Fuzzy Reasoning and Systems Thinking in a Decision Aid for Designers

John Darzentas, Jenny Darzentas and Thomas Spyrou
Research Laboratory of Samos
Dept. of Mathematics
University of the Aegean

Abstract

This paper presents a methodology for the development of a decision aiding system, where the designer is a computer systems designer. the problem space is human computer interaction and the decision aiding is to actually help the designer to choose amongst a number of models and methods available to him. A blending of Systems Thinking and fuzzy representation of meaning was used in the methodology.

Key words: Decision Aiding, Design, Human Computer Interaction, Soft Systems Methodology, Fuzzy Reasoning

1. Introduction

For designers of computer systems seeking help for usability problems, there exist two main sources of formalised assistance: guidelines and models.

Guidelines, such as those contained in in-house manuals provided by software houses to their design teams [2], military handbooks written to attempt to guarantee some standards in systems designed for use by military personnel, [8] or research papers suggesting principles derived from, for instance, empirical studies conducted to examine particular aspects of usability [3,11], etc. Guidelines are good starting points, reading them can alert designers to the presence of many issues, even on occasions of some alternative approaches, and designers are glad to have them. [12] However, by themselves guidelines are not sufficient. This is generally agreed upon, even by those writing them. Guidelines cannot deal with explicit design scenarios or with choices that are highly dependent on context, as many important choices in human interface design are.[9,13] In addition, the capabilities of technology change so fast, that some guidelines become very quickly out-of-date.

The other type of assistance comes from researchers who have concentrated on developing formal models for the design of components of the interface. Generally these require that the designers translate the design requirements into some intermediate language, examine this for inconsistencies and usability implications, and then eventually translate these statements into a programming language. In this way are caught many design flaws and inconsistencies that would otherwise persist further into the design process. A related use of these models is to compare alternate user interface approaches by analysing them in terms of the models.

The Amodeus project [1] has sponsored the development of several models of this type. including models about users, systems, tasks, as well as methods for structuring and capturing design commitment and rationale.

Once the design community has been alerted to the presence of these models and methods, collectively referred to as modelling techniques, there still remains the task of transferring modelling to the design community. A designer's decision aid was conceived as a means of assisting in this transfer, its objective to guide the designers to select the most appropriate modelling technique(s) for their design problem.

Since the modelling approaches can only be used as whole units and not be amalgamated at present, it was important to define what each approach can do in an operational way and devise a «language» for the designer to be able to correlate his problem to the abilities of the modelling approaches in order for the decision aiding system to evaluate the appropriateness of each modelling approach to designer problems.

The actual problem to be tackled by the decision aid is relatively complex, although not so difficult to express and generalise: namely the evaluation of modelling techniques to solve design problems in a multi-variable multi-parameter context which includes the designer problem relationship. The techniques are multi-disciplinary and although most of them can deal with most aspects of a generic design space at various levels, they all identify themselves as more efficient at specific areas of the design space than others.

A methodology for the elicitation of expert knowledge about the potential of models, and for representing the meaning of that potential to the client and from there recommending to the client the most suitable technique(s) is presented in the next section. It is based primarily on principles of Systems Thinking, and in particular Soft Systems Methodology(SSM) 5,6], for eliciting and structuring knowledge relevant to the generic design space and on fuzzy sets through test score semantics 15] for representing the meaning of relationships and for reasoning. Section 3 describes in more detail the role played by SSM in creating the system of relevant subsystems and in section 4, the application of test score semantics is given.

2. Developing an intelligent decision aid.

In the case presented here the decision maker is the designer of a computer system, the problem space is the design space of the human computer interaction subsystem, and the decisions to be aided relate to the choice amongst an array of modelling approaches available for assisting aspects of usability design. The decision space is understood to contain situations leading to decisions about the appropriate use of these modelling techniques in design practice. In other words decisions about which modelling techniques to use and where and when.

The task of identifying and defining the potential of these techniques to design was approached in a 'top down' fashion based on Checkland's Soft Systems Methodology (SSM) [5,6]. Use was made of rich picture diagramming to identify activity subsystems relevant to the modelling techniques [10]. The primary benefit of using such an approach is that of gaining insight from learning and debating about the problem situation i.e. the relationship between the design action situations in the design space and the modelling techniques.

The relationships between the relevant «activity» subsystems and the modelling techniques contain descriptions of the potential of modelling techniques in question in relation to a (sub)problem within the design space. These relationships make up «discourses» about what each modelling technique can do for that problem. The meaning of each of these discourses is evaluated using test score semantics [15] where the proposals contained in the discourse about a modelling technique are treated as fuzzy constraints.

Using the system the designers will, in effect, be led to express their problem in ways which enable the decision aid to match them with the most suitable modelling approach(es).

The problem encompasses the design space within which are situated the array of models and the designer with his design problem seeking assistance from a variety of avenues, one of which is interaction with the decision aiding system. This interaction results in a description of the problem in terms of the activity subsytems which link back into the models. However, this output most often shows that several of the models would have something to say about the problem in hand. Such output is not succinct enough to be of use to the designer. He requires more intelligent aid. He would like to know when and where a technique is useful. To refine output, the use of fuzzy reasoning , in the form of Test Score Semantics is used. The resulting output is given to the designer.

Figure 1 describes the process whereby the designer through his interaction with the decision aiding system produces a description of his design space. It is important to note that the designer's design space description is one way of expressing his problem, amongst many, which is driven by the modelling techniques. In other words the subsystems relevant to the problem will be chosen from a set of subsystems which were identified (elicited) initially because of their relationships to these modelling techniques and to decisions relating to the use of these techniques. The design space (Figure 1) consists of sets of relevant activity subsystems directly relating to specific modelling techniques. The addition operator shown in figure 1 signifies that there are, as expected, overlaps amongst these sets of relevant subsystems. The design problem space description shown in network form in figure 1. is actually what is kept in the design problem space shown in figure 2. This is what is used to produce the recommendations for decision (via test score semantics).

The problem is defined, at the current level, by the designer on the basis of his interaction with the system. A design space description via relevant subsystems and their relationships to themselves and to the modelling techniques is provided by the design space knowledge module as seen in Figure 2 which shows a high level functional architecture of the system. The designer selects the set of subsystems relevant to his problem to provide the design problem space knowledge module. The design space module is based on representations in a frame - rule based environment. Sets of rules and metarules manage the relevant subsystems network. The design problem space is basically retrieved from the general design space contained in the first knowledge module and it is the one used by the decision models module.

Note that modelling techniques are considered as whole units, in other words the modelling techniques are not broken down into operational parts nor there is any attempt to redefine amalgamated techniques from parts of others. The relationships are identified and defined between subsystems of the design space and whole modelling techniques.

The decision models module contains the function which performs the decision aiding process based on fuzzy sets and in particular on test score semantics. This module, having the set of relevant subsystems selected by the designer describing his problem, requests that all the links (constraints) between the subsystems and the modelling techniques etc. are evaluated either by the designer or retrieved from a data base with expert evaluations for some of the links or both. Then following the test score semantics approach this module proceeds with the aggregation of these evaluations followed by the defuzzification of that aggregation towards the final recommendation.

The interface module controls the interfacing between the user (designer) and the system. It contains six functions which are responsible for displaying the current problem representation, browsing and inputting user selections/rejections, evaluations of subsystems and constraints (relationships); and communicating the final recommendations.

Operationally the system works as follows: The designer is requested by the system to identify and describe his problem by selecting a subset of relevant subsystems and corresponding relations from the overall system of subsystems (figure 3.) representing the design space. This is achieved by the functions 3.1, 3.2, 3.3 and 3.4. His selections now form the current description of the problem, and this is carried out by functions 1.1 and 1.2. This current problem description consists of the selected relevant subsystems and their relationships. On the basis of those the system through function 2.1 proceeds to ask the designer to evaluate the relationships (links), now taken as fuzzy constraints. Function 3.5 inputs these evaluations. The system at this point through function 2.2 continues applying the test score semantics approach by aggregating the partial scores (evaluations) and defuzzifying the overall test score through function 2.3. This is finally communicated to the designer via function 3.6.

To summarise, the user is asked to express his area of concern through the relevant subsystem based description of the design space. This is achieved by highlighting those relevant subsystems which he identifies as pertinent to his concern. These relevant subsystems will show their links to the modelling techniques. The links are stored in the knowledge base at varying levels of resolution. The user is required to evaluate with a score those links in relation to how much they (taken now as constraints) are satisfied.

3. System of Relevant Subsystems

The system of relevant activity subsystems is the main vehicle for providing a representation of the design space useful for the purpose of aiding the designer in his decision making as to which modelling approach(es) to use to tackle his problem.

This system is defined here as the space which consists of activity subsystems S_j, and their relationships as follows:

$$[S_j, \mathbf{R}_{S_j}^{mt_i}, \Diamond, mt_i, x] \qquad\qquad [S_j, \mathbf{R}_{S_jS_k}^{mt_i}, S_k, mt_i, x]$$

where, S_j is the activity subsystem j, j = 1..N, mt_i is the modelling technique i, i = 1..7 currently, $\mathbf{R}_{S_j}^{mt_i}$ is the relationship identified within S_j in relation to modelling technique i. ($\Diamond$ denotes that the relationship is actually an attribute of S_j which stems out of the properties of mt_i).

$\mathbf{R}_{S_k}^{mt_i}$ is the relationship between S_i, S_k again in relation to mt_i, usually corresponding to pre and post-conditions. Finally x is an empirical measure of how much $\mathbf{R}_{S}^{mt_i}$ is satisfied by mt_i.

The relevant activity subsystems S_j are elicited using (SSM). Figure 4 gives a representation of the formal definition of this system.

As an illustration, figure 3 displays part of a conceptual model of the design space as «extracted» from the texts and communications [4,7,14] of those involved in the attempt to provide an environment for decision aiding. It describes the relevant subsystems of the design space, as activity subsystems, always in relation to the problem of choosing modelling approach(es) amongst those available, to apply to solve specific design problems.

The conceptual model presented in figure 3. consists of a number of activity subsystems and their relationships to the modelling approaches. There are also implied relationships between the activity subsystems identified with respect to the modelling techniques, usually in the form of pre/post-conditions. Figure 5 shows a subset of the relevant activity subsystems identified for the PUM (Programmable User Modelling) and CTA (Cognitive Task Analysis) modelling approaches after the application of SSM and through the rich pictures of those techniques as described elsewhere [7].

Figure 3 actually evolved from figure 5 by grouping the activities identified accordingly: for instance the highlighted activities in the form of the «is applied» activity subsystem in figure 3.

The system of relevant activity subsystems is operationally useful in the sense that although the relevant subsystems cannot be mutually exclusive in a mathematical sense, they should be individual semantically and operational so the user can usefully distinguish amongst them. Also they should cover the design space as much as possible.

Presentation of the relevant subsystems and their links will form a «picture» which is shown to the user initially when he is requesting aid for his decisions as to which modelling approach(es) should he consult at his current stage of design. The contents of the conceptual model in that «picture» is also a knowledge model representing knowledge about the design space in general with the possibility of introducing problem specificity.

A form of this «picture» given in Figure 6 which shows the content of figure 3 where each activity subsystem is expressed with a question which represents a more designer problem oriented view. The question of decision aiding arises when more than one modelling technique are associated to the designer's problem as expressed through figure 6.

Consider the case where a designer using the system is shown figure 6. The designer is asked to express his problem by answering the questions corresponding to subsystems he considers as relevant to his concern. Figure 7 shows a hypothetical subset of figure 3 selected by a designer showing the corresponding relationships considered as relevant to his problem after interaction with the system.

The interaction between the designer and the system to produce the results shown in Figure 7 would be carried out as follows. When the designer selects certain subsystems from figure 6, the system, by activating a set of rules, determines which relationships will have to be verified by questioning the designer. It is important to note that each attribute is associated to one or more activity subsystems, only in relation to modelling techniques, here $R^{mt_1}_{AB_1}$ and $R^{mt_2}_{AB_2}$

The resulting attributes associated with subsystems selected by the designer, can be separated into groups according to the modelling techniques they are associated to. Each of these groups can now be evaluated in order to provide recommendation as to which modelling technique(s) are more appropriate to be used for the particular problem. The evaluation of these groups of attributes is carried out with the aid of test score semantics and is described in the next section.

4 Application of test score semantics

Test score semantics have been introduced by Zadeh [15] for representing the meaning of semantic entities such as propositions, which are viewed as sets of fuzzy (elastic) constraints, which in turn implicitly constrain certain linguistic variables in the propositions.

The representation of meaning of a proposition through test score semantics consists of the following steps:

a) identification of the variables in the proposition; b) identification of the constraints induced by the proposition; c) association of a test score to each constraint representing the degree to which that constraint is satisfied; d)Aggregation of the partial scores into a smaller number of scores or a scalar.

In the present work the proposition p of interest may be:

p: Amongst the array of the available modelling techniques there are some which tackle the problem in its present form; etc.

or even:

p: I need the most appropriate modelling technique for my problem at this stage.

It is assumed that p consists of a collection of a number of implicit fuzzy constraints which here are the relationships R between the relevant activity subsystems and the modelling techniques. Also one obvious variable relevant here is «modelling technique». It is considered that the constraints induced by p are expressed by the relationships R which as fuzzy relations are kept in an explanatory database [15].

Conceptually what is achieved is the evaluation of discourses relevant to a design problem, through a representation of their meaning. The different discourses correspond to different modelling techniques to approach the same design problem i.e. to different values of the linguistic variable «modelling technique». As a result the evaluations may provide the designer with a «picture» of discourse which is more relevant to the present design problem and hence recommendable at this stage.

The relationships R are usually expressed in text form and are considered as a collection of fuzzy constraints. That is a number of fuzzy relations constituting the meaning of the relationship between the subsystem Sj and the modelling techniques in terms of relevancy of those modelling techniques identified and S_j.

Assume that the user requesting decision aid has settled with the set of subsystems shown in Figure 7, as being relevant to his concern at the current stage of design. The real decision problem will be to evaluate the meaning of the usefulness of each modelling technique which through the relationships to the subsystems selected by the user, will appear suitable. Following the test score semantics procedure each relationship (fuzzy constraint) is evaluated in the explanatory database on the basis of specific templates for the relations. The user (designer) may also need to provide a score ts_i for each relationship, which will describe the degree to which according to him (and not the expert who compiled the database) the relationship is satisfied. These two scores per constraint may be combined to one via the minimum or other operator.

According to this approach his test scores assigned to every relationship will give overall aggregated test scores for the groups of attributes discussed in the previous section, which correspond to each modelling technique. The highest of these overall test scores may be taken as a very good indication that the corresponding modelling technique is the most appropriate currently.

However it must be noted that the suggested approach is an attempt to evaluate the meaning of relationships in terms of a proposition expressing concern. In other words it is an attempt to identify the most «meaningful» action to be taken by the designer in terms of using a modelling technique to proceed with solving his problem. In that context it is worth mentioning that Zadeh suggests that the overall score by itself does not represent the meaning of the proposition of concern, but one has to consider the actual process leading to that score. As a result the overall scores here cannot always reflect the appropriateness of a modelling technique over another in relation to a design situation. For example a modelling technique may be moderately appropriate but it may satisfy (moderately) a large number of links (fuzzy constraints), while another one may strongly satisfy one or two constraints only. Usually the fact that only a few constraints are very much satisfied, is enough to overpower the case of the great number of constraints moderately satisfied in a fuzzy environment. In the case of fuzzy quantifiers the aggregation of the partial scores is achieved using the sigma-count of «modelling technique» Σcount(mod. tech.), where small partial scores are ignored. Zadeh gives a number of standardised rules for the aggregation of the partial scores emphasising also that aggregation could be left to the discretion of the constructor of the test procedure.

The importance of the approach is that it offers a more natural way to evaluating options within a problem situation, in this case a design (sub)problem, by maintaining the more natural linguistic expressions relating to the problem. The choice of the most appropriate aggregation is a matter of experimentation in the actual design environments, allowing always, if possible, for subjective intervention by the designer(s).

$[A, \mathbf{R}_{A_1}^{mt_1}, \Diamond, mt_1, x]$	$[ts_1]$	badly
$[A, \mathbf{R}_{A_2}^{mt_1}, \Diamond, mt_1, x]$	$[ts_2]$	not so well
$[A, \mathbf{R}_{A_3}^{mt_2}, \Diamond, mt_2, x]$	$[ts_3]$	
$[B, \mathbf{R}_{B_1}^{mt_1}, \Diamond, mt_1, x]$	$[ts_4]$	very well
$[B, \mathbf{R}_{B_2}^{mt_2}, \Diamond, mt_2, x]$	$[ts_5]$	
$[A, \mathbf{R}_{AB_1}^{mt_1}, B, mt_1, x]$	$[ts_6]$	
$[A, \mathbf{R}_{AB_2}^{mt_2}, B, mt_2, x]$	$[ts_7]$	

Table 1

In the case presented in Figure 7 the partial scores t_1, t_2, t_3 corresponding to the attributes of A and B with respect to modelling technique 1 are given in table 1. Here the minimum operator aggregator has also been used for the aggregation.

Hence for each modelling technique having a connection through a number of relationships to designer's problem, there is an overall score. The highest of these scores is an indication that the corresponding modelling technique would be the most appropriate.

5. Summary and Conclusions

This paper presented a methodology for decision aiding, where the designer is a computer systems designer, the problem space is human computer interaction and the decision aiding is to actually aid the designer to choose amongst a number of modelling techniques available for designing interfaces.

A blending of Systems Thinking and fuzzy representation of meaning was used in a methodology for developing an intelligent decision aiding system for designers of computer systems and especially of human computer interaction subsystems.

Systems Thinking offered a means to operationally define the subsystems relevant to the generic design space. The links of those subsystems to the modelling approaches were elicited primarily from relevant texts and communications with experts. In this way the primary purpose of «translating» the design space into a mode where the role of the modelling approaches in the design space is described in an operational way was achieved. Then the application of test score semantics can evaluate that role in the part of the design space selected by the designer.

Finally the system makes recommendations to the designer on the basis of the evaluations.

The evaluation of the links between the subsystems and the modelling approaches by treating them as fuzzy constraints allows for greater flexibility in representation and also sustains considerably more of the designer's understanding of his problem in order to be able to aid him in his decision for choice.

The approach is generic in the sense that it can be applied to develop intelligent decision support in many problem areas where the correspondence between tools/methodologies and problems must be established and evaluated in order for the problem solvers/decision-makers to decide *what* they should use to tackle their problems and *when* and *where*.

Acknowledgements: *This work was funded by the ESPRIT Basic Research Action 7040 AMODEUS (Assaying Means of Design Expressions for Users and Systems)*

References

[1] AMODEUS: ESPRIT Basic Research Action 3066 1989-1992 AMODEUS (Assimilating Models of Designers Users and Systems) and ESPRIT Basic Research Action 7040 AMODEUS 1992-1995 (Assaying Means of Design Expressions for Users and Systems) Documentation available by anonymous ftp (ftp.mrc-apu.cam.ac.uk).

[2] Apple Computer Human Interface Guidelines: the Apple Desktop interface Addison-Wesley, 1987.

[3] Bidgoli, H. Designing a user-friendly interface for a decision support system Information Age, pp.148-154, 1990.

[4] Buckingham Shum, S., Jorgensen, A.H., Hammond, N., Aboulafia, A.(Eds), Amodeus-2 HCI Modelling and Design approaches: Executive Summaries and Worked Examples Amodeus Project Document: TA/WP16, 1994.

[5] Checkland, P.B., Systems Thinking, Systems Practice. Wiley, 1981.

[6] Checkland, P.B., Scholes, Soft Systems Methodology in action. Wiley, 1990.

[7] Darzentas, J., Darzentas, J.S., Spyrou, T., Defining the Design «Decision Space»: Rich Pictures and Relevant Subsystems. Amodeus Project Document: TA/WP 21, 1994.

[8] Department of Defence Human engineering procedures guide, Military Handbook Washington DC MIL-HDBK-XXX , August 1986

[9] Gould, J.D. How to Design Usable Systems Handbook of Human-Computer Interaction M.Helander (ed) Elsevier 1988

[10] Lewis, P.J., Rich Picture Building in the Soft Systems Methodology. *European Journal of Information Systems*, Vol 1, no.5, 1992, pp. 551-360.

[11] Merchant, A.N. Expert Systems: a design methodology pp21-41, Vol 16, 1992 Computers, Environments and Urban Systems.

[12] Moisier, J.N Smith, S.L. Application guidelines for designing user interface software *Behaviour and Information Technology* 5, pp. 39-46, 1986

[13] Nielsen, J. The Usability Engineering Life Cycle, *IEEE Computer* Vol 25, No 3, pp. 12-22, 1992

[14] Young, R., Buckingham Shum, S., Duke, D., Paterno, F., May, J., Bellotti, V. Thoughts and Questions about Transfer and Assaying. Amodeus Project Document TA/IR5, 1994.

[15] Zadeh, L.A., Knowledge Representation in Fuzzy Logic, IEEE Transactions on Knowledge and Data Engng, 1, (1), pp. 89-100, 1989.