Proceedings of the ASME 2008 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference
IDETC/CIE 2008
August 3-6, 2008, Brooklyn, New York, USA

DETC2008-49338

MODULARIZATION METHOD FOR ADAPTABLE PRODUCTS

Lieke Arts
Department of BioMechanical Engineering
Faculty of Mechanical, Maritime and Materials Engineering
Delft University of Technology
Mekelweg 2
Delft, 2628 CD
The Netherlands
Email: l.arts-wb@student.tudelft.nl

Magdalena K. Chmarra
Department of BioMechanical Engineering
Faculty of Mechanical, Maritime and Materials Engineering
Delft University of Technology
Mekelweg 2
Delft, 2628 CD
The Netherlands
Email: m.k.chmarra@tudelft.nl

Tetsuo Tomiyama
Department of BioMechanical Engineering
Faculty of Mechanical, Maritime and Materials Engineering
Delft University of Technology
Mekelweg 2
Delft, 2628 CD
The Netherlands
Email: t.tomiyama@tudelft.nl

ABSTRACT

Adaptable products are gaining interests. Those products are able to adapt themselves to new environments, new states or new user defined tasks. There is not yet a standard design methodology for designing those products. This paper focuses on making large complex products (e.g. printers) more adaptable. Large-scale complex systems need to have modular architecture to some extent in order for engineers to be able to clearly comprehend the product. Therefore, a method to cluster components of an adaptable system is developed based on Design Structure Matrix (DSM) which stores information about connections between components. For each scenario or action plan to perform adaptability, the importance of component interconnections is rated in a separate DSM structure. By combining the original DSM with the adaptability DSM the engineers can group components. Finally, an example of a coffee maker is illustrated.

INTRODUCTION

There is a strong need for adaptable products; products that can adapt themselves to different tasks, states or environmental conditions. An example of a product that can be made adaptable is a printer that has to cope with different tasks and environments but has to maintain a high output quality.

In the literature, various definitions of adaptability can be found. For example, Gu et al. defined two types of adaptability: design adaptability and product adaptability [1]. Design adaptability applies to designing products in such a way that additional changes in the design do not cost much time and effort. Product adaptability, however, aims at a product that can adapt itself to different conditions.

Olewnik et al. described adaptability as ‘system parameters that can be changed to enhance performance of the system in predictable situations; these parameters can be changed when the system is not in use (passive) or in real time (active)’ [2]. Robust systems, however, are capable of accommodating unforeseeable changes in the operating environment. A system that is both robust and adaptable is called a flexible system.

In this paper, adaptability is defined as system parameters (characteristics) that can be changed (adapted) to improve performance of the system in predictable situations. The parameters can be changed in real time (active) or when the system is not in use (passive).The parameters can be changed only by the system itself and not by the user.

Designing an adaptable system is not trivial; many aspects should be changed or adjusted during the design process in order to obtain an adaptable product. For example, some of the product parameters can change (e.g. from constant values into variables), and it can be needed to modify the product architecture. Besides, appropriate action plans have to be determined to plan the adaptability. Moreover, the control of the product can change, and the software will need to be adjusted.

Research in the field of the decision making and measures of the flexibility has been done. Guidelines for deciding to what extent the product should be made adaptable in order to keep the product profitable have been developed [3]. However, a complete design methodology for adaptable products has not been yet developed. The goal of this paper is to come up with a method to analyze and modularize the product architecture of an existing product when adaptability is incorporated.

Complex products have, to some extent, a modular structure. Therefore the method proposed in this paper is based on modularity of a product. In the following sections of the paper product architecture and grouping of the components into modules will be discussed in the perspective of adaptability.

MODULARITY AND ADAPTABLE SYSTEMS

In the architecture of a product, there are always tradeoffs that have to be made. It has not yet been investigated how these tradeoffs should be made for adaptable products. Since complex adaptable products are difficult to manage, modular architectures (to some extent) can help engineers to design and comprehend the product.

Modularity

The concept of modularity is widely used to deal with complex products [4]. When products are divided into modules that are to some extent independent of each other, the product's design process can be divided over multiple engineers. For very complex products this is even necessary, since the human is incapable of understanding and designing all issues of the product.

In a modular product, there is a tendency to minimize the interactions between components that are not in the same module. When connections are contained in one module only (e.g. module 1), components outside that module do not need any information about connections and parameters of the components of that module (module 1). Therefore, information can be hidden into that particular module (module 1). Very often this is not the case and the information hiding should be planned carefully.

There are always some connections that are not contained in a module. These connections have to be managed and thus the interfaces between different modules have to be carefully defined. Only then the modules will be able to function together.

There are different methods for designing a modular product. Six primary methods are discussed by Holmqvist and Persson [5]. The methods discussed include complete design methods and grouping methods for creating modules. One of the discussed methods is the Design Structure Matrix (DSM).

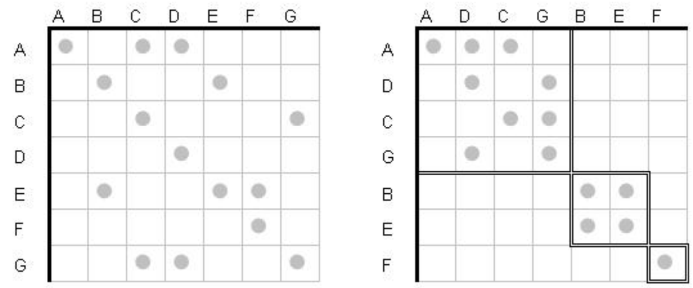


Figure 1. ORIGINAL DSM AND CLUSTERED DSM

Design Structure Matrix

The concept of the DSM is discussed by Browing [6]. The concept as a modularization method was described by Pimmler and Eppinger [7]. The DSM has been used widely for many purposes. The DSM can be a tool for structuring and clustering the product components. The individual components of a product are listed in the matrix and the interactions between these components are rated. With help of clustering algorithms modules can be identified using the information in the DSM. Figure 1 shows the concept of the DSM for an imaginary product consisting of 7 components (A-G).

Rating interactions. The quality of the DSM depends greatly on the accuracy of the interaction information. Information about the interactions between components can be stored in the DSM in different ways. For example, Figure 1 shows an original (left) and a clustered (right) DSM. As shown, the interactions between components may vary in different directions. In the current example component C and D depend on component A, but component A does not depend on C, D or any other component.

In Figure 1 the interactions between components are stored binary, showing whether there is or there is not a connection between the components. The interactions, however, can also be rated. For example, a scale from -2 to 2 can give information about the strength and the desirability of the connection. To include information about the nature of the interactions the connections can also be rated in different domains. The four DSMs presented in Fig. 2 are: the spatial DSM ($\mathbf{D_s}$), which shows the associations of physical space and alignment, the energy DSM ($\mathbf{D_e}$), which shows the needs for energy exchange, the information DSM ($\mathbf{D_i}$), which shows the need for data or signals to pass between components and the material DSM ($\mathbf{D_m}$), which shows the need for material exchange. The combination of these four DSMs forms the complete DSM of a product as shown in Eqn. (1). The dimensions of the combined DSM ($\mathbf{D}$) are n x n x m where n is the number of components and m the number of domains the interactions are rated on.

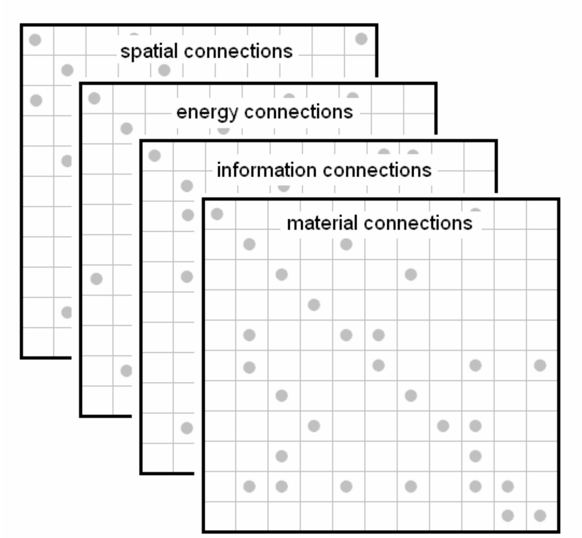


Figure 2. DSM CONTAINING INTERACTIONS ON FOUR DIFFERENT DOMAINS

$$\mathbf{D} = \left\{\mathbf{D_s}, \mathbf{D_e}, \mathbf{D_i}, \mathbf{D_m}\right\} \tag{1}$$

Clustering algorithms. Once the complete product information is contained in the DSM, a clustering algorithm can be used to identify modules. Clustering is an optimization process where as many interactions as possible should be contained inside a module. The modules, however, should not become too big since the advantages of modularity disappear when the modules are too big. There is always a tradeoff between those two objectives: to minimize the size of the modules and to keep as many interactions as possible inside the modules.

Several algorithms for clustering a DSM have been developed in the past. The genetic optimization algorithm is an example of a clustering method that is frequently used for clustering the DSM [8-11]. A more analytical method to cluster the DSMs is developed by Xiaogang et al. [12].

DSM FOR ADAPTABLE SYSTEMS

When redesigning a product for adaptability, the product parameters and/or their characteristics can change. Due to this change, some of the connections between components may become more important for the functionality of the product than other connections. To incorporate this in the design of the adaptable product, a method is derived based on the DSM.

Action plans

Initially, the engineers have to choose the parameters of the product that will be made adaptable. Subsequently, action plans for the adaptation of those parameters have to be derived taking various scenarios into account (e.g. user defined tasks, possible environment change or product state change). For each

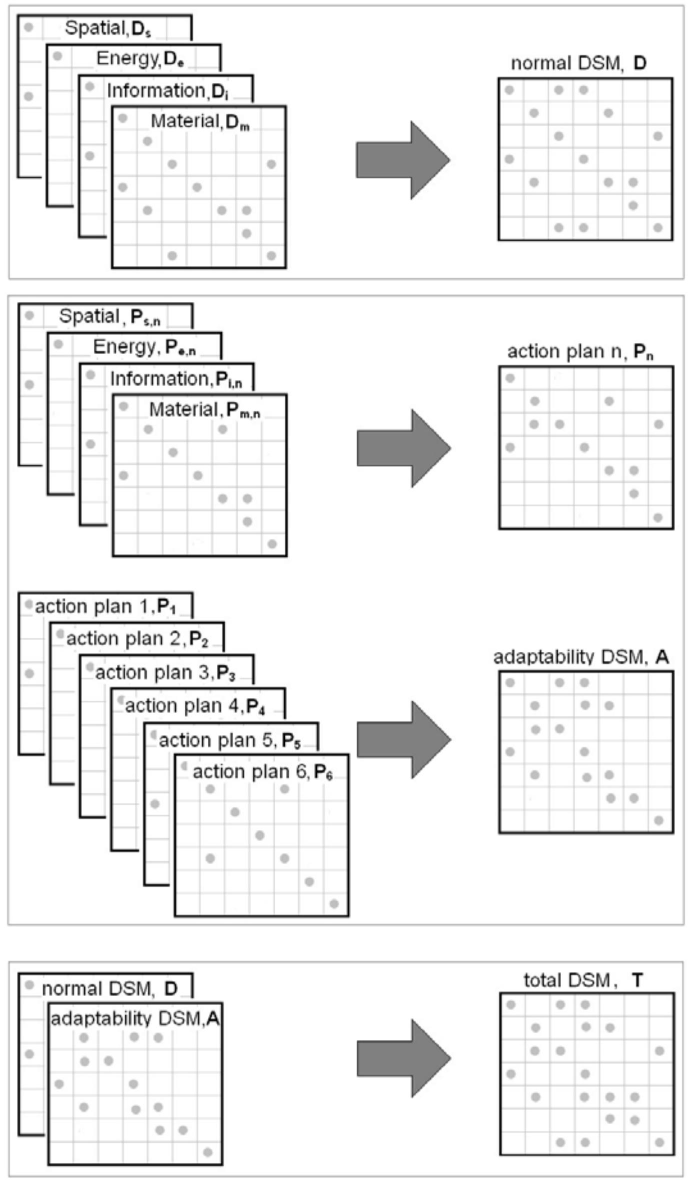


Figure 3. BUILDING THE TOTAL DSM WITH THE ADAPTABILITY DSM AND THE NORMAL DSM

scenario, the action plan has to be made. The action plans describe in detail when and how the product will adapt itself. An action plan will be executed by the product itself when the circumstances or tasks ask for that particular action plan.

Expanding the DSM

Now the original DSM of the product is expanded to deal with the action plans needed by the product to adapt itself to different tasks and changes in the environment or product state. The original DSM is still of importance since the connections in the original product will still be in the adaptable product. To be able to deal with the adaptability of the product new layers

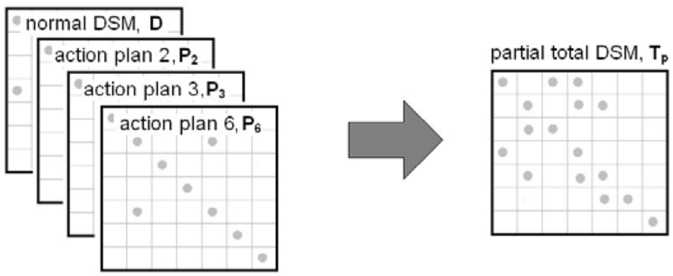


Figure 4. EXAMPLE OF THE CONSTRUCTION OF A PARTIAL TOTAL DSM WITH ONLY ACTION PLANS 2, 3 AND 6

are added to the original DSM which will be called the *normal DSM* (**D**).

For each action plan the component connections of importance for that particular action plan have to be investigated by the engineers. The important connections for the action plan have to be listed in separate DSMs on the four domains: spatial, energy, information and material. Combining the four domains gives one complete DSM for each action plan, as shown in Eqn. (2). The DSM for action plan number n is $\mathbf{P_n}$. Rating the interactions of the action plans should be done by the engineer responsible for the action plans because this process requires expert knowledge. It is possible that interactions in an *action plan DSM* are already contained in the *normal DSM*. In that case they are emphasized more in the *action plan DSM*. In the *action plan DSM* also new connections can be stored.

$$\mathbf{P} = \left\{ \mathbf{P_s}, \mathbf{P_e}, \mathbf{P_i}, \mathbf{P_m} \right\} \tag{2}$$

There are typically multiple action plans when a product is made adaptable. Dependent on the desired level of adaptability the amount of action plans might rise up.

When each action plan has been translated into a single DSM, the *adaptability DSM* can be created by combining the information from all action plan DSMs. The *adaptability DSM* (**A**) will consist of the *action plan DSMs* ($\mathbf{P_n}$) for the total amount of action plans N as shown in Eqn. (3).

$$\mathbf{A} = \left\{ \mathbf{P_1}, \mathbf{P_2}, ..., \mathbf{P_N} \right\} \tag{3}$$

The *total DSM* (**T**) for the adaptable product is the combination of the original, non-adaptable, product DSM (Normal DSM, **D**) and the *adaptability DSM* (**A**) as shown in Eqn. (4). In this *total DSM* all connections of importance are contained. The construction of the *total DSM* is shown graphically in Fig. 3.

$$\mathbf{T} = \left\{ \mathbf{A}, \mathbf{D} \right\} \tag{4}$$

Combining the DSMs

All the interaction information is stored in individual matrices that have to be combined into the total DSM. This is done in order to make the running of clustering algorithms possible. The different DSM structures can easily be added since the dimensions of each DSM will be the same

The *normal DSM* (**D**) and the *action plans DSMs* ($\mathbf{P_n}$) are obtained by combining the spatial, energy, information and material DSMs. The level of importance of each domain can be 'valuated' using weights as shown in Eqn. (5) and Eqn. (6). In this paper the same weight is proposed for the same domain DSMs of the *normal DSM* and the *action plan DSM*. Thus, for example, the weights w_e will be the same for the *normal energy DSM* ($\mathbf{D_e}$) and for the *action plan energy DSM* ($\mathbf{P_e}$).

$$\mathbf{D} = w_s \cdot \mathbf{D_s} + w_e \cdot \mathbf{D_e} + w_i \cdot \mathbf{D_i} + w_m \cdot \mathbf{D_m} \tag{5}$$

$$\mathbf{P_n} = w_s \cdot \mathbf{P_{s,n}} + w_e \cdot \mathbf{P_{e,n}} + w_i \cdot \mathbf{P_{i,n}} + w_m \cdot \mathbf{P_{m,n}} \tag{6}$$

The *action plan DSMs* are combined to form the *adaptability DSM* as shown in Eqn. (7). The different *action plan DSMs* can also be weighted so the importance of each action plan for the product can be incorporated in the *adaptability DSM*. Finally, the total DSM is given by Eqn. (8).

$$\mathbf{A} = w_1 \cdot \mathbf{P_1} + w_2 \cdot \mathbf{P_2} + ... + w_N \cdot \mathbf{P_N} \tag{7}$$

$$\mathbf{T} = \mathbf{D} + \mathbf{A} \tag{8}$$

Special attention should be given to interactions that are rated as -2 (undesired connection) by one of the DSMs and as +2 (required connection) by another DSM. Those connections cannot be added. They have to be individually addressed by the engineers. In some cases it might be necessary to change the design or to divide the component into two similar or different components in order to deal with the contradicting interests. Product reconfiguration is another solution to this problem. The chance that this conflict occurs depends on how often the rating '-2' is given in the DSMs. For some products this rating will be given to more connections than for other products.

A '-1' rating in the DSM provides information that some negative effects are present, but functionality of the components is not prevented. When a -1 rating conflicts with a positive rating in another DSM the possible negative side-effects should be carefully documented so they can be addressed in a later stage of the design process.

More insight in the types of connections can be gained when the DSMs of the four domains are combined separately for all action plans and the 'normal product'. The *total spatial DSM* ($\mathbf{T_s}$) for example, consists of the normal spatial DSM and the spatial DSMs of all the action plans. This is shown in Eqn. (9) for the spatial domain. As a result, the knowledge about the domains is captured separately.

$$\mathbf{T}_s = \mathbf{D}_\mathbf{s} + w_1 \cdot \mathbf{P}_{\mathbf{s,1}} + w_2 \cdot \mathbf{P}_{\mathbf{s,2}} + \ldots + w_N \cdot \mathbf{P}_{\mathbf{s,N}} \tag{9}$$

Weight choice. The weights that the DSMs are multiplied with are not trivial. They should be carefully chosen for each product and its DSMs to arrive at a DSM that contains the most accurate information for the engineer's current purpose. Since the weights depend on the product, engineer and company, there is still lack of a general method for choosing the weights. For the *action plan weights* w_n the sum of all weights is a measure for the importance of the *adaptability DSM* with respect to the *normal DSM*. The sum of all the *action plan weights* is the *adaptability weight* w_A as shown in Eqn. (10). The choice of the weights should not be set for a product unless this choice is carefully motivated. It can, however, be helpful to the engineers to see what impact the variation of specific weights can have to the clustering of the *total DSM* or *partial total DSM*.

$$w_A = \sum_{n=1}^{N} w_n \tag{10}$$

Clustering the total DSM

Once the *total DSM* is generated, a clustering algorithm can be used to group the product components into clusters. Existing clustering routines can be used for the clustering of the *total DSM* because it has the same structure as a normal DSM.

The clusters identified are the proposed modules for the adaptable product. Clustering the complete *total DSM* (**T**) gives the optimal clustering for the total adaptable product. More insight in the contributions of the individual *action plan DSMs* ($\mathbf{P_n}$) can be gained when a *partial total DSM* ($\mathbf{T_p}$) is constructed of the *normal DSM* and a few selected *action plan DSMs*. Now the clustering of the complete *total DSM* can be compared with the clustering of the DSM with only a few action plans involved. An example of a *partial total DSM* is given in Fig. 4.

Real time product reconfiguration. In an adaptable product, it might be the case that the product has to reconfigure itself during real time. There are two possible reasons for the product to reconfigure:

1. One or more connections between components are undesired ('-2' rating) for one of the action plans but necessary ('+2' rating) for another action plan.
2. The optimal clustering for a *partial total DSM* containing some action plans is significantly different than the optimal clustering of another *partial total DSM* containing other action plans.

If a different configuration is needed for *action plan 1* than for *action plan 4*, for example, the product might reconfigure itself when going from executing *action plan 1* to *action plan 4*. *States* can be defined consisting of action plans that have

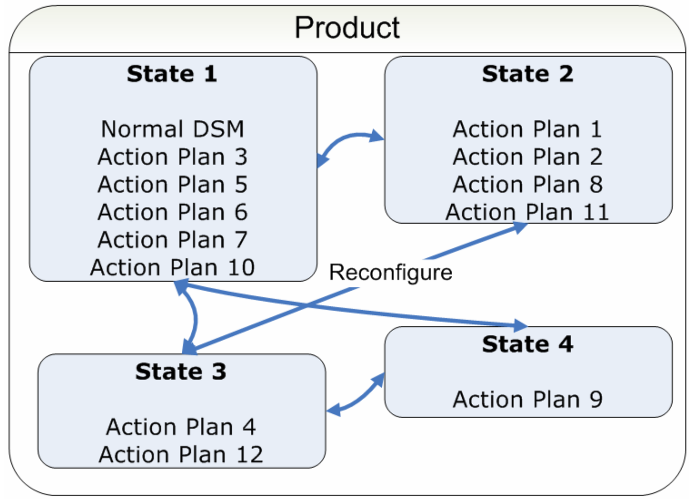


Figure 5. CONSTRUCTION OF PRODUCT STATES WITH DIFFERENT ACTION PLANS

similar clustering needs. Each *state* will contain multiple action plans. For every *state* a *partial total DSM* can be made. Action plans for which the clustering is similar should be in the same *state* of the product. An example of the division of a product into *states* with different action plans is shown in Fig. 5. Between product states the product should reconfigure. It might be the case that it is impossible to go directly from one state to another. For example, in Fig. 5, it is impossible to go directly from *state 2* to *state 4*.

Adding new components

Often, when action plans for an adaptable product are developed, new components have to be added to the product in order to achieve the desired adaptability. Those new components should be included in the DSM as well. The *normal DSM* (original, non-adaptable), will only have zeroes for those new components. The DSM corresponding to the action plan responsible for the new components will have to contain all the necessary information about the connections of those new components. Finally, all DSMs (including the *normal DSM* and every *action plan DSM*) will contain the same new amount of components.

Possible use of the method

The method that is developed is intended to be used by a team of engineers designing an adaptable product. It can be especially useful when adjusting an existing complex product. The connections between components can be investigated with this method. When an existing modular structure is used new interfaces between modules can be identified as new connections between components outside a module. Also, a new modular structure for the adaptable product can be derived. States of the product can be defined for action plans that have similar connection needs between the components.

The developed method is intended to give the design team a way to capture information about the primary connections. The propagation of changes in the product due to changes in the parameters or connections is not contained in this method; those issues should be addresses separately by the design team.

EXAMPLE

To explain the developed method, a coffee maker that works with coffee pods, and that is capable of making only one cup of coffee, is used as an example. A schematic drawing of the coffee maker is shown in Fig. 6a. The water from the water reservoir is heated by an electrical heating element. The heated water flows up in the product to the dripper. The dripper lets the hot water drip onto the coffee pod which will be placed by the user in the coffee holder. The hot water will turn into coffee and flow out at the bottom of the coffee holder. The coffee enters the cup through the coffee tube. The flow meter and water valve make sure only enough water passes through the product for one cup of coffee. The one-way valve prevents water from flowing back into the water reservoir. Parts of the coffee maker, such as the on/off switch, the button to activate the product, the indicator light, the state and temperature sensors and the power cable are not presented in Fig. 6 in order to keep the figure simple.

Adaptability action plans

Two action plans are used to make this product adaptable. The first action plan will make the product capable of making two cups of coffee. For this, an additional coffee tube has to be installed. Also, a valve to control through which coffee tube the coffee has to run has to be installed and an additional button for the user is needed. In this way the user can choose between one or two cups of coffee. The coffee holder should be capable of holding the double amount of coffee which is needed for two cups of coffee. For making two cups of coffee the double amount of water has to run through the system, which should be taken care of by the water valve. To make one cup of coffee, the coffee valve should close one of the coffee tubes. For this action plan extra components are needed in the coffee maker. The product with the extra components is shown in Fig. 6b.

The second action plan involves detecting product deterioration. The product state sensor is used to detect if one of the coffee tubes is blocked (by dirt or by coffee leaked from the coffee pod). In case one of the coffee tubes is blocked, the other coffee tube can be used instead. When two cups of coffee have to be made, the same coffee tube has to be used twice. In this case, the speed with which the water flows out of the dripper has to be adjusted in order to get two cups of coffee of the same strength. To achieve this, the temperature of the heating element has to be adjusted by the product. This is part of the action plan, as well as the control of the coffee valve. When two cups of coffee have to be made, the water valve has to be controlled by the product in such a way that it closes after

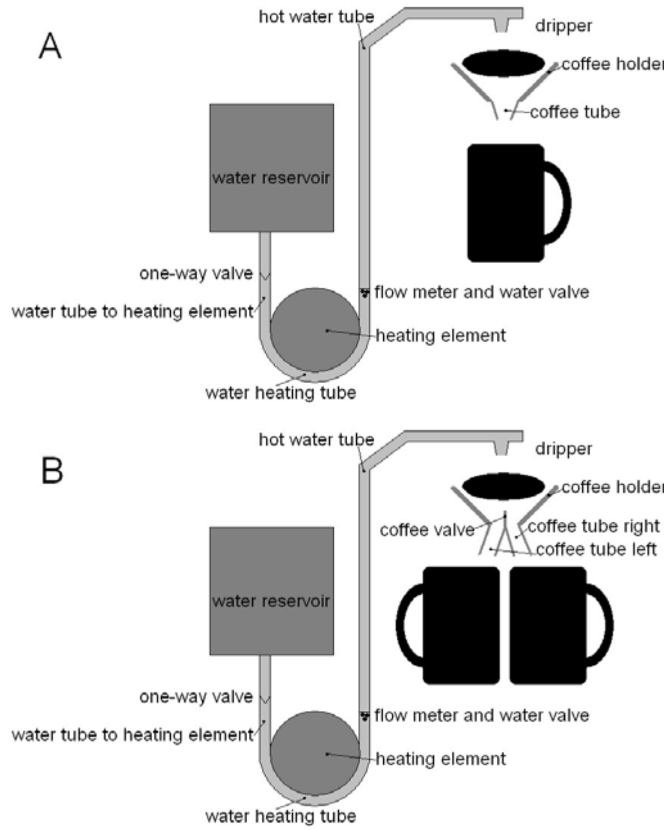


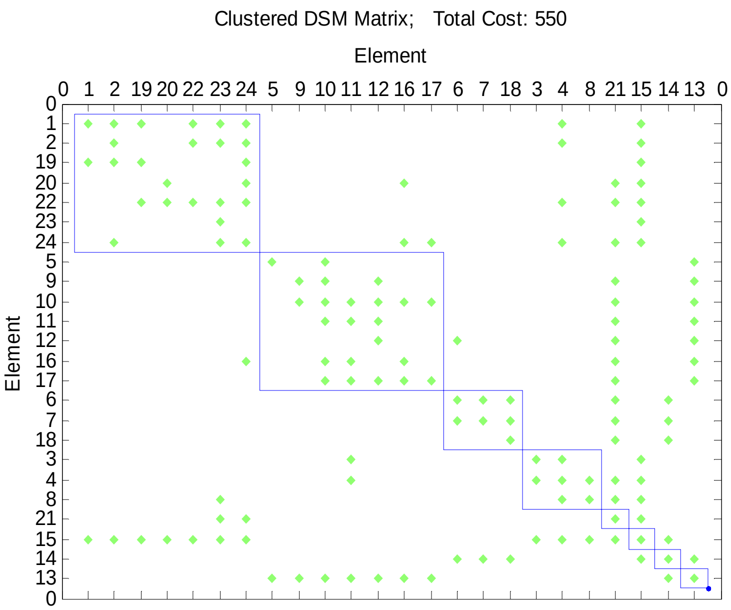
Figure 6. SCHEMATIC DRAWING OF COFFEE MAKER, A: ORIGINAL COFFEE MAKER, B: ADAPTABLE COFFEE MAKER

enough water for one cup of coffee has passed. When the user has replaced the cup, the water for the second cup should pass the water valve. This action plan is carried out by the controller that has to anticipate to this new situation.

Constructing the DSMs

With the proposed method, the grouping of the components can now be done for both the normal and the adaptable coffee maker. For simplicity, the rating of the connections in the DSM is done in a binary way. Therefore the weights in constructing the total DSMs can be omitted. When combining the binary DSMs into a new DSM, the DSM will still be binary. When combining DSMs, the combined DSM will contain all connections that are in the DSMs it consists of. The clustering algorithm used was implemented by Thebeau [13] and downloaded from the DSM homepage at: `http://www.dsmweb.org/content/view/51/38/`.

The clustering algorithm minimizes a cost function. For the relations between elements that are in a cluster the entries are summed and multiplied with the cluster size C_s to a user specified power p as shown in Eqn. (11). For the relations



Cluster Member List

Cluster #1

Power cable (1)
Voltage converter (2)
On/off switch (19)
Button (1 cup) (20)
Electric cable (22)
Indicating light (23)
Control unit (24)

Cluster #2

Water reservoir (5)
One-way valve (9)
Water tube to heating element (10)
Water heating tube (11)
Hot water tube to dripper (12)
Water flow meter (16)
Water valve (17)

Cluster #3

Dripper (6)
Coffee holder (7)
Coffee tube (left) (18)

Cluster #4

Heat conductive grease (3)
Heating element (4)
Temperature sensor (8)

Cluster #5

State sensor (21)

Cluster #6

Housing electrical elements (15)

Cluster #7

Housing general (14)

Cluster #8

Housing of water (13)

Figure 7. CLUSTERED DSM AND CLUSTER MEMBER LIST OF NORMAL (NON-ADAPTABLE) COFFEE MAKER

between elements that do not belong to any cluster the entries are summed and multiplied with the total matrix size n to the power of p as is shown in Eqn. (12). The total cost of the clustering solution is finally the sum of the two as is expressed by Eqn. (13). The value of p is for now set to 1 for our research.

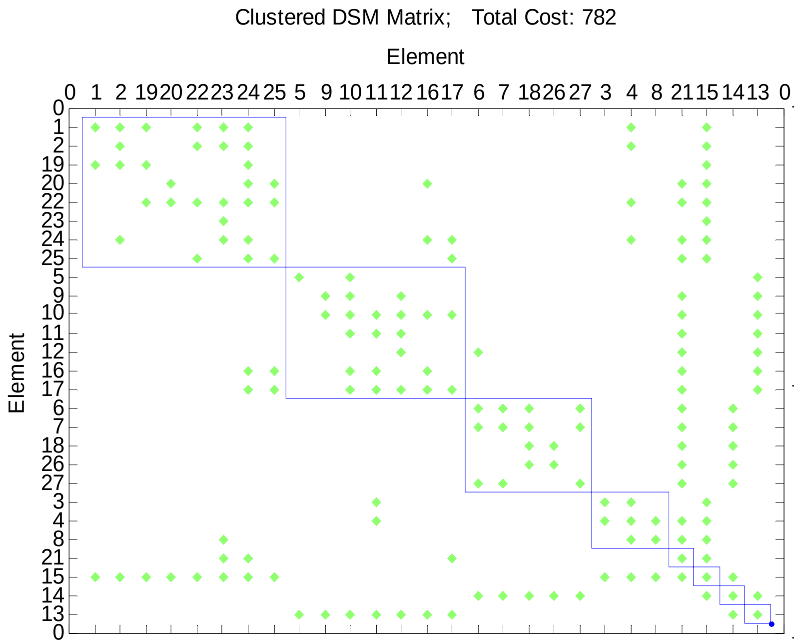
$$Cost_{cluster} = \sum_{i=1,\, j=1}^{i=n,\, j=n} \left(T(i,j)+T(j,i)\right)\cdot Cs^{p} \qquad (11)$$

$$Cost_{non-cluster} = \sum_{i=1,\, j=1}^{i=n,\, j=n} \left(T(i,j)+T(j,i)\right)\cdot n^{p} \qquad (12)$$

$$Cost_{total} = Cost_{cluster} + Cost_{non-cluster} \qquad (13)$$

Normal DSM. The *normal DSM* (**D**) of the product was created for the non-adaptable coffee maker. The coffee maker was decomposed into 24 components. Those components were listed and the interactions between the components were rated on the four domains. The *normal DSM* was created by combining the connections on the four different domains. The *normal DSM* was clustered minimizing the cost function in Eqn. (13). The housing of the coffee maker and the state sensors were not included in the clustering process, because these are ‘system parameters’. Those system parameters interact with a lot of other components so they are not suitable to be grouped. The result of the clustering of the *normal DSM* for the coffee maker is shown in Fig. 7. Aside from the ‘system parameters’ that were not included in the clustering process, four clusters were identified.

Action plan DSMs. In order to make the *adaptability DSM* first the action plans have to be converted into separate *action plan DSMs* ($\mathbf{P}_1$ and $\mathbf{P}_2$). *Action plan 1* included making the product able to also make two cups of coffee. In order for the product to be able to carry out the first action plan, extra components have to be added to the product and also to the product DSM. As a consequence, $\mathbf{P}_1$ should contain all the connections involved with those new components. Also the important connections for this action plan between the original components have to be listed. For example, the control unit should now control the water valve to be able to make one or two cups of coffee. The connection between the water valve and the control unit was already in the *original DSM* (**D**) but is also in $\mathbf{P}_1$. In this case, this particular connection in $\mathbf{P}_1$ has no influence on the total DSM because the connection was already in **D** and the DSM is binary.



Cluster Member List

Cluster #1

Power cable (1)
Voltage converter (2)
On/off switch (19)
Button (1 cup) (20)
Electric cable (22)
Indicating light (23)
Control unit (24)
Button 2 cups (25)

Cluster #2

Water reservoir (5)
One-way valve (9)
Water tube to heating element (10)
Water heating tube (11)
Hot water tube to dripper (12)
Water flow meter (16)
Water valve (17)

Cluster #3

Dripper (6)
Coffee holder (7)
Coffee tube (left) (18)
Coffee tube right (26)
Coffee valve (27)

Cluster #4

Heat conductive grease (3)
Heating element (4)
Temperature sensor (8)

Cluster #5

State sensor (21)

Cluster #6

Housing electrical elements (15)

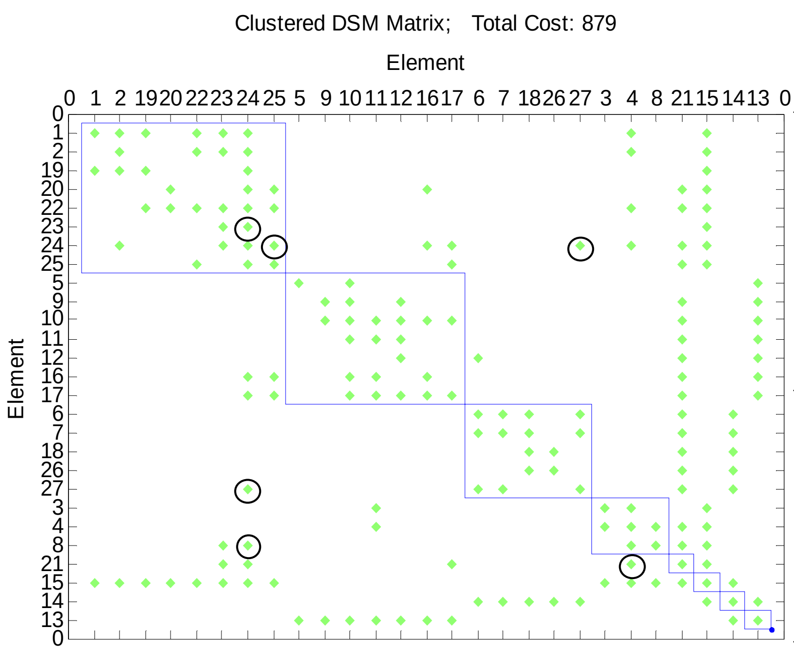
Cluster #7

Housing general (14)

Cluster #8

Housing of water (13)

Figure 8. CLUSTERED DSM AND CLUSTER LIST OF PARTIAL TOTAL DSM INCLUDING ACTION PLAN 1



Cluster Member List

Cluster #1

Power cable (1)
Voltage converter (2)
On/off switch (19)
Button (1 cup) (20)
Electric cable (22)
Indicating light (23)
Control unit (24)
Button 2 cups (25)

Cluster #2

Water reservoir (5)
One-way valve (9)
Water tube to heating element (10)
Water heating tube (11)
Hot water tube to dripper (12)
Water flow meter (16)
Water valve (17)

Cluster #3

Dripper (6)
Coffee holder (7)
Coffee tube (left) (18)
Coffee tube right (26)
Coffee valve (27)

Cluster #4

Heat conductive grease (3)
Heating element (4)
Temperature sensor (8)

Cluster #5

State sensor (21)

Cluster #6

Housing electrical elements (15)

Cluster #7

Housing general (14)

Cluster #8

Housing of water (13)

Figure 9. CLUSTERED DSM AND CLUSTER LIST OF TOTAL DSM, CONNECTIONS OF $\mathbf{P_2}$ HIGHLIGHTED

The second action plan enables the product to function when one of the coffee tubes is blocked. For the second action to be carried out plan the first action plan has to be already executed so the connections between the new components already are described in $\mathbf{P_1}$ and are not needed in the DSM of *action plan 2* ($\mathbf{P_2}$). For the second action plan, one of the important extra connections is a connection between the control unit and the heating element, in order to control the strength of the coffee. Other new connections in $\mathbf{P_2}$ include connections between the state sensor and the coffee tubes and a connection between the control unit and the coffee valve. The new connections of *action plan 2* are highlighted in Fig. 9.

The *total DSM* will contain the *normal DSM* and the *action plan DSMs*. The clustering of the partial total DSM ($\mathbf{T_p}$) including only *action plan 1* is shown in Fig. 8. The clustering of the *total DSM* including both action plans is shown in Fig. 9. Constructing the *partial total DSM* with only *action plan 2* is not possible because *action plan 1* is needed for *action plan 2*.

It follows from the clustering of the *partial total DSM* in Fig. 7 that the clustering remains the same with respect to the clustering of the original coffee maker in Fig. 6; only the new components are added to the clusters. Because more connections are included in the DSM, the total cost of the clustered matrix is also higher.

The clustering of the *total DSM* (with *action plan 2*) is not different from the clustering of the *partial total DSM* with only *action plan 1*, although extra connections were added. The new connections did not cause a change in optimal clustering. This means the division in clusters is optimal in such a way, that a few extra connections, even outside the current clusters, do not change the clustering. So implementing *action plan 2* does not cause a significant design change in the product architecture, although the control of the product does change significantly.

The connections between the components were not weighted for this simple example. When the connections would be weighted, the outcome of the clustering algorithm might be different. The outcome of the clustering algorithm now was a division in clusters that is intuitively logical. The example shows only the basics of the proposed method, not yet including any of the possible extensions.

The example given is a bit of a special case because the action plans incorporated in the product are in this case the same as just extra functionality. Further research will have to show the use of the method in large-scale complex products.

CONCLUSION

Adaptability is a relatively new concept in design. A standard design methodology has not yet been developed for adaptable products. The concept of product adaptability is not yet fully understood, since a standard definition of the concept is still lacking. In this paper a method was developed to deal with some aspects in the product architecture of the adaptable product. The method is especially useful for understanding the connections and for documenting the connections between components in an adaptable product.

In the example, especially the usefulness of the method is shown in the addition of extra components due to new, adaptable, functionality of the product. The weights of the different action plans have not yet been taken into account. This becomes more important in products that are more complex.

The method that was developed is intended for large-scale complex products. When the connections in such a product are investigated, the rating of the connections is more important than with a small product like the coffee maker from the example. The rating of those connections will require expert knowledge of multiple engineers. In this case, the choice of the weight factors of the different action plans is becoming important. In a complex product more components have to be included and the connections are more complex. The clustering will be different depending on the subjective rating of the interactions and the weight. Finally the clustering routine will influence the outcome of the clustering procedure.

Research has to show to what extent the rating of the interaction and the weights of the adaptability action plan DSMs influence the clustering of components. Aside from the interconnections between components, there might be different reasons for some components to be in the same module. Those reasons have to be incorporated into this method by the engineers. It is recommended for the use of this method, to use different weights and compare the outcomes of the clustering routines. In this way, the engineers using the method can gain insight in the influence of the weights and in the connections between components and clusters. The engineer can subsequently make a decision upon the most profitable clustering and thus the best weights to be used for that product.

ACKNOWLEDGMENTS

This work has been carried out as part of the OCTOPUS project with Océ-Technologies B.V. under the responsibility of the Embedded Systems Institute. This project is partially supported by the Netherlands Ministry of Economic Affairs under the Bsik program.

REFERENCES

[1] P. Gu, M. Hashemian, and A. Y. C. Nee, "Adaptable design," *CIRP annals, International Institution for Production Engineering*, vol. 53, pp. 539-557, 2004.

[2] A. Olewnik, T. Brauen, S. Ferguson, and K. Lewis, "A framework for flexible systems and its implementation in multiattribute decision making," presented at ASME 2001 Design Engineering Technical conferences and Computers and Information in Engineering Conference, Pittsburgh, PA, 2001.

[3] A. Olewnik and K. Lewis, "A Decision Support Framework for Flexible System Design," *Journal of Engineering Desgn*, vol. 17, pp. 75-97, 2006.

[4] C. Y. Baldwin, K. B. Clark, R. Division of, and S. Harvard Business, *Modularity in the Design of Complex Engineering Systems*: Springer, 2004.

[5] T. K. P. Holmqvist and M. L. Persson, "Analysis and improvement of product modularization methods: Their ability to deal with complex products," *Systems Engineering*, vol. 6, pp. 195-209, 2003.

[6] T. R. Browning, "Applying the design structure matrix to system decomposition and integration problems: a review and new directions," *Engineering Management, IEEE Transactions on*, vol. 48, pp. 292-306, 2001.

[7] T. U. Pimmler and S. D. Eppinger, "Integration analysis of product decompositions," presented at ASME Design Theory and Methodology Conference, Minneapolis, MN, 1994.

[8] T. L. Yu, A. Yassine, and D. E. Goldberg, "A genetic algorithm for developing modular product architecture," presented at ASME, 2003.

[9] V. B. Kreng and T.-P. Lee, "Modular product design with grouping genetic algorithm--a case study," *Computers & Industrial Engineering*, vol. 46, pp. 443-460, 2004.

[10] R. I. Whitfield, J. S. Smith, and A. H. B. Duffy, "Identifying Component Modules," *Seventh International Conference on Artificial Intelligence in Design AID'02*, pp. 15–17, 2002.

[11] J. Lancaster and K. Cheng, "A fitness differential adaptive parameter controlled evolutionary algorithm with application to the design structure matrix" *International Journal of Production Research*, vol. 99999, pp. 1 - 15, 2007.

[12] X. Xiaogang, L. Chao, Y. Jian, and C. Yahua, "An analytical method based on design structure matrix for modular identification," *Computer-Aided Industrial Design and Conceptual Design, 2006. CAIDCD'06. 7th International Conference on*, pp. 1-4, 2006.

[13] R. E. Thebeau, "Knowledge Management of System Interfaces and Interactions for Product Development Processes," in *System Design & Management Program*, vol. Masters: MIT, 2001.