

THE MATRIX OF FUNCTION AND FUNCTIONALITY IN PRODUCT DEVELOPMENT PROCESS

Karakasic, M.^{*}; Zadnik, Z.^{**}; Kljajin, M.^{*.#} & Duhovnik, J.^{***}

^{*} J. J. Strossmayer University of Osijek, Mechanical Engineering Faculty in Slavonski Brod, Trg Ivane Brlić – Mažuranić 2, 35000 Slavonski Brod, Croatia

^{**} Jungheinrich d. o. o., Korenova cesta 11, 1241 Kamnik, Slovenia

^{***} University of Ljubljana, Faculty of Mechanical Engineering, Aškerčeva 6, 1000 Ljubljana, Slovenia

E-Mail: mirko.karakasic@sfsb.hr, ziga.zadnik@jungheinrich.si, mkljajin@sfsb.hr, joze.duhovnik@lecad.fs.uni-lj.si ([#] Corresponding author)

Abstract

This work presents a developed mathematical model that transforms functions and functionalities as sets from the functional domain and design elements domain. The sets are interconnected by a matrix model which we called the matrix of function and functionality (MFF). By the MFF model we are searching cross-sections of function and functionality sets. The cross-sections we called submatrices. Submatrices contain design parameters that designers need for development of new product conceptual variants. MFF is conceived as a tool used in conceptual design phase. MFF collects and binds data from the requirement list and transforms them into functional requirements. Compared to the morphological matrix, MFF is structured as multidimensional matrix, which according to the mathematical model presented in this work, generates a functional structure as multilayer functional structure. MFF simulates the iterativity of the design process in conceptual phase, generating new conceptual variants. According to the mathematical model of MFF, a computer application was made and also presented in this work.

(Received in November 2017, accepted in May 2018. This paper was with the authors 2 months for 2 revisions.)

Key Words: Product Development, Conceptual Design, Design Process, Function, Functionality, MFF (Matrix of Function and Functionality)

1. INTRODUCTION

Conceptual design phase has important role as initial stage of product design and product development process [1, 2]. At this stage is important to examine long-term aims, rules and also to develop a policy routing of the design process [2-4]. Design process in the conceptual phase is usually specified by complexity [5], unpredictability and creativity [6], which mainly rests on an individual's intuition that is hard to repeat in creating new product concepts. It is also crucial to know that the higher level of abstraction is important for deflection from the usual and concrete. The problem is if the abstraction level is too large and when the input data in design process are unarranged, complex and for design process unacceptable. During conceptual design there are no geometric dimensions [2], exactly defined product shape, tolerances and design structure. Initial data are stored in the form of customer requirements, which need to be transformed into data acceptable to the further design process. These data are not unambiguous and exact as is the case for data in phase of detailed design.

The problem of transformation customer requirements into functional requirements and mapping of customer, functional, physical and process domain involves axiomatic design [7-9]. Since the conceptual design phase connects the customer requirements and the stage of detailed design, it's aim is to transform the information and data obtained from the market, into a form that is acceptable for designers for further guiding of the design process. According to the theory of technical systems [10], each product represents the use of a function and therefore, it reflects solving initial functional requirements [11]. To contribute to

solve this problem, we defined a set of rules by which we transformed customer requirements into product functions [12].

It is important to insure quality solutions by functions. Also is important to insure possibility for repeatability of solutions in creation of new product concepts. So process of creation of new product concepts with functions, needs development of new methods and models with distinct description in practical and theoretical way [13]. Transmission of knowledge is definitely the missing shackle [14], and it is extremely important to show the models in a practical implementation (e.g. computer programming environment in stage of conceptual design. It is the way to show that the theoretical epistemologies can be approved.

In the conceptual design stage are developed different matrix models, e.g. TRIZ [15-17], DSM [18, 19], QFD [17, 20], SOS [21] and morphological box [22]. Every method has its own way to solve the problem of data transformation from customer requirements, functional modelling, creating of conceptual variants and data management [23, 24]. After analysing the above mentioned methods, we selected a morphological matrix as fundamental guidance for the new research.

The morphological matrix represents the inner, genuinely structural characteristic of the formation. Matrix is using the intuitive classification of data without a special mathematical model or solving rules for connection between functional domain (functions) and design elements domain (technical systems). Implementation of intuition is required, which is neither infinite nor repeatable. It is not possible to connect simultaneously customer, functional, physical and design elements domain. Also it is not possible to arrange functions in functional structure. This is possible by using functional flow method [23]. With morphological matrix is not possible to divide functions into categories and unambiguously describe them by physical sizes from physical domain. Theoretically number of solution is 10×10 matrix yields 10^9 solutions. For this number is necessary to have a lot of time to evaluate and select. This technique does not show the best way to find the best possible solution. There is no fundamental principle for a selection the best solutions and design forms.

As a contribution to solving above mentioned problems, we developed a matrix of function and functionality (MFF) method. This matrix method transforms customer requirements into the functional requirements according to rules developed in [12]. The process of transformation is mapping of customer domain into the functional domain. With four function categories technical systems are described as the part of design elements domain. These technical systems we called functionalities because they are described with physical sizes and design parameters. Thus, the connection between functional and design elements domain is achieved. When we search solutions for new set of functions, we look functionalities who can solve these functions. Then the physical sizes from four categories of functions by which are described functionalities, if they are solution, is possible through the submatrices add to these functions and unambiguously describe them. In this way, with MFF, a mapping between the functional and the physical domain was achieved. Functions now become measurable and acceptable for analysis in detailed design stage. It is also possible to generate conceptual variants and their functional description in matrix form. With the computer application is possible to simulate iterativity of design process at the conceptual stage through the generation a large number of conceptual variants.

2. THE MFF METHOD

2.1 The base concept of MFF

The MFF is a method which can be useful for designers in a conceptual phase of design process, to find and analyse different solutions of functional requirements in a way which can be repeated according to a precisely defined mathematical model described in this paper.

Repeatability of solutions, guided by a mathematical model, is not possible using a morphological matrix.

For the next stages of design process (e.g. detailed design), MFF also enables easier generation of product's conceptual variants, as well as their functional and physical structure. Analysing axiomatic design [9], by MFF model, is achieved mapping and connection between the customer, functional, physical and design elements domain (Fig. 1). In domain of design elements, we represent all detail forms, which are directly dependent on the manufacturing technology. Additionally, we can also represent forms of the physical structure of the products which make direct relation with the mechanics, electrical, optics and nuclear laws and relations. Functionalities are elements of this domain. According to the set of rules [12, 25], customer domain is mapping in functional domain. Functional requirements are elements of functional domain. Physical sizes are part of design parameters and they unambiguously describe functions. Physical domain contains design parameters.

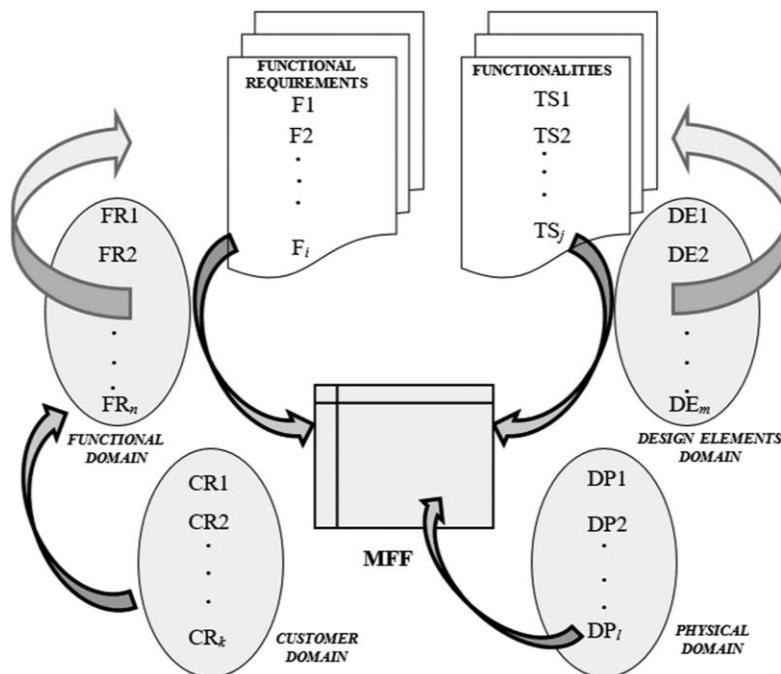


Figure 1: Binding between domains by the MFF model.

The MFF model consists of three parts (Fig. 2). The first part is *functional requirements part*. The part defines the basic functions. They are transformed from customer requirements, presented in requirement list, and representing the initial information on a new product. For the future development, it is of utmost importance. The second part is *functionality part*. Functionalities are individual possible solutions and working principles, shown as technical systems. Each functionality is an individual unity, outwardly presented with a name, while inwardly it contains specifically defined functions and functional parameters. They present information frame important for design teams. *Searching for solutions and evaluation* presents the third part of MFF. The solving framework is shown by the MFF, size $n \times m$. Inside the matrix, the possible solutions are searched based on the relationships among the functions and functionalities.

A general model of MFF is presented in Fig. 2. The relations between the functions and functionalities are achieved by submatrices (cross-section fields inside MFF). They represent the possible solutions of functions and also connection between functional, physical and design elements domain. The functional requirements in MFF model are marked with F_i . Functionalities are described by means of technical systems marked with TS_j .

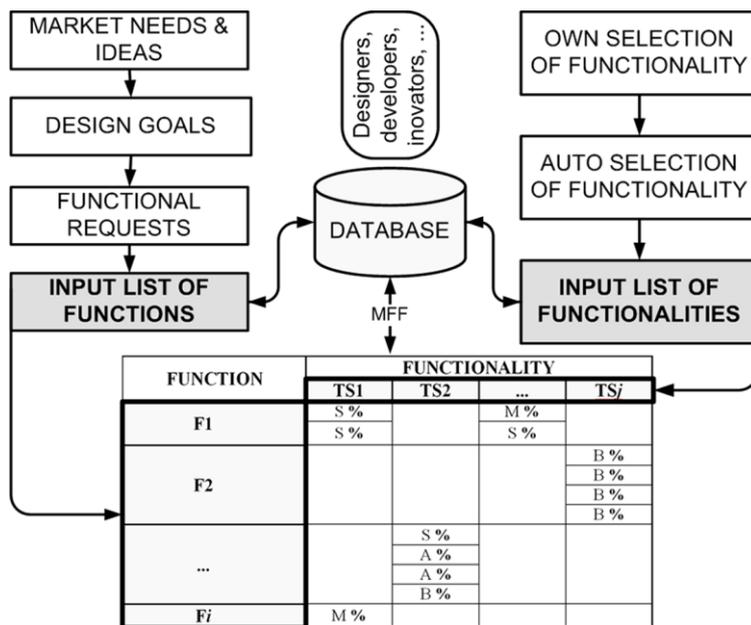


Figure 2: General model of the MFF.

When, by the MFF model, process of conceptual design starts, the first step consists of transformation customer requirements into the functional requirements. This process of mapping a customer domain into a functional domain takes place inside the input list of functions (Fig. 2). When the transformation process is complete, the functions are stored in the MFF. Functionalities which are described by main (M), supplementary (S), auxiliary (A) and binding (B) functions, are stored in database and present elements of design elements domain. Through the input list of functionalities database is connected with MFF. Input list of functionalities is a filter that has the task of searching database and storing possible function`s solution in the MFF. If function from input list has solution in specific functionality, then appears submatrix which tells which function (M, A, S, B) in a given percentage resolves the function from the input list of functions (Fig. 2). Since functions in submatrices are described by physical sizes it is possible to add these parameters to functions in input list of functions. This is a process of mapping a physical domain into a functional domain. The process of making a concept is over when each function from the input list of functions is described with physical sizes and functionalities. Conceptual solutions (variants) are stored in a matrix form by MFF. The number of matrices corresponds to the total number of conceptual solutions.

According to the MFF model described above, the search for function`s solutions results with creation of conceptual variants. This process is iterative and each new iteration can give a new conceptual solution. In this way, using the MFF, the simulation of the design process has been realized at the conceptual phase. In this simulation process, which has related to iterativity, it is possible to vary functions and function parameters at the beginning of each new simulation process. The result is visible as a set of conceptual solutions derived from several simulation processes. Each new simulation process gives a new design solution, or a new conceptual variant. It is therefore possible to obtain a new design solution that makes the final solution (product) inside design elements domain.

2.2 Mathematical model of MFF

In the mathematical model of MFF (Fig. 3), functional requirements $F_i, i = 1, \dots, n$, are marked with corresponding marks and the following applies:

$$F_1, F_2, F_3, \dots, F_n; \forall n \in \mathbb{N} \quad (1)$$

Functionalities $TS_j, j = 1, \dots, m$, are marked with corresponding marks and the following applies:

$$TS_1, TS_2, TS_3, \dots, TS_m; \forall m \in \mathbb{N} \tag{2}$$

| FUNCTIONAL REQUIREMENTS | FUNCTIONALITIES / SOLUTIONS | | | |
|--|-----------------------------|-------------------------------|--------------------------------|-------------------|
| | $TS_1 \rightarrow$ | $\leftarrow TS_2 \rightarrow$ | $\leftarrow \dots \rightarrow$ | $\leftarrow TS_m$ |
| Functional requirement - F_1 ↓ [Suggested solution R_{F_1}] | $X_{F_1TS_1}$ | $X_{F_1TS_2}$ | $X_{F_1TS_3}$ | $X_{F_1TS_m}$ |
| Functional requirement - F_2 ↑↓ [Suggested solution R_{F_2}] | $X_{F_2TS_1}$ | $X_{F_2TS_2}$ | ... | $X_{F_2TS_m}$ |
| Functional requirement - F_3 ↑↓ [Suggested solution R_{F_3}] | $X_{F_3TS_1}$ | ... | ... | $X_{F_3TS_m}$ |
| ⋮ ↑↓ | ⋮ | ... | ⋮ | ⋮ |
| Functional requirement - F_n ↑ [Suggested solution R_{F_n}] | $X_{F_nTS_1}$ | $X_{F_nTS_2}$ | $X_{F_nTS_3}$ | $X_{F_nTS_m}$ |

Figure 3: Final mathematical form of the MFF.

Individual cross-sections $F_i \cap TS_j$ or $TS_j \cap F_i$ (solutions, i.e. submatrices) are defined as in Fig. 4 and the following applies:

$$F_i \cap TS_j = TS_j \cap F_i$$

$$F_i \cap TS_j = \{X_{F_i,TS_j} \mid X_{F_i,TS_j} \in F_i \wedge X_{F_i,TS_j} \in TS_j\}$$

$$i = 1, \dots, n; j = 1, \dots, m \tag{3}$$

All cross-sections represent solutions or submatrices. They are one functional requirement. So it is possible to write:

$$R_{F_i} = \{X_{F_i,TS_1}, X_{F_i,TS_2}, X_{F_i,TS_3}, \dots, X_{F_i,TS_m}\}, \tag{4}$$

where $i = 1, \dots, n$.

All cross-sections represent solutions or submatrices. They are one functionality. It is possible to write:

$$R_{TS_j} = \{X_{F_1,TS_j}, X_{F_2,TS_j}, X_{F_3,TS_j}, \dots, X_{F_n,TS_j}\}, \tag{5}$$

where $j = 1, \dots, m$.

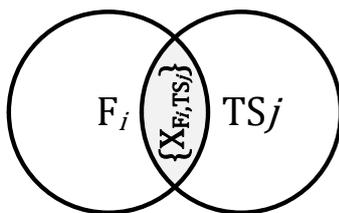


Figure 4: Cross-section (solution, submatrix) $F_i \cap TS_j = X_{F_i,TS_j}$.

The whole matrix of cross-section (solutions, submatrices) is written:

$$\mathbf{R} = \begin{bmatrix} \mathbf{X}_{F_1,TS_1} & \mathbf{X}_{F_1,TS_2} & \cdots & \mathbf{X}_{F_1,TS_m} \\ \mathbf{X}_{F_2,TS_1} & \mathbf{X}_{F_2,TS_2} & \cdots & \mathbf{X}_{F_2,TS_m} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{X}_{F_n,TS_1} & \mathbf{X}_{F_n,TS_2} & \cdots & \mathbf{X}_{F_n,TS_m} \end{bmatrix} \quad (6)$$

In m dimensional vector space, rank r of $m \times n$ dimensional matrix \mathbf{R} is less than or equal to number m or n :

$$\text{rank}(\mathbf{R}) = r \leq \min\{m, n\} \quad (7)$$

A $n \times 1$ matrix can be transformed in the column vector:

$$\mathbf{M}_{TS_j} = \begin{bmatrix} \mathbf{X}_{F_1,TS_j} \\ \mathbf{X}_{F_2,TS_j} \\ \mathbf{M} \\ \mathbf{X}_{F_n,TS_j} \end{bmatrix}; j = 1, \dots, m \quad (8)$$

A $1 \times m$ matrix can be transformed in the row vector:

$$\mathbf{M}_{F_i} = \begin{bmatrix} \mathbf{X}_{F_i,TS_1} \\ \mathbf{X}_{F_i,TS_2} \\ \mathbf{M} \\ \mathbf{X}_{F_i,TS_m} \end{bmatrix}; i = 1, \dots, n \quad (9)$$

The transpose form is:

$$\mathbf{M}_{F_i}^T = [\mathbf{X}_{F_i,TS_1}, \mathbf{X}_{F_i,TS_2}, \dots, \mathbf{X}_{F_i,TS_m}] \quad (10)$$

The set of all possible solutions of MFF is written:

$$\mathbf{R}_{MFF} = \left\{ \begin{bmatrix} \mathbf{X}_{F_1,TS_1}, \mathbf{X}_{F_1,TS_2}, \mathbf{X}_{F_1,TS_3}, \dots, \mathbf{X}_{F_1,TS_m} \\ \vdots & \vdots & \vdots & \dots & \vdots \\ \mathbf{X}_{F_n,TS_1}, \mathbf{X}_{F_n,TS_2}, \mathbf{X}_{F_n,TS_3}, \dots, \mathbf{X}_{F_n,TS_m} \end{bmatrix} \right\} \quad (11)$$

The sum of all possible solutions of MFF is:

$$\mathbf{R}_{\sum MFF} = \sum_{i=1}^n \sum_{j=1}^m \mathbf{X}_{F_i,TS_j} \quad (12)$$

2.3 Submatrices model

The submatrices present the possible solutions of MFF. They form a link between the functions and functionalities. Each submatrix is a set of individual functions of functionality, between which a set of functional requirements seeks a solution. If there is a solution, then this solution makes the cross-section between the set of functional requirements and the set of functionalities. This cross-section is a solution or submatrix. If there is no cross-section between the functional requirements and the functionalities, then there is no solution (no submatrix).

F_i is a set of functional requirements, and TS_j is a set of functionalities:

$$\begin{aligned} F_i &= \{N_1, N_2, N_3, \dots, N_u\} \\ TS_j &= \{M_1, M_2, M_3, \dots, M_v\} \end{aligned} \quad (13)$$

where:

$$\begin{aligned} N_i &\in F_i, i = 1, \dots, u \\ M_j &\in TS_j, j = 1, \dots, v \end{aligned} \tag{14}$$

Each functional requirement is described by words, elements $N_i, i = 1, \dots, u$, represent words. For example, let us take functional requirement F_1 "Torque moment transmission". Using Eq. (13), it is possible to write:

$$F_1 = \{N_1, N_2, N_3\}, \tag{15}$$

where: N_1 – torque, N_2 – moment, N_3 – transmission. Eq. (15) takes the following form:

$$F_1 = \{\text{torque, moment, transmission}\} \tag{16}$$

Each functionality is described with a different type of function. If $M_j, j = 1, \dots, v$, are elements of TS_j set, then they make the following types of functions: *main function (MF)*, *supplementary function (SF)*, *auxiliary function (AF)* and *binding function (BF)*. Now it follows:

$$TS_j = \left\{ \begin{array}{l} MF_1, \\ SF_1, SF_2, SF_3, \dots, SF_p, \\ AF_1, AF_2, AF_3, \dots, AF_r, \\ BF_1, BF_2, BF_3, \dots, BF_s \end{array} \right\} \tag{17}$$

where: $MF_i, i = 1; SF_j, j = 1, \dots, p; AF_k, k = 1, \dots, r; BF_l, l = 1, \dots, s$. It is possible to write:

$$\begin{aligned} MF_i &= \{N_1, N_2, N_3, \dots, N_a\} \\ SF_j &= \{N_1, N_2, N_3, \dots, N_b\} \\ AF_k &= \{N_1, N_2, N_3, \dots, N_c\} \\ BF_l &= \{N_1, N_2, N_3, \dots, N_d\} \end{aligned} \tag{18}$$

The functions MF_i, SF_j, AF_k and BF_l present sub-sets of TS_j . It follows:

$$MF_i \subseteq TS_j; SF_j \subseteq TS_j; AF_k \subseteq TS_j; BF_l \subseteq TS_j \tag{19}$$

The final representation of TS_j set is shown in Fig. 5.

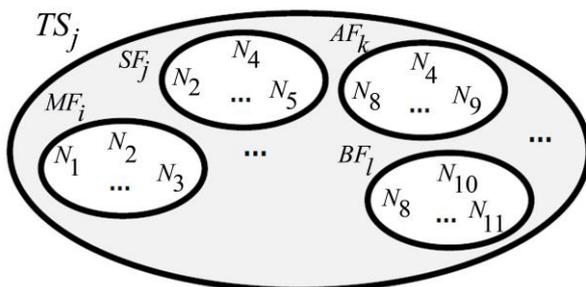


Figure 5: Graphical representation of complete TS_j set.

For example, let the function SF_1 "Torque moment generation" be a supplementary function on TS_1 . The function $SF_1 \subseteq TS_1$ contains three elements. It is possible to write:

$$SF_1 = \{\text{torque, moment, generation}\}, \tag{20}$$

where: N_1 – torque, N_2 – moment, N_4 – generation.

After the mathematical description of F_i and TS_j sets, it is possible to determine $F_i \cap TS_j$ which makes the solution X_{F_i,TS_j} (Fig. 4). Within the MFF, this solution represents a submatrix.

If we go back to the previous two examples, it is possible to find the cross-section between the function F_1 and supplementary function SF_1 , which is a sub-set of the TS_1 . It follows:

$$F_1 \cap TS_1 = \{N_1, N_2, N_3\} \cap \{N_1, N_2, N_4\} = \{torque, moment, transmission\} \cap \{torque, moment, generation\} \quad (21)$$

The solution of cross-section is a set X_{F_1,TS_1} :

$$F_1 \cap TS_1 = \{N_1, N_2\} = \{torque, moment\} = X_{F_1,TS_1} \quad (22)$$

In the submatrices solutions are displayed as a percentage value. Each cross-section $F_i \cap TS_j$ has n elements; each TS_j set has m elements. The solutions are calculated by:

$$R_{F_i \cap TS_j} = \frac{F_i}{TS_j} = \frac{n}{m} \quad (23)$$

The solution value $\frac{n}{m}$ for the functional requirement, certain functionality addresses in particular percent value. MFF shows all the solutions that are different from 0 %. For 0 % of solution value, MFF does not display the submatrices.

In the previous example, the solution value is:

$$R_{F_1 \cap TS_1} = \frac{F_1}{TS_1} = \frac{2}{3}, \quad (24)$$

respectively, function F_1 resolves functionality TS_1 with the solution value of 66 %.

3. IMPLEMENTATION

According to the mathematical model presented in this work, a computer application has been developed (Fig. 6).

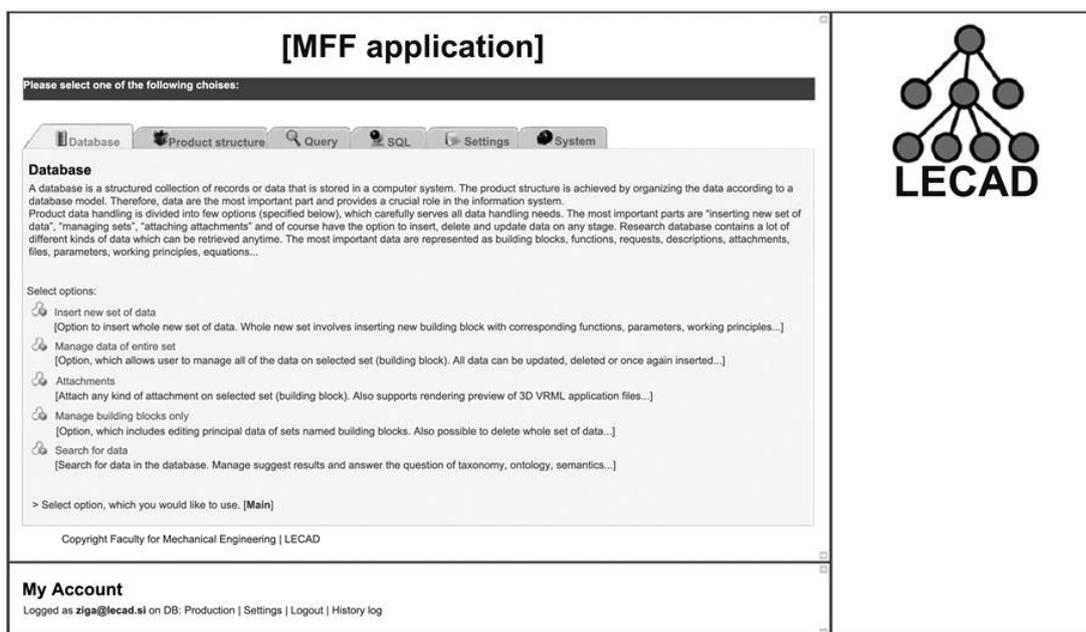


Figure 6: MFF application.

Application is divided in six sections. The most important from this standpoint are *database* and *product structure* sections. Inside of these sections we generate new concepts described with functions and physical sizes.

Process of conceptual design starts with a transformation of customer requirements into the functional requirements through the input list of functions (Fig. 7). Also functionalities stored in the application database, is possible to prepare and edit through the input list of functionalities (Fig. 7). When functions and functionalities are prepared for conceptual design, through the lists, we put them in MFF and begin generation of conceptual variants.

INPUT LIST OF FUNCTIONS

Clear Filter Confirm Close

Add new function

Acceptance of dr| Add

Acceptance of driver`s mass

Table of functions

| Referent system | Reference on Seat | |
|-----------------|---------------------------------------|----------|
| No.: | Function: | Options: |
| 1 | Vibration reduction | ↙ ✕ ⚙ |
| 2 | Acceptance of driver`s mass and... | ↙ ✕ ⚙ |
| 3 | Regulation of driver`s seat possition | ↙ ✕ ⚙ |
| 4 | Adjusting ride height | ↙ ✕ ⚙ |
| 5 | Adjusting suspension hardness | ↙ ✕ ⚙ |
| 6 | Seat rotation around one axis | ↙ ✕ ⚙ |

INPUT LIST OF FUNCTIONALITIES

Clear Filter Confirm Close

Add functionality:

Select engine method:

Select

List selection

Maximum search by function

Referent system | Reference on Seat

| No.: | Functionality: | Defined by: | Options: |
|------|-------------------|-------------|----------|
| 1 | Seat | LS method | ✕ ⚙ |
| 2 | Spring | LS method | ✕ ⚙ |
| 3 | Bolt | LS method | ✕ ⚙ |
| 4 | Guiding rail | MS method | ✕ ⚙ |
| 5 | Bearing | LS method | ✕ ⚙ |
| 6 | Frame | MS method | ✕ ⚙ |
| 7 | Adjustable spring | LS method | ✕ ⚙ |

Figure 7: Input list of functions and input list of functionalities.

The procedure of generation of two conceptual variants of assembly *seat positioning* is described in this chapter. This assembly is a part of a product at a higher hierarchical level – *tricycle*. Therefore *seat positioning* is a subassembly in tricycle`s product shape structure.

3.1 Creation of first conceptual variant

First concept is determined by function "*Regulation of seating and driving positions*" (Fig. 8). The function is on the second level of the functional structure of tricycle. This function is necessary to solve with six partial functions, which were generated by transformation of selected customer requirements (Fig. 8). Using the input list of functions, functions are implemented in the matrix.

Now begins the arranging process of the matrix, i.e. generation of first conceptual variant. Depending on the number of functions, this process can be very long, and involves functional modelling procedure. This process implies description of functions from input list, by parameters (physical sizes) from the submatrices, whose functionalities are solution of functions from input list. Functionality *Weld* was added later in MMF to check propriety of functional solutions, but in arranging process was rejected, because there is no solution (no submatrices) for functions from the input list (Fig. 8). Also is rejected functionality *Spring*, although its main function (M) solves function "*Vibration reduction*" with 100 % value.

In Table I are presented functions from input list and accepted functionalities whose functions in submatrices are solving them. It`s possible to see category of function in submatrix and percentage value of solution.

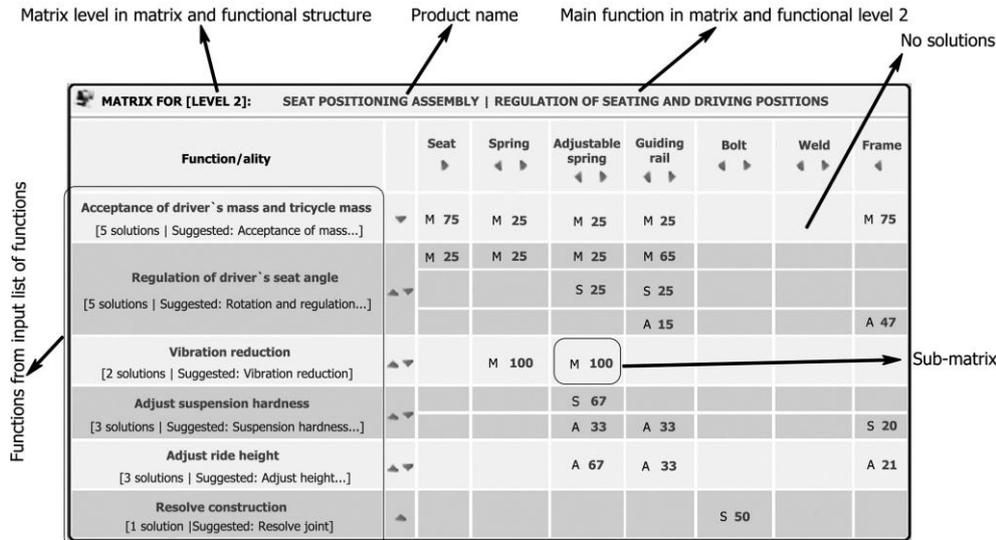


Figure 8: No arranged MFF of the first conceptual variant.

Table I: MFF solution of functions from the input list.

| Function from input list | Functionality | Function category | Solution percentage value |
|---|-------------------|----------------------------|---------------------------|
| Acceptance of driver's mass and tricycle mass | Seat | Main function (M) | 75 |
| | Frame | Main function (M) | 75 |
| Regulation of driver's seat angle | Guiding rail | Main function (M) | 65 |
| Vibration reduction | Adjustable spring | Main function (M) | 100 |
| Adjust suspension hardness | Adjustable spring | Supplementary function (S) | 67 |
| Adjust ride height | Adjustable spring | Auxiliary function (A) | 67 |
| Resolve construction | Bolt | Supplementary function (S) | 50 |

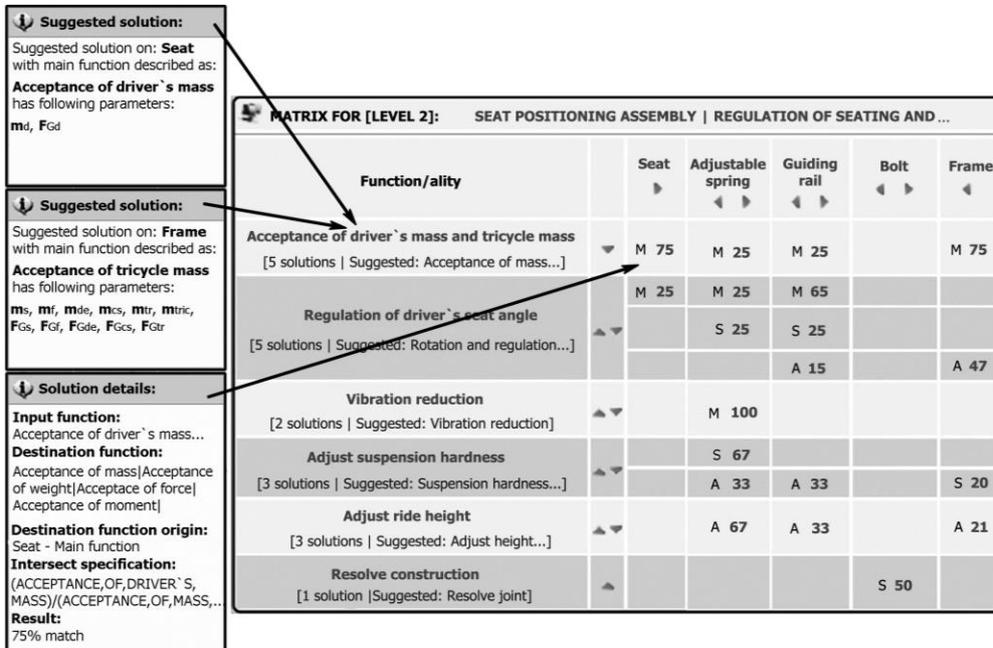


Figure 9: Arranged MFF of the first conceptual variant.

When rough structure of concept is defined, it is necessary to start with mapping process of physical sizes (parameters) from the physical domain into the functional domain. This process is necessary to do for every function from input list. For example, function

"Acceptance of driver`s mass and tricycle mass" has five solutions (Fig. 8), but only two are accepted for the first conceptual variant. In a mapping process with parameters describing main functions from both functionalities (*Seat* and *Frame*), it is necessary to describe previous function. These parameters are collected in popup windows of suggested solutions (Fig. 9).

| ID: | Name: | Mark: | Min: | Max: | Unit: | Additional: | Winning parameter: |
|--|--------------------------------|-----------------|------|------|-------|-------------|--------------------|
| MAIN FUNCTION: Acceptance of driver`s mass and tricycle mass | | | | | | | |
| 1 | Driver`s mass | md | 40 | 150 | kg | | ↓ |
| 2 | Driver`s weight | F _{Gd} | 393 | 1472 | N | | ↓ |
| 3 | Seat`s mass | ms | 5 | 7 | kg | | ↓ |
| 4 | Frame mass | mf | 20 | 25 | kg | | ↓ |
| 5 | Control system mass | mcs | 3 | 5 | kg | | ↓ |
| 6 | Damping element mass | mde | 3 | 4 | kg | | ↓ |
| 7 | Tricycle mass | mtrc | 32 | 45 | kg | | ↓ |
| 8 | Transmission ratio system mass | mtr | 1 | 3 | kg | | ↓ |

Figure 10: Description of function with physical sizes.

Application automatically maps parameters in function "Acceptance of driver`s mass and tricycle mass" and gives possibility for numerical description of parameters (Fig. 10). They are very important, because they present design constraints and design requirements. If construction demands, it is possible to add new parameters or delete previous parameters (Fig. 10). Now all described functions are stored in arranged MFF. When concept *seat positioning* is arranged in MFF, this matrix will be implemented in general matrix and functional structure of tricycle (Fig. 11). After the detail design of this concept, according to MFF (Fig. 9), final shape and subassembly *seat positioning* is presented in Fig. 12.

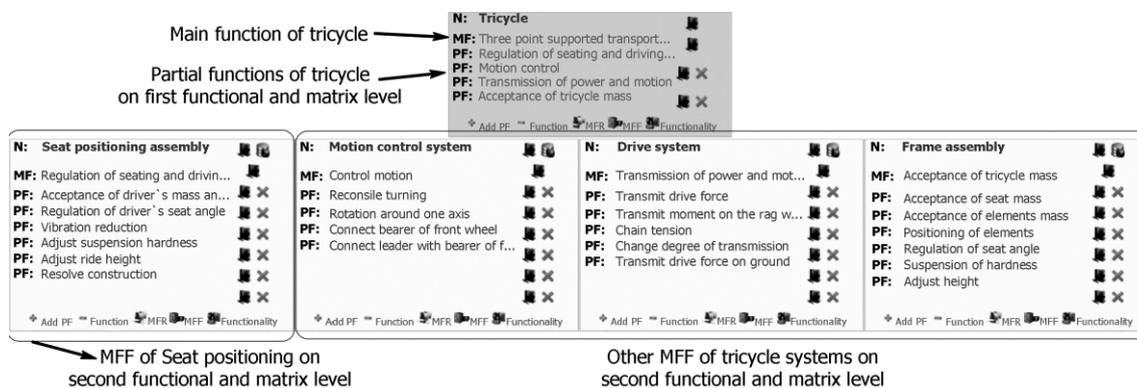


Figure 11: General tricycle`s functional structure, with first concept, generated through the MFF.

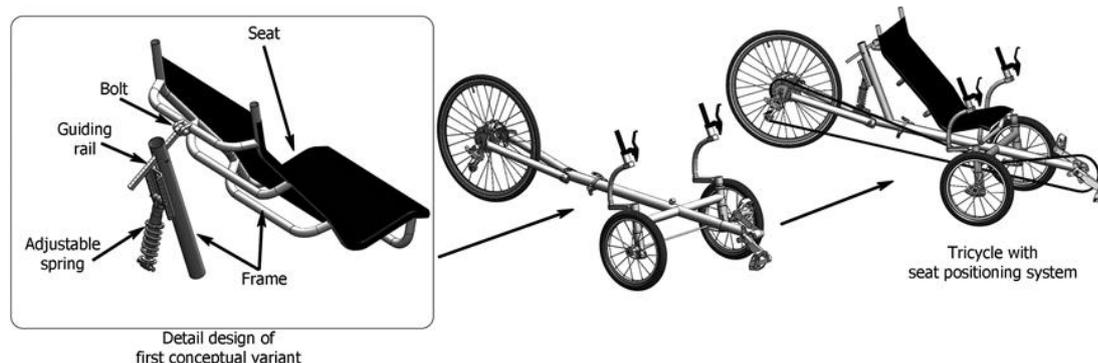


Figure 12: Detail design of the first conceptual variant.

3.2 Creation of second conceptual variant

Second concept is determined by function "Fix driving and sitting position" (Fig. 13). This function is different from main function of the first conceptual variant. According to this function, six functions for its solution are necessary too. It's possible to see that functionalities *Weld* and *Adjustable spring* are not acceptable for the concept. Process of solution creating is similar like creation of the first conceptual variant.

When the second concept of *seat positioning* is arranged in MFF, this matrix will be implemented in general matrix and functional structure of tricycle (Fig. 14). After the detail design of this concept, according to MFF (Fig. 13), final shape and subassembly *seat positioning* is presented in Fig. 15.

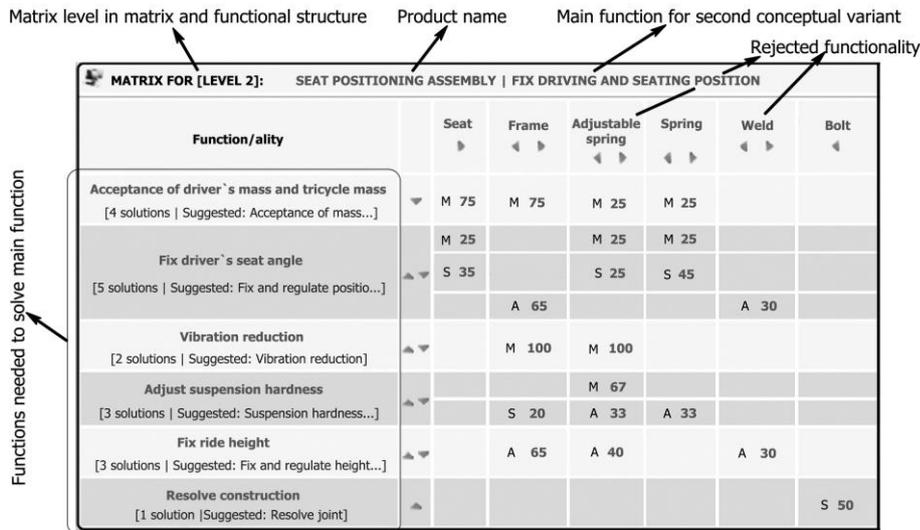


Figure 13: Arranged MFF of the second conceptual variant.

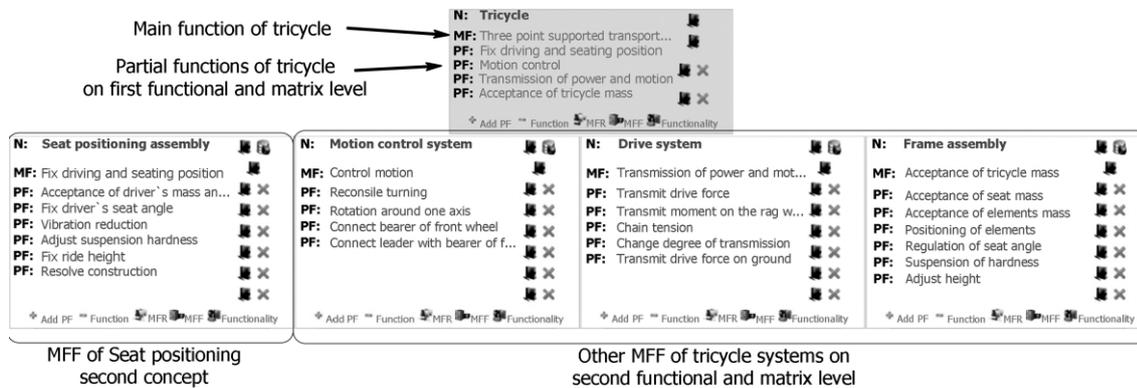


Figure 14: General tricycle's functional structure, with the second concept, generated through the MFF.

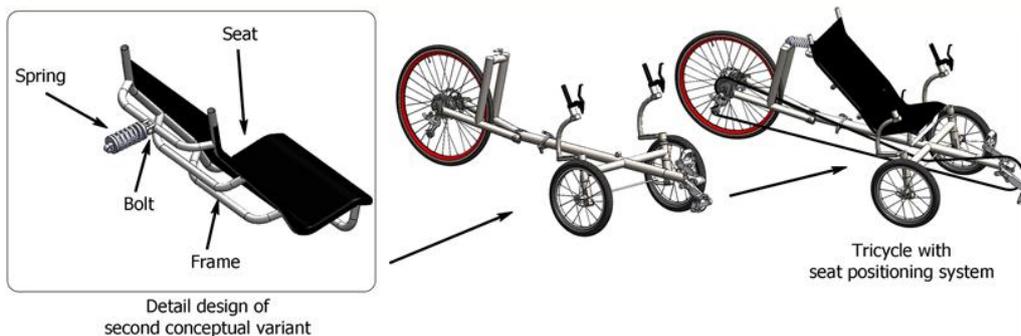


Figure 15: Detail design of the second conceptual variant.

4. CONCLUSION

By analysing matrix methods used at the conceptual design stage, first of all morphological matrix, the problem of associating, displaying and transforming of customer requirements into acceptable data for further guidance of design process was noticed. In order to contribute to the solution of this problem, a MFF model was developed. Using the MFF model, according to the theory of technical systems and axiomatic design, binding between customer, functional, physical and design elements domain was achieved. The mathematical description of the model and the developed computer application are presented in this work.

The beginning of the design process requires a new set of customer requirements. Through the set of rules, the process of mapping customer requirements into the set of functions represents a transformation a customer domain into the functional domain. Solutions of a new set of functions are obtained through the MFF by linking with the functions who describe functionalities. These functions are divided into the four categories (main, supplementary, auxiliary and binding function). On this way functionalities are solutions of new set of functions. Every new set of functions represent a new conceptual variant of new future product. Physical sizes from four categories of functions, through the submatrices, are possible to add to these set of functions. In this way, with MFF, a mapping between the physical and functional domain was achieved. When functions are described by physical sizes, they create functional and matrix structure on new product in matrix form.

Future research involves the development of mathematical model that would generate criteria for evaluation a set of conceptual variants. It's a problem of decision making process. In creation of new concepts important is to make decision which concept is better for future detail design process. Idea is with mathematical model transform functions in criteria for decision making process. Also is necessary to develop model that can create weight factors and map them to evaluation criteria. According to this, it is necessary to upgrade present computer application. On this way, with this models and this application it will be possible to simulate a creation of new conceptual variants and also will be possible to simulate a decision making process in conceptual design stage.

REFERENCES

- [1] Weisbrod, G.; Kroll, E. (2018). Idea-configuration-evaluation (ICE): development and demonstration of a new prescriptive model of the conceptual engineering design process based on parameter analysis and C-K theory, *Research in Engineering Design*, Vol. 29, No. 2, 203-225, doi:[10.1007/s00163-017-0263-6](https://doi.org/10.1007/s00163-017-0263-6)
- [2] Hofer, M. J.; Frank, M. C. (2018). Automated manufacturing process selection during conceptual design, *Journal of Mechanical Design*, Vol. 140, No. 3, Paper 031701, 12 pages, doi:[10.1115/1.4038686](https://doi.org/10.1115/1.4038686)
- [3] Hubka, V.; Eder, W. E. (1988). *Theory of Technical Systems*, Translation of the 2nd German edition, Springer-Verlag, Berlin
- [4] Duhovnik, J. (2004). Design goal as a customer need, *Proceedings of the CIMA 2004 – Conférence Internationale de Mécanique Avancée*, 9 pages
- [5] Ameri, F.; Summers, J. D.; Mocko, G. M.; Porter, M. (2008). Engineering design complexity: an investigation of methods and measures, *Research in Engineering Design*, Vol. 19, No. 2-3, 161-179, doi:[10.1007/s00163-008-0053-2](https://doi.org/10.1007/s00163-008-0053-2)
- [6] Kreye, M. E.; Goh, Y. M.; Newnes, L. B. (2011). Manifestation of uncertainty – a classification, *Proceedings of the 18th International Conference on Engineering Design (ICED 11)*, Vol. 6, 96-107
- [7] Liu, J.; Chen, B.; Xie, Y. (2016). An improved axiomatic design approach in distributed resource environment, Part 1: Toward functional requirements to design parameters transformation, *Procedia CIRP*, Vol. 53, 35-43, doi:[10.1016/j.procir.2016.06.104](https://doi.org/10.1016/j.procir.2016.06.104)

- [8] Chen, B.; Liu, J.; Xie, Y. (2016). An improved axiomatic design approach in distributed resource environment, Part 2: Algorithm for functional unit chain set generation, *Procedia CIRP*, Vol. 53, 44-49, doi:[10.1016/j.procir.2016.06.106](https://doi.org/10.1016/j.procir.2016.06.106)
- [9] Suh, N. P. (2001). *Axiomatic Design: Advances and Applications*, Oxford University Press, New York
- [10] Eder, W. E. (2016). Theory of technical systems – Educational tool for engineering, *Universal Journal of Educational Research*, Vol. 4, No. 6, 1395-1405, doi:[10.13189/ujer.2016.040617](https://doi.org/10.13189/ujer.2016.040617)
- [11] Simpson, T. W.; Bobuk, A.; Slingerland, L. A.; Brennan, S.; Logan, D.; Reichard, K. (2012). From user requirements to commonality specifications: an integrated approach to product family design, *Research in Engineering Design*, Vol. 23, No. 2, 141-153, doi:[10.1007/s00163-011-0119-4](https://doi.org/10.1007/s00163-011-0119-4)
- [12] Zadnik, Ž.; Karakašić, M.; Kljajin, M.; Duhovnik, J. (2009). Function and functionality in the conceptual design process, *Strojnikski vestnik – Journal of Mechanical Engineering*, Vol. 55, No. 7-8, 455-471
- [13] Reich, Y. (2010). My method is better!, *Research in Engineering Design*, Vol. 21, No. 3, 137-142, doi:[10.1007/s00163-010-0092-3](https://doi.org/10.1007/s00163-010-0092-3)
- [14] Hales, C.; Wallace, K. (2011). Design research reflections – 30 years on, *Proceedings of the 18th International Conference on Engineering Design (ICED 11)*, Vol. 2, 163-172
- [15] Nikulin, C.; Zuniga, M.; Akhloufi, M.; Manzi, C.; Wiche, C.; Pinones, E. (2018). Enhancing creativity for development of automation solutions using OTSM-TRIZ: A systematic case study in agronomic industry, *Advances in Mechanical Engineering*, Vol. 10, No. 1, 15 pages, doi:[10.1177/1687814017751950](https://doi.org/10.1177/1687814017751950)
- [16] Kamarudin, K. M.; Ridgway, K.; Hassan, M. R. (2016). Modelling constraints in the conceptual design process with TRIZ and F3, *Procedia CIRP*, Vol. 39, 3-8, doi:[10.1016/j.procir.2016.01.034](https://doi.org/10.1016/j.procir.2016.01.034)
- [17] Mayda, M.; Borklu, H. R. (2014). Development of an innovative conceptual design process by using Pahl and Beitz's systematic design, TRIZ and QFD, *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, Vol. 8, No. 3, 12 pages, doi:[10.1299/jamdsm.2014.jamdsm0031](https://doi.org/10.1299/jamdsm.2014.jamdsm0031)
- [18] Eisenbart, B.; Gericke, K.; Blessing, L. T. M.; McAlone, T. C. (2017). A DSM-based framework for integrated function modelling: concept, application and evaluation, *Research in Engineering Design*, Vol. 28, No. 1, 25-51, doi:[10.1007/s00163-016-0228-1](https://doi.org/10.1007/s00163-016-0228-1)
- [19] Feng, X.; Zang, X.; Liu, Y. (2016). DSM/DMM-based methodology for handling changes in product development project, *MATEC Web of Conferences*, Vol. 40, Paper 07004, 4 pages, doi:[10.1051/mateconf/20164007004](https://doi.org/10.1051/mateconf/20164007004)
- [20] Marini, C. D.; Fatchurrohman, N.; Azhari, A.; Suraya, S. (2016). Product development using QFD, MCDM and the combination of these two methods, *IOP Conference Series: Materials Science and Engineering*, Vol. 114, Paper 012089, 8 pages, doi:[10.1088/1757-899X/114/1/012089](https://doi.org/10.1088/1757-899X/114/1/012089)
- [21] Rosenstein, D.; Reich, Y. (2011). Hierarchical system concept generation, *Proceedings of the 18th International Conference on Engineering Design (ICED 11)*, Vol. 10, 24-34
- [22] Dragomir, M.; Banyai, D.; Dragomir, D.; Popescu, F.; Criste, A. (2016). Efficiency and resilience in product design by using morphological charts, *Energy Procedia*, Vol. 85, 206-210, doi:[10.1016/j.egypro.2015.12.218](https://doi.org/10.1016/j.egypro.2015.12.218)
- [23] Yildirim, U.; Campean, F.; Williams, H. (2017). Function modeling using the system state flow diagram, *Artificial Intelligence for Engineering Design, Analysis and Manufacturing*, Vol. 31, No. 4, 413-435, doi:[10.1017/S0890060417000294](https://doi.org/10.1017/S0890060417000294)
- [24] Sun, J.; Chai, N.; Pi, G.; Zhang, Z.; Fan, B. (2017). Modularization of product service system based on functional requirement, *Procedia CIRP*, Vol. 64, 301-305, doi:[10.1016/j.procir.2017.03.038](https://doi.org/10.1016/j.procir.2017.03.038)
- [25] Karakašić, M.; Zadnik, Ž.; Kljajin, M.; Duhovnik, J. (2010). Functional structure generation within multi-structured matrix forms, *Technical Gazette*, Vol. 17, No. 4, 465-473