MODULARITY WITHIN A MATRIX OF FUNCTION AND FUNCTIONALITY (MFF)

Žiga Zadnik[1], Vanja Čok[1], Mirko Karakašić[2], Milan Kljajin[2], Jože Duhovnik[1],
(1) University of Ljubljana, SI (2) University of Osijek, CR

ABSTRACT
The objective of this paper is to present the concept of modularity in the development of a product by means of a descriptive matrix of function and functionality (MFF), based on the generative model and the criteria for describing products, functions and functionalities. The purpose of using the modularity of the descriptive MFF is to improve the initial design process, where only the most basic information is available, such as functions and functionalities, and to use the general functionality method, which is not quite possible with the morphological matrix. The modularity inside the MFF is based on the mutual relation between the function and the functionality, representing the data definition. In relation to the morphological matrix it is built and defined on the basis of a mathematical model and pre-set rules [1], not just on the basis of design intuition. This work represents a method for solving the modularity with regard to the shape and the function. This should facilitate the generation of the functional and shape structures of new and variant products. The developed MFF modularity model was implemented into a prototype web application and confirmed on a concrete product – the Active Lounge Chair 1.

Keywords: Design process, Functional matrix, Functional modelling, Functionality matrix, Modular design

1 INTRODUCTION
Market requirements are the basis for defining basic functional requirements, which in turn represent the initial information on a potential new product [2]. At the beginning of the design process, the functional requirements are usually unarranged, incomplete and sporadically presented, which makes it necessary to arrange, complement and expand them. The product structure can be presented as a functional structure, which at the same time is the basis for defining the shape (physical structure) of the product [3]. In [1, 4, 5] matrix models were developed and presented; they enable the generation of a functional structure of the product, described in matrices. This structure is then the basis for generating a product's shape structure.
In order to reduce the time required for arranging and improving the functional requirements, a modularity model of the matrix of function and functionality (MFF) was later developed. The basic morphological matrix [6] forms the basis for the development of the MFF modularity model. Using a small number of rows and columns, the model can yield a large number of solutions, which often makes them poor and unsuitable. The objective of MFF modularity development is to upgrade and update the deficiencies of the morphological matrix in the following areas: the use of a mathematically based model for creating the links between the function and the functionality, the possibility of the automatic suggestion of solutions, and use of sub-matrices with the modularity.
The MFF modularity model represents a tool that enables the designer to manage the design process faster and better, particularly in the initial concept phases. MFF is a synonym for the tabular presentation of the links between the functions and the functionalities. These functionalities are represented by technical systems [7] or shape models that in part, or in whole, fulfil the required functions. The matrix is used for the development of brand-new products as well as for the development of variant design.
Due to the fact that the MFF in itself connects functions and functionalities, the latter should be carefully and uniquely defined. In [8], the authors approach describing the functions by defining the terminology that is related to the names of the functions, while others describe the functions of technical systems by means of physical laws [9]. With a view to unique identification, rules were defined [1], by means of which the functions, functionalities and products are described. The reference

points for designing these rules are those presented in [10]. The functions are described by parameters, based on physical laws, which form the basis for the development of a mathematical model through which the connection with functionalities is established.
Today's market requires ever shorter development times for new products, which triggers the need for a modular architecture of products. Such a modular architecture makes it possible to combine one or several functions in the functional structure with one element that solves them [11]. Such an approach has several advantages; the main one being an increased number of product variants [12]. Erixon [13] developed the Modular Function Deployment method, using the Module Indication Matrix. The established rules (1) also include modularity rules in terms of the function and modularity with regard to the shape. These rules are at the same time implemented into the MFF model and presented on a concrete product, called the Active Lounge Chair 1 – (ALC 1).
Research and development activities within the product-development process have their own characteristic and distinctive features, dominated by unpredictability, creativity, mentality and abstraction. Due to these features it is difficult to thoroughly describe, develop and implement the design process in the initial phases of computer-tools development [3]. From this point of view, we have developed a computer web application, within which the MFF modularity model has been implemented. The application uses a central relational database that includes functionally described technical systems of various complexities, which in turn feeds all the functions, the corresponding parameters and the parameter values of all sorts of products.

2 MFF MODULARITY MODEL

2.1 Modularity with regard to function and shape

Modularity with regard to shape

Modularity with regard to shape is referred to as the appearance of a product in one or more variants (versions). According to the shape-modularity principle [2], products can be pooled into modular assemblies. They are checked in terms of the number of their functions that fulfil individual product variants, i.e., we are determining how many functions are fulfilled by a particular variant. In the case that a product variant includes all the functions of another variant, as well as the functions that other variants do not possess, that variant can replace the other one. A comparatively larger number of functions, fulfilled by a particular variant in comparison to another variant with a smaller number of functions, reflects a greater complexity of the variant. For the final confirmation of the variant with a larger number of functions it is later necessary to upgrade it and carry out an economic analysis, which has not been dealt with in this part because it is too extensive.

Table 1: Modularity with regard to shape

	FUNCTIONALITY			
FUNCTION	Variant 1	Variant 2	Variant 3	Variant 4
Function 1	X	X	X	X
Function 2	X	X	X	X
Function 3	X	X	X	
Function 4	X	X		
Function 5		X		
Function 6				

Table 1 shows that variant 2 entirely replaces variant 1, as it fulfils all the functions that are fulfilled by variant 1. Compared to variant 1, variant 2, in turn, solves some other – additional – functions that the adopted variant does not solve. It can be argued that variant 2 is more sophisticated, compared to variant 1, and that it solves more functions. A back-to-back examination of variants 3 and 4 also reveals that variant 3 entirely replaces variant 4.

Table 2: Modularity with regard to shape

	FUNCTIONALITY			
FUNCTION	Variant 1	Variant 2	Variant 3	Variant 4
Function 1		X	X	X
Function 2	X	X	X	X
Function 3	X	X	X	
Function 4	X		X	X
Function 5		X	X	
Function 6				

If individual variants (1 and 2, for example) have no common function, a new variant (variant 3) should be generated. This variant should fulfil all the functions not common to variants 1 and 2 (Table 2). An economic analysis has not been dealt with at this point because it is too extensive.

Modularity with regard to function:
Within the functional structure, more than one product can have identical or similar functions for performing the same or similar process. For such cases it is necessary to check the technical system overload by introducing modularity with regard to function. The modularity function consequently pools the functions for the larger number of variants. Function pooling represents the introduction of modularity according to the principle of function, where the use of various technical systems for identical functions is protected. For two products with identical functions, and in the case of non-established functions, it is vital to confirm their potential diversity. Only one product should be selected if no additional function has been confirmed for two functionally identical products.

Table 3: Modularity with regard to function:

	FUNCTIONALITY			
FUNCTION	Variant 1	Variant 2	Variant 3	Variant 4
Function 1	X	X	X	X
Function 2	X	X	X	X
Function 3	X	X		
Function 4	X		X	X
Function 5		X	X	
Function 6		X	X	

In Table 3, variants 1 and 2 solve functions 1, 2 and 3. These are two different technical systems that solve their common functions.

2.2 Modularity with regard to MFF

The concept of modularity in the development of a product by means of a descriptive MFF matrix is based on the generative model and criteria for describing products, functions and functionalities. The purpose of using the modularity by shape and function of the descriptive MFF matrix is to improve the initial design process where only the most basic information is available, such as functions and functionalities, and to use the general functionality method. Two aspects are taken into the account: fulfilling as many requirements as possible and fulfilling the requirements as effectively as possible. Both aspects are achieved through consistent combination of solution elements, bindings and modularization. Modularity within the MFF is based on the mutual relation between the function and functionality, which represents the data definition. The presentation is aimed at direct users, developers and researchers of technical systems and recognised technical processes. It is based on the connection between the recognised natural processes in nature and searching for comparable or satisfying technical processes at a certain level of knowledge development.
MFF represents a tabular presentation of the links between the functional requirements and the functionalities. Modularity can be devised if we know the key elements, such as the basic list of functional requirements and a list of functionalities, whose details will be dealt with later. The developed modularity model inside the MFF was created by expanding the matrix of functions and functionalities model, shown in (10), and by examining the functionality as it depends on various

functions. The basis for generating and arranging the MFF is the functional structure of a product, which at the same time represents the matrix input. When developing a new product, it is not possible to know and be familiar with a detailed functional structure right at the beginning. Such a structure can be obtained and built only from a rough functional structure, which is subject to constant changes during the design process, as shown in (5).

Within the MFF, functional requirements are introduced into the relation, on one side, and functionalities, on the other, as shown in Figure 1. The functional requirements represent the basic functions, while the functionalities are represented via technical systems. Both functions and technical systems can be either simple or more complex, which depends on the initial description of the individual systems and on the result of a rearrangement. In the theoretical part of the model, simplified and generalised marks for the functions and technical systems will be used for the purpose of showing the MFF modularity. The functions and functional requirement are represented by F*i* and are located in the first column, while individual technical systems are represented by TS*j* and are located in the subsequent columns.

In the MFF model, technical systems are marked with the general marks TS1, TS2,…,TS*j* *j*=1,…,*n*, while in the case of implementations and concrete examples the marks are of course replaced by the real names of technical systems. Columns with the names of technical systems, described in the matrix, are placed under the first functionality row. With each new technical system entry, a new column appears in the matrix. Its name matches the newly entered technical system. The same analogy also applies to the presupposed input functions that are to be solved. Hence, if a new modularity by shape or function, or a new functional requirement is added, it is analogously added to the matrix row. All the changes are dynamic and are continuously adapted and updated to the last updated MFF status.

The functions, defined in the MFF matrix, are described in the first column in the table, labelled Function. Each function corresponds to its row. In order to present the model in a simple way, the function names are marked with general marks, such as: F1, F2,…, F*i*; *i*=1,…,*m*, while in the concrete examples within an implementation, they are followed by concrete and real names.

The functions are defined on the basis of the required functional requirements. For systematics and modularity reasons, they are described in the relevant input lists. The MFF vision is that solving the matrix should gradually lead to defining more and more information for a particular functional requirement or function, and that it is solved at the end of the process with a suitable functionality. With a view to fulfilling the function, the differences between particular variants are arranged and the modularity is built.

FUNCTION	FUNCTIONALITY / SOLUTION			
	TS1 →	**← TS2 →**	**← TS3 →**	**← TS*j***
Functional requirement - F1 [Suggested solution] ↓	M **88** ↓ ↑ S **49**		M **30** ↓ ↑ A **40**	
Functional requirement – F2 [Suggested solution] ↑↓		S **85** ↓ ↑ S **45** ↓ ↑ B **23** ↓ ↑ B **15**		M **100** ↓ ↑ A **100** ↓ ↑ B **56** ↓ ↑ B **33**
Functional requirement – F3 ↑↓				
Functional requirement – F4 [Suggested solution] ↑↓	M **100**	M **88** ↓ ↑ S **49**		M **88** ↓ ↑ S **49**
Functional requirement - F*i* [Suggested solution] ↑	M **80**			

Figure 1: Modularity model of the matrix of functions and functionality

The result of arranging implies the modularity and/or the growth of the product's complexity. The fulfilment of variants for particular functions should always be ensured. In this case we are determining the number of functions that are fulfilled by a particular variant. This is how to confirm the gradation from the biggest to the smallest possible fulfilment of a function, and to establish a possible connection. We can look for modularity by shape or specifically determine the modularity by functions. This is achieved by providing a fulfilment for a particular variant in one of the adjacent variants. It establishes the modularity by functions, which makes it possible to use different technical systems for identical functions.

For a function that we do not know a lot about at the beginning of the design process, it is possible to determine a suitable solution by means of solving and describing, and to specify in more detail the type of function and all the corresponding parameters, winning parameters, intervals, etc. According to (1) the functions are divided into four different types of functions: main, supplementary, auxiliary and binding functions. They are all described by parameters, winning parameters and intervals.

The links between the functions and the functionalities that solve them are created by means of the so-called sub-matrices. These sub-matrices in the presented MFF model are coloured and highlighted in grey (Figure 1). Figure 1 shows several different sub-matrices, within which we will explain the solving, arranging, modularity and complexity of solving. As a rule, sub-matrices are not logically distributed at the beginning as their internal distribution is determined by how the design process develops and by the presupposed number of functions and functionalities. Parts of the matrix significantly deviating from the main diagonal are usually evidence that the determined function does not have an accurate basis, that it is specifically oriented and cannot be directly applied in a particular variant. This is a way to determine an unjustified description of function and to develop opportunities for further arranging and modularity. The key feature of the MFF is its arranging ability and the modularity of the sub-matrices, which makes it possible to arrange and sort the whole matrix during the design process. According to the given computer algorithm, the MFF presupposes a hierarchical order at the beginning of the process. It is based on the matching percentage, which can be re-arranged later. Attached is the possibility of two-level-row and one-level-column sorting, representing matrix dynamics within the design process.

The MFF model within the design process always includes all the sub-matrices that are of key importance for the development of further designing. Sub-matrices involving at least one possible solution on at least one function within the presupposed building block or functionality are full and display a partial and complete result for this sub-matrix, while the unsolved sub-matrices are not displayed. The result is displayed in the form of percentage values – numbers in a sub-matrix cell. The value is calculated on the basis of a verbal algorithm of the functional requirement’s crossed values and the function on the functionality. Each displayed value corresponds to the informative type of the current function. The function type is based on the description and is determined from the characterised character set M, S, A, B (initial letters for main, supplementary, auxiliary and binding functions).

The number of functions within the sub-matrix is analogous to the number of possible solutions in the functionality column. Results-wise, only the functions with a specific, possible solution are displayed. The functions that are not solving a given situation are not included in the display.

Besides the complete display of possible results between the functional requirements and the functionalities, the MFF model also includes an automatic suggestion for the end solution – the suggested solution in Figure 1 in each row of the first column. It is presupposed that a possible solution is the one that most closely corresponds to the given functional requirement. The end solution is selected on the basis of the individual percentage values; solutions’ values, making the end solution the one with the highest calculated percentage value. In the case that there are several solutions with identical percentage values, the higher solution is selected, i.e., the solution ranked higher according to the hierarchical type of function. For example, if there are several identical percentage solutions, the highest-ranking solution on the Main Function type would be selected. The automatically suggested solution is only a suggestion within the design process. In any case, the end and final decisions should always, and in all cases, be taken by the user, i.e., the designer.

The main function’s name for each system is shown with a mark (M1). According to the description rules, each building block can only have one function. It can happen that the MFF includes several technical systems with identical names of either the main, supplementary, auxiliary, or binding function. In the prototype model implementation, the real names of the existing status of the description are used for all the functions of the technical system. The supplementary functions’ names in the matrix model are shown in Figure 1 and marked with 4, S1; S2,…,Sk; k = 1,...,p. A technical system can have more than one supplementary function; it is even possible that a technical system has no supplementary function if it was not planned in the actual descriptions. The auxiliary functions are shown with marks: A1, A2,…,Ak; k = 1,…,p. A technical system can have one or more auxiliary functions. In analogy to the explanation above, it can happen that a technical system has no auxiliary function. The binding functions are shown with marks: B1, B2,…,B*k*; *k*=1,…,*p*. In contrast to the supplementary and auxiliary functions, a binding function without a single binding function is not possible because it would make the description incomplete and the technical system would not fulfil the appropriate criteria or rules of the pre-defined rules on describing functions, functionalities and technical systems. A cumulative p value cannot be the same for the supplementary, auxiliary and binding functions. Each technical system can have a different number of functions. For definition and uniqueness reasons, each function of a particular technical system in the MFF matrix is described by parameters, winning parameters and value intervals. However, it is not certain that it will be displayed as it has been mentioned above that it is displayed only when it solves a given functional requirement with a significant probability. Depending on the complexity of the function, it can be described by one or more parameters. In no case can it happen that a function could be left with no parameters, since a function without parameters is no longer a function.

3 IMPLEMENTATION

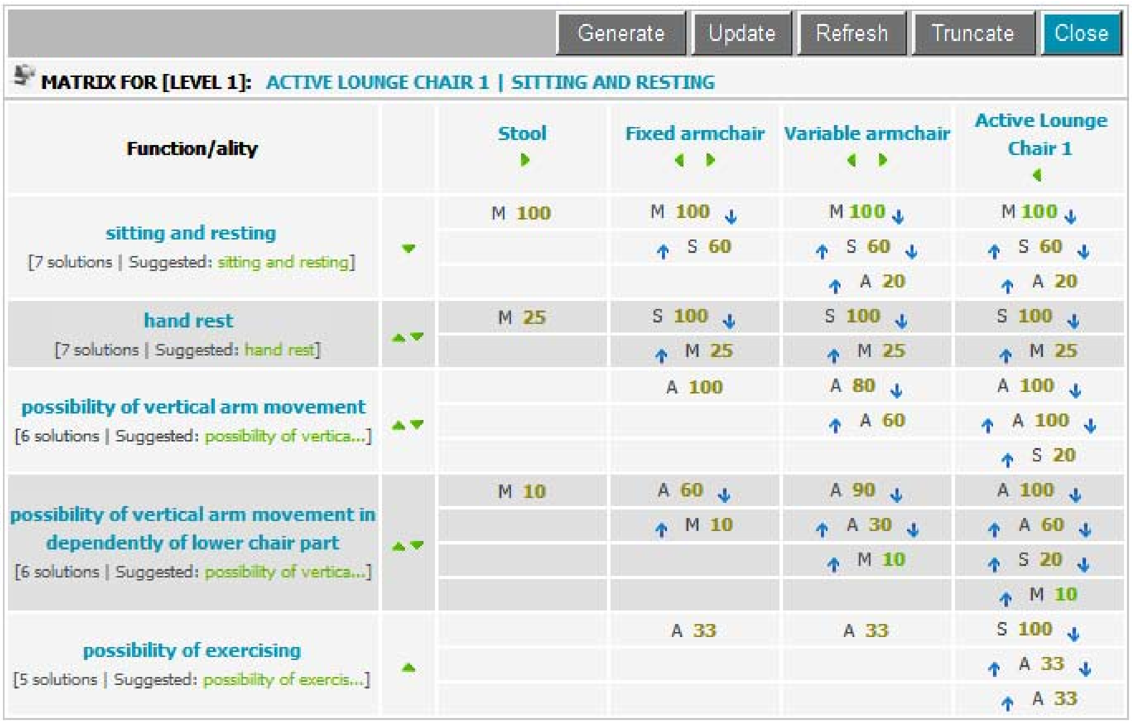
The MFF modularity model will be presented and implemented on a selected product, called the Active Lounge Chair 1 (ALC 1). The goal of the implementation is not to design a complete Active Lounge Chair 1, but to clearly show and prove the modularity for part of the chair (the arm rest with an exercise mechanism for the hands, Figure 2). Due to the fact that the functional requirements mark the beginning of the design process, the original idea is to direct the designer through the modularity of the shape, function and MFF matrix, and to come to possible new solutions. They will be based on rules and a mathematical model, not only on the basic design intuition.

The Active Lounge Chair 1 represents a product whose basic functions are *sitting*, *resting* and *exercising*. It is aimed at a wide range of users of all ages. The key component parts of the Active Lounge Chair 1 are: the sitting part, the leg/foot rest, the arm/hand rest, the upper body rest, and the hand and foot exercise mechanism, as shown in Figures 2 and 4, where each of the component parts allows and fulfils a precisely defined function. Figures 2 and 4 are composed of several individual pictures that precisely and clearly show the design thinking behind the chair concept, particularly the arm rest with the exercise option.

Figure 2: Active Lounge Chair 1 – ALC 1

Before presenting a concrete matrix and modularity within the MFF it should be made clear that the basic, theoretical model of the MFF functioning has been developed and confirmed in a prototype computer web application. The power of managing, running and creating the product data is provided by a central relational database with a relevant database-management system. The model is complemented and upgraded by a number of additional modules, but they will not be dealt with at this point because they are too extensive.

The MFF in Figure 3 represents the real modular matrix *ALC 1* concept. The matrix involves several possible solutions (schematically shown in Figure 4), cross-corresponding to several functions. The main possible functionalities are *Stool*, *Fixed Armchair*, *Variable Armchair* and *Active Lounge Chair 1*, among which it is possible to manipulate the desired functions or functional requirements: *sitting and resting*, *hand rest*, *possibility of vertical arm movement*, *possibility of vertical arm movement independently of lower chair part* and *possibility of exercise*.


Generate | Update | Refresh | Truncate | Close

MATRIX FOR [LEVEL 1]: ACTIVE LOUNGE CHAIR 1 | SITTING AND RESTING

Function/ality	Stool	Fixed armchair	Variable armchair	Active Lounge Chair 1
sitting and resting [7 solutions | Suggested: sitting and resting]	M 100	M 100	M 100	M 100
		S 60	S 60	S 60
			A 20	A 20
hand rest [7 solutions | Suggested: hand rest]	M 25	S 100	S 100	S 100
		M 25	M 25	M 25
possibility of vertical arm movement [6 solutions | Suggested: possibility of vertica...]		A 100	A 80	A 100
			A 60	A 100
				S 20
possibility of vertical arm movement in dependently of lower chair part [6 solutions | Suggested: possibility of vertica...]	M 10	A 60	A 90	A 100
		M 10	A 30	A 60
			M 10	S 20
				M 10
possibility of exercising [5 solutions | Suggested: possibility of exercis...]		A 33	A 33	S 100
				A 33
				A 33

Figure 3: Implementation of the matrix of function and functionality on the example of ALC 1

A product can appear in one or more variants, which can be pooled into modular assemblies according to the principle of shape modularity or the principle of function modularity. The basic feature of shape modularity is to establish how many functions are fulfilled by each product variant. For example, Figure 3 reveals that the *Fixed armchair* variant completely replaces the *Stool* variant, as it solves the *Stool's* main function (*sitting and resting*), as well as another function: *hand rest* and the *possibility of vertical movement*, which is by default not fulfilled by the *Stool*. The function solution within the technical system is shown as a percentage value in cells, i.e., cross-intersections in the matrix. The displayed value can be highlighted in various colours, depending on the quality of the sought-after data that can be found within different function types. The probability of a suitable solution hierarchically follows in colours from the most probable green to brown and the least probable grey. Compared to the *Variable armchair* variant, the *Active Lounge Chair 1* variant solves some other, additional functions that the former variant does not solve by default. It can be argued that the *Active Lounge Chair 1* variant, compared to all three other variants, is more sophisticated and fulfils more functions. The *Active Lounge Chair 1* is actually the only modular end solution that fulfils all the set functions according to the shape-modularity principle. It can also be argued that if a product variant includes all the functions of another variant, as well as the functions that the other variant does not possess, that variant can replace and substitute it.

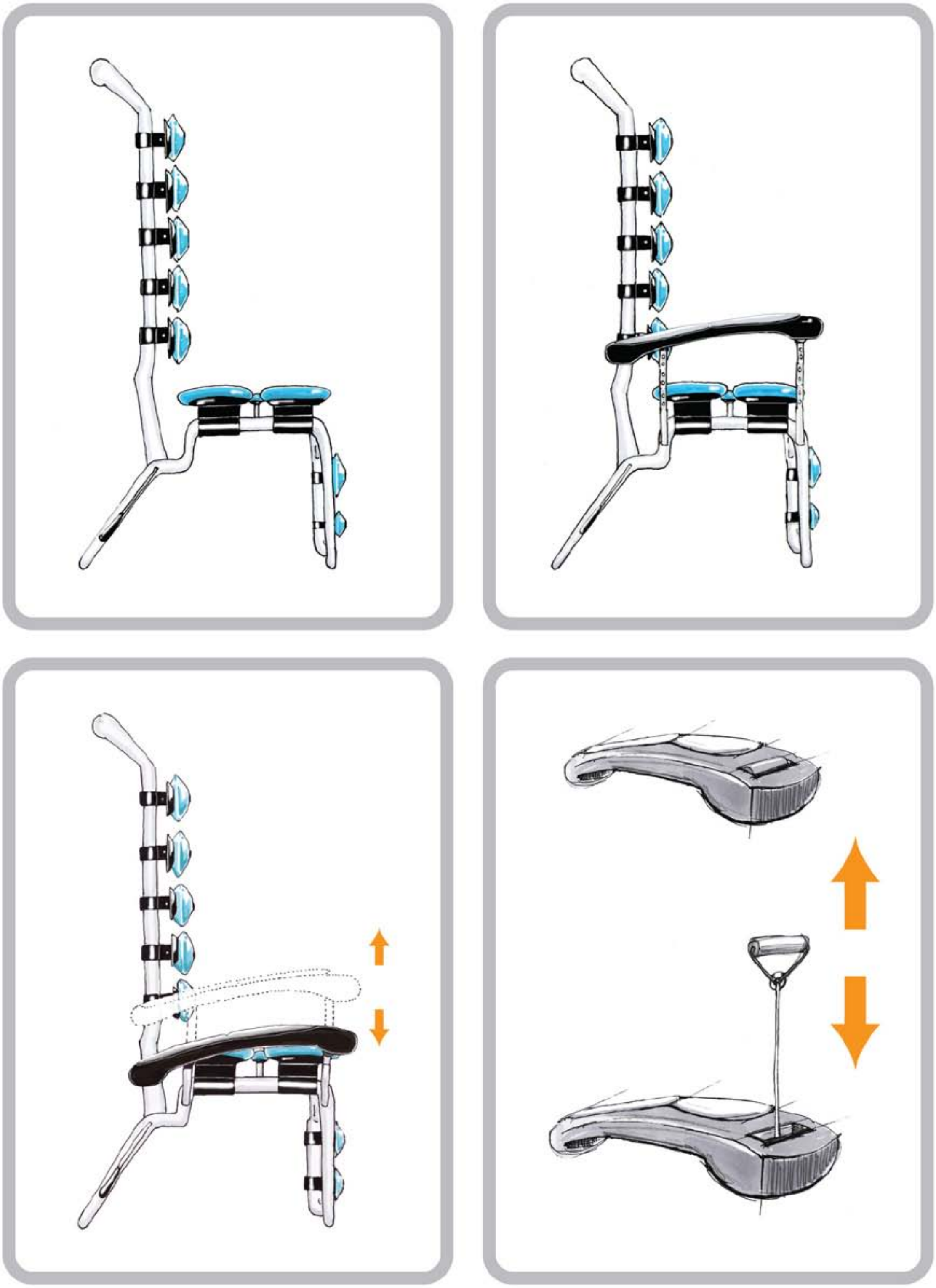


Figure 4: Sketches of possible solutions for the exercise arm rest

On the other hand, modularity by functions can be specifically determined by providing fulfilment for a particular variant in one of the adjacent variants. It provides the possibility of using different technical systems for identical functions, which means that the real function *hand rest* corresponds to all the functionalities, except *Stool*. On the other hand, the functional requirement *sitting and resting* corresponds to all the set solutions. The Active Lounge Chair 1 functional variant completely covers all of the other three variants and so they can be replaced by the said variant. The replacement should be confirmed by an econometric study and technical fulfilment alone is not the only condition.

4 CONCLUDING REMARKS

The goal of the design process is to create new, conceptual product variants. Variants in the process are created by combining functional models and solving technical systems. The MFF modularity model is presented in order to simplify and upgrade the design process. The development of the presented descriptive mathematical model is based on the basic morphological matrix. By means of a developed mathematical model it later enables MFF matrices to create new links between functions and functionalities. Functionalities represent the technical systems of various sophistications, of course described by functions and parameters. On the basis of pre-set rules (description rules, verbal and mathematical rules), the MFF allows the generation of new conceptual variants of products for which we can say that they are not based merely on design intuition.

Due to the increasing competition on the market, enterprises have been increasingly faced with the requirement for a more precise design of the product, adapted to the customers' specific requirements. Enterprises are forced to supply the market with the greatest possible variety of products along with the smallest possible differences between individual variants. The differences are mostly about design, manufacturing and maintenance. For all these reasons, there is an increasing demand for products with an increasingly modular architecture. For this reason, this part presents a theoretical model that implements modularity with regard to shape, function and MFF.

In the case of modularity with regard to shape, products are pooled into modular assemblies. They are checked in terms of the number of functions that fulfil an individual product variant, i.e., we are determining the number of functions that are fulfilled by a particular variant. The variant that includes all the functions of another function plus some new, additional functions, is selected as the end product. The model does not include any econometric analyses that would confirm the economic feasibility of such a selected variant. Using modularity with regard to shape, it is possible to check the complexity of the product variants and the complexity of the process itself by means of the designer's self-checking. The variant that fulfils more functions is more complex and more sophisticated compared to the variant with a smaller number of functions.

In the case of modularity with regard to function, the functions are pooled for the larger number of variants. This ensures the use of various technical systems for identical functions. Two products with identical functions and not yet established functions require confirmation of their potential diversity. Only one product should be selected if no additional function has been confirmed for two functionally identical products.

The MFF model, as well as the modularity model with regard to function and shape, has been included and implemented into a prototype computer web application. By means of a developed central relations database it manages the design data for the development of new conceptual product variants.

The presentation is aimed at direct users, developers and researchers of technical systems and recognised technical processes. It is based on the connection between the recognised natural processes in nature and searching for comparable or satisfying technical processes at a certain level of knowledge development. The mission of the developed models is to contribute to, and find within, the initial design processes the appropriate fundamentals for better and faster design management.

ACKNOWLEDGEMENT

The work presented in this paper is financially supported by the Ministry of Higher Education, Science and Technology of the Republic of Slovenia and the Ministry of Science, Education and Sports of the Republic of Croatia through a bilateral project.

REFERENCES

[1] Zadnik Ž., Karakašić M., Kljajin M., Duhovnik J. Functional and Functionality in the Conceptual Design process. *Strojniški vestnik*, 2009, 55 (7-8), pp. 455-471.

[2] Kušar J., Duhovnik J., Tomaževič R., Starbek M. Finding and Evaluating Customers Needs in the Product-Development Process. *Strojniški vestnik*, 2007, 53 (2), pp. 78-104.

[3] Kurtoglu T. A Computational Approach to Innovative Conceptual Design. *PhD Thesis*, 2007 (The University of Texas at Austin).

[4] Karakašić M., Zadnik Ž., Kljajin M., Duhovnik J. Product Function Matrix and its Request Model. *Strojarstvo*, 2009, 51 (4), pp. 293-301.

[5] Karakašić M., Zadnik Ž., Kljajin M., Duhovnik J. Functional structure generation within multi-structured matrix forms. *Technical Gazette*, 2010, 17 (4), pp.465-473.

[6] Kljajin M., Ivandić Ž., Karakašić M., Galić Z. Conceptual Design in the Solid Fuel Oven Development. In *Proceedings the 4th DAAAM International Conference on Advanced Technologies for Developing Countries*, Slavonski Brod, September 2005, pp. 109-114.

[7] Hubka V., Eder W.E. *Theory of Technical Systems*, 1988 (Springer-Verlag Berlin, Heidelberg)

[8] Hirtz J., Stone R.B., McAdams D.A., Szykman S., Wood K.L. A Functional Basis for Engineering Design: Reconciling and Evolving Previous Efforts. *NIST Technical Note 1447,* 2002 (Department of Commerce United States of America, National Institute of Standards and Technology).

[9] Žavbi R., Duhovnik, J. Conceptual design chains with basic schematics based on an algorithm of conceptual design. *Journal of Engineering Design*, 2001, 12 (2), pp. 131-145.

[10] Duhovnik J., Tavčar J. Product Design Test using the Matrix of Functions and Functionality. In *Proceedings of AEDS 2005 Workshop*, Pilsen-Czech Republic, November 2005.

[11] Pavlić D. Sustav za konfiguriranje proizvoda modularne arhitekture. 2003 (Fakultet strojarstva i brodogradnje u Zagrebu).

[12] O´Grady P., Liang W.Y., Tseng T.L., Huang, C.C., Kusiak A. Remote Collaborative Design With Modules. *Technical Report 97-0,* 1997 (University of Iowa).

[13] Erixon G. Modular Function Deployment-A Method for Product Modularisation. *PhD Thesis,* 1998 (Department of Manufacturing Systems, The Royal Institute of Technology).

Contact: Žiga Zadnik
Faculty of Mechanical Engineering/University of Ljubljana
LECAD Laboratory
Aškerčeva 6
1000, Ljubljana
SI-Slovenia
Phone: +386 1 4771 740
Fax: +386 1 4771 156
E-mail: ziga.zadnik@fs.uni-lj.si
URL: http://www.fs.uni-lj.si

Žiga Zadnik is an early stage researcher at Faculty of Mechanical Engineering, University of Ljubljana, Slovenia. He received a BSc. in Mechanical Engineering in 2008. His current research interests include design theory, product development, functional and functionality modelling, matrix methods, PDM systems and programming.

Vanja Čok is an early stage researcher at Faculty of Mechanical Engineering, University of Ljubljana, Slovenia. She graduated at Academy of fine arts and design in Ljubljana, department for industrial design in 2010. Her current research interests are product design, interactions of emotions and design, Kansei engineering and design for all (user experience).

Mirko Karakašić is a full researcher at the Mechanical Engineering Faculty, J. J. Strossmayer University of Osijek, Croatia. He received his PhD degree in 2010 in the field of design theory, functional modelling and matrix methods. He is author and co-author of many scientific and professional paper published in journals and proceedings of scientific professional conferences.

Milan. Kljajin is a full professor for Machine Design in Mechanical Engineering and Head of Laboratory for Product Development – LECAD Slavonski Brod at the Faculty of Mechanical Engineering in Slavonski Brod, Croatia. His scientific interests refer to theory in mechanical engineering design and its application in practice, recycling, Eco design and diagnostic of failure to improve design.

Jože Duhovnik is a full Professor of computer-aided design at the Faculty of Mechanical Engineering, University of Ljubljana, Slovenia. His pedagogic and research work is oriented towards design theory, development technic, project management, information flow in CAD, and geometric modelling. He is founder and head of the CAD Laboratory at the Faculty of Mechanical Engineering since 1983.