

RETRIEVING ADAPTABLE SOLUTIONS FROM A DESIGN REPOSITORY

Mehmet H. Göker

Keywords: Case-Based Reasoning, Adaptation, Similarity, Design Repository

1 Introduction

In order to be able to create Case-Based Reasoning [1] systems that support the engineer in adapting a technical object to conform with the customer requirements, a measure of adaptability, i.e. a measure which shows how feasible it is to adapt a certain object to the new requirements has to be incorporated into Case-Based Design systems.

One major task in developing such a measure is to determine which attributes influence each other in which way. Model based approaches for this are only feasible for small problems and rather difficult to create and maintain in real design environments. On the other hand, experts are easily capable of estimating the magnitude and direction of influences between the relevant attributes of a system.

On the following pages, the influence matrix is used to analyze the dependencies between attributes of a technical object. Based on this analysis, the attributes of an object are classified into two groups: the independent (base) attributes and the dependent (derived) attributes. This classification is utilized to develop the 'Adaptation Matrix' which allows to predict the outcome of small adaptations for attributes with numeric values. It is shown how the elements of the adaptation matrix can be used to create weighting factors for a similarity metric that can be utilized to retrieve adaptable solutions.

2 Attributes and the Description of Technical Objects

2.1 Attribute Types and the System Boundary

In engineering design methodology [2], technical objects are described using attribute-value pairs, i.e. properties. Attributes are considered to be characteristic identifiers for the relevant technical object class [3]. If all attributes of an object class have fixed values associated with them, an instance or real technical object is described. The number and type of the properties that are needed to handle and describe a technical object are very much dependent on the context or the system boundary.

The system boundary separates the attributes that describe the inner workings of an object from the external attributes that are used in handling and utilizing it. It is dependent on the goal the user of the object has in mind and is thus subjective. Attributes that are hidden behind the system boundary for one application may be very relevant and thus visible for other purposes.


The attributes that can be directly modified are *independent* or *base* attributes of a technical system, the attributes that cannot be modified directly but can only be changed through other attributes are the *dependent* or *derived* attributes [4].

The classification of attributes into independent (base) and dependent (derived) attributes is always a function of the current system boundary and the specification level in which the technical object is being processed. To determine which attributes are dependent and which attributes are the independent under the given circumstances, the influence-matrix described in [5, 6] can be slightly modified and used.

2.2 Determination of the Independent and Dependent Attributes with the Influence-matrix

The influence-matrix is a powerful tool to analyze the dependencies between the parameters of a complex technical object. Table 1 shows an influence-matrix for a system with four parameters P_1 to P_4 .

Table 1: Sample Influence Matrix

	P_1	P_2	P_3	P_4	AS
P_1		3	3	0	6
P_2	2		3	0	5
P_3	0	2		0	2
P_4	0	1	1		2
PS	2	6	7	0	

The fields of the matrix are filled out by evaluating the influence of a parameter in the first column of each row on the parameters in each column and entering a corresponding value into the respective field. It is imperative to keep in mind the goal of the user and the context while estimating the influences. The influence is directed and not symmetric. A very high influence is represented by 3, high influence with 2, low influence with 1 and no influence with 0 points. In the example shown in Table 1, P_1 has a high influence on P_2 and P_3 , but no influence on P_4 etc.

In order to capture design-relevant information we extend the influence matrix and add the direction of the influence as a prefix (i.e. if P_1 had a high inverse influence on P_2 we would enter -3 in the corresponding field in the matrix). To be able to account for exponential influences we multiply the influence value with the factorial of the exponential (i.e. if P_1 has a high influence to the third power on P_2 we would enter $3 \times (3!) = 18$ in the corresponding field in the matrix).

The sum of the absolute values (i.e. without directions) over each row yields the *active-sum* (AS), the sum of the absolute values over the columns the *passive-sum* (PS) for each system parameter. The higher the active-sum of a parameter is, the more influence it has on other parameters, the higher the passive-sum of a parameter is, the more it is influenced by others.

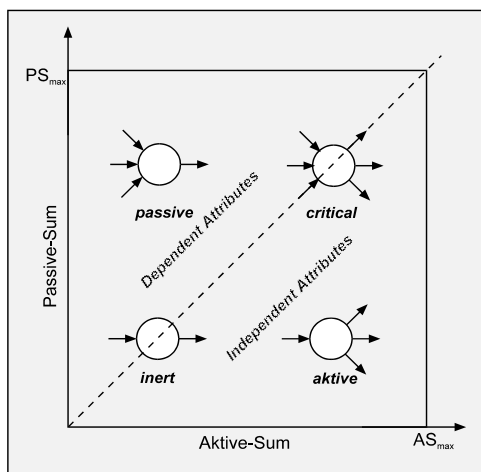


Figure 1: Regions in the Influence Graph

By placing each parameter in a graph with the active-sum as the ordinate and the passive sum as the abscissa (Figure 1), statements regarding the behavior of each parameter within the system can be made. Critical parameters are significantly influenced by and significantly influence other parameters. These attributes are mostly temporary attributes of complex nature that are re-used in other processes. While active parameters significantly influence other parameters, passive parameters are being influenced. Inert parameters have no interaction with other parameters. In the example given above, P_2 would be a critical, P_1 an active, P_3 a passive and P_4 a inert parameter of the system. The attributes that are below the angular bisector of the influence graph are independent (base) attributes for the given context, whereas the attributes above the angular bisector are dependent (derived) attributes. In the example above, parameters P_1 and

P_4 would be base and P_2, P_3 derived attributes. The angular bisector indicates the system boundary under the given circumstances.

While it is almost impossible to determine the dependencies between attributes of a real-world complex system in a mathematically exact manner, domain experts are able to estimate the influences between attributes of a system, as well as the direction and magnitude of this influence. This allows them to perform an influence analysis and judge the adaptability of an object rather intuitively where models of the system are non-existent.

3 Application of the Influence Matrix in Case-Based Reasoning Systems

3.1 Independent and Dependent Attributes in Case-Based Reasoning

Customers typically base their requirements on dependent (derived) properties. An end-customer asks for 'Price' or 'Extras' and does -under normal circumstances- not bother with the alternative manufacturing processes that influence the price. These attributes remain hidden behind the system boundary. A request for an object is typically based on dependent (derived) properties.

The task of an engineer is to select, modify or create the independent properties that result in the dependent properties of the technical object. This surfaces in the problem of similarity vs. adaptability in case-based reasoning. While the customer wants the end-product to be similar in the dependent properties the engineer wants to know which properties to modify in order to achieve this 'similar' product. The attributes that the engineer has to modify are the independent (base) attributes. However, a similarity calculation that is based on the dependent attributes does not necessarily give the most adaptable solution. The problem with adaptation is, that one has to cross the system boundary in order to perform it.

3.2 The Adaptation Matrix

The analysis with the influence matrix results in values that represent the influence levels of the independent attributes on the dependent attributes. The factors in the influence matrix do only give the rate and direction of change a modification in one attribute causes in the others. As this influence is represented to be linear, we are only able to make statements regarding the effects of small changes in the attributes.

If we define the incremental changes in the independent attribute B_m and the dependent attribute D_n to be

$$b_m = \frac{\Delta B_m}{|B_{\max} - B_{\min}|} \quad (\text{Eq. 1})$$

$$d_n = \frac{\Delta D_n}{|D_{\max} - D_{\min}|} \quad (\text{Eq. 2})$$

where B_{\max}, B_{\min} and D_{\max}, D_{\min} are the maximal and minimal values of the attributes and ΔB_m and ΔD_n are small changes, the following equation (Eq. 3) can be defined between the incremental changes in the independent and dependent attributes of a system.

$$\begin{Bmatrix} d_1 \\ d_2 \\ d_3 \\ \dots \\ d_n \end{Bmatrix} = \begin{bmatrix} a_{11} & a_{12} & \dots & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & \dots & \dots \\ a_{31} & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & \dots & a_{nm} \end{bmatrix} \begin{Bmatrix} b_1 \\ b_2 \\ b_3 \\ \dots \\ b_m \end{Bmatrix} \quad (\text{Eq. 3})$$

$$a_{ij} = \frac{A_{ij}}{\sum_{k=1}^m |A_{ik}|} \quad (\text{Eq. 4})$$

The elements of the matrix, a_{11} to a_{mn} are determined as shown in Eq.4. In this equation, A_{ij} are the elements of the influence matrix that correspond to the influence of the dependent attribute D_i on the independent attribute B_j . Note that not all elements of the influence matrix are used, but only the values corresponding to the columns of the independent attributes on the lines of the dependent attributes.

Using this matrix the effects of small changes in the independent attributes on the dependent attributes can be predicted and small, linear adaptations can be performed.

3.3 The Adaptation Matrix and the Calculation of Weighting Factors for Retrieval

During the retrieval of solutions from the case-base it is of utmost importance to retrieve the most adaptable solution rather than the one most similar in appearance. To do this, a measure for the adaptability of a case is needed.

One way of obtaining this information is to acquire and utilize knowledge regarding how to adapt attributes and how to resolve conflicts resulting from these adaptations [7]. However, even though this approach does yield exact results, there is a high overhead involved in acquiring, maintaining and utilizing the adaptation knowledge. Especially in domains where this adaptation knowledge is not readily available or the system is too complex to model, this approach will be very difficult to implement.

Another approach is to improve the standard similarity metrics used in retrieval with a weighting scheme that models adaptability [7]. This can be achieved by using the elements of the adaptation matrix as a set of weighting factors to capture the adaptability information in the similarity metric.

Assuming that the similarity of a case to a set of new requirements is going to be calculated based on the dependent attributes, we can define the global similarity metric to be:

$$Sim(q,c) = \{w_1 \quad w_2 \quad \dots \quad w_n\} \left\{ \begin{array}{c} \delta_1 \\ \delta_2 \\ \delta_3 \\ \dots \\ \delta_n \end{array} \right\} \quad (Eq. 5)$$

This equation calculates the similarity between the requested, ideal object q and an existing design c from the case-base. δ_n is the similarity metric that is applicable for the dependent attribute D_n , w_n the weighting factor for this attribute, and n the number of dependent attributes that are used to calculate the global similarity. Sim calculates the overall similarity based on the local similarity measures δ_n for each dependent attribute D_n . δ_n compares the value for the attribute D_n in the objects q and c and determines a similarity value between 0 (dissimilar) and 1 (identical) [8].

Having a closer look at the formulas for the values for b_n and d_n in (Eq. 3) we see that they can actually be considered to be simple, linear similarity functions.

If we replace d_n in (Eq. 3) with δ_n , and b_n with similarity metrics applicable to the independent attributes (β_m) we get the following equation.

$$Sim(q, c) = \left\{ w_1 \quad w_2 \quad \dots \quad w_n \right\} \begin{bmatrix} a_{11} & a_{12} & \dots & \dots & a_{1m} \\ a_{21} & a_{22} & \dots & \dots & \dots \\ a_{31} & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ \dots & \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & \dots & a_{nm} \end{bmatrix} \begin{Bmatrix} \beta_1 \\ \beta_2 \\ \beta_3 \\ \dots \\ \beta_m \end{Bmatrix} \quad (\text{Eq. 6})$$

This equation bases the similarity calculation with respect to the dependent attributes on the ease of adaptation using the base attributes. The weighting factors $w_1..w_n$ set the relative importance of the dependent attributes to each other whereas the elements of the adaptation matrix $a_{11}..a_{nm}$ are the weighting factors for the base attributes.

3.4 Example: Attributes of an Object made of Expensive Material

Let us assume, the object shown in Figure 2 is made of a expensive material and that the attributes that are relevant under the given circumstances are:

- | | |
|----------------------------------|--|
| a : Depth and length of the base | h : Height of base |
| r : Radius of hole | w : Width of the groove |
| d : Depth of the groove | m : Mass of object |
| ρ : Density of material | ψ : Value per unit volume of material |
| V : Volume of object | Q : Value of object |

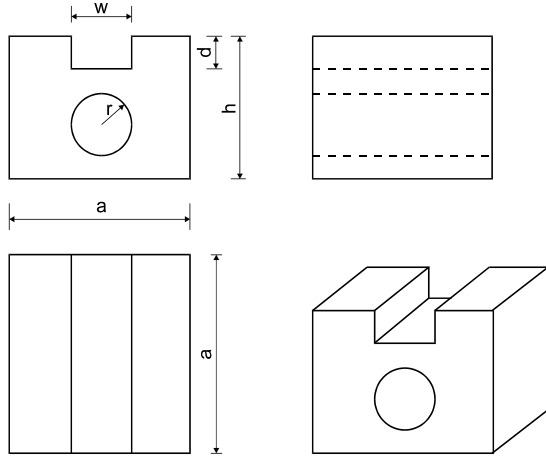


Figure 2: Object made out of expensive material

Table 2: Influence matrix for object made of expensive material

	a	h	d	w	r	V	m	Q	ρ	ψ	A S
a		0	0	0	0	9	9	9	0	0	27
h	0		0	0	0	3	3	3	0	0	9
d	0	0		0	0	-1	-1	-1	0	0	3
w	0	0	0		0	-2	-2	-2	0	0	6
r	0	0	0	0		-4	-4	-4	0	0	12
V	0	0	0	0	0		3	3	0	0	6
m	0	0	0	0	0	0		0	0	0	0
Q	0	0	0	0	0	0	0		0	0	0
ρ	0	0	0	0	0	0	3	0		0	3
ψ	0	0	0	0	0	0	0	3	0		3
PS	0	0	0	0	0	19	25	25	0	0	

If we create a influence matrix for these attributes, we will get a result similar to the one shown in Table 2. The negative numbers show a decreasing effect (such as the effects of d, w and r on the volume, mass and price) , and the effect of a and r are squared due to their exponential influence on the volume and the mass of the object. For the example given above, the similarity measure would be :

$$Sim(q, c) = \left\{ w_V \quad w_m \quad w_P \right\} \begin{bmatrix} 9 & 3 & -1 & -2 & -4 & 0 & 0 \\ 19 & 19 & -19 & -19 & -19 & 0 & 0 \\ 9 & 3 & -1 & -2 & -4 & 3 & 0 \\ 25 & 25 & -25 & -25 & -25 & 25 & 0 \\ 9 & 3 & -1 & -2 & -4 & 3 & 0 \\ 25 & 25 & -25 & -25 & -25 & 0 & 25 \end{bmatrix} \times \begin{Bmatrix} \beta_a \\ \beta_h \\ \beta_d \\ \beta_w \\ \beta_r \\ \beta_V \\ \beta_P \end{Bmatrix} \quad (\text{Eq. 7})$$

where w_V , w_m and w_P are the weighting factors for the volume, mass and price of the object and β_a to β_ψ the similarity metrics for the independent attributes. Even though these weighting

factors have been determined based on a subjective estimation of influences, the resulting similarity metric is able to retrieve solutions that can be adapted to the requirements easier.

4 Conclusion

Based on the influence matrix of the attributes of a system, a method that allows to analyze the effect of small changes in the independent properties on the dependent properties is developed. The method can be used to measure adaptability as well as to predict the outcome of small adaptations for attributes with numeric values. It is also shown, how the elements of the adaptation matrix can be used as weighting factors in a similarity metric that retrieves adaptable solutions from a case-base.

References

- [1] A. Aamodt, E. Plaza., „Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches“, AICOM Vol.7 Nr.1, pp.39-59, March 1994
- [2] G. Pahl, W. Beitz, "Konstruktionslehre, Methoden und Anwendung", 4th edition , Springer Verlag, Berlin, Heidelberg, 1997
- [3] M.Göker, "Einbinden von Erfahrung in das konstruktionsmethodische Vorgehen ", Dissertation, Technische Hochschule Darmstadt, 1996, VDI Fortschrittsberichte, Reihe 1, Nr. 268, VDI Verlag, Duesseldorf, 1996
- [4] K. Ehrlenspiel, "Kostengünstig Konstruieren: Kostenwissen, Kosteneinflüsse, Kostensenkung", Berlin, Springer, 1985
- [5] F. Vester, "Ausfahrt Zukunft, Strategien für den Verkehr von Morgen, Eine Systemuntersuchung", W. Heyne Verlag, München, 1990 and the respective "Supplement", Studiengruppe für Biologie und Umwelt GmbH, München, 1990.
- [6] G. Probst, "Organisation: Strukturen, Lenkungsinstrumente und Entwicklungsperspektiven", Verlag Moderne Industrie, Landsburg, Lech, 1993
- [7] B. Smyth, M. Keane, "Experiments on Adaptation-Guided Retrieval in Case Based Design", in Case-Based Reasoning Research and Development, M. Veloso & A. Aamodt eds., Proceedings of the First International Conference on Case-Based Reasoning ICCBR95, pp313-324, Berlin, Springer Verlag, 1995
- [8] M. Göker, "Similarity and Evaluation of Technical Objects", Proceedings of the 11th International Conference on Engineering Design ICED-97, pp31-36, A. Riitahuhta ed., Schriftenreihe WDK25, Tampere, Finland 1997

Mehmet H. Göker
DaimlerChrysler Research and Technology 3
Machine Learning FT3/KL
P.O. Box 2360, D89013 Ulm
Germany
Phone: +49 731 5052856
Fax: +49 731 5054210
E-mail: mehmet.goeker@daimlerchrysler.com