varying degrees of membership (Bezdek, 1981).	Open
T_8	Decomposition / classification tree (DCT)	Depicts the functional view of a product family from an abstract level to individual instances (Jiao and Tseng, 1999a).	Open
T_9	Zigzag decomposition	Decomposes functional requirements (FR) and design parameters (DP) in the functional and the physical domains to create the FR and DP hierarchies (Suh, 2001).	Open
T_{10}	Design matrix (DM)	Represent the relationships between two domains, for example, between functions and physical design parameters (Suh, 2001).	Open
T_{11}	Cluster identification algorithm	Finds optimal machine cells and part families provided that the machine-part incidence matrix has the diagonal block structure embedded to solve standard group technology problems (Kusiak and Chow, 1987).	Open
T_{12}	Modular structure	Reveals the overall schematic of arranging modules for synthesizing a solution (Jiao, 1998)	Open
T_{13}	Utility analysis	Integrates demand analysis and design optimization to evaluate users' satisfaction levels for products (Yoshimura and Takeuchi, 1994).	Open
T_{14}	Pragmatic approach to product costing	Estimates the cost of a product at early design stages where only a schematic design may be available (Jiao and Tseng, 1999b).	Open
T_{15}	Cost-utility analysis	Evaluates various building blocks according to their contribution to maintaining the economy of scale and providing functional variety (Jiao and Tseng, 1999a).	Open
T_{16}	Fuzzy ranking approach	Handles linguistic and ordinary quantitative information in solving the multicriteria decision-making problem faced during the early stage of the design process (Jiao and Tseng, 1998).	Open
T_{17}	Configuration structure	Describes how various product variants are derived from the combination of the physical modules and the interconnections across different levels of assembly (Jiao and Tseng, 1999a).	Open
T_{18}	Function structure / diagram	Graphically represents a functional model where its overall function is represented by a collection of sub-functions connected by the flows on which they operate (Stone and Wood, 2000).	Open
T_{19}	Product line commonality index (PCI)	Captures the level of component commonality in a product family (Kota, Sethuraman and Miller, 2000).	Open
T_{20}	Generic bill-of-material (GBOM)	Specifies all variants of a product family in a single hierarchical AND/OR tree structure (Li, Huang and Newman, 2008).	Open
T_{21}	Dominant flow heuristic	Defines a module based on a flow that passes through a set of sub-functions (Stone, Wood and Crawford, 2000).	Open
T_{22}	Branching flow heuristic	Defines a module based on branches of a parallel function chain (Stone, Wood and Crawford, 2000).	Open
T_{23}	Conversion-transmission heuristic	Defines a module based on a conversion sub-function or a conversion transmission pair or chain of sub-functions (Stone, Wood and Crawford, 2000).	Open
T_{24}	Modularity matrix	Represents the functional outputs of modules for each product variant (Dahmus, Gonzalez-Zugasti and Otto, 2001).	Open
T_{25}	Pugh matrix	Compares alternative design concepts against customer requirements, with evaluations, made relative to a base or favored concept, in a process independent of the HoQ analysis (Chen, Hoyle and Wassenaar, 2013).	Open
T_{26}	Aggregate project plan	Classify projects based on the number of resources they consume and on how they will contribute to the company's product line (Wheelwright and Clark, 1992).	Open
T_{27}	New product development map	Presents the evolution of current product lines in a summarised yet strikingly clear way so that all functional areas in the organization can respond to a common vision (Wheelwright and Sasser, 1989).	Open
T_{28}	Quality function deployment (QFD)	Translates customer requirements into technical requirements for each stage of product development and production (Chan and Wu, 2002).	Open

(continued)

Table A2. (continued)

Id.	Codes	Definition / Function	Type
T_{29}	Conjoint analysis	Determines what combination of attributes is most influential on customers' decision making (Chen, Hoyle and Wassenaar, 2013).	Open
T_{30}	Generational variety index (GVI)	Indicates the amount of redesign required for a component to meet the future market requirements (Martin and Ishii, 2002).	Open
T_{31}	Design structure matrix (DSM)	Represents the relationships among elements of the same domain, for example, between components (Browning, 2001).	Open
T_{32}	Coupling index (CI)	Indicates the strength of coupling among the components in a product (Martin and Ishii, 2002).	Open
T_{33}	Modularisation function	Measures the degree of modularity of product architecture by taking into consideration the following variables: number of components, the degree of coupling, and the substitutability of new-to-the-firm components (Mikkola and Gassmann, 2003).	Open
T_{34}	Pareto optimization	Optimizes solutions for problems involving more than one objective function simultaneously (Rai and Allada, 2003).	Open
T_{35}	Agent-based models (ABM)	Simulates the actions and interactions of autonomous agents in assessing their effects on the system as a whole (Rai and Allada, 2003).	Open
T_{36}	Weighted sum for multi-objective optimization	Scalarizes the set of objectives into a single objective by multiplying each objective with a user-supplied weight (Rai and Allada, 2003).	Open
T_{37}	Quality loss function (QLF)	Estimates the loss of quality resulting from the deviation of a product characteristic from its target value (Rai and Allada, 2003).	Open
T_{38}	Modularity degree matrix	Evaluates the willingness to the modularity of product architectures based on the measures of the degree of modularity and the appropriateness for modularity (Asan, Polat and Serdar, 2004).	Open
T_{39}	Qualitative interviews	Derives meaning through interpretations, not necessarily 'facts' from participant talk (Malhotra and Birks, 2007).	Open
T_{40}	Design for "X"	Provides system-level design guidelines for product development (Pahl *et al.*, 2007).	Open
T_{41}	Objective matrix	Evaluates different types or dimensions of performance measures altogether by composing them into a single composite measure (Asan, Polat and Serdar, 2004).	Open
T_{42}	Cross-impact systems and matrices (SMIC)	Describes the potential modes of interaction between a given set of variables and the assessment of the strength of these interactions (Asan, Polat and Serdar, 2004).	Open
T_{43}	Simplex model	Solves linear programming problems (Hilier and Lieberman, 2015).	Open
T_{44}	Robustness chart	Evaluates the robustness of a product family in fulfilling the customer requirements over time (Jiang and Allada, 2005).	Open
T_{45}	Market segmentation grid (MSG)	Articulates leveraging platform strategies in a given market (Kumar, Chen and Simpson, 2009).	Open
T_{46}	Comprehensive metric for commonality (CMC)	Assesses the commonality of a product family based on the components in each product, their size, geometry, material, manufacturing process, assembly, cost, and the allowed diversity in the family (Thevenot and Simpson, 2007).	Open
T_{47}	Cost-worth metric	Evaluates modules against cost and worth criteria (Otto and Hölttä-Otto, 2007).	Open
T_{48}	Customer needs metric	Measures how well the customer needs are met by the platform (Otto and Hölttä-Otto, 2007).	Open
T_{49}	Carryover metric	Measures how well a function can be incorporated into different products without change and no technology upgrades (Otto and Hölttä-Otto, 2007).	Open
T_{50}	Common module metric	Evaluates how well the modules are shared in a platform-based product family (Otto and Hölttä-Otto, 2007).	Open
T_{51}	Specification variety metric	Measures the relative number of functions with different specifications within a product family (Otto and Hölttä-Otto, 2007).	Open
T_{52}	Partitioning for reliability metric	Measures how far the number of modules is from this ideal in order to prevent errors (Otto and Hölttä-Otto, 2007).	Open
T_{53}	Failure modes and effects analysis (FMEA)	Identifies possible failures and estimates the related risks (Pahl *et al.*, 2007).	Open
T_{54}	Partitioning for service metric	Measures how well the serviceable functions are isolated into modules (Otto and Hölttä-Otto, 2007).	Open
T_{55}	AT&T model	Assesses the environmental friendliness of a platform (Otto and Hölttä-Otto, 2007).	Open
T_{56}	Ease of assembly metric	Measures the effectiveness of a design from a purely assembly theoretical point of view (Otto and Hölttä-Otto, 2007).	Open
T_{57}	Make-buy metric	Measures the relative number of possible outsourced components within a product family (Otto and Hölttä-Otto, 2007).	Open
T_{58}	Organizational alignment metric	Demonstrates the degree of alignment of the architecture with the development organization (Otto and Hölttä-Otto, 2007).	Open
T_{59}	Testability metric	Evaluates if the value of the tested module corresponds precisely to field requirements (Otto and Hölttä-Otto, 2007).	Open
T_{60}	Unknown isolation metric	Evaluates how well the module's architecture can accommodate requirements changing (Otto and Hölttä-Otto, 2007).	Open
T_{61}	Change potential number	Estimates the readiness of the company to deal with the change as well as flexibility of the product (Otto and Hölttä-Otto, 2007).	Open
T_{62}	Function and form alignment metric	Measures the degree of function-form independence (Otto and Hölttä-Otto, 2007).	Open
T_{63}	Interface flexibility metric	Evaluates the redesign effort of different interaction types (Otto and Hölttä-Otto, 2007).	Open
T_{64}	Interface adjustment factor (IAF)	Indicates the unexpected interactions going through the interface of each module (Otto and Hölttä-Otto, 2007).	Open
T_{65}	One DOF adjustments metric	Quantifies the degree of freedom of product architectures that require service or swap-outs of modules (Otto and Hölttä-Otto, 2007).	Open
T_{66}	Limited extremes metric	Compares the new requirements of the architecture under development to the requirements of the current model in the market (Otto and Hölttä-Otto, 2007).	Open

(continued)

Table A2. (continued)

Id.	Codes	Definition / Function	Type
T_{67}	Commonality diversity index (CDI)	Scores the difference between real and ideal trade-off within and across a family of products with different depths of analysis (Thevenot and Simpson, 2007).	Open
T_{68}	Bill of materials (BOM)	Represents the hierarchical structure of a product family in a tabulating form (Jiao and Tseng, 2000).	Open
T_{69}	Value analysis	Identifies unnecessary costs in a product and eliminates them without impairing its quality and efficiency (Alizon, Shooter and Simpson, 2007).	Open
T_{70}	Core needs-based product platforms	Identifies common modules (platforms) from the relationship between customer need frequency and customer need weight (Stone *et al.*, 2008).	Open
T_{71}	Instance bill-of-materials (IBOM)	Represents a product variant derived from the corresponding platform (Li, Huang and Newman, 2008).	Open
T_{72}	Genetic algorithm (GA)	Finds optimized solutions to search problems based on the mechanism of natural selection and natural genetics (Meng, Jiang and Huang, 2007).	Open
T_{73}	Degree of variety (DV)	Calculates the contribution of elements to the product variety (Park *et al.*, 2008).	Open
T_{74}	Cooperative coevolutionary algorithm	Imitates the coevolutionary process of two or more species that evolve while interacting with and adapting to each other (Li, Huang and Newman, 2008).	Open
T_{75}	Naïve Bayesian Model	Builds a predictive model based on a fraction of customer survey data, used to train the computer learning model (Tucker and Kim, 2008).	Open
T_{76}	Branch and bound algorithm	Finds an optimal solution for mixed integer nonlinear programming problems with discrete and continuous variables (Tucker and Kim, 2008).	Open
T_{77}	Data to Knowledge (D2K)	Classifies the survey results and maps the data into one of several predefined classes (Tucker and Kim, 2008).	Open
T_{78}	Gross margin	Measurement of how effectively the company turns its revenue into profit (Cox and Schleier, 2010).	Open
T_{79}	Generalized reduced gradient (GRG) algorithm	Finds optimized solutions for nonlinear problems (Zacharias and Yassine, 2008).	Open
T_{80}	Commonality degree (CD)	Measures the commonality of multi-level variables of product family variants (Li and Huang, 2009).	Open
T_{81}	Non-dominated sorting genetic algorithm (NSGA-II)	Finds optimized solutions to search multi-objective problems based on the mechanism of natural selection and natural genetics (Li and Huang, 2009).	Open
T_{82}	Thebaud clustering algorithm	Groups the components into clusters by minimizing the function of total coupling cost (Bonjour *et al.*, 2009).	Open
T_{83}	Recognizable matrix	Specifies all variants of a product family and its configuration performance in a tabulating form (Zhu *et al.*, 2010).	Open
T_{84}	Confusion matrix	Describes the classification accuracy by showing the predicted and actual classifications (Zhu *et al.*, 2010).	Open
T_{85}	Artificial neural network (ANN)	Simulates the network of neurons that make up a human brain in order to make a machine learn things, recognize patterns, and make decisions in a human-like way (Zhu *et al.*, 2010).	Open
T_{86}	K-means clustering	Classifies a given data set through a certain number of clusters fixed apriori (Zhu *et al.*, 2010).	Open
T_{87}	Attribute-module matrix (AMM)	Depicts the coupling degree among product attributes and modules (Liu, Wong and Lee, 2010).	Open
T_{88}	Commonality index (CI)	Measures the overall level of commonality of a product family based on the commonality index of all platforming elements and their expected sharing degree (Liu, Wong and Lee, 2010).	Open
T_{89}	Evolutionary dynamic weighted aggregation (EDWA)	Finds optimized solutions to search multi-objective problems with a concave Pareto front based on the mechanism of natural selection and natural genetics (Liu, Wong and Lee, 2010).	Open
T_{90}	Variety index (VI)	Estimates the design variation or effort on modules to meet customer-perceived varieties (Liu, Wong and Lee, 2010).	Open
T_{91}	Non-recurring engineering (NRE) cost	Quantifies the one-time investment of the research, design, and testing of one new product (Liu, Wong and Lee, 2010).	Open
T_{92}	Module identification function (MIM)	Visually displays the strength of dependencies between components in a DSM (Yan and Stewart, 2010).	Open
T_{93}	Module strength indicator (MSI)	Indicates how robust a module is from a particular viewpoint (Yan and Stewart, 2010).	Open
T_{94}	Decision tree	Represents the branching structure of a decision process (Dong, Shao and Xiong, 2011).	Open
T_{95}	Geometric Brownian motion (GBM)	Models continuous-time stochastic trends subject to random noise (Dong, Shao and Xiong, 2011).	Open
T_{96}	CPLEX optimizer	Solves linear programming, mixed integer programming, quadratic programming, and quadratically constrained programming problems (Dong, Shao and Xiong, 2011).	Open
T_{97}	Minimal description length (MDL)	Tries to find the shortest valid descriptions for the data (Arciniegas and Kim, 2011).	Open
T_{98}	Impact metric (IM)	Captures the impact of changing a component in a given platform, combining the MDL representation for each component and the CI score (Arciniegas and Kim, 2011).	Open
T_{99}	Cladogram	Illustrates phylogenetic relationships and shows points at which various species are presumed to have diverged from common ancestral forms (ElMaraghy and AlGeddawy, 2012).	Open
T_{100}	Liaison graph	Depicts the physical relationship among components in a graphical formalism (ElMaraghy and AlGeddawy, 2012).	Open
T_{101}	Real options analysis (ROA)	Applies option valuation techniques to capital budgeting and strategic decisions (Jiao, 2012).	Open
T_{102}	Probability of design success	Measures the probability of a design parameter satisfying a given functional requirement (Suh, 1998).	Open

(continued)

Table A2. (continued)

Id.	Codes	Definition / Function	Type
T_{103}	Monte Carlo simulation	Models probabilistic systems and establishes the odds for a variety of outcomes (Montgomery and Runger, 2011).	Open
T_{104}	Stratified state aggregation	Partitions a one-dimensional space at each time step, independent of the number of state variables (Jiao, 2012).	Open
T_{105}	Multi-period option pricing model	Allows multi-period views of the underlying asset price and the price of the option for multiple periods as well as the range of possible results for each period (Jiao, 2012).	Open
T_{106}	Game theory	Deals with the general features of competitive situations formally and abstractly (Hilier and Lieberman, 2015).	Open
T_{107}	Multinomial logit (MNL)	Predicts the probabilities of the different possible outcomes of a categorically distributed dependent variable, given a set of independent variables (Chen, Hoyle and Wassenaar, 2013).	Open
T_{108}	Financial real option	Consists of a payoff function of exercising a financial option on its expiration date (Jiao, 2012).	Open
T_{109}	Focus group	Consists of a gathering of deliberately selected people who participate in a planned discussion intended to elicit consumer perceptions about a topic or area of interest in an environment that is non-threatening and receptive (Malhotra and Birks, 2007).	Open
T_{110}	Product family penalty function (PFPF)	Measures the dissimilarity among the different parameter settings for each design variable used to define the product family (Simpson *et al.*, 2012).	Open
T_{111}	Hierarchy graph	Illustrates the hierarchical relationships among the elements retrieved from a decomposed DSM (Hsiao *et al.*, 2013).	Open
T_{112}	Disassembly effort index (DEI)	Evaluates the assembly relationships among components (Hsiao *et al.*, 2013).	Open
T_{113}	Analytic network process (ANP)	Consists of a generalization of AHP, which represents a decision-making problem as a network of elements (including criteria and other alternatives) that are grouped into clusters (Hsiao *et al.*, 2013).	Open
T_{114}	Sensitivity analysis	Determines how different values of an independent variable affect a dependent variable under a given set of assumptions (Hsiao *et al.*, 2013).	Open
T_{115}	Simulated annealing	Approximates global optimization in an ample search space for an optimization problem (Agard and Bassetto, 2013).	Open
T_{116}	Scaling by minimizing a convex function (SMACOF)	Transforms the DSM constituents into two-dimensional vectors (Li *et al.*, 2013).	Open
T_{117}	Partition coefficient (PC)	Evaluates the clustering quality of a decomposed DSM (Moon, Kumara and Simpson, 2006).	Open
T_{118}	Dendrogram	Illustrates the fusions or divisions which have been made at each successive stage of hierarchical classification (Selim, Askin and Vakharia, 1998).	Open
T_{119}	Modular function deployment	Uses three interlinked matrices to describe customer requirements, engineering specifications, and product family strategy (Borjesson and Hoelttae-Otto, 2014).	Open
T_{120}	R-IGTA algorithm	Generates modules through simultaneously clustering similarity-based (MIM) and coupling-based (DSM) matrices (Borjesson and Hoelttae-Otto, 2014).	Open
T_{121}	Algorithm for modular principle selection	Establishes the specific modular architecture alternatives for each subfunction proposed in the original functional structure (Mesa *et al.*, 2014).	Open
T_{122}	Average linkage clustering algorithm (ALCA)	Forms machine cells for cellular manufacturing applications based on the similarity coefficients (Seifoddini, 1989).	Open
T_{123}	Strength Pareto evolutionary algorithm (SPEA-II)	Locates and maintain a front of non-dominated solutions, ideally a set of Pareto optimal solutions, by using an evolutionary process to explore the search space (Jiao, 2012).	Open
T_{124}	Environmental performance metric	Denotes the environmental performance of a product based on its reusability and recyclability (Yang, Yu and Jiang, 2014).	Open
T_{125}	Shared surplus measure	Expresses the ratio between the customer-perceived product utility and the costs to produce it (Jiao and Zhang, 2005).	Open
T_{126}	Principal component analysis (PCA)	Reduces the dimensionality of multivariate data while preserving the relevant information as much as possible (Meng *et al.*, 2014).	Open
T_{127}	Support vector machine (SVM)	Predicts the unknown outputs of a system based on a training dataset (Meng *et al.*, 2014).	Open
T_{128}	Breadth-first search (BSF)	Searches for tree or graph data structures (Fan *et al.*, 2015).	Open
T_{129}	Node degree metric	Indicates the number of links related to a specific node that composes a product structure (Fan *et al.*, 2015).	Open
T_{130}	Usage time metric	Computes the number of times a specific module is used in a product family (Fan *et al.*, 2015).	Open
T_{131}	Number of different products metric	Computes the number of different products that use a specific module in a product family (Fan *et al.*, 2015).	Open
T_{132}	Node betweenness metric	Measures the number of shortest paths via a particular node in the network (Fan *et al.*, 2015).	Open
T_{133}	Degree of variety of a module (DVM)	Classifies modules into common or different modules (Yu *et al.*, 2015).	Open
T_{134}	Platform priority matrix (PPM)	Grades the strategic importance of product modularity drivers (Scalice *et al.*, 2015).	Open
T_{135}	Module indication matrix (MIM)	Captures the strategic intent of technical solutions (Borjesson and Hoelttae-Otto, 2014).	Open
T_{136}	Improved strength Pareto evolutionary algorithm (ISPEA-II)	Locates and maintain a front of non-dominated solutions, ideally a set of Pareto optimal solutions, by using a more effective evolutionary process than SPEA-II, to explore the search space (Wei *et al.*, 2015).	Open

(continued)

Table A2. (continued)

Id.	Codes	Definition / Function	Type
T_{137}	Fuzzy ranking mechanism	Extracts the best compromise solution from the Pareto-optimal front (Wei *et al.*, 2015).	Open
T_{138}	Mathematical modeling	Constructs a mathematical model that represents the essence of the problem to be solved (Hilier and Lieberman, 2015).	Open
T_{139}	Multi-objective particle swarm optimization algorithm (MOPSO)	Solves multi-objective problems by moving a collection of particles that move around a continuous search space influenced by their own best past location and the best past location of the whole swarm or a close neighbor (Jiao, 2012).	Open
T_{140}	Mixed-discrete particle swarm optimization algorithm (MDPSO)	Solves multi-objective problems by moving a collection of particles that move around a discrete or continuous search space influenced by their own best past location and the best past location of the whole swarm or a close neighbor (Chowdhury *et al.*, 2016).	Open
T_{141}	Structural complexity metric (SC)	Captures the overall complexity of components and interfaces in the product platforms or variants (Kim *et al.*, 2016).	Open
T_{142}	Trapezoidal fuzzy arithmetic	Translates the linguistic terms into fuzzy numbers to support the quantitative indices calculation (Li *et al.*, 2016).	Open
T_{143}	Change propagation index (CPI)	Measures the degree of physical change propagation caused by an element when an external change is imposed on the system (Suh et al. 2007).	Open
T_{144}	Scalable index (SI)	Measures the degree of scaling parameters of a component (Li *et al.*, 2016).	Open
T_{145}	Clonal selection algorithm (CSA)	Clusters data by imitating the principles of clonal selection in an immune system (Aydin and Ulutas, 2016).	Open
T_{146}	Functional coupling	Indicates how strong is the functional relationship among components (Ma *et al.*, 2016).	Open
T_{147}	Technological coupling	Indicates how strong is the technological relationship among components (Ma *et al.*, 2016).	Open
T_{148}	Structural coupling	Indicates how strong is the structural relationship among components (Ma *et al.*, 2016).	Open
T_{149}	Sourcing cost	Calculates the sourcing cost associated to a modular product family (Ma *et al.*, 2016).	Open
T_{150}	Process cost	Calculates the process cost associated to a modular product family (Ma *et al.*, 2016).	Open
T_{151}	Company strategic landscape (CSL)	Presents the main elements of a business environment that relate to the product structuring (Pakkanen, Juuti and Lehtonen, 2016).	Open
T_{152}	Cause-and-effect diagram	Identifies possible causes for an effect or problem (Pakkanen, Juuti and Lehtonen, 2016).	Open
T_{153}	Brainstorming	Produces ideas or ways of solving problems through a spontaneous group of discussion (Pakkanen, Juuti and Lehtonen, 2016).	Open
T_{154}	Product family master plan (PFMP)	Provides an object-oriented modeling formalism for product families and highlights customer, engineering, and part views (Pakkanen, Juuti and Lehtonen, 2016).	Open
T_{155}	K-matrix	Maps the relationship between technical and customer views (Pakkanen, Juuti and Lehtonen, 2016).	Open
T_{156}	Product structuring blueprint (PSBP)	Describes the name of the product family in question, the generic elements it includes, the solution principles for each generic element and type of each solution and variation needs (Pakkanen, Juuti and Lehtonen, 2016).	Open
T_{157}	Business impact analysis (BIA)	Gives a rough estimate of profit in a workshop environment (Pakkanen, Juuti and Lehtonen, 2016).	Open
T_{158}	Graph-based decomposition	Graphically represents the components and their interactions (Hou *et al.*, 2017).	Open
T_{159}	Structural stiffness	Measures the displacement of a given structural element (Hou *et al.*, 2017).	Open
T_{160}	Manufacturability	Measures manufacturability of a given model based on the total expenses of its sub-components (Hou *et al.*, 2017).	Open
T_{161}	Assembling ability	Measures the effectiveness of a design from an assembly point of view (Hou *et al.*, 2017).	Open
T_{162}	Design checklist	Supports the decision-making on what type of platform architecture (modular or integral) should be developed (Shamsuzzoha and Helo, 2017).	Open
T_{163}	Aggregate manufacturing cost (AMC)	Represents the manufacturing cost incurred in offering the optimal number of product profiles (Goswami, Daultani and Tiwari, 2017).	Open
T_{164}	Penalty for violating time to the market constraint (PTMC)	Represents the penalty in case the manufacturer is not able to meet the time to market requirements (Goswami, Daultani and Tiwari, 2017).	Open
T_{165}	Customer satisfaction loss cost (CSLC)	Refers to the loss incurred in offering an existing product attribute far away from the optimal level (Goswami, Daultani and Tiwari, 2017).	Open
T_{166}	Common index (CI)	Measures the degree of sharing parameters of a component (Li *et al.*, 2016).	Open
T_{167}	Product premium index	Measures the degree of market penetration based on the product's attribute level (Goswami, Daultani and Tiwari, 2017).	Open
T_{168}	CAx activity	Builds conceptual parameterized CAD & CAE models and integrates it with PLM & PDM software (Johannesson *et al.*, 2017).	Open
T_{169}	Time-driven activity-based costing (TDABC)	Estimates the cost of a product based on the unit cost of supplying capacity and the time required to perform it (Kaplan and Anderson, 2007).	Open
T_{170}	Cohesion degree	Measures the similarity among physical components within modules (Cheng, Xiao and Wang, 2018).	Open
T_{171}	Coupling degree	Represents the degree of the interaction between modules (Cheng, Xiao and Wang, 2018).	Open
T_{172}	Correlation impact degree	Describes the impact of one module on another module (Cheng, Xiao and Wang, 2018).	Open
T_{173}	Instability index	Evaluates the change probability of a component in the future generation (Wang *et al.*, 2018).	Open
T_{174}	Change propagation tree	Depicts the possible change propagation path (Wang *et al.*, 2018).	Open
T_{175}	Comprehensive connection	Grades the structural interaction among components (Wang *et al.*, 2018).	Open

(continued)

Table A2. (continued)

Id.	Codes	Definition / Function	Type
T_{176}	Material similarity	Grades the materials' compatibility among components (Wang *et al.*, 2018).	Open
T_{177}	Maintenance similarity	Grades the maintenance similarity among components (Wang *et al.*, 2018).	Open
T_{178}	Manufacturing technology and process similarity	Grades the manufacturing technology and process similarity among components (Wang *et al.*, 2018).	Open
T_{179}	Low carbon performance	Measures the performance of low carbon in modular planning (Wang *et al.*, 2018).	Open
T_{180}	Adaptive memetic algorithm (AMA)	Combines the global and local search through GA to explore the solution space (Wang *et al.*, 2018).	Open
T_{181}	Total constant commonality index (TCCI)	Measures the absolute level of component commonality in a product family (Thevenot and Simpson, 2007).	Open
T_{182}	Modularity score	Calculates the score of clustering solution for each product variant in the family (Baylis, Zhang and McAdams, 2018).	Open
T_{183}	Venn diagram	Visually illustrates the platforming strategy (Baylis, Zhang and McAdams, 2018).	Open
T_{184}	Jaccard similarity coefficient	Measures the similarity between sample sets (Mesa *et al.*, 2014).	Open
T_{185}	Ward's linkage algorithm	Groups components into clusters by minimizing within-group dispersion based on a classical sum-of-squares criterion (Everitt *et al.*, 2011).	Open
T_{186}	Morphology analysis	Creates alternative architectures through the modules' combination (Ko and Kuo, 2010).	Open
T_{187}	Expectation maximization clustering	Assumes the data set can be modeled as a linear combination of multivariate normal distributions and finds the distribution parameters that maximize a model quality measure (Abbas, 2008).	Open
T_{188}	Descriptive statistics	Describes the measures of central tendency and variability of a given data set (Montgomery and Runger, 2011).	Open
T_{189}	Requirements list	Lists the product development requirements into a single document (Pahl *et al.*, 2007).	Open
T_{190}	Classification scheme	Serves as catalog during the searching for design solutions (Pahl *et al.*, 2007).	Open
T_{191}	Elimination and preference	Selects design solutions through multicriteria decision-making (Pahl *et al.*, 2007).	Open
T_{192}	Modularity Index (MI)	Evaluates modularity in product architectures (Jung and Simpson, 2017).	Open
T_{193}	Adherence index	Indicates the level of utilization of basic, auxiliary, and adaptive modules within a module-based machine variant (Gauss, Lacerda and Sellitto, 2019)	Open

Table A3. Structured classes of design problems.

Classes of design problems	Design problems	Artifacts: Techniques	Artifacts: Methods	Evaluation approach	Product classification	Primary studies
Cp_1	$Pb_{1.1}$	T_3	M_8	Et_1	Pt_1	R_8
			M_{37}		Pt_2	R_{37}
		T_4	M_1			R_1
		T_{26}	M_5	Et_5	Pt_1	R_5
		T_{27}				
		T_{38}	M_8	Et_1		R_8
		T_{44}	M_9	Et_5		R_9
		T_{58}	M_{12}		Pt_1, Pt_2, Pt_3	R_{12}
		T_{79}	M_{20}	Et_1	Pt_1	R_{20}
		T_{134}	M_{46}			R_{46}
		T_{151}	M_{54}		Pt_3	R_{54}
		T_{152}				
		T_{162}	M_{58}		Pt_2	R_{58}
		N/A	M_{70}		Pt_1	R_{70}
	$Pb_{1.2}$	T_3	M_{32}			R_{32}
		T_5	M_1		Pt_2	R_1
		T_7	M_{31}	Et_5	Pt_4	R_{31}
		T_{10}	M_{27}	Et_1	Pt_1	R_{27}
		T_{45}	M_{11}	Et_5		R_{11}
			M_{31}		Pt_4	R_{31}
			M_{37}	Et_1	Pt_2	R_{37}
			M_{47}	Et_5		R_{47}
			M_{71}	Et_1	Pt_3	R_{71}
		T_{77}	M_{19}		Pt_1	R_{19}
		T_{86}	M_{32}			R_{32}
		T_{113}				
		N/A	M_{44}		Pt_2	R_{44}
			M_{51}	Et_5	Pt_3	R_{51}
		T_{185}	M_{70}	Et_1	Pt_1	R_{70}
Cp_2	$Pb_{2.1}$	T_3	M_{19}		Pt_1	R_{19}
			M_{54}		Pt_3	R_{54}
			M_{56}			R_{56}
			M_{70}		Pt_1	R_{70}
		T_{18}				
		T_{28}	M_{30}		Pt_2	R_{30}
		T_{29}	M_{70}		Pt_1	R_{70}
		T_{39}	M_8		Pt_1	R_8
			M_{16}		Pt_4	R_{16}
			M_{30}		Pt_2	R_{30}
			M_{32}		Pt_1	R_{32}
		T_{109}	M_{30}		Pt_2	R_{30}
	$Pb_{2.2}$	T_3	M_1			R_1
			M_8		Pt_1	R_8
			M_{16}		Pt_4	R_{16}
			M_{41}		Pt_2	R_{41}
		T_6	M_1			R_1
			M_{48}			R_{48}
		T_{29}	M_{41}			R_{41}
	$Pb_{2.3}$	T_1	M_1			R_1
		T_2	M_2	Et_5	Pt_1	R_2
		T_3	M_{72}		Pt_3	R_{72}
		T_6	M_{39}	Et_1		R_{39}
		T_{10}	M_{56}			R_{56}
		T_{18}	M_4	Et_5	Pt_1	R_4
			M_8	Et_1		R_8
			M_9	Et_5		R_9
			M_{10}		Pt_3	R_{10}
			M_{14}	Et_1	Pt_1	R_{14}
			M_{15}		Pt_2	R_{15}
			M_{16}		Pt_4	R_{16}
			M_{20}		Pt_1	R_{20}
			M_{23}			R_{23}
			M_{24}		Pt_3	R_{24}
			M_{30}		Pt_2	R_{30}
			M_{39}		Pt_3	R_{39}
			M_{46}		Pt_1	R_{46}
			M_{60}		Pt_2	R_{60}
			M_{72}	Et_5	Pt_3	R_{72}
		T_{21}				
		T_{22}				
		T_{23}				

(continued)

Table A3. (continued)

Classes of design problems	Design problems	Artifacts: Techniques	Artifacts: Methods	Evaluation approach	Product classification	Primary studies
Cp_2	$Pb_{2.3}$	T_{28}	M_5	Et_1	Pt_1	R_5
			M_{17}			R_{17}
			M_{45}			R_{45}
			M_{55}			R_{55}
		T_{39}	M_{66}		Pt_2	R_{66}
		T_{121}	M_{39}		Pt_3	R_{39}
		N/A	M_{54}			R_{54}
			M_{67}			R_{67}
	$Pb_{2.4}$	T_{28}	M_{65}		Pt_1	R_{65}
		N/A	M_9	Et_5		R_9
			M_{67}	Et_1	Pt_3	R_{67}
	$Pb_{2.5}$	T_{10}	M_{39}			R_{39}
			M_{67}			R_{67}
		T_{18}	M_4	Et_5	Pt_1	R_4
			M_8	Et_1		R_8
			M_{10}	Et_5	Pt_3	R_{10}
			M_{14}	Et_1	Pt_1	R_{14}
			M_{15}		Pt_2	R_{15}
			M_{16}		Pt_4	R_{16}
			M_{20}		Pt_1	R_{20}
			M_{23}			R_{23}
			M_{24}		Pt_3	R_{24}
			M_{30}		Pt_2	R_{30}
		T_{31}	M_{66}			R_{66}
	$Pb_{2.6}$	T_7	M_1			R_1
		T_{28}	M_{17}		Pt_1	R_{17}
			M_{20}			R_{20}
			M_{45}			R_{45}
		T_{29}	M_5	Et_5		R_5
		T_{86}	M_{56}	Et_1	Pt_3	R_{56}
			M_{67}			R_{67}
		T_{122}	M_{39}			R_{39}
		T_{184}				
		T_{187}	M_{72}	Et_5	Pt_3	R_{72}
		T_{188}				
	$Pb_{2.7}$	T_8	M_1	Et_1	Pt_2	R_1
			M_{72}	Et_5	Pt_3	R_{72}
		T_{18}	M_2		Pt_1	R_2
			M_4			R_4
			M_8	Et_1		R_8
			M_{24}		Pt_3	R_{24}
			M_{30}		Pt_2	R_{30}
			M_{37}			R_{37}
			M_{40}		Pt_1	R_{40}
			M_{46}			R_{46}
			M_{51}	Et_5	Pt_3	R_{51}
		T_{118}	M_{39}	Et_1		R_{39}
		T_{189}	M_{72}	Et_5		R_{72}
Cp_3	$Pb_{3.1}$	T_{19}	M_{55}	Et_1	Pt_1	R_{55}
		T_{30}	M_5	Et_5		R_5
			M_{15}	Et_1	Pt_2	R_{15}
			M_{31}	Et_5	Pt_4	R_{31}
			M_{55}	Et_1	Pt_1	R_{55}
		T_{32}	M_5	Et_5		R_5
			M_{17}	Et_1		R_{17}
			M_{26}	Et_5		R_{26}
			M_{31}		Pt_4	R_{31}
			M_{53}	Et_1	Pt_1	R_{53}
		T_{70}	M_{16}		Pt_4	R_{16}
		T_{73}	M_{17}		Pt_1	R_{17}
			M_{45}			R_{45}
		T_{90}	M_{23}			R_{23}
		T_{91}				
		T_{97}	M_{26}	Et_5		R_{26}
		T_{98}				
		T_{110}	M_{31}		Pt_4	R_{31}
		T_{112}	M_{32}	Et_1	Pt_1	R_{32}
		T_{114}				
		T_{124}	M_{40}			R_{40}
		T_{129}	M_{44}		Pt_2	R_{44}
		T_{130}				
		T_{131}				
		T_{132}				

(continued)

Table A3. (continued)

Classes of design problems	Design problems	Artifacts: Techniques	Artifacts: Methods	Evaluation approach	Product classification	Primary studies
Cp_3	$Pb_{3.1}$	T_{133}	M_{45}	Et_1	Pt_1	R_{45}
		T_{138}	M_7	Et_5		R_7
			M_{56}	Et_1	Pt_3	R_{56}
		T_{147}	M_{53}		Pt_1	R_{53}
		T_{148}				
		T_{149}				
		T_{150}				
		T_{146}				
		T_{159}	M_{57}		Pt_2	R_{57}
			M_{62}			R_{62}
		T_{160}	M_{57}			R_{57}
		T_{161}				
		T_{169}	M_{61}	Et_5	Pt_3	R_{61}
		T_{170}	M_{64}	Et_1		R_{64}
		T_{171}				
		T_{172}				
		T_{175}	M_{65}	Et_1	Pt_1	R_{65}
		T_{176}				
		T_{177}				
		T_{178}				
		T_{179}				
		N/A	M_{30}	Et_1	Pt_2	R_{30}
	$Pb_{3.2}$	T_9	M_1			R_1
			M_{56}		Pt_3	R_{56}
			M_{67}			R_{67}
			M_{72}	Et_5		R_{72}
		T_{153}	M_{54}	Et_1		R_{54}
		N/A	M_{60}		Pt_2	R_{60}
		T_{186}	M_{71}		Pt_3	R_{71}
		T_{190}	M_{72}	Et_5		R_{72}
		T_{191}				
	$Pb_{3.3}$	T_{10}	M_1	Et_1	Pt_2	R_1
			M_5	Et_5	Pt_1	R_5
			M_{15}	Et_1	Pt_2	R_{15}
			M_{17}		Pt_1	R_{17}
			M_{26}	Et_5		R_{26}
			M_{27}	Et_1		R_{27}
			M_{36}			R_{36}
			M_{45}			R_{45}
			M_{48}		Pt_2	R_{48}
			M_{56}		Pt_3	R_{56}
			M_{66}		Pt_2	R_{66}
			M_{72}	Et_5	Pt_3	R_{72}
		T_{24}	M_4		Pt_1	R_4
			M_9			R_9
			M_{10}		Pt_3	R_{10}
			M_{40}	Et_1	Pt_1	R_{40}
			M_{71}		Pt_3	R_{71}
		T_{28}	M_{55}		Pt_1	R_{55}
			M_{65}			R_{65}
		T_{31}	M_8			R_8
		T_{119}	M_{38}			R_{38}
		T_{155}	M_{54}		Pt_3	R_{54}
	$Pb_{3.4}$	T_{11}	M_1		Pt_2	R_1
			M_{72}	Et_5	Pt_3	R_{72}
		T_{21}	M_4		Pt_1	R_4
			M_8	Et_1		R_8
			M_9	Et_5		R_9
			M_{10}		Pt_3	R_{10}
			M_{16}	Et_1	Pt_4	R_{16}
			M_{20}		Pt_1	R_{20}
			M_{23}			R_{23}
			M_{30}		Pt_2	R_{30}
			M_{46}		Pt_1	R_{46}
			M_{51}	Et_5	Pt_3	R_{51}
		T_{22}	M_4		Pt_1	R_4
			M_8	Et_1		R_8
			M_9	Et_5		R_9
			M_{10}		Pt_3	R_{10}
			M_{16}	Et_1	Pt_4	R_{16}
			M_{20}		Pt_1	R_{20}
			M_{23}			R_{23}
			M_{30}		Pt_2	R_{30}

(continued)

Table A3. (continued)

Classes of design problems	Design problems	Artifacts: Techniques	Artifacts: Methods	Evaluation approach	Product classification	Primary studies
Cp_3	$Pb_{3.4}$	T_{22}	M_{46}	Et_1	Pt_1	R_{46}
			M_{51}	Et_5	Pt_3	R_{51}
		T_{23}	M_4		Pt_1	R_4
			M_8	Et_1		R_8
			M_9	Et_5		R_9
			M_{10}		Pt_3	R_{10}
			M_{16}	Et_1	Pt_4	R_{16}
			M_{20}		Pt_1	R_{20}
			M_{23}			R_{23}
			M_{30}		Pt_2	R_{30}
			M_{46}		Pt_1	R_{46}
			M_{51}	Et_5	Pt_3	R_{51}
		T_{72}	M_{40}	Et_1	Pt_1	R_{40}
		T_{120}	M_{38}			R_{38}
		T_{136}	M_{48}		Pt_2	R_{48}
	$Pb_{3.5}$	T_{12}	M_1		Pt_2	R_1
		T_{40}	M_8		Pt_1	R_8
			M_{30}		Pt_2	R_{30}
		N/A	M_5	Et_5	Pt_1	R_5
			M_{16}	Et_1	Pt_4	R_{16}
			M_{51}	Et_5	Pt_3	R_{51}
			M_{54}	Et_1		R_{54}
			M_{72}			R_{72}
	$Pb_{3.6}$	T_6	M_{15}		Pt_2	R_{15}
		T_{31}	M_5	Et_5	Pt_1	R_5
			M_9			R_9
			M_{14}	Et_1		R_{14}
			M_{15}		Pt_2	R_{15}
			M_{17}		Pt_1	R_{17}
			M_{24}		Pt_3	R_{24}
			M_{25}	Et_5	Pt_1	R_{25}
			M_{26}			R_{26}
			M_{27}	Et_1		R_{27}
			M_{31}	Et_5	Pt_4	R_{31}
			M_{32}	Et_1	Pt_1	R_{32}
			M_{35}		Pt_3	R_{35}
			M_{38}		Pt_1	R_{38}
			M_{45}			R_{45}
			M_{48}		Pt_2	R_{48}
			M_{50}			R_{50}
			M_{52}		Pt_1	R_{52}
			M_{53}			R_{53}
			M_{55}			R_{55}
			M_{64}		Pt_3	R_{64}
			M_{65}		Pt_1	R_{65}
			M_{66}		Pt_2	R_{66}
			M_{67}		Pt_3	R_{67}
			M_{69}		Pt_1	R_{69}
		T_{100}	M_{27}			R_{27}
			M_{33}	Et_5		R_{33}
			M_{65}	Et_1		R_{65}
		T_{158}	M_{57}		Pt_2	R_{57}
		T_{174}	M_{65}		Pt_1	R_{65}
	$Pb_{3.7}$	T_7	M_{35}		Pt_3	R_{35}
			M_{45}		Pt_1	R_{45}
		T_{11}	M_8			R_8
		T72	M_{15}		Pt_2	R_{15}
			M_{24}		Pt_3	R_{24}
			M_{26}	Et_5	Pt_1	R_{26}
			M_{27}	Et_1		R_{27}
			M_{53}			R_{53}
		T_{81}	M_{57}		Pt_2	R_{57}
		T_{82}	M_{66}			R_{66}
			M_{69}		Pt_1	R_{69}
		T_{99}	M_{27}			R_{27}
			M_{33}	Et_5		R_{33}
			M_{67}	Et_1	Pt_3	R_{67}
		T_{115}	M_{34}		Pt_2	R_{34}
		T_{116}	M_{35}		Pt_3	R_{35}
		T_{118}	M_{36}		Pt_1	R_{36}
		T_{120}	M_{38}			R_{38}
		T_{135}	M_{46}			R_{46}
		T_{136}	M_{48}		Pt_2	R_{48}

(continued)

Table A3. (continued)

Classes of design problems	Design problems	Artifacts: Techniques	Artifacts: Methods	Evaluation approach	Product classification	Primary studies
Cp_3	$Pb_{3.7}$	T_{145}	M_{52}	Et_1	Pt_1	R_{52}
		T_{180}	M_{65}			R_{65}
		N/A	M_{32}			R_{32}
			M_{60}		Pt_2	R_{60}
			M_{64}		Pt_3	R_{64}
	$Pb_{3.8}$	T_3	M_{37}		Pt_2	R_{37}
		T_{13}	M_1			R_1
		T_{14}	M_{72}		Pt_3	R_{72}
		T_{15}	M_1		Pt_2	R_1
			M_{72}		Pt_3	R_{72}
		T_{16}	M_1		Pt_2	R_1
		T_{47}	M_{12}	Et_5	Pt_1, Pt_2, Pt_3	R_{12}
		T_{48}				
		T_{49}				
		T_{50}				
		T_{51}				
		T_{52}				
		T_{53}				
		T_{54}				
		T_{55}				
		T_{56}				
			M_{30}	Et_1	Pt_2	R_{30}
		T_{57}	M_{12}	Et_5	Pt_1, Pt_2, Pt_3	R_{12}
		T_{59}				
		T_{60}				
		T_{61}				
		T_{63}				
		T_{64}				
		T_{65}				
		T_{93}	M_{24}	Et_1	Pt_3	R_{24}
		T_{117}	M_{35}			R_{35}
		T_{137}	M_{48}		Pt_2	R_{48}
		T_{192}	M_{72}		Pt_3	R_{72}
	$Pb_{3.9}$	T_3	M_{51}	Et_5	Pt_3	R_{51}
		T_7				
		T_{24}				
		T_{74}	M_{62}	Et_1	Pt_2	R_{62}
		T_{90}	M_{51}	Et_5	Pt_3	R_{51}
		T_{99}	M_{33}		Pt_1	R_{33}
			M_{67}	Et_1	Pt_3	R_{67}
		T_{142}	M_{51}	Et_5		R_{51}
		T_{143}				
		T_{144}				
		T_{166}				
		T_{173}	M_{65}	Et_1	Pt_1	R_{65}
		N/A	M_4	Et_5		R_4
			M_{44}	Et_1	Pt_2	R_{44}
			M_{52}		Pt_1	R_{52}
			M_{54}		Pt_3	R_{54}
			M_{57}		Pt_2	R_{57}
			M_{60}			R_{60}
	$Pb_{3.10}$	T_8	M_1			R_1
			M_{72}		Pt_3	R_{72}
		T_{17}	M_1		Pt_2	R_1
			M_{11}	Et_5	Pt_1	R_{11}
			M_{23}	Et_1		R_{23}
			M_{24}		Pt_3	R_{24}
		T_{20}	M_3		Pt_2	R_3
			M_7	Et_5	Pt_1	R_7
			M_{18}	Et_1, Et_3	Pt_3	R_{18}
			M_{21}			R_{21}
		T_{31}	M_{49}	Et_5	Pt_4	R_{49}
		T_{87}	M_{23}	Et_1	Pt_1	R_{23}
		T_{92}	M_{24}		Pt_3	R_{24}
		T_{111}	M_{32}		Pt_1	R_{32}
		T_{128}	M_{44}		Pt_2	R_{44}
		T_{154}	M_{54}		Pt_3	R_{54}
		T_{156}				
		T_{168}	M_{60}		Pt_2	R_{60}
		T_{183}	M_{69}		Pt_1	R_{69}
	$Pb_{3.11}$	T_{19}	M_2	Et_5		R_2
		T_{25}	M_4			R_4

(continued)

Table A3. (continued)

Classes of design problems	Design problems	Artifacts: Techniques	Artifacts: Methods	Evaluation approach	Product classification	Primary studies
Cp_3	$Pb_{3.11}$	T_{33}	M_6	Et_1	Pt_3	R_6
		T_{41}	M_8		Pt_1	R_8
		T_{42}				
		T_{46}	M_{11}	Et_5		R_{11}
			M_{13}			R_{13}
		T_{47}	M_{12}		Pt_1, Pt_2, Pt_3	R_{12}
		T_{62}				
		T_{66}				
		T_{67}	M_{13}	Et_1	Pt_1	R_{13}
			M_{14}			R_{14}
		T_{141}	M_{50}		Pt_2	R_{50}
		T_{157}	M_{54}		Pt_3	R_{54}
		N/A	M_{60}		Pt_2	R_{60}
			M_{67}		Pt_3	R_{67}
		T_{193}	M_{72}			R_{72}
Cp_4	$Pb_{4.1}$	T_{14}	M_{28}		Pt_2	R_{28}
		T_{29}	M_{63}		Pt_1	R_{63}
			M_{68}			R_{68}
		T_{32}	M_{49}	Et_5	Pt_4	R_{49}
		T_{37}	M_7		Pt_1	R_7
			M_9			R_9
		T_{75}	M_{19}	Et_1		R_{19}
		T_{78}				
		T_{80}	M_{21}	Et_1, Et_3	Pt_3	R_{21}
		T_{88}	M_{23}	Et_1	Pt_1	R_{23}
		T_{95}	M_{25}	Et_5		R_{25}
		T_{102}	M_{28}	Et_1	Pt_2	R_{28}
		T_{103}				
		T_{104}				
		T_{105}				
		T_{106}				
		T_{107}	M_{41}			R_{41}
			M_{59}			R_{59}
			M_{63}		Pt_1	R_{63}
			M_{68}			R_{68}
		T_{108}	M_{28}		Pt_2	R_{28}
		T_{125}	M_{41}			R_{41}
		T_{138}	M_{49}	Et_5	Pt_4	R_{49}
			M_{63}	Et_1	Pt_1	R_{63}
			M_{68}			R_{68}
		T_{163}	M_{59}		Pt_2	R_{59}
		T_{164}				
		T_{165}				
		T_{167}				
		T_{169}	M_{61}	Et_5	Pt_3	R_{61}
		T_{181}	M_{69}	Et_1	Pt_1	R_{69}
		T_{182}				
		N/A	M_9	Et_5		R_9
			M_{25}			R_{25}
			M_{60}	Et_1	Pt_2	R_{60}
		T_{138}	M_{70}		Pt_1	R_{70}
	$Pb_{4.2}$	T_{72}	M_{22}			R_{22}
		T_{83}				
		T_{84}				
		T_{85}				
		T_{86}				
		T_{126}	M_{42}		Pt_2	R_{42}
		T_{127}				
	$Pb_{4.3}$	T_{34}	M_7	Et_5	Pt_1	R_7
			M_{69}	Et_1		R_{69}
		T_{35}	M_7	Et_5		R_7
		T_{36}				
		T_{43}	M_9			R_9
		T_{72}	M_{42}	Et_1	Pt_2	R_{42}
			M_{63}		Pt_1	R_{63}
			M_{68}			R_{68}
		T_{74}	M_{18}	Et_1, Et_3	Pt_3	R_{18}
		T_{76}	M_{19}	Et_1	Pt_1	R_{19}
		T_{81}	M_{21}	Et_1, Et_3	Pt_3	R_{21}
			M_{28}	Et_1	Pt_2	R_{28}
			M_{29}	Et_5	Pt_4	R_{29}
			M_{47}		Pt_2	R_{47}

(continued)

Table A3. (continued)

Classes of design problems	Design problems	Artifacts		Evaluation approach	Product classification	Primary studies
		Techniques	Methods			
Cp_4	$Pb_{4.3}$	T_{81}	M_{49}	Et_5	Pt_4	R_{49}
		T_{89}	M_{23}	Et_1	Pt_1	R_{23}
		T_{94}	M_{25}	Et_5		R_{25}
		T_{96}				
			M_{43}	Et_1		R_{43}
		T_{101}	M_{28}		Pt_2	R_{28}
		T_{123}				
		T_{138}	M_3			R_3
		T_{139}	M_{28}			R_{28}
		T_{140}	M_{49}	Et_5	Pt_4	R_{49}
		N/A	M_{41}	Et_1	Pt_2	R_{41}
			M_{59}			R_{59}
			M_{60}			R_{60}
			M_{70}			R_{70}
	$Pb_{4.4}$	T_{74}	M_{18}	Et_1, Et_3	Pt_3	R_{18}
		T_{81}	M_{21}			R_{21}
			M_{47}	Et_5	Pt_2	R_{47}
			M_{49}		Pt_4	R_{49}
		T_{140}				
		N/A	M_{41}	Et_1	Pt_2	R_{41}
			M_{60}			R_{60}
		T_{74}	M_{18}	Et_1, Et_3	Pt_3	R_{18}
		T_{81}	M_{21}			R_{21}
			M_{47}	Et_5	Pt_2	R_{47}
			M_{49}		Pt_4	R_{49}
		T_{140}				
		N/A	M_{41}	Et_1	Pt_2	R_{41}
			M_{60}			R_{60}
	$Pb_{4.5}$	T_{20}	M_{72}	Et_5	Pt_3	R_{72}
		T_{17}	M_6	Et_1		R_6
			M_{14}		Pt_1	R_{14}
		T_{68}	M_6		Pt_3	R_6
			M_{14}		Pt_1	R_{14}
		T_{71}	M_3		Pt_2	R_3
			M_{18}	Et_1, Et_3	Pt_3	R_{18}
			M_{21}			R_{21}

Table A4. Incidence matrix.

Primary studies	Design problems																								Classes of design problems			
	$Pb_{1.1}$	$Pb_{1.2}$	$Pb_{2.1}$	$Pb_{2.2}$	$Pb_{2.3}$	$Pb_{2.4}$	$Pb_{2.5}$	$Pb_{2.6}$	$Pb_{2.7}$	$Pb_{3.1}$	$Pb_{3.2}$	$Pb_{3.3}$	$Pb_{3.4}$	$Pb_{3.5}$	$Pb_{3.6}$	$Pb_{3.7}$	$Pb_{3.8}$	$Pb_{3.9}$	$Pb_{3.10}$	$Pb_{3.11}$	$Pb_{4.1}$	$Pb_{4.2}$	$Pb_{4.3}$	$Pb_{4.4}$	$Pb_{4.5}$	Cp_1	Cp_2	Cp_3	Cp_4
R_{9}	1				1	1						1	1		1						1		1			1	1	1	1
R_{1}	1	1		1	1			1	1		1	1	1	1			1		1							1	1	1	
R_{37}	1	1							1								1									1	1	1	
R_{8}	1		1	1	1		1		1			1	1	1		1				1						1	1	1	
R_{54}	1		1		1						1	1		1				1	1	1						1	1	1	
R_{20}	1				1		1	1					1													1	1	1	
R_{5}	1				1			1		1		1		1	1											1	1	1	
R_{46}	1				1				1				1			1										1	1	1	
R_{32}		1	1							1					1	1			1							1	1	1	
R_{51}		1							1				1	1				1								1	1	1	
R_{70}	1	1	1																		1		1			1	1		1
R_{19}		1	1																		1		1			1	1		1
R_{12}	1																1			1						1		1	
R_{31}		1								1					1											1		1	
R_{44}		1								1								1	1							1		1	
R_{71}		1									1	1														1		1	
R_{27}		1										1			1	1										1		1	
R_{11}		1																	1	1						1		1	
R_{47}		1																					1	1		1			1
R_{58}	1																									1			
R_{23}					1		1			1			1						1		1		1				1	1	1
R_{14}					1		1								1					1					1		1	1	1
R_{72}					1			1	1		1	1	1	1			1		1	1					1		1	1	1
R_{60}					1						1					1		1	1	1	1		1	1			1	1	1
R_{16}			1	1	1		1			1			1	1													1	1	
R_{30}			1		1		1		1	1			1	1			1										1	1	
R_{56}			1		1			1		1	1	1															1	1	
R_{48}				1								1	1		1	1	1										1	1	
R_{67}					1	1	1	1			1				1	1		1		1							1	1	
R_{4}					1		1		1			1	1					1		1							1	1	
R_{24}					1		1		1						1	1	1		1								1	1	
R_{15}					1		1			1		1			1	1											1	1	
R_{10}					1		1					1	1														1	1	
R_{66}					1		1					1			1	1											1	1	
R_{45}					1			1		1		1			1	1											1	1	
R_{17}					1			1		1		1			1												1	1	
R_{55}					1					1		1			1												1	1	
R_{65}						1				1		1			1	1		1									1	1	
R_{40}									1	1		1	1														1	1	
R_{2}									1											1							1	1	
R_{41}				1																	1		1	1			1		1
R_{39}					1		1	1	1																		1		
R_{7}										1									1		1		1					1	1
R_{61}										1											1							1	1
R_{69}															1	1			1		1		1					1	1
R_{25}															1						1		1					1	1
R_{21}																			1		1		1	1	1			1	1
R_{49}																			1		1		1	1				1	1
R_{18}																			1				1	1	1			1	1
R_{3}																			1				1		1			1	1
R_{6}																				1					1			1	1

(continued)

Table A4. Incidence matrix.

Primary studies	Design problems																									Classes of design problems			
	$Pb_{1.1}$	$Pb_{1.2}$	$Pb_{2.1}$	$Pb_{2.2}$	$Pb_{2.3}$	$Pb_{2.4}$	$Pb_{2.5}$	$Pb_{2.6}$	$Pb_{2.7}$	$Pb_{3.1}$	$Pb_{3.2}$	$Pb_{3.3}$	$Pb_{3.4}$	$Pb_{3.5}$	$Pb_{3.6}$	$Pb_{3.7}$	$Pb_{3.8}$	$Pb_{3.9}$	$Pb_{3.10}$	$Pb_{3.11}$	$Pb_{4.1}$	$Pb_{4.2}$	$Pb_{4.3}$	$Pb_{4.4}$	$Pb_{4.5}$	Cp_1	Cp_2	Cp_3	Cp_4
R_{26}										1		1			1	1												1	
R_{57}										1					1	1		1										1	
R_{53}										1					1	1												1	
R_{64}										1					1	1												1	
R_{62}										1								1										1	
R_{38}												1	1		1	1												1	
R_{36}												1				1												1	
R_{35}															1	1	1											1	
R_{33}															1	1		1										1	
R_{52}															1	1		1										1	
R_{50}															1					1								1	
R_{34}																1												1	
R_{13}																				1								1	
R_{28}																					1		1						1
R_{59}																					1		1						1
R_{63}																					1		1						1
R_{68}																					1		1						1
R_{42}																						1	1						1
R_{22}																						1							1
R_{29}																							1						1
R_{43}																							1						1
Frequency	11	12	8	5	24	3	13	9	12	21	7	22	15	8	26	7	8	11	15	13	16	2	21	6	6	20	34	58	26
Relative Frequency (1	15.3%	16.7%	11.1%	6.9%	33.3%	4.2%	18.1%	12.5%	16.7%	29.2%	9.7%	30.6%	20.8%	11.1%	36.1%	9.7%	11.1%	15.3%	20.8%	18.1%	22.2%	2.8%	29.2%	8.3%	8.3%	27.8%	47.2%	80.6%	36.1%

1) Frequency divided by the total number of primary studies (72 studies).

APPENDIX B – ARTICLE 2

Table B1. Protocol for systematic literature review.

1.0	Conceptual framework	1.1	Increased demand for a greater variety of consumer products has forced many companies to rethink their strategies to offer more product variants. For manufacturers, producing a variety of products can satisfy this increasing demand and help companies gain more market share; however, increased variety can lead to higher design and production costs as well as longer lead times for new variants. As a result, a trade-off arises between cost-effectiveness and satisfying diverse customer demand (Simpson *et al.*, 2014).
		1.2	As a general reference, based on data from practical situations, it can be said that the production costs decrease by about 15 to 25% whenever the production scale is duplicated (Antunes *et al.*, 2008). In the same way, it can be stated that there is an increase of 20 to 25% of the cost per unit produced each time the variety of manufactured items is duplicated (Antunes *et al.*, 2008).
		1.3	Research has found that the trade-off between product variety and cost-effectiveness can be appropriately managed by exploiting PBPF development, an area that has been widely studied for the past two decades (Simpson *et al.*, 2014).
		1.4	The product family is a set of products that share one or more common "elements" (e.g., components, modules, subsystems, fabrication processes, assembly operations) yet target a variety of different market segments (Simpson *et al.*, 2014). A product family refers to a set of similar products that are derived from a common platform and yet possess specific functionalities to meet particular customer requirements (Meyer and Lehnerd, 1997).
		1.5	The platform is a set of common components, modules, or parts from which a stream of derivative products can be efficiently developed and launched (Meyer and Lehnerd, 1997).
		1.6	In essence, a PFA means the underlying architecture of a firm's product platform, within which various product variants can be derived from basic product designs to satisfy a spectrum of customer needs related to various market niches (Jiao and Tseng, 1999a). Product architecture can be defined as how the functional elements of a product are arranged into physical units and how these units interact (Ulrich, 1995).
		1.7	The functional elements represent the minimum set of independent requirements that characterize entirely the attributes desired in a product (Jiao and Tseng, 1999a; Suh, 2001; Jiao, Simpson and Siddique, 2007). This voice-of-the-customer is a critical factor for specifying new platforms (Ferguson, Olewnik and Cormier, 2011), and usually derives from a marketing-centric perspective (Chen, Hoyle and Wassenaar, 2013). The involvement of customer preferences in engineering design decisions has received remarkable attention recently (Simpson *et al.*, 2014).
		1.8	The physical components implement the functional elements of the product (Ulrich, 1995; Suh, 2001), and its objective is to achieve the best performance within the budget available (Kumar, Chen and Simpson, 2009). A strategy strongly linked to the traditional engineering-centric perspective (Chen, Hoyle and Wassenaar, 2013). Current research in the field of product family design mostly focuses on the tradeoff between increased commonality among products and the resulting decreased ability to meet performance targets for each product variant (Luo, 2011).
		1.9	The two perspectives related to each element of a PFA may lead to a disconnected decision making that cannot assure optimal or near-optimal decisions into product family design (Luo, 2011; Michalek *et al.*, 2011). Where the engineering-centric perspective does not consider customer preferences and demand into the product development, and the marketing-centric perspective does not account for the engineering attributes related to performance and cost into product design (Kumar, Chen and Simpson, 2009).
		1.10	Such highly interconnected relationships between the two domains imply that any required action in one domain can potentially influence the outcomes in the other domain. Therefore, in the design of an optimal or near-optimal product line, the marketing and engineering requirements often cannot be pursued separately or even sequentially (Luo, 2011).
		1.11	Most literature on product line design tackles the optimal selection of products by maximizing the surplus - the margin between the customer-perceived utility and the price of the product. Other objectives widely used in selecting products among a broad set of potential products within a target market include maximization of profit, net present value, a seller's welfare, market share, and share of choices, to name but a few (Jiao, Simpson and Siddique, 2007).
		1.13	Most studies lack data-driven models of market preferences and consequently do not investigate broader business indicators such as profit and market share (Kumar, Chen and Simpson, 2009). As a result, few existing methods examine broader enterprise measures and market considerations with product family design efforts in their formulation (Kumar, Chen and Simpson, 2009).

(continued)

Table B1. (continued).

2.0	Context	2.1	Engineering, production, and operations management domains.
		2.3	Product Development Process - PDP (Ulrich and Eppinger, 2012).
		2.2	Type of products: consumer goods (durables), intermediate goods, capital goods.
3.0	Time horizon	3.1	Product family design and platform-based product development: a state-of-the-art review (Jiao, Simpson and Siddique, 2007) - Up to 2007.
		3.2	A review of mass customization across marketing, engineering, and distribution domains toward the development of a process framework (Ferguson, Olewnik and Cormier, 2014) - From 2007 to 2014.
		3.3	Advances in Product Family and Product Platform Design (Simpson *et al.*, 2014) - From 2007 to 2014.
		3.4	This study: up to 2020.
4.0	Theoretical currents	4.1	Market-driven product family design
5.0	Language	5.1	English
6.0	Research question	6.1	Which methods address market considerations into product family design?
		6.2	Which methods encompass broader business indicators into product family design?
		6.3	What kind of design problems do these methods account for?
		6.4	For which kind of products have these methods been developed?
		6.5	How has the performance of these methods been assessed?
		6.6	What are the main steps of these methods?
		6.7	What is the execution order of these steps?
		6.8	Which techniques are used to execute each step of these methods?
		6.9	Is there a common underlying structure among these methods?
7.0	Review strategy	7.1	Configurative (meta-synthesis).
8.0	Selecting criteria		
8.1	Including criteria	8.1.1	Methods and techniques addressing market considerations and broader business indicators into product family design (Following the conceptual framework of this protocol).
		8.1.2	Document type: Articles.
		8.1.3	All-access types.
8.2	Excluding criteria	8.2.1	Manufacturing and production for product families.
		8.2.2	Supply chain issues of product families.
		8.2.3	Service design.
		8.2.4	Software development.
		8.2.5	Design support systems.
		8.2.6	Theoretical development and synthesis of product family design.
		8.2.7	Fundamental issues on product family design.
		8.2.8	Literature review.
		8.2.9	Document type: Conference paper, review, book, book chapter, and conference review.
		8.2.10	Subjects areas such as computer science, mathematics, decision science, materials science, environmental science economics, econometrics, finance, energy, physics, and astronomy.
9.0	Search terms	9.1	Market-driven.
		9.2	Customer-oriented.
		9.3	Marketing.
		9.4	Design.
		9.5	Product family.
		9.6	Product platform.
		9.7	(“Market-driven” OR “Customer-oriented” OR “Marketing”) AND “design” AND ("Product family" OR "Product platform")
10.0	Data-bases	10.1	Web of Science.
		10.2	Scopus.

Table B2. Mixed coding scheme.

Id.	Codes	Definition / Function	Type
Cp_i	Classes of design problems for PBPF:		
Cp_1	Product family planning and positioning	Deals with market objectives, along with technology developments guided by corporate strategies (Ulrich and Eppinger, 2012).	Categorical
Cp_2	Market-driven product family design	Deals with the transition of customers' needs to functional requirements (Simpson *et al.*, 2014).	Categorical
Cp_3	Product family modeling	Comprehends the definition of product family instances in terms of design parameters and functional requirements (Simpson *et al.*, 2014).	Categorical
Cp_4	Product family configuration	Deals with structural configuration problem wherein the modules formulating the variant are optimally selected (Simpson *et al.*, 2014).	Categorical
Pb_i	Design problems:		
$Pb_{1.1}$	Strategic product family planning	How to incorporate strategic axes into product family design (Jiao and Tseng, 1999a)?	Categorical
$Pb_{1.2}$	Market segmentation	How to decompose the market into several segments taking into account the industry type, customer consumption levels, regional characteristics, among other factors (Fan *et al.*, 2015)?	Categorical
$Pb_{1.3}$	Definition and modeling of product positioning criteria	What criteria to use for positioning the product family into the marketplace (Jiao, Simpson and Siddique, 2007)?	Open
$Pb_{1.4}$	Selection of variants to compound the product line	How to choose the right variants to compound the product line (Miao *et al.*, 2017)?	Open
$Pb_{2.1}$	Identification of customer needs	How to derive meaning through interpretations of customers' perceptions about the existing products (Cheng *et al.*, 2017)?	Categorical
$Pb_{2.2}$	Determination of relative importance among customer needs	How to determine the most influential needs on customer decision making (Du, Jiao and Chen, 2014; Wei *et al.*, 2015)?	Categorical
$Pb_{2.3}$	Formulation of functional requirements	How to translate the market-centric information into engineering specifications (Jung and Simpson, 2016; Johannesson *et al.*, 2017)?	Categorical
$Pb_{2.4}$	Definition of functional requirements target values and ranges	How to arrange similar customers in terms of their desired values (Park *et al.*, 2008; Zacharias and Yassine, 2008; Mesa *et al.*, 2014; Bejlegaard *et al.*, 2018)?	Categorical
$Pb_{2.5}$	Mapping of dependencies among functional requirements	How to determine the functional requirements' hierarchy (Alizon, Shooter and Simpson, 2007; Bonjour *et al.*, 2009; Yan and Stewart, 2010)?	Categorical
$Pb_{2.6}$	Representation of functional requirements	How to represent the functional view of a product family (Jiao and Tseng, 1999a; Kota, Sethuraman and Miller, 2000; Yang, Yu and Jiang, 2014)?	Categorical
$Pb_{3.1}$	Definition and modeling of the product family and platforming criteria	What approach and criteria to use for defining the product family instances and platforms (Messac, Martinez and Simpson, 2002; Miao *et al.*, 2017)?	Categorical
$Pb_{3.2}$	Formulation of design parameters	How to determine the physical effect with the ability to fulfill one or more functional requirements (Gauss, Lacerda and Sellitto, 2019)?	Categorical
$Pb_{3.3}$	Definition of design parameters specification ranges	How to define the design parameters specification ranges to accomplish its respective functional requirements (Ma and Kim, 2016)?	Open
$Pb_{3.4}$	Mapping of product family architecture	How to map the relationships between functional requirements and design parameters (Navarrete *et al.*, 2013; Borjesson and Hoelttae-Otto, 2014)?	Categorical
$Pb_{3.5}$	Decomposition of the system into functional modules	How to decompose the product family architecture into design modules (Jiao and Tseng, 1999a)?	Categorical
$Pb_{3.6}$	Creation of rough geometric layouts	How to identify the interactions among physical components (Pakkanen, Juuti and Lehtonen, 2016)?	Categorical
$Pb_{3.7}$	Mapping of structural dependencies among components	How to model the structural dependencies among components (Yu *et al.*, 2015; Kim *et al.*, 2016; Baylis, Zhang and McAdams, 2018)?	Categorical
$Pb_{3.8}$	Decomposition of the system into physical modules	How to decompose a set of structural relationships into physical modules (Bonjour *et al.*, 2009)?	Categorical
$Pb_{3.9}$	Specification of product family instances	How to obtain the product family instances that best satisfy the overall design requirements (Messac, Martinez and Simpson, 2002)?	Open
$Pb_{3.10}$	Building of the MBPF configuration structure	How to build a hierarchical structure for end product configuration (Li, Huang and Newman, 2008)?	Categorical
$Pb_{3.11}$	Evaluation of module-based product family design	How to evaluate product families as a whole and generate measures of deviation from the ideal (Otto and Hölttä-Otto, 2007)?	Categorical
$Pb_{4.1}$	Definition and modeling of configuration criteria	What criteria to use for modeling the combinatorial and parametric problem (Li, Huang and Newman, 2008; Li and Huang, 2009)?	Categorical
$Pb_{4.3}$	Combination of modules to generate product family variants	How to determine the right combination of modules to formulate the product family variants (Xiong, Du and Jiao, 2018)?	Categorical

(continued)

Table B2. (continued).

Id.	Codes	Definition / Function	Type
Ct_i	Criteria:		
Ct_1	Utility	The level of users' satisfaction with a product (Yoshimura and Takeuchi, 1994).	Categorical
Ct_2	Cost	The amount of expenditure incurred to produce a product (Wouters and Morales, 2014).	Categorical
Ct_3	Commonality	The sharing of intellectual and material assets across products to minimize manufacturing complexity (Erens and Verhulst, 1997).	Categorical
Ct_4	Redesign effort	The amount of redesign effort required for future designs of the product (Martin and Ishii, 2002).	Categorical
Ct_5	Interaction or coupling	The strength of coupling between the components in a product (Martin and Ishii, 2002).	Categorical
Ct_6	Modularity	The decomposition of a system into independent modules that can be treated as logical units (Newcomb, Bras and Rosen, 1998).	Categorical
Ct_7	Quality	The state of being free from defects (Chan and Wu, 2002).	Categorical
Ct_8	Demand	The quantity of a product the customer intends to purchase (Antunes *et al.*, 2008).	Categorical
Ct_9	Profit	The economic benefit of a product to an enterprise (Dong, Shao and Xiong, 2011; Chen, Hoyle and Wassenaar, 2013).	Categorical
Ct_{10}	Price	The amount of money the customer is willing to pay for a product (Cox and Schleier, 2010).	Categorical
Ct_{11}	Variety	The level of distinctiveness of the product's offering in the marketplace (Chen, Hoyle and Wassenaar, 2013).	Categorical
P_{ti}	Product classification:		
P_{t1}	Consumer goods (durables)	Consist of durable products that people buy for their use (OECD, 2008).	Categorical
P_{t2}	Intermediate goods	Comprehend those products used in the production of other goods (OECD, 2008).	Categorical
P_{t3}	Capital goods	Consist of machines and equipment used to produce products or provide services (OECD, 2008).	Categorical
E_{ti}	Evaluation approach:		
E_{t1}	Observational	(i) Case study elements: study the existing or created artifact in-depth in the business environment. (ii) Field study: monitor the use of the artifact in multiple projects (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
E_{t2}	Analytical	(i) Static analysis: examine the structure of the artifact for static qualities. (ii) Architecture analysis: study the fit of the artifact in the technical architecture of the complete technical system. (iii) Optimization: demonstrate the optimal properties inherent to the artifact or demonstrate the limits of the optimization in the artifact behavior. (iv) Dynamic analysis: study the artifact during use to evaluate its dynamic qualities (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
E_{t3}	Experimental	(i) Controlled experiment: study the artifact in a controlled environment to determine its qualities. (ii) Simulation: execute the artifact with artificial data (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
E_{t4}	Testing	(i) Functional test (black box): implement the artifact interfaces to discover potential failures and identify defects. (ii) Structural test (white box): perform coverage tests of some metrics for implementing the artifact (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
E_{t5}	Descriptive	(i) Informed argument: use the information of knowledge bases (e.g., relevant research) to construct a convincing argument about the utility of the artifact. (ii) Scenarios: construct detailed scenarios for the artifact to demonstrate its utility (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
M_i	Methods:		
M_1	Product line development with customer interaction	Finds a variety of close to optimal solutions to compound the product line (Márkus and Váncza, 1998).	Open
M_2	Design for variety (DFV)	Presents two indices to measure a product's architecture. The first index is the Generational Variety Index (GVI), a measure for redesign effort required for future designs of the product. The second index is the Coupling Index (CI), a measure of the coupling among the product components (Martin and Ishii, 2002).	Categorical
M_3	Product family design using physical programming	Employs the Physical Programming method, enabling designers to formulate the product family optimization problem in terms of physically meaningful terms and parameters (Messac, Martinez and Simpson, 2002).	Open
M_4	Product platform concept exploration method (PPCEM)	Discuss how the strategic incorporation of product platforms into the design process can leverage the design effort of individually customized products (Farrell and Simpson, 2003).	Open
M_5	Prescribing the content and timing of product upgrades	Prescribes the content and timing of upgrades to maximize total profit over the life cycle of the product family (Wilhelm, Damodaran and Li, 2003).	Open
M_6	Integrated modular product design	Consists of an integrated method that includes additional tools and stages for complete modular architecture design. The borders of the modular design process are expanded by adding strategic issues, appropriateness to modularity, the degree of modularity and modularity strategies (Asan, Polat and Serdar, 2004).	Categorical
M_7	Structural component-based product family design	Creates product family variants for different market requests through a structural graph representing the design priorities and constraints (Hsiao and Liu, 2005).	Open

(continued)

Table B2. (continued).

Id.	Codes	Definition / Function	Type
M_8	Customer value analysis	Customize existing product designs to meet individual customers' needs (Du, Jiao and Tseng, 2006).	Open
M_9	Mapping product design specification for mass customization	Generates product design specifications from customer functional requirements with a mass customization perspective (Krishnapillai and Zeid, 2006).	Open
M_{10}	Designing a family of development-intensive products	Models the decision-making on the product line design by taking into account the quality degradation and the development costs (Krishnan and Zhu, 2006).	Open
M_{11}	Functional modeling of modular product family design	Supports the identification of both shared and individual functional modules across a family of products (Zhang, Tor and Britton, 2006).	Categorical
M_{12}	Market segmentation for product family positioning based on fuzzy clustering	Proposes a fuzzy clustering-based market segmentation approach (Zhang, Jiao and Ma, 2007).	Open
M_{13}	Platform leveraging in a market segmentation grid for an existing product line	Optimizes a component platform portfolio, so that market segment grid platform leveraging is maximized (Farrell and Simpson, 2008).	Open
M_{14}	Dynamic approach to product architecture optimisation	Determines the optimal product architecture configuration in the multiproduct hierarchy by directly incorporating what customers want in the design and formulation of a family of products (Tucker and Kim, 2008).	Categorical
M_{15}	Optimal platform investment for product family design	Suggests the optimal initial investment in the platform, the commonality level between variants, and the number of variants to be produced in order to maximize market coverage using both analytical and simulation techniques (Zacharias and Yassine, 2008).	Categorical
M_{16}	Integration of marketing research techniques into house of quality and product family design	Integrates marketing research techniques into the house of quality and product family design (Kazemzadeh *et al.*, 2009).	Open
M_{17}	Evolutionary product line design balancing customer needs and product commonality	Develops a multi-objective optimization model to balance the diverging forces between marketing and engineering in evolutionary product line design (Chen, Jiao and Tseng, 2009).	Open
M_{18}	Market-driven product family design (MPFD)	Examines the impact of increasing the variety in the product offerings across different market segments and explore the cost-savings associated with commonality decisions (Kumar, Chen and Simpson, 2009).	Open
M_{19}	Customer-oriented optimal configuration of product scheme based on Pareto genetic algorithm	Uses a Pareto genetic algorithm for configuration optimization through which optimal solutions for customer requirements can be obtained (Yifei *et al.*, 2015).	Open
M_{20}	Predictive data-driven product family design (PDPFD)	Determines the optimal product family architectures with customer preference data (Ma and Kim, 2016).	Open
M_{21}	Coordinated optimisation of platform-driven product line planning by bilevel programming	Uses a bilevel mixed 0–1 nonlinear programming model to formulate coordinated optimisation for platform-driven product line planning (Miao *et al.*, 2017).	Open
T_i	Techniques:		
T_1	Mathematical modeling	Constructs a mathematical model that represents the essence of the problem to be solved (Hilier and Lieberman, 2015).	Categorical
T_2	Agent-based models (ABM)	Simulates the actions and interactions of autonomous agents in assessing their effects on the system as a whole (Rai and Allada, 2003).	Categorical
T_3	Conjoint analysis	Determines what combination of attributes is most influential on customers' decision making (Chen, Hoyle and Wassenaar, 2013).	Categorical
T_4	Aggregate project plan	Classify projects based on the number of resources they consume and on how they will contribute to the company's product line (Wheelwright and Clark, 1992).	Categorical
T_5	New product development map	Presents the evolution of current product lines in a summarised yet strikingly clear way so that all functional areas in the organization can respond to a common vision (Wheelwright and Sasser, 1989).	Categorical
T_6	Design matrix (DM)	Represent the relationships between two domains, for example, between functions and physical design parameters (Suh, 2001).	Categorical
T_7	Generational variety index (GVI)	Indicates the amount of redesign required for a component to meet the future market requirements (Martin and Ishii, 2002).	Categorical
T_8	Design structure matrix (DSM)	Represents the relationships among elements of the same domain, for example, between components (Browning, 2001).	Categorical

(continued)

Table B2. (continued).

Id.	Codes	Definition / Function	Type
T_9	Coupling index (CI)	Indicates the strength of coupling among the components in a product (Martin and Ishii, 2002).	Categorical
T_{10}	Market segmentation grid (MSG)	Articulates leveraging platform strategies in a given market (Kumar, Chen and Simpson, 2009).	Categorical
T_{11}	Physical programming	Addresses the inherent multiobjective nature of design problems, where multiple conflicting objectives govern the search for the best solution (Messac, Martinez and Simpson, 2002).	Open
T_{12}	Generalized reduced gradient (GRG) algorithm	Finds optimized solutions for nonlinear problems (Zacharias and Yassine, 2008).	Categorical
T_{13}	Simulated annealing	Approximates global optimization in an ample search space for an optimization problem (Agard and Bassetto, 2013).	Categorical
T_{14}	Branch and bound algorithm	Finds an optimal solution for mixed integer nonlinear programming problems with discrete and continuous variables (Tucker and Kim, 2008).	Categorical
T_{15}	Survey	Provides statistical descriptions of people by asking questions, usually of a sample (Forza, 2002).	Categorical
T_{16}	Modularity degree matrix	Evaluates the willingness to the modularity of product architectures based on the measures of the degree of modularity and the appropriateness for modularity (Asan, Polat and Serdar, 2004).	Categorical
T_{17}	Qualitative interviews	Derives meaning through interpretations, not necessarily 'facts' from participant talk (Malhotra and Birks, 2007).	Categorical
T_{18}	Function structure / diagram	Graphically represents a functional model where its overall function is represented by a collection of sub-functions connected by the flows on which they operate (Stone and Wood, 2000).	Categorical
T_{19}	Dominant flow heuristic	Defines a module based on a flow that passes through a set of sub-functions (Stone, Wood and Crawford, 2000).	Categorical
T_{20}	Branching flow heuristic	Defines a module based on branches of a parallel function chain (Stone, Wood and Crawford, 2000).	Categorical
T_{21}	Conversion-transmission heuristic	Defines a module based on a conversion sub-function or a conversion transmission pair or chain of sub-functions (Stone, Wood and Crawford, 2000).	Categorical
T_{22}	Cluster identification algorithm	Finds optimal machine cells and part families provided that the machine-part incidence matrix has the diagonal block structure embedded to solve standard group technology problems (Kusiak and Chow, 1987).	Categorical
T_{23}	Design for "X"	Provides system-level design guidelines for product development (Pahl *et al.*, 2007).	Categorical
T_{24}	Cross-impact systems and matrices (SMIC)	Describes the potential modes of interaction between a given set of variables and the assessment of the strength of these interactions (Asan, Polat and Serdar, 2004).	Categorical
T_{25}	Objective matrix	Evaluates different types or dimensions of performance measures altogether by composing them into a single composite measure (Asan, Polat and Serdar, 2004).	Categorical
T_{26}	Hierarchy graph	Illustrates the hierarchical relationships among the elements retrieved from a decomposed DSM (Hsiao *et al.*, 2013).	Categorical
T_{27}	Quality function deployment (QFD)	Translates customer requirements into technical requirements for each stage of product development and production (Chan and Wu, 2002).	Categorical
T_{28}	Focus group	Consists of a gathering of deliberately selected people who participate in a planned discussion intended to elicit consumer perceptions about a topic or area of interest in an environment that is non-threatening and receptive (Malhotra and Birks, 2007).	Categorical
T_{29}	Reverse engineering	Deconstructs an object to reveal its design, architecture, or extract knowledge from it (Thevenot and Simpson, 2007).	Open
T_{30}	Benchmarking	Measures the performance of a product against those considered to be the best (Thevenot and Simpson, 2007).	Open
T_{31}	Fractional factorial design	Provides the smallest number of runs for which k factors can be studied in a complete factorial design (Montgomery and Runger, 2011).	Open
T_{32}	Multiple regression	Models the relationship between multiple regressors or predictor variables (Montgomery and Runger, 2011).	Open
T_{33}	ANOVA	Analyses the differences among group means in a sample (Montgomery and Runger, 2011).	Open
T_{34}	Descriptive statistics	Describes the measures of central tendency and variability of a given data set (Montgomery and Runger, 2011).	Open
T_{35}	Modularity matrix	Represents the functional outputs of modules for each product variant (Dahmus, Gonzalez-Zugasti and Otto, 2001).	Categorical
T_{36}	Fuzzy clustering means (FCM)	Assigns data points to clusters allowing each data point to belong to multiple clusters with varying degrees of membership (Bezdek, 1981).	Categorical
T_{37}	Analytical hierarchy process (AHP)	Compares alternatives through a scale of absolute judgments that represents, how much more, one element dominates another concerning a given attribute (Saaty, 2008).	Categorical
T_{38}	Naïve Bayesian Model	Builds a predictive model based on a fraction of customer survey data used to train the computer learning model (Tucker and Kim, 2008).	Categorical
T_{39}	Data to Knowledge (D2K)	Classifies the survey results and maps the data into one of several predefined classes (Tucker and Kim, 2008).	Categorical
T_{40}	Gross margin	Measurement of how effectively the company turns its revenue into profit (Cox and Schleier, 2010).	Categorical
T_{41}	K-means clustering	Classifies a given data set through a certain number of clusters fixed apriori (Zhu *et al.*, 2010).	Categorical
T_{42}	Ward's hierarchical clustering	Merges attributes into clusters based on the residual error within the differences of the instance attribute from those of another instance or group (Kazemzadeh *et al.*, 2009).	Open

(continued)

Table B2. (continued).

Id.	Codes	Definition / Function	Type
T_{43}	Direct observation	Collects and analyzes information obtained directly or indirectly by watching and observing others in natural or planned environments (Kazemzadeh *et al.*, 2009).	Open
T_{44}	Part-worth utility	Measures how much each attribute and level influence the customer's decision making (Du, Jiao and Tseng, 2006).	Open
T_{45}	Commonality percentage index (CPI)	Measures how well the product family design utilizes common technical requirements (Kazemzadeh *et al.*, 2009).	Open
T_{46}	Cost reduction index (CRI)	Accounts for the savings in production cost when the company decides in favor of meeting customized technical requirements for customers in a particular segment rather than general technical requirements (Kazemzadeh *et al.*, 2009).	Open
T_{47}	Satisfaction percentage index (SPI)	Measures customer satisfaction (Kazemzadeh *et al.*, 2009).	Open
T_{48}	Discrete choice analysis	Estimates market demand based on customers' purchasing decisions (Chen, Jiao and Tseng, 2009).	Open
T_{49}	Internal product line commonality	Consists of the inverse of the average 1-norm distance among the constituent products of a product line (Chen, Jiao and Tseng, 2009).	Open
T_{50}	External product line commonality	Consists of the inverse of the average 1-norm distance between the constituent products of two generations of a product line (Chen, Jiao and Tseng, 2009).	Open
T_{51}	Genetic algorithm (GA)	Finds optimized solutions to search problems based on the mechanism of natural selection and natural genetics (Meng, Jiang and Huang, 2007).	Categorical
T_{52}	Nested logit model (NL)	Expresses the choice-behavior of individual customers and can be used whenever some choice alternatives are similar to others (Chen, Hoyle and Wassenaar, 2013).	Open
T_{53}	Pareto genetic algorithm	Locates and maintain a front of non-dominated solutions, ideally a set of Pareto optimal solutions, by using an evolutionary process to explore the search space (Jiao, 2012).	Open
T_{54}	Exponential smoothing model	Smoothes time series data using the exponential window function (Ma and Kim, 2016).	Open
T_{55}	Expectation maximization clustering (EM)	Assumes the data set can be modeled as a linear combination of multivariate normal distributions and finds the distribution parameters that maximize a model quality measure (Abbas, 2008).	Open
T_{56}	Multinomial logit (MNL)	Predicts the probabilities of the different possible outcomes of a categorically distributed dependent variable, given a set of independent variables (Chen, Hoyle and Wassenaar, 2013).	Categorical

Table B3. Structured classes of design problems.

<table>
<tr><th rowspan="2">Classes of design problems</th><th rowspan="2">Design problems</th><th colspan="2">Artifacts</th><th rowspan="2">Evaluation approach</th><th rowspan="2">Product classification</th><th rowspan="2">Primary studies</th></tr>
<tr><th>Techniques</th><th>Methods</th></tr>
<tr><td rowspan="48">Cp_1</td><td rowspan="9">$Pb_{1.1}$</td><td>T_4</td><td rowspan="2">M_2</td><td rowspan="3">Et_5</td><td rowspan="2">Pt_1</td><td rowspan="2">R_2</td></tr>
<tr><td>T_5</td></tr>
<tr><td rowspan="2">T_{10}</td><td>M_3</td><td rowspan="2">Pt_2</td><td>R_3</td></tr>
<tr><td>M_4</td><td rowspan="2">Et_1</td><td>R_4</td></tr>
<tr><td>T_{12}</td><td>M_{15}</td><td rowspan="5">Pt_1</td><td>R_{15}</td></tr>
<tr><td>T_{14}</td><td>M_5</td><td>Et_3, Et_5</td><td>R_5</td></tr>
<tr><td>T_{15}</td><td rowspan="2">M_6</td><td rowspan="3">Et_1</td><td rowspan="2">R_6</td></tr>
<tr><td>T_{16}</td></tr>
<tr><td>N/A</td><td>M_7</td><td>R_7</td></tr>
<tr><td rowspan="12">$Pb_{1.2}$</td><td rowspan="4">T_{10}</td><td>M_3</td><td>Et_5</td><td rowspan="4">Pt_2</td><td>R_3</td></tr>
<tr><td>M_4</td><td rowspan="9">Et_1</td><td>R_4</td></tr>
<tr><td>M_{13}</td><td>R_{13}</td></tr>
<tr><td>M_{18}</td><td>R_{18}</td></tr>
<tr><td>T_{33}</td><td>M_{16}</td><td>Pt_1</td><td>R_{16}</td></tr>
<tr><td>T_{36}</td><td rowspan="2">M_{12}</td><td rowspan="2">Pt_2</td><td rowspan="2">R_{12}</td></tr>
<tr><td>T_{37}</td></tr>
<tr><td>T_{39}</td><td>M_{14}</td><td rowspan="3">Pt_1</td><td>R_{14}</td></tr>
<tr><td>T_{41}</td><td rowspan="2">M_{16}</td><td rowspan="2">R_{16}</td></tr>
<tr><td>T_{42}</td></tr>
<tr><td rowspan="2">N/A</td><td>M_1</td><td>Et_5</td><td>Pt3</td><td>R_1</td></tr>
<tr><td>M_7</td><td>Et_1</td><td>Pt_1</td><td>R_7</td></tr>
<tr><td rowspan="21">$Pb_{1.3}$</td><td rowspan="6">T_1</td><td>M_1</td><td>Et_5</td><td>Pt3</td><td>R_1</td></tr>
<tr><td>M_5</td><td>Et_3, Et_5</td><td>Pt_1</td><td>R_5</td></tr>
<tr><td>M_{10}</td><td rowspan="3">Et_1</td><td rowspan="3">Pt_2</td><td>R_{10}</td></tr>
<tr><td>M_{13}</td><td>R_{13}</td></tr>
<tr><td>M_{18}</td><td>R_{18}</td></tr>
<tr><td rowspan="2">M_{21}</td><td rowspan="5">Et_5</td><td rowspan="2">Pt_1</td><td rowspan="2">R_{21}</td></tr>
<tr><td>T_3</td></tr>
<tr><td>T_{32}</td><td rowspan="2">M_{20}</td><td rowspan="2">Pt_2</td><td rowspan="2">R_{20}</td></tr>
<tr><td>T_{34}</td></tr>
<tr><td rowspan="6">T_{40}</td><td>M_1</td><td>Pt3</td><td>R_1</td></tr>
<tr><td>M_5</td><td>Et_3, Et_5</td><td>Pt_1</td><td>R_5</td></tr>
<tr><td>M_{10}</td><td rowspan="2">Et_1</td><td rowspan="3">Pt_2</td><td>R_{10}</td></tr>
<tr><td>M_{18}</td><td>R_{18}</td></tr>
<tr><td>M_{20}</td><td rowspan="2">Et_5</td><td>R_{20}</td></tr>
<tr><td>M_{21}</td><td rowspan="4">Pt_1</td><td>R_{21}</td></tr>
<tr><td>T_{48}</td><td rowspan="3">M_{17}</td><td rowspan="4">Et_1</td><td rowspan="3">R_{17}</td></tr>
<tr><td>T_{49}</td></tr>
<tr><td>T_{50}</td></tr>
<tr><td>T_{52}</td><td>M_{18}</td><td rowspan="2">Pt_2</td><td>R_{18}</td></tr>
<tr><td>T_{54}</td><td>M_{20}</td><td rowspan="4">Et_5</td><td>R_{20}</td></tr>
<tr><td>T_{56}</td><td>M_{21}</td><td>Pt_1</td><td>R_{21}</td></tr>
<tr><td rowspan="6">$Pb_{1.4}$</td><td>T_2</td><td>M_1</td><td>Pt3</td><td>R_1</td></tr>
<tr><td>T_{12}</td><td>M_{20}</td><td>Pt_2</td><td>R_{20}</td></tr>
<tr><td rowspan="2">T_{51}</td><td>M_{17}</td><td>Et_1</td><td rowspan="2">Pt_1</td><td>R_{17}</td></tr>
<tr><td>M_{21}</td><td>Et_5</td><td>R_{21}</td></tr>
<tr><td rowspan="2">N/A</td><td>M_{13}</td><td rowspan="2">Et_1</td><td rowspan="2">Pt_2</td><td>R_{13}</td></tr>
<tr><td>M_{18}</td><td>R_{18}</td></tr>
</table>

(continued)

Table B3. (continued).

Classes of design problems	Design problems	Artifacts: Techniques	Artifacts: Methods	Evaluation approach	Product classification	Primary studies
Cp_2	$Pb_{2.1}$	T_{15}	M_6	Et_1	Pt_1	R_6
			M_8		Pt_2	R_8
			M_{14}		Pt_1	R_{14}
			M_{16}			R_{16}
		T_{17}	M_6			R_6
			M_8		Pt_2	R_8
			M_{16}		Pt_1	R_{16}
		T_{28}	M_8		Pt_2	R_8
		T_{43}	M_{16}		Pt_1	R_{16}
	$Pb_{2.2}$	T_3	M_8		Pt_2	R_8
			M_{16}		Pt_1	R_{16}
		T_{15}	M_6			R_6
			M_8		Pt_2	R_8
			M_{16}		Pt_1	R_{16}
		T_{27}	M_7			R_7
		T_{31}	M_8		Pt_2	R_8
			M_{16}		Pt_1	R_{16}
		T_{32}	M_8		Pt_2	R_8
		T_{33}				
		T_{44}				
			M_{16}		Pt_1	R_{16}
	$Pb_{2.3}$	T_{18}	M_6			R_6
			M_{11}	Et_5	Pt_3	R_{11}
			M_{15}	Et_1	Pt_1	R_{15}
		T_{27}	M_2	Et_5		R_2
			M_7	Et_1		R_7
			M_{16}			R_{16}
		T_{29}	M_8		Pt_2	R_8
		T_{30}				
	$Pb_{2.4}$	T_3	M_2	Et_5	Pt_1	R_2
		T_{27}	M_9		Pt_2	R_9
			M_{15}	Et_1	Pt_1	R_{15}
		T_{41}	M_{20}	Et_5	Pt_2	R_{20}
		N/A	M_3			R_3
			M_4	Et_1		R_4
	$Pb_{2.5}$	T_{18}	M_6		Pt_1	R_6
			M_{11}	Et_5	Pt_3	R_{11}
			M_{15}	Et_1	Pt_1	R_{15}
		T_{27}	M_{16}			R_{16}
	$Pb_{2.6}$	T_{18}	M_6			R_6

(continued)

Table B3. (continued).

<table>
<tr><th rowspan="2">Classes of design problems</th><th rowspan="2">Design problems</th><th colspan="2">Artifacts</th><th rowspan="2">Evaluation approach</th><th rowspan="2">Product classification</th><th rowspan="2">Primary studies</th></tr>
<tr><th>Techniques</th><th>Methods</th></tr>
<tr><td rowspan="47">Cp_3</td><td rowspan="8">$Pb_{3.1}$</td><td rowspan="4">T_1</td><td>M_4</td><td rowspan="3">Et_1</td><td>Pt_2</td><td>R_4</td></tr>
<tr><td>M_8</td><td rowspan="2">Pt_2</td><td>R_8</td></tr>
<tr><td>M_{18}</td><td>R_{18}</td></tr>
<tr><td>M_{21}</td><td rowspan="6">Et_5</td><td rowspan="3">Pt_1</td><td>R_{21}</td></tr>
<tr><td>T_7</td><td rowspan="2">M_2</td><td rowspan="2">R_2</td></tr>
<tr><td>T_9</td></tr>
<tr><td>T_{11}</td><td>M_3</td><td rowspan="7">Pt_2</td><td>R_3</td></tr>
<tr><td>T_{34}</td><td>M_9</td><td>R_9</td></tr>
<tr><td rowspan="3">$Pb_{3.2}$</td><td rowspan="3">N/A</td><td>M_3</td><td>R_3</td></tr>
<tr><td>M_4</td><td rowspan="2">Et_1</td><td>R_4</td></tr>
<tr><td>M_8</td><td>R_8</td></tr>
<tr><td rowspan="2">$Pb_{3.3}$</td><td>T_{12}</td><td>M_{20}</td><td rowspan="4">Et_5</td><td>R_{20}</td></tr>
<tr><td>N/A</td><td>M_3</td><td>R_3</td></tr>
<tr><td rowspan="4">$Pb_{3.4}$</td><td>T_6</td><td>M_2</td><td rowspan="2">Pt_1</td><td>R_2</td></tr>
<tr><td rowspan="2">T_{27}</td><td>M_7</td><td>Et_1</td><td>R_7</td></tr>
<tr><td>M_9</td><td rowspan="2">Et_5</td><td>Pt_2</td><td>R_9</td></tr>
<tr><td>T_{35}</td><td>M_{11}</td><td>Pt_3</td><td>R_{11}</td></tr>
<tr><td rowspan="9">$Pb_{3.5}$</td><td rowspan="3">T_{19}</td><td>M_6</td><td>Et_1</td><td>Pt_1</td><td>R_6</td></tr>
<tr><td>M_{11}</td><td>Et_5</td><td>Pt_3</td><td>R_{11}</td></tr>
<tr><td>M_{15}</td><td rowspan="2">Et_1</td><td rowspan="2">Pt_1</td><td>R_{15}</td></tr>
<tr><td rowspan="3">T_{20}</td><td>M_6</td><td>R_6</td></tr>
<tr><td>M_{11}</td><td>Et_5</td><td>Pt_3</td><td>R_{11}</td></tr>
<tr><td>M_{15}</td><td rowspan="2">Et_1</td><td rowspan="2">Pt_1</td><td>R_{15}</td></tr>
<tr><td rowspan="3">T_{21}</td><td>M_6</td><td>R_6</td></tr>
<tr><td>M_{11}</td><td>Et_5</td><td>Pt_3</td><td>R_{11}</td></tr>
<tr><td>M_{15}</td><td rowspan="2">Et_1</td><td rowspan="3">Pt_1</td><td>R_{15}</td></tr>
<tr><td rowspan="2">$Pb_{3.6}$</td><td>T_{23}</td><td>M_6</td><td>R_6</td></tr>
<tr><td>N/A</td><td>M_2</td><td rowspan="3">Et_5</td><td>R_2</td></tr>
<tr><td rowspan="3">$Pb_{3.7}$</td><td>T_6</td><td>M_9</td><td>Pt_2</td><td>R_9</td></tr>
<tr><td rowspan="2">T_8</td><td>M_2</td><td rowspan="4">Pt_1</td><td>R_2</td></tr>
<tr><td>M_7</td><td rowspan="3">Et_1</td><td>R_7</td></tr>
<tr><td rowspan="3">$Pb_{3.8}$</td><td>T_{22}</td><td>M_6</td><td>R_6</td></tr>
<tr><td rowspan="2">N/A</td><td>M_7</td><td>R_7</td></tr>
<tr><td>M_9</td><td rowspan="2">Et_5</td><td rowspan="4">Pt_2</td><td>R_9</td></tr>
<tr><td rowspan="7">$Pb_{3.9}$</td><td>T_{11}</td><td>M_3</td><td>R_3</td></tr>
<tr><td>T_{12}</td><td rowspan="2">M_4</td><td rowspan="2">Et_1</td><td rowspan="2">R_4</td></tr>
<tr><td>T_{13}</td></tr>
<tr><td>T_{51}</td><td>M_{21}</td><td rowspan="2">Et_5</td><td>Pt_1</td><td>R_{21}</td></tr>
<tr><td>T_{55}</td><td>M_{20}</td><td rowspan="3">Pt_2</td><td>R_{20}</td></tr>
<tr><td rowspan="2">N/A</td><td>M_8</td><td rowspan="14">Et_1</td><td>R_8</td></tr>
<tr><td>M_{18}</td><td>R_{18}</td></tr>
<tr><td>$Pb_{3.10}$</td><td>T_{26}</td><td>M_7</td><td rowspan="6">Pt_1</td><td>R_7</td></tr>
<tr><td rowspan="5">$Pb_{3.11}$</td><td>T_{24}</td><td rowspan="2">M_6</td><td rowspan="2">R_6</td></tr>
<tr><td>T_{25}</td></tr>
<tr><td>T_{45}</td><td rowspan="3">M_{16}</td><td rowspan="3">R_{16}</td></tr>
<tr><td>T_{46}</td></tr>
<tr><td>T_{47}</td></tr>
<tr><td rowspan="6">Cp_4</td><td rowspan="4">$Pb_{4.1}$</td><td>T_1</td><td rowspan="2">M_{19}</td><td rowspan="2">Pt_3</td><td rowspan="2">R_{19}</td></tr>
<tr><td>T_{37}</td></tr>
<tr><td>T_{38}</td><td rowspan="3">M_{14}</td><td rowspan="3">Pt_1</td><td rowspan="3">R_{14}</td></tr>
<tr><td>T_{40}</td></tr>
<tr><td rowspan="2">$Pb_{4.2}$</td><td>T_{14}</td></tr>
<tr><td>T_{53}</td><td>M_{19}</td><td>Pt_3</td><td>R_{19}</td></tr>
</table>

Table B4. Incidence matrix.

Primary studies	Design problems																						Classes of design problems			
	$S_{1.1}$	$S_{1.2}$	$S_{1.3}$	$S_{1.4}$	$S_{2.1}$	$S_{2.2}$	$S_{2.3}$	$S_{2.4}$	$S_{2.5}$	$S_{2.6}$	$S_{3.1}$	$S_{3.2}$	$S_{3.3}$	$S_{3.4}$	$S_{3.5}$	$S_{3.6}$	$S_{3.7}$	$S_{3.8}$	$S_{3.9}$	$S_{3.10}$	S3.$_{11}$	$S_{4.1}$	$S_{4.2}$	Cp_1	Cp_2	Cp_3	Cp_4
R_7	1	1				1	1							1			1	1		1				1	1	1	
R_3	1	1						1			1	1	1						1					1	1	1	
R_4	1	1						1			1	1							1					1	1	1	
R_6	1				1	1	1		1	1					1	1		1			1			1	1	1	
R_{15}	1						1	1	1						1									1	1	1	
R_2	1						1	1			1			1		1	1							1	1	1	
R_{16}		1			1	1	1		1												1			1	1	1	
R_{20}			1	1				1					1						1					1	1	1	
R_{14}		1			1																	1	1	1	1		1
R_{18}		1	1	1							1								1					1		1	
R_{21}			1	1							1								1					1		1	
R_1		1	1	1																				1			
R_5	1		1																					1			
R_{10}			1																					1			
R_{12}		1																						1			
R_{13}		1	1	1																				1			
R_{17}			1	1																				1			
R_8					1	1	1				1	1							1						1	1	
R_{11}							1		1					1	1										1	1	
R_9								1			1			1			1	1							1	1	
R_{19}																						1	1				1
Occurrence	7	9	8	6	4	4	7	6	4	1	7	3	2	4	3	7	3	3	6	1	2	2	2	17	12	13	2
Relative Frequency [1]	33.3%	42.9%	38.1%	28.6%	19.0%	19.0%	33.3%	28.6%	19.0%	4.8%	33.3%	14.3%	9.5%	19.0%	14.3%	33.3%	14.3%	14.3%	28.6%	4.8%	9.5%	9.5%	9.5%	81.0%	57.1%	61.9%	9.5%

1) Frequency divided by the total number of primary studies (21 studies).

APPENDIX C – ARTICLE 3

Table C1. Learning log.

Id.	Cycle	Entered by	DSR Step	Subject	Situation	Recommendations & Comments	Implemented in
L_1	1	L. Gauss	2. Systematic literature review (SLR).	Stones' heuristics are subjected to bias.	Stones' heuristics used to cluster the design problems into classes are subject to bias (Stone, Wood and Crawford, 2000).	The modularity index (MI), a measure to evaluate the quality of a clustering solution, has been added. In future research, MI can be used to optimize the clustering of design problems into classes (Jung and Simpson, 2017).	DSR methodology
L_2	1	A. Dresch	3. Awareness of the problem.	SLR is not enough.	SLR might not be the unique technique to support the awareness of the problem, otherwise, the result of this step might be subjected to academic bias.	The SLR has been complemented by qualitative interviews with scholars and practitioners.	DSR methodology
L_3	1	L. Gauss	3. Awareness of the problem.	Techniques influence the functional model.	It has been noted during the artifact construction that the selection of the techniques (Step 6.2) might influence the sub-functions compounding the functional model (Step 5.1). Although it appears in the research strategy, it was not considered in the research method.	It has been added a feedback flow coming from step 6.2 to 5.1.	DSR methodology
L_4	1	L. Gauss	5. Proposition of artifacts.	The adoption of discrete choice analysis changed the method's steps.	From the selection of Nested Logit technique to model the customers' choice probabilities in step $S_{1.3}$, emerged the need to include additional method's steps.	The steps $S_{2.7}$, $S_{2.8}$ and $S_{2.9}$ have been added.	MDM version 2
L_5	1	L. Gauss	5. Proposition of artifacts.	Discarded the modules' classification in Cp_3.	It has been noted that the classification of the modules does not alter the final modular product family structure.	The modules' classification step in Cp_3 has been eliminated from the MDM structure.	MDM version 2
L_6	1	L. Gauss	8. Evaluation of the artifacts.	Differentiating attributes (DA) dependent on market segments (Ms).	It has been identified that the differentiating attributes (DA) may vary depending on the market segment (Ms).	It has been considered the mapping between the differentiating attributes (DA) and market segments (Ms) in step $S_{2.2}$.	MDM version 2
L_7	1	L. Gauss	8. Evaluation of the artifacts.	Feedback from the definition of differentiating attributes (DA) to market segmentation.	It has been noted that the survey results in step $S_{2.2}$ could be used to refine the market segmentation in step $S_{1.3}$.	It has been added a feedback flow coming from step $S_{2.2}$ to $S_{1.3}$.	MDM version 2
L_8	1	L. Gauss	8. Evaluation of the artifacts.	Engineering attributes (E) as numerical, binary, and categorical variables.	It has been perceived that the engineering attributes (E) might be of three types: numerical, binary, and categorical.	No action has been taken.	MDM version 1
L_9	1	L. Gauss D. Lacerda	8. Evaluation of the artifacts.	Theoretical assumptions of multiple regression are not attended in contexts of low data availability.	It has been found multicollinearity problems regarding the regression coefficients. The reason for that lies in the ratio between the number of engineering attributes (E) and required competing alternatives (J) to properly adjust the coefficients.	The multiple regression technique has been substituted by the Analytic Hierarchy Process (AHP) (Saaty, 2008) for modeling the customers' choice probabilities in contexts of low data availability.	MDM version 4
L_{10}	1	L. Gauss D. Lacerda	8. Evaluation of the artifacts.	Lack of technique to estimate the market size (M_k).	Although the market size (M_k) consists of an important entity of the method, there were no techniques assigned to its estimation.	It has been added the following techniques in step $S_{1.1}$: Delphi (Dalkey, 1969), Three-point estimate (Premachandra, 2001), and Domain knowledge (Jiao and Tseng, 1999a).	MDM version 5
L_{11}	1	L. Gauss	8. Evaluation of the artifacts.	The threshold to help the decision-making on investment in the product family design.	It has been identified the lack of a threshold to help the decision-making on investment in the product family design.	The expected profit (V_e) has been added to that end. Therefore, if $V \geq V_e$, then it would be reasonable to invest in the product family design.	MDM version 2

(continued)

Table C1. (continued).

Id.	Cycle	Entered by	DSR Step	Subject	Situation	Recommendations & Comments	Implemented in
L_{12}	2/3	V. Leivas F. Lima	8. Evaluation of the artifacts.	Aesthetics requirements might prevent the adoption of modularity.	It is believed that the adoption of modularity might prevent product variety when aesthetics attributes are required.	It has been assumed as a limitation of the method, therefore no action has been taken.	MDM version 3
L_{13}	2	L. Gehlen	8. Evaluation of the artifacts.	Difficulty to modularize convenience goods.	It has been noted that convenience goods such as food and pet food might be difficult to modularize due to its integral architecture.	No action has been taken since it was already considered out of the MDM scope.	MDM version 2
L_{14}	2	N. Fagherazzi	8. Evaluation of the artifacts.	Open architecture with the possibility to include other techniques.	It has been perceived that, depending on the context, other techniques might emerge during the MDM life-cycle.	The open architecture of the technique has been formalized.	MDM version 6
L_{15}	2	T. Vargas	8. Evaluation of the artifacts.	Willingness to modularization.	There is no step to evaluate the willingness to modularization of a product family architecture.	It has been added a conditional decision after step $S_{3.4}$ leading to the end of the process if an integral product family architecture is identified.	MDM version 6
L_{16}	2	V. Lubke	8. Evaluation of the artifacts.	Low heterogeneity of a single market segment might not require the use of modularity.	It is believed that the low heterogeneity of customers' needs of a single market segment might not require the use of modularity to provide variety.	It has been emphasized in the method's delimitation that the multiple market segments can amplify the benefits of design modular product families.	MDM version 6
L_{17}	3	K. Hamdar	8. Evaluation of the artifacts.	Cultural barriers preventing the method's implementation.	Like in any other method, the cultural barriers and the resistance to change might prevent the implementation of the MDM in organizations.	It has been better defined the delimitation of MDM regarding the socio-technical aspects of an enterprise.	MDM version 5
L_{18}	3	C. Vieiro L. Quitzrau	8. Evaluation of the artifacts.	The possibility to design a new product family from the existing product structure.	It has been observed that the proposed method is oriented to design completely new product families. However, there are many companies where the products have been evolving for decades, therefore it would not be reasonable to discard this experience.	The MDM has been reorganized to redesign the existing families from a modular point of view as well as to design new modules, new families, and new generations of families.	MDM version 4
L_{19}	3	L. Quitzrau A. Marques	8. Evaluation of the artifacts.	The estimation of customers' desired attributes (A) in the future.	It has been observed that the proposed method is oriented to design product families from the existing customers' desired attributes (A). However, since the products are designed in the present to be launched in the future, the MDM should estimate how the customers' desired attributes in the future should be.	It is believed that from the techniques adopted by the MDM it is possible to capture future needs. However, to explicit that, it was added an external input to the step $S_{2.1}$ and its description has been improved in that direction.	MDM version 4
L_{20}	3	A. Franceschini	8. Evaluation of the artifacts.	Lack of techniques to define the technological trends to be adopted in the product family design.	It has been identified the lack of techniques to define the technological trends to be adopted in the product family design.	It has been added the technology roadmap (Phaal and Muller, 2009) in the step $S_{1.1}$.	MDM version 4
L_{21}	3	A. Marques	8. Evaluation of the artifacts.	Evaluation of the competing alternatives (J) life-cycle before considering them into the product family design.	It has been noted the possibility to include competing alternatives (J) at the end of the life-cycle. This situation might influence the development process to be based on obsolete engineering attribute values (E_v).	It has been included a piece of advice to choose competing alternatives (J) at the beginning of the life cycle in steps $S_{2.4}$ and $S_{3.5}$.	MDM version 6
L_{22}	4	M. Salerno T. Betts T. Kull J. Hsuan	8. Evaluation of the artifacts.	Unsuitability for small and some midsize enterprises.	It is believed that small and some midsize enterprises might not have the knowledge base and organizational structure to support the method's implementation.	It has been reinforced in the method's delimitation its suitability for large companies.	MDM version 6

(continued)

Table C1. (continued).

Id.	Cycle	Entered by	DSR Step	Subject	Situation	Recommendations & Comments	Implemented in
L_{23}	4	M. Salerno T. Betts J. Hsuan	8. Evaluation of the artifacts.	Unsuitability for engineering-to-order (ETO) enterprises.	Although MDM can be used to change the production strategy of an enterprise, it is believed that it is not suitable for companies where the orders are not typically repeated on a large scale, i.e. Engineering-to-order (ETO).	The method's delimitation has been updated excluding the ETOs companies from its boundaries.	MDM version 6
L_{24}	4	K. Otto R. Scalice R. Bagno	8. Evaluation of the artifacts.	The method's evaluation process.	It has been claimed that the results obtained through practical applications, in real or made-up cases, are more relevant than the experts' opinions.	This research has tried to mix the practical applications along with the experts' opinions to make the evaluation process more robust.	DSR methodology
L_{25}	4	R. Bagno	8. Evaluation of the artifacts.	Product design performed from a deterministic perspective for static scenarios.	It has been noted the assumption of dealing with product family design from a deterministic perspective and only considering static scenarios. This is not incorrect but needs to be clear in the delimitation of the method.	It has been assumed as a limitation of the MDM so far. Additionally, the method's delimitation has been updated to make it more explicit.	MDM version 6 and future research on product family design
L_{26}	5	L. Gauss	8. Evaluation of the artifacts.	Market segmentation concurrently performed with the strategic product family planning.	It has been perceived that the market segmentation $(S_{1.3})$ can be performed concurrently with the strategic product family planning $(S_{1.1})$.	No action has been taken.	MDM version 5
L_{27}	5	L. Gauss	8. Evaluation of the artifacts.	Random index (RI) for comparison matrices with more than 15 criteria in AHP.	It has been noted that the consistency ratio (CR) increases as the number of criteria grow. Besides that, the traditional random index (RI) proposed by Saaty (2008) does not support matrices with more than 15 criteria.	It has been adopted the consistency system proposed by Alonso and Lamata (2006).	MDM version 6
L_{28}	5	L. Gauss	8. Evaluation of the artifacts.	Unnecessary construction of choice sets in contexts of low data availability.	It has been perceived that in contexts of low data availability, the utilization of AHP in step $S_{2.7}$ does not require the construction of the choice set in step $S_{2.5}$.	It has been added a bypass from the step $S_{2.4}$ to $S_{2.6}$.	MDM version 6
L_{29}	5	L. Gauss	8. Evaluation of the artifacts.	Difficulty to obtain the engineering attribute values (E_v) of complex competing alternatives (J).	It has been identified the difficulty to obtain all engineering attribute values (E_v) of competing alternatives (J), mainly when they are complex, i.e. robotic palletizers. As a result, these missing values might penalize the choice probability of its corresponding alternatives.	To relax this penalization, for those competing alternatives (J) with missing values, the same engineering attribute values (E_v) of the variant configured by the mathematical model have been considered.	MDM version 6
L_{30}	5	L. Gauss	8. Evaluation of the artifacts.	Difficulty to obtain the price (P) of complex competing alternatives (J).	It has been identified the difficulty to obtain the price (P) of competing alternatives (J), mainly when they are complex, i.e. robotic palletizers.	In such cases where the price is not available, it has been suggested estimating the price thought the multiplication between a typical markup adopted in the market and the product's variable cost (C_v), i.e. $P_i = markup.C_{vi}$.	MDM version 6
L_{31}	5	L. Gauss	8. Evaluation of the artifacts.	The need for data scaling when calculating the utility function (W) from AHP.	When calculating the utility (W) from AHP, it has been noted that the difference in scale among the engineering attribute values (E_v) might influence the results. In other words, without data scaling the utility function becomes deeply dependent on those attributes of higher magnitude.	When using the AHP, it has been considered scaling the engineering attribute values (E_v) before calculating the utility (W) and the choice probability (P_r).	MDM version 6
L_{32}	5	L. Gauss	8. Evaluation of the artifacts.	Mathematical imponderability when using negative weights (w) obtained through the AHP to calculate the utility function (W).	It has been noted that some engineering attributes (E) might assume a negative direction $(-)$, i.e. "the lower the better". In such cases, using negative weights (w) retrieved from AHP to calculate the utility (W) might results in counterintuitive values.	The weights (w) have been kept positive, and the inverse of the engineering attributes values $(1/E_{vi})$ have been considered in the utility (W) calculation.	MDM version 6

(continued)

Table C1. (continued).

Id.	Cycle	Entered by	DSR Step	Subject	Situation	Recommendations & Comments	Implemented in
L_{33}		L. Gauss	8. Evaluation of the artifacts.	Sufficiency of the current definition of modular architecture.	It has been identified that there might exist different levels of intensity in the relationship among the engineering attributes (E) and design parameters (DP). This intensity might require new decomposition techniques as well as a new definition of what a modular architecture is.	To accomplish the level of intensity an additional decomposition technique has been added, the Cluster identification algorithm (Kusiak and Chow, 1987). The definition of modular architecture has been suggested as future research opportunities.	MDM version 6 and future research on modularity
L_{34}	5	L. Gauss	8. Evaluation of the artifacts.	Variable cost estimation (C_v) when historical data on product costing is not available.	It has been perceived that companies not always have historical data on product costing, an issue that might prevent the utilization of the reasoning behind the pragmatic approach to product costing (Jiao and Tseng, 1999b). This situation tends to aggravate when new generations of product families are under development.	The three-point estimate (Premachandra, 2001) and the request for quotation (Gümüş, 2014) techniques, have been added to the MDM framework.	MDM version 6
L_{35}	5	L. Gauss	8. Evaluation of the artifacts.	New design parameters (DP) might emerge from the conceptual layout's formulation.	It has been identified that new design parameters (DP) might emerge from the conceptual layout's formulation.	It has been added a feedback flow coming from step $S_{3.7}$ to $S_{3.1}$.	MDM version 6
L_{36}	5	L. Gauss	8. Evaluation of the artifacts.	Unnecessary combination of design parameter instances (DPI) to generate physical module candidates.	It has been noted that the systematic combination of design parameter instances (DPI) to generate physical module candidates, and then combining these module candidates into product family variants (PF_v) was unnecessary. That is because of the design parameter instance (DPI) can be directly combined to generate the product family variants (PF_v).	The step of combining the DPI into physical modules candidates ($S_{3.11}$) has been eliminated from the MDM structure.	MDM version 6
L_{37}	5	L. Gauss	8. Evaluation of the artifacts.	The weights (w) obtained through the AHP cannot be used to calculate the choice probability (P_r) in multinomial logit models.	It has been identified a mathematical imponderability when using the weights (w) obtained through the AHP to calculate the choice probability (P_r) in multinomial logit models. The problem is when the utility function (W) tends to 0, the resulting choice probability (P_r) does not tend to 0. It allows an extremely low utility variant (PF_v), or even a low utility competing alternative (J) having a choice probability higher than 0%.	When using the AHP, the utility has been assumed to be equals to the choice probability, i.e. $W = P_r$.	MDM version 6
L_{38}	5	L. Gauss	8. Evaluation of the artifacts.	The tendency of the configuration model to adjust the price as much as possible during the optimization process.	It has been noted that although the price (P) has a negative influence on the choice probability (Pr), the model presented the tendency to adjust it as high as possible to maximize the partial profit (V_{Ms}). That is because the marginal increase in partial profit derived from the increment in price is more substantial than the one retrieved from the augmentation in choice probability. In contexts of low data availability, where the weights (w) are estimated through the AHP, this undesired effect tends to aggravate as the number of customers' desired attributes (A) increases.	To overcome this limitation in contexts of low data availability, the alternative solution adopted was to consider the price as a parameter instead of a decision variable of the configuration model. In this sense, the price should be intentionally defined based on the strategy of product family positioning established at the step $S_{1.1}$.	MDM version 6
L_{39}	5	L. Gauss	8. Evaluation of the artifacts.	Synthesis of the corporate strategy after the market segmentation.	It has been identified that the results of the market segmentation should integrate the list of objective measures for product family development.	The activity of synthesizing the corporate strategy into objective measures for product family development has been relocated from step $S_{1.1}$ to $S_{1.2}$.	MDM version 6

(continued)

Table C1. (continued).

Id.	Cycle	Entered by	DSR Step	Subject	Situation	Recommendations & Comments	Implemented in
L_{40}	5	L. Gauss D. Lacerda	8. Evaluation of the artifacts.	Lack of practical guidance on artifact's evaluation.	It has been noted the lack of practical guidance on the levels of artifact evaluation. Although it has been tackled by FEDS (Venable, Pries-Heje and Baskerville, 2016) it still required more discussion in terms of evaluation dimensions.	The definition of the artifact's evaluation level has been suggested as future research directions on DSR.	Future research on DSR.
L_{41}	5	L. Gauss	8. Evaluation of the artifacts.	Different time exposure to artifact's evaluation.	It has been identified a decreasing tendency of the level of agreement (k_{free}) among the testing cycles 2, 3 and 4. One of the reasons might be time exposure for the artifact's evaluation that reduced from 12 hours in cycle 2, passing to 1 hour in cycle 3, achieving 15 min. in cycle 4.	Keep the same procedure and time exposure for the experts and scholars in the evaluation cycles.	Future research on DSR.

Table C2. Characterization of participants.

Id.	Respondents	Characterization
	Students:	
STD.01	Rafael Kreling	Process Engineering Manager at Docile (Brazil), with B.Sc. in Controls and Automation Engineering, and M.Sc. in Business Management.
STD.02	Rogério Marciniak	Engineering Manager at Visconti (Brazil), with B.Sc. in Electrical Engineering, and M.B.E. in Advanced Manufacturing.
STD.03	Ricardo Semler Barroso	Production Planning and Control Analyst at Metal Work (Brazil), with B.Sc. in Production Engineering, and M.B.E. in Advanced Manufacturing.
STD.04	Lúcio de Lima Gehlen	Sr. Electrical Design Engineer at Hercosul (Brazil), with B.Sc. in Controls and Automation Engineering, and M.B.E. in Advanced Manufacturing.
STD.05	Thiarles Silva de Vargas	Continuous Improvement Specialist at Paquetá (Brazil), with B.Sc. in Production Engineering, and M.B.E. in Advanced Manufacturing.
STD.06	Volker Lübke	Industrial Director at Gedore (Brazil), with B.Sc. in Mechanical Engineering, and M.B.E. in Advanced Manufacturing.
STD.07	Nelso Luis Fagherazzi	Executive Director at Infasul (Brazil), with B.Sc. in Mechanical Engineering, and M.B.E. in Advanced Manufacturing.
STD.08	Vítor Costa Leivas	Manufacturing Engineering Supervisor at Fimac (Brazil), with B.Sc. in Mechanical Engineering, and M.B.E. in Advanced Manufacturing.
	Experts:	
EXP.01	Luiz Eduardo Quitzrau	Product Engineering Supervisor at John Deere (Brazil), with B.Sc. in Mechanical and Electrical Engineering, and M.Sc. in Mechanical Engineering.
EXP.02	Carlos Frederico Viero	Former Engineering Director at Comil (Brazil), with B.Sc. in Mathematics, and Ph.D. in Production and Systems Engineering.
EXP.03	Paulo Azola	Product Development Manager at Embraer (Brazil), with B.Sc. in Mechanical Engineering, and Specialization in Aeronautical Engineering.
EXP.04	Alexandre Marques da Rosa	Product Engineering Manager at Ciber (Brazil), with B.Sc. in Mechanical Engineering, and Specialization in Health and Safety Engineering.
EXP.05	Georgia Forneck	Head of product information at Ciber (Brazil), with B.Sc. in Business Administration, and Specialization in Quality & Innovation.
EXP.06	Rafael Loose	Product Design Coordinator at Atlas (Brazil), with B.Sc. in Product Design, and M.B.A in Marketing.
EXP.07	Kayam Hamdar	Product Development Engineer at Electrolux (Brazil), with B.Sc. in Mechanical and Electrical Engineering.
EXP.08	Sarah Amin de Lima	Architecture Leader Air & Water Business at Whirlpool (Brazil), with B.Sc. in Mechanical Engineering, and M.B.A. in Project Management.
EXP.09	André Schwarz Franceschini	Head of Research and Development at Ciber (Brazil), with B.Sc. in Mechanical Engineering, and M.Sc. in Mechanical Engineering.
EXP.10	Flavio Lima	Vice-President of Quality & Total Customer Satisfaction America at Renault (Brazil), with B.Sc. in Mechanical Engineering.
	Scholars:	
SCH.01	Flavio Issao Kubota	Adjunct Professor of Mechanical Engineering at the Federal University of Parana - UFPR (Brazil), with Ph.D. in Production Engineering.
SCH.02	Mario Sergio Salerno	Full Professor of Production Engineering at the University of São Paulo - USP (Brazil), with Ph.D. in Production Engineering.
SCH.03	Teresa Betts	Associate Professor of Logistics and Supply Chain Management at Murray State University (USA), with Ph.D. in Production Operations Management.
SCH.04	Thomas Kull	Associate Professor of Supply Chain Management at Arizona State University (USA), with Ph.D. in Operations and Sourcing Management.
SCH.05	Juliana Hsuan	Associate Professor of Operations Management at Copenhagen Business School (Denmark), with Ph.D. in Operations Management.
SCH.06	Katja Hölttä-Otto	Associate Professor of Mechanical Engineering at Aalto University (Finland), with Ph.D. in Mechanical Engineering.
SCH.07	Régis Kovacs Scalice	Adjunct Professor of Mechanical Engineering at the Federal University of Santa Catarina - UFSC (Brazil), with Ph.D. in Mechanical Engineering.
SCH.08	Raoni Barros Bagno	Full Professor of Production Engineering at the Federal University of Minas Gerais - UFMG (Brazil), with Ph.D. in Mechanical Engineering.
SCH.09	Hoda ElMaraghy	Distinguished Professor of Mechanical Engineering at the University of Windsor (Canada), with Ph.D. in Mechanical Engineering.

Table C3. Mixed coding scheme.

Id.	Codes	Definition	Type
	Top terms:		
1.0	Pragmatic validity	The evidence that the design produces the desired results (van Aken, Chandrasekaran and Halman, 2016).	Categorical
2.0	Practical relevance	The contribution of design in addressing a significant field problem or exploiting a promising opportunity (van Aken, Chandrasekaran and Halman, 2016).	Categorical
	Constructs:		
1.1	External environment	The context in which the artifact can be used and its performance limits (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
1.2	Internal environment	The organization of internal mechanisms to achieve a particular goal in the external environment (Simon, 1962).	Categorical
1.3	Artifacts' evaluation	Set of procedures to attest if the artifact produces the desired outcomes (Dresch, Lacerda and Antunes Jr, 2015).	Categorical
2.1	General utility	The overall ability to address the problem under investigation.	Categorical
	Dimensions:		
1.1.1 (Q07)	Company size	The size of the company, i.e. small, midsize and large companies.	Categorical
1.1.2 (Q09)	Production strategy	The production strategy used for delivering goods, i.e. make-to-stock (MTS), assemble-to-order (ATO), make-to-order (MTO), and engineering to order (ETO) (Slack and Brandon-Jones, 2019).	Categorical
1.1.3 (Q11)	Product development phases	Phases compounding the product development process, i.e. planning, conceptual design, system-level design (Ulrich and Eppinger, 2012).	Categorical
1.1.4 (Q13)	Product type	Type of goods manufactured by an enterprise, i.e. consumer (durables), intermediate, and capital goods (OECD, 2008).	Categorical
1.1.5 (Q15)	Single market segment	A specific group of customers having similar characteristics (Kahn, 2012).	Categorical
1.1.6 (Q17)	Multiple market segments	Multiple groups of customers by which the market is segmented.	Categorical
1.2.1 (Q19)	Steps' sufficiency	The need for the method's steps to exist.	Categorical
1.2.2 (Q21)	Steps' execution order	Adequacy of the execution order of the method's steps.	Categorical
1.2.3 (Q23)	Adequacy of feedback flows	Adequacy of the feedback between the method's steps.	Categorical
1.2.4 (Q25)	Applicability of techniques	The ability of techniques to execute the method steps.	Categorical
1.2.5 (Q27)	Suitability of qualitative techniques	The reasoning of using qualitative techniques in contexts where the data are scarce or the cost to obtain it cannot be afforded by the company (low data availability).	Categorical
1.2.6 (Q29)	Suitability of quantitative techniques	The reasoning of using quantitative techniques in contexts where the data are available or the cost to obtain it can be afforded by the company (high data availability).	Categorical
1.2.7 (Q31)	Applicability of existing tools	The ability of existing tools (software) to operationalize the techniques.	Categorical
1.2.8 (Q33)	Missing steps	Steps that have not been considered in the method but should be.	Categorical
1.3.1 (NA)	Artifact's evaluation	The process of determining whether the artifact completely accomplish their function (Venable, Pries-Heje and Baskerville, 2016).	Categorical
2.1.1 (Q34)	Customers' choice modeling	The ability to mathematically model the customers' choice (Chen, Hoyle and Wassenaar, 2013).	Categorical
2.1.2 (Q36)	Market-driven variants	The capacity for generating product family variants that accomplish the customers' desired attributes.	Categorical
2.1.3 (Q38)	Balance of market needs and profitability	The ability to balance the accomplishment of market needs and the resulting profitability to achieve them.	Categorical
2.1.4 (Q40)	Product family economic potential	The ability to capture the economic potential of a product family.	Categorical
2.1.5 (Q42)	Trade-off between variety and cost	The capacity for providing product variety without sacrificing production efficiency (Simpson *et al.*, 2014).	Categorical
2.1.6 (Q44)	Utility	The ability to build a product family structure by balancing the fulfillment of market needs and its resulting profitability, as well as providing its economic potential to the decision-maker.	Categorical
	Moderating variables:		
MV.01	Aesthetics requirements	Products desired attributes related to aesthetics requirements, i.e. consumer durables.	Open
MV.02	Cultural barriers	Cultural barriers (resistance of change) that prevent the method's implementation.	Open
MV.03	Manufacturing under the customer's drawings	Products manufactured under the customer's drawings, i.e. intermediate goods.	Open
MV.04	Low heterogeneity	Low heterogeneity of a single segment might not require the use of modularity to provide variety.	Open
MV.05	Uncertainty of estimated data	The error of data estimated by domain knowledge.	Open
MV.06	Other existing techniques	Other techniques or heuristics that might be used to execute the steps of the method.	Open

(continued)

Table C3. (continued).

Id.	Codes	Definition	Type
	Moderating variables:		
MV.07	Convenience goods	Goods that are purchased frequently and with a minimum of effort, i.e. food.	Open
MV.08	Willingness to modularization	Step to evaluate the willingness of product modularity in product planning.	Open
MV.09	Method's complexity	The difficulty or cost of executing the method's steps and techniques.	Open
MV.10	Lack of a design supporting system	Lack of a design support system (software) encapsulating the method's steps and techniques.	Open
MV.11	Organizational immaturity	Lack of knowledge base and organizational structure to support the method's implementation. Usually found in small and midsize enterprises (SME).	Open
MV.12	Redundant steps	Different steps that have similar or the same function, i.e. functional and physical decomposition.	Open
MV.13	Future customers' needs	Products' desired attributes that will be required by customers in the future.	Open
MV.14	Customers' satisfaction feedback	Feedback evaluating the customers' satisfaction with the product family variants generated by the proposed method.	Open
MV.15	Bottom-up techniques	Techniques used to design new modules, variants, and families from the existing product structures.	Open
MV.16	Future customers' needs	Step or technique to predict future customers' needs.	Open
MV.17	Life-cycle of competing alternatives	Step to evaluate the life-cycle of competing alternatives to composing the choice set.	Open
MV.18	Modularization as a function of production volume	Step to evaluate if it is worth it to modularize the product family based on its production volume.	Open
MV.19	Regulatory standards	Step to deal with regulatory standards in product development.	Open
MV.20	Management of change	Step to deal with cultural issues preventing the implementation.	Open
MV.21	Strategic pricing definition	Step to strategically define the price of product family variants in the view of the trade-of between the gross margin and market share.	Open
MV.22	Configuration management	Step to manage the product family configuration structure.	Open
MV.23	Fixed costs	The assumption of not considering the fixed costs may adversely affect results.	Open
MV.24	Price of competing alternatives	Difficulty to obtain the price of competing alternatives.	Open
MV.25	Clarify the design strategy	Make the design strategy clear, i.e. top-down or bottom-up.	Open
MV.26	Uncertainty of cost estimation	The error associated with estimating costs at early design stages.	Open
MV.27	Technological trends	Step or technique to define the technological trends to be adopted in the product family design.	Open
MV.28	Complex products	The complexity of the good being designed in terms of the number of components and assemblies, i.e. smartphones, automobile modules (engine, gearbox, etc.).	Open
MV.29	Low scale orders	Orders that are not typically repeated on a large scale, i.e. Engineering-to-order (ETO).	Open
MV.30	Qualitative techniques	Techniques that capture the same value as the quantitative ones but with less time and effort requirements.	Open
MV.31	Modularity maturity level	Step to analyze the modularity maturity level of the company to define which subsequent steps of the method should be used.	Open
MV.32	Uncertainty of market size estimation	The error associated with estimating market size.	Open
MV.33	Practical application	The results obtained through practical applications are more relevant than the experts' opinions.	Open
MV.34	Static scenarios under a deterministic perspective	The assumption of dealing with product family design from a deterministic perspective and only considering static scenarios.	Open
MV.35	Divergent key performance indicators (KPI)	Disconnected performance measures leading departments of the company to opposite directions.	Open

APPENDIX D – ARTICLE 3

Market-Driven Modularity (MDM) – Online-Based Questionnaire

** Required*

Respondent Characterization:

Q01. Email address *

Q02. What is your full name? *

Q03. What is your education level? *

- ☐ Bachelor of Science (B.Sc.)
- ☐ Master of Science (M.Sc.)
- ☐ Doctor of Philosophy (Ph.D.)
- ☐ Other:

Q04. What is your professional/research area? *

Q05. How long have you worked/researched in this area? *

- ☐ Less than 5 years
- ☐ Between 5 and 10 years
- ☐ Between 10 and 15 years
- ☐ More than 15 years

Q06. Which type of product do you currently work with? (only required for practitioners)

- ☐ Consumer goods (durables), i.e. durable products that people buy for their use
- ☐ Intermediate goods, i.e. products used in the production of other goods
- ☐ Capital goods, i.e. equipment used to produce products or provide services
- ☐ Other:

External Environment of Usage:

Q07. Do you agree that the MDM method can be adopted by small, midsize, and large companies? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q08. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q09. Do you agree that the MDM method can be used by companies that operate under the MTS, ATO, MTO and ETO strategies? *

Make-to-stock (MTS), Assemble-to-order (ATO), Make-to-order (MTO), and Engineering-to-order (ETO).

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q10. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q11. Do you agree that the MDM method can be incorporated in the early stages of the product development process, such as planning, conceptual design, and system-level design? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q12. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q13. Do you agree that the MDM method can be used for designing consumer (durables), intermediate, and capital goods? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q14. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q15. Do you agree that the MDM method can be used to conceptually design a product family for a single segment? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q16. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q17. Do you agree that the MDM method can be used to conceptually design a product family for multiple segments? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q18. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Internal Functional Environment:

Q19. Do you agree that all steps of the MDM method are required? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q20. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q21. Do you agree that the execution order of the MDM steps is adequate? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q22. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q23. Do you agree that the feedback flows between the MDM steps are adequate? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q24. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q25. Do you agree that all steps of the MDM method can be performed by the suggested techniques? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q26. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q27. Do you agree with the proposition of the MDM method of using more qualitative techniques in contexts of low data availability? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q28. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q29. Do you agree with the proposition of the MDM method of using more quantitative techniques in contexts of high data availability? *

- ☐ Disagree
- ☐ Partially agree

- ☐ Totally agree

Q30. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q31. Do you agree that the techniques suggested by the MDM method can be performed by tools (software) available in organizations? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q32. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q33. Is there any step that has not been considered by the MDM method that in your opinion it should be? *

General Utility of the Method:

Q34. Do you agree that through the MDM method, it is possible to model customer preferences? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q35. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q36. Do you agree that through the MDM method, it is possible to generate product alternatives (variants) oriented to customer preferences? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q37. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q38. Do you agree that through the MDM method, it is possible to structure a product family from the selection of alternatives (variants) that balance the fulfillment of market needs and the resulting profitability to achieve them? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q39. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q40. Do you agree that through the MDM method, it is possible to assess the economic potential of a product family? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q41. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q.42 Do you agree that through the MDM method, it is possible to mitigate the trade-off between variety and cost? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q43. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

Q44. Do you agree that the MDM method is useful for organizations that develop, produce and market products? *

- ☐ Disagree
- ☐ Partially agree
- ☐ Totally agree

Q45. Regarding the previous question, in case of disagreement (1), or partial agreement (2), please inform the reasons?

APPENDIX E – ACCEPTANCE LETTER OF ARTICLE 1

Dear Mr. Gauss,

We are pleased to inform you that your manuscript, "Module-Based Product Family Design: Systematic Literature Review and Meta-Synthesis", has been accepted for publication in Journal of Intelligent Manufacturing.

You will receive an e-mail from Springer in due course with regard to the following items:

1. Offprints
2. Colour figures
3. Transfer of Copyright

Please remember to quote the manuscript number, JIMS-D-19-00710R1, whenever inquiring about your manuscript.

With best regards,

Andrew Kusiak
Editor-in-Chief

Reviewer #1: the corrections are suitable it can be accepted.

Reviewer #3: accept

Reviewer #5: It can be found that the authors have largely improved the paper and hence, the paper can be considered for publication.

As a result of the significant disruption that is being caused by the COVID-19 pandemic we are very aware that many researchers will have difficulty in meeting the timelines associated with our peer review process during normal times. Please do let us know if you need additional time. Our systems will continue to remind you of the original timelines but we intend to be highly flexible at this time.

This letter contains confidential information, is for your own use, and should not be forwarded to third parties.

Recipients of this email are registered users within the Editorial Manager database for this journal. We will keep your information on file to use in the process of submitting, evaluating and publishing a manuscript. For more information on how we use your personal details please see our privacy policy at https://www.springernature.com/production-privacy-policy. If you no longer wish to receive messages from this journal or you have questions regarding database management, please contact the Publication Office at the link below.

In compliance with data protection regulations, you may request that we remove your personal registration details at any time. (Use the following URL: https://www.editorialmanager.com/jims/login.asp?a=r). Please contact the publication office if you have any questions.