Decision-Theoretic Evaluation of Design Choices in a Case-Based Environment

Qing Cheng and Costas Tsatsoulis
Center for Excellence in Computer Aided Systems Engineering
Department of Electrical Engineering and Computer Science
The University of Kansas, Lawrence, KS 66045, USA
tsatsoul@eecs.ukans.edu

Abstract

We present a unique methodology that combines case-based design, explanation-based reasoning, and decision theory for the evaluation of design alternatives. Our technique uses: (1) case-based reasoning to generate design alternatives, (2) explanation-based reasoning to assign subjective probabilities to the possible values of missing design specifications, and to assign subjective utilities based on multiple criteria to the design alternatives, and (3) decision theory to find the best alternative. Our technique integrates case-based design and decision theory, and offers a way to deal with missing and incomplete design specifications and to document and explain design decision making and intent. The methodology has been implemented in an intelligent, case-based system in the domain of pharmaceutical design.

Introduction

Design specifications are often high-level, vague, conflicting and incomplete, making design decision making difficult and error prone. The goal of our work is to provide an intelligent assistant to the designer that will help during the evaluation of design choices. The assistant should accept incomplete, high-level specifications, generate potential design alternatives, evaluate them using a systematic approach, and then present and justify its rationale. We are using a case-based design formalism to generate design alternatives, and techniques adapted from decision theory to provide a systematic method to evaluate choices. We have implemented an explanation-based technique which assigns probability to events, and utility to outcomes. Finally, the intelligent assistant can justify its choices and assist the user in making the best decision between competing design alternatives. Our methodology has been implemented in a case-based design assistant that helps chemists design pharmaceuticals. Drug design is an appropriate domain of application: The number of possible compounds that have to be explored during the design phase is enormous, and the evaluation of design choices is extremely important, since it allows the chemist to focus on a small subset of such compounds. The interactions of chemicals or their effect on the human body are often not known and the specifications are necessarily incomplete. Finally, a compound may have multiple effects, both positive and negative, and its usefulness in a specific situation needs to be evaluated by weighing the utility of the drug to the target population versus the predicted risks.

This work was supported in part by a Kansas Technology Enterprise Corporation grant to the Center of Excellence for Computer-Aided Systems Engineering of the University of Kansas.

A Brief Introduction to Expected Utility Models

To facilitate the comprehension of our methodology, we present here a very brief overview of expected utility theory, concentrating on the parts of it that are used in our model. For details see (Raiffa 1968) and (French 1988). Decision theory emphasizes making a choice among a set of alternatives. The criterion for optimal choice is the maximum *expected utility* of the projected outcomes, which would allow the decision maker to select among them. There are two fundamental ways to approach utility theory: a *normative model* or a *descriptive model.*

The descriptive model of utility theory conjectures how things are or are behaving (French 1988). There are two broad families of descriptive models: semi-orders and probabilistic choice models. Semi-orders attempt to model intransitivities of indifference that arise from inability to discriminate finely between options. The probability choice models mirror the propensity of a decision maker to choose between two alternatives. In other words, in many situations a decision maker will not *always* choose a over b, but may sometimes choose a and sometimes b.

The normative model of utility theory describes how decisions should be made: given the probability of the events, that is the quantified probability distribution of uncertain states of nature, and the utility of the outcomes, that is the preference we have towards each alternative, we can calculate the expected utility of actions. Usually, the

79

probability and utility values are treated as subjective judgments, expressing the knowledge, experience, and intuition of the expert (Raiffa 1968). Subjective probability, which obeys the theorems of normal probability, is used to scale the uncertainty of events, and is the result of experience gained before the elicitation of the probability judgment, rather than experimental results of observed frequencies. Subjective or judgmental utility is used to scale preferences for possible consequences, and, like subjective probability, exists in the mind of the decision maker. Subjective utility is determined based on a *betting semantic*, which assigns a value between 0 and 1 to each consequence based on the decision maker's preference for achieving this outcome. Then, the *expected utility* of an action a_i can be calculated as:

$$E[u(a_i)] = \sum_{k=1}^{n} u(x_{i,k}) \times P(s_k)$$

where $E(u(a_i))$ is the expected utility of action a_i, $u(x_{i,k})$ is the subjective utility of outcome (i,k), and $P(s_k)$ is the subjective probability of event s_k.

While decision theory provides a rational framework for choosing among alternatives, it has several limitations. Descriptive models provide excellent representations of empirical evidence, but are limited in their use in fully automated systems. Normative models are rigorous, but judgment and preference have to be expressed in terms of probability and utility which is not always easy or possible, the generation and prediction of possible outcomes is not always feasible, and the process of selecting between alternatives cannot be adequately explained since it is purely mathematical (Kahneman et al. 1982; Scholz 1983; Wright 1984; French 1988). What is needed is a new approach to normative, bayesian utility models that would resolve some of their old inconsistencies.

Methodological Approach

In very general terms our methodology can be described as follows: Given some design specifications, a partial matcher retrieves cases which satisfy the known requirements. From the missing requirements we construct a decision model. Using domain-specific explanation models we assess the values of the subjective probabilities, and subjective utilities in the decision model. Finally, using simple decision theoretic techniques we compute the expected utilities, and retrieve the most appropriate case that satisfies the specifications. Further interaction with the user allows the system to explain its decisions and reasoning.

Specifications Definition and Case Matching

We predefine a template of all possible specifications related to the top-level design of a pharmaceutical. The user is asked to fill in as many of the attributes of a problem as possible; the collection of these attribute-value pairs forms the high-level specifications for the problem. The user is also required to specify which features are considered *important,* indicating that an exact match is required for these values. Next, the system matches the description of the problem to the cases in memory. The cases with a degree of similarity above a certain threshold are retrieved.

Consider the following example of a problem definition[1] (internal representations and code are simplified and translated in pseudo-natural language for brevity and understandability):

```
ID:              423
Type:            #<Standard-Class CASE>
Name:            NEW-PROBLEM
Purpose:         CARDIOVASCULAR
Behavior:        UNKNOWN
Features:
        Name:    THERAPEUTIC-GROUP
        Value:   CARDIOVASCULAR

        Name:    MODEOF-ACTION
        Value:   INHIBIT-PACEMAKER-DEPOLARIZATION

        Name:    ACTION-CLASS
        Value:   ANTI-ARRHYTHMIA-AGENT

        Name:    THERAPEUTIC-RANGE
        Value:   10

        Name:    EFFECT-TYPE
        Value:   NEGATIVE-IONOTROPIC

        Name:    PATIENT-CONDITION
        Value:   (NURSING RENAL-IMPAIRMENT)

        Name:    MARKET-DESTINATION
        Value:   LOW-INCOME
Important features: (LIPID-SOLUBILITY
                     ACTION-CLASS EFFECT-TYPE)
```

This problem defines the general class of the site where the drug must have an effect (cardiovascular), its action (inhibit pacemaker depolarization), the general category where the action belongs (anti-arrhythmia), its effectiveness (therapeutic range), how it should operate (by producing a negative ionotropic effect), the expected condition of the patient, and the income of the targeted market group. The BEHAVIOR of the drug is not yet known since it must be designed. This is an incomplete description of specifications. As we will see, other possible features would be "metabolism type" or "lipid solubility". As a matter of fact, "lipid solubility" has been defined by the user as in important feature, although the designer has made no attempt to define requirements for this value.

[1] All case descriptions are implemented in MEM-1, a generic CBR shell developed at the Center of Excellence for Computer-Aided Systems Engineering (CECASE) of the University of Kansas.

Building a Decision Tree

After the definition of the problem, the system will retrieve a set of cases that partially match the problem specifications. Then, the system uses the missing features to generate a decision tree. The missing features are used as decision points, and the outcomes of the retrieved cases are used as possible outcomes in the decision making process (for details see (Wei & Tsatsoulis 1993)).

In our example, the system retrieved four cases. The solutions of these cases become the possible outcomes of the decision tree; since we only know of four alternative pharmaceuticals that satisfy our specifications, these are the only ones we can evaluate. This "closed world assumption" is a way of solving one of the major problems of traditional decision theory, that of the generation of outcomes. In decision theory the outcomes of actions have to be predicted or generated by simulation, and this is usually a very difficult problem. This problem is resolved by assuming that all possible outcomes have to be members of our case base.

The decision variables are the missing problem features (specifications), and their possible values are the values these attributes take in the cases retrieved. This is again a closed world assumption, and helps resolve the problem decision theory systems face with trying to define all decision variables. The number of decision variables can be so large that it could be impossible to select the right variables and construct a decision model; by limiting the variables and the values they may take to the known ones, we help alleviate this problem.

In our example we retrieved four cases, which provided the system with four possible outcomes. We also had two missing attributes. The decision tree is shown in figure 1:

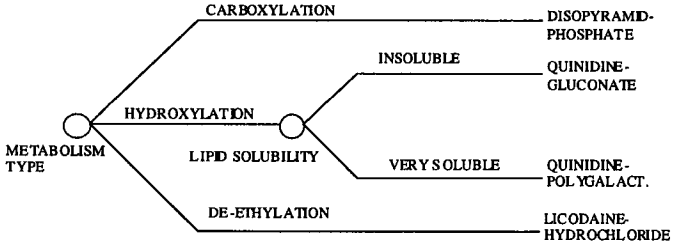

Figure 1: Initial example decision tree

The outcomes at the end of the tree indicate the compounds used in previous designs. There are no decision points, since we are not taking any actions; just chance nodes, where an unknown or undefined specification (here "metabolism rate" and "lipid solubility") could take one of a number of values. Now we need to assign subjective probabilities and utilities to allow us to make decisions between the four alternative outcomes.

Assigning Subjective Probabilities and Utilities

Assigning Subjective Probabilities. We use a simple probability model to assess subjective probabilities of the uncertain features at the chance nodes of the decision tree. The probability of the value of an uncertain feature is based on two factors: (1) Raw probability, defined as the occurrence of a feature value in all categories of a pharmaceutical, and, (2) Similarity scaling, which associates frequency of occurrence with the similarity metric of a case. We assume that the more frequently a value of a feature occurs in the case base, the more *possible* it will be that it will occur as part of the solution to the new problem. We also assume that the more similar a case is to the current problem, then the more possible it will be for the feature to take a value from the old case. The subjective probability of a feature value is:

$$P(v_i) = \frac{RP(v_i) \times S_i}{\sum_{j=1}^{n} RP(v_j) \times S_j}$$

where $P(v_i)$ is the subjective probability of feature value v_i, RP is its raw probability, S_i the similarity value of the case that contained value v_i, and the denominator is the sum of all other sibling nodes of v_i, including itself. The value for $P(v_i)$ follows the rules of subjective probability, since for all possible values of an uncertain feature the sum of all subjective probability values is 1 (since we assume the set to be exhaustive). The new decision tree is shown in figure 2.

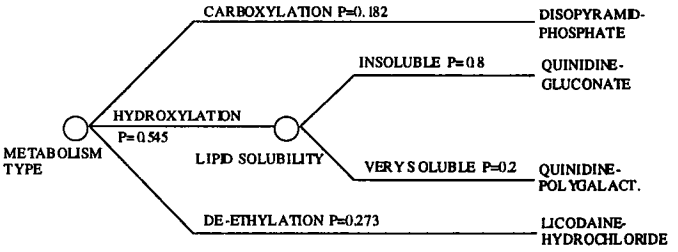

Figure 2: Decision tree after subjective probabilities have been assigned.

Assigning Subjective Utilities. The designs of the retrieved cases become the outcomes of the decision tree. The utilities of the outcomes are assigned using explanation-based reasoning and multi-attribute value functions. Our system defines the criteria and objectives which need to be evaluated, and then explains and predicts the behavior of the compound across these evaluation dimensions. The criteria can be concrete, for example a salient tangible feature such as "degree of solubility", or intangible and generic, for example a concept like "marketability", which requires substantial explanation. We assume that the user of the system will wish to evaluate a compound based on many criteria, and we use the theory of multi-attribute values, to reach a decision (Roberts 1979).

The explanation model is constructed so that it combines multiple influence factors with an objective concept. Each influence factor provides a preference value over the outcomes of all cases, and helps to partially explain an objective. The influence factors express the sufficient conditions for the satisfaction of an objective, and are mutually exclusive. In our system, explanation rules describe

the qualitative relationship between an influence factor and a concept or objective that needs to be explained. We have defined and implemented six different types of explanation rules (In the following F is a factor influencing a supporting objective and O is the objective that needs to be explained):

```
1. Incremental
       F increase -> O increase
       F decrease -> O decrease
2. Decremental
       F increase -> O decrease
       F decrease -> O increase
3. Existent
       F exists -> O increases/decreases
       F is f_i -> O increases/decreases
       F is in range (...) ->
              O increases/decreases
4. Value Order
       F has values (f1 f2 ... fn) ->
              O increases/decreases
5. Value-Map
       F has value -> O has value
              f1              o1
                   ......
              fn              on
6. Other
       Special rules
```

For example, the following are the some of the rules explaining the objective "absorption":

```
Concept Name: Absorption
Meaning: "The rate and extent of drug absorption in the gastrointestinal tract"
Explanations
Factor:        ionization-degree
Rule:          Decremental
A highly ionized drug has little absorption
      ...
Factor:        administration-route
Rule:          Value-order
Absorption is fastest and most complete if the drug is administered directly in the organ
```

The rules qualitatively relate factors with objectives. For example, the first rule indicates that there is an inverse relationship between ionization and absorption in the gastrointestinal tract. Objectives are selected by the user, since this allows experimentation with different combinations. Each objective is explained by all influence factors that affect it, and all objectives are combined to provide a utility value for the each outcome. Each outcome is ranked resulting in a matrix:

Result	Rank			
	1	2	3	4
outcome 1	W_{11}	W_{12}		
outcome 2			W_{23}	
...	...	...	...	...
outcome n		W_{n2}		

where W_{ij} is defined as:

$$W_{ij} = \sum_{k=1}^{n} W_{ijk}$$

where W_{ijk} is the weight of factor k which placed outcome i as j-th in rank. Then we calculate the *preference ranking* as:

$$R_i = \sum_{j=0}^{max-rank} W_{ij} \times j$$

where j is the *converted rank* of the objectives, and the preference ranking is the summation over all converted ranks [2]. Finally, subjective utility is assigned based on the preference ranking as:

$$U_i = \frac{R_i}{\sum_{j=1}^{M} R_j}$$

where U_i is the utility of case i, and the preference ranking is normalized by the sum of all preference ranking values.

For example, consider the objective "absorption", discussed above. Consider also two retrieved cases which contain chemicals with the following absorption-related factors, which are assigned a ranking by explanation:

	FACTOR	VALUE	RANK
Case 1	ionization-degree	highly-ionized	(2)
	lipid-solubility	insoluble	(1)
	administration-route	IV	(1)
Case 2	ionization-degree	un-ionized	(1)
	lipid-solubility	insoluble	(1)
	administration-route	oral	(2)

In other words, Case 1 is ranked second based on "ionization degree", first based on "lipid solubility", and first based on "administration route". Note that we allow ties in rank. Then, the ranking matrix becomes:

Result	Rank / Converted Rank	
	1/1	2/0
Case 1	0.2+0.3	0.5
Case 2	0.5+0.3	0.2

The values in the cells are the weights of the explanation rules. Case 1 has as weight W_{11} the weight of the rules that ranked it first. The rules were associated with the criteria "lipid solubility" and "administration route", and their weights were 0.3 and 0.2, which are added in the first cell of the table. The combination of preference rankings generates:

$R_{case\text{-}1} = (0.2+0.3)x1 + 0 = 0.5$

$R_{case\text{-}2} = (0.5+0.3)x1 + 0 + 0 = 0.8$

---

[2] The raw ranks are transformed into *converted ranks* such that rank 1 becomes converted rank m-1, where m is the maximum ranking number, rank 2 becomes converted rank m-2, and so on, and rank m becomes converted rank 0. This way we penalize low ranks and strengthen high ranks.

If "absorption" were the only objective, then we could immediately calculate the utilities as:

$U_{case\text{-}1} = 0.5/1.3 = 0.38$

$U_{case\text{-}2} = 0.8/1.3 = 0.62$

Returning to our old example of drug design, the descriptions of the drug compound in the retrieved cases will be used as the factors which will help evaluate each case based on the user-defined objectives. The user can select any combination of nine objectives: good bioavailability, complete and fast absorption, stable chemical properties, complete and easy distribution, easy elimination, quick action speed, long effective duration, high marketability, and low risk. These criteria were selected as the most basic ones that guide the early stages of drug design.

In our example, the user selected "complete and easy distribution" and "low risk" as the two evaluation objectives with weights 0.4 and 0.6 respectively. The following is an edited transcripted section of the system's evaluation of these two objectives:

```
Preference on DISTRIBUTION by PROTEIN-BIND is:
Case DISOPYRAMIDE-PHOSPHATE
       (PROTEIN-BIND MEDIUM)  1.0
Case LIDOCAINE-HYDROCHLORIDE
       (PROTEIN-BIND MEDIUM)  1.0
       ...
EXPLANATION: Only the unbound drug is dis-
tributed easily and is biologically active
       ...
Thus, based on the objective DISTRIBUTION
which means : The extend of the drug in system
circulation that permeates membranes and
reaches the potential site of action.
Explained by PROTEIN-BIND and LIPID-SOLUBILITY
Preferences are:
Case DISOPYRAMIDE-PHOSPHATE          0.7
Case LIDOCAINE-HYDROCHLORIDE         1.0
Case QUINIDINE-GLUCONATE             0.0
Case QUINIDINE-POLYGALACTURONATE     0.6
       ...
```

Next the system will analyze the cases based on the objective "low risk":

```
Preference on RISK by SIDE-EFFECTS is:
       ...
Case QUINIDINE-GLUCONATE has
   MINOR side effect: (NAUSEA HEADACHE)
   THREATENING side effect: (HEART-FAILURE)
   risk severity: 2.1
Case QUINIDINE-POLYGALACTURONATE
   MINOR side effect: (NAUSEA HEADACHE)
   THREATENING side effect: (HEART-FAILURE)
   risk severity: 2.1
...
Thus, based on the objective RISK which means:
...
Explained by SIDE-EFFECT, PATIENT-CONDITION
and DRUG-INTERACTION
Preferences are:
Case DISOPYRAMIDE-PHOSPHATE          0.7
Case LIDOCAINE-HYDROCHLORIDE         0.6
Case QUINIDINE-GLUCONATE             0.3
Case QUINIDINE-POLYGALACTURONATE     0.3

By overall criteria RISK and DISTRIBUTION:
Explanation-based subjective utilities are:
Case DISOPYRAMIDE-PHOSPHATE          0.366
Case LIDOCAINE-HYDROCHLORIDE         0.341
Case QUINIDINE-GLUCONATE             0.122
Case QUINIDINE-POLYGALACTURONATE     0.171
```

Finally, using the now completed decision tree shown in figure 4, we can calculate the expected utilities of all choices and select the compound LIDOCAINE-HYDROCHLORIDE as the most appropriate starting point for our subsequent design actions.

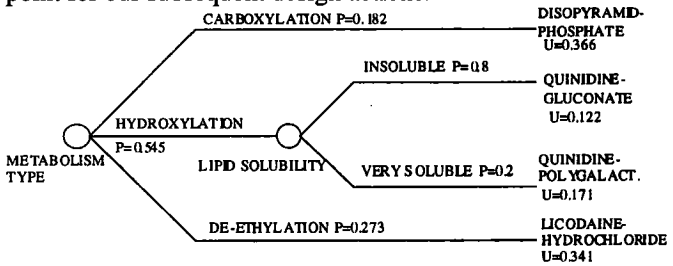

Figure 4: Final decision tree

Conclusions

We have developed a methodology that allows for the rigorous evaluation of design choices based on user-defined criteria, and is a unique effort to combine case-based reasoning, explanation-based reasoning, and utility theory. CBR can now deal with incomplete problem descriptions, and can retrieve cases based on multiple, dynamic criteria, allowing cases to be applicable to many diverse problems; case-based design has an automated way to evaluate alternatives, and explain the rationale and intent behind design decisions; decision theory is offered a way of establishing decision variables and predicted outcomes, and of assigning subjective probabilities and utilities based on historical information and domain knowledge.

References

French, S. 1988. *Decision Theory: an Introduction to the Mathematics of Rationality.* Ellis Horwood Limited.

Kahneman, D., P. Slovic and A. Tversky. eds. 1982. *Judgment Under Uncertainty: Heuristics and Biases.* Cambridge, MA: Cambridge University Press.

Raiffa, H. 1968. *Decision Analysis: Introductory Lectures on Choices Under Uncertainty,* Addison-Wesley.

Roberts, F.S. 1979. *Measurement Theory*, Addison-Wesley.

Scholz, R. W. ed. 1983. *Decision Making Under Uncertainty.* North Holland.

Wei, H-Y and C. Tsatsoulis. 1993. *Uncertain Case-Based Reasoning.* Technical Report, CECASE, Dept. of EECS, The University of Kansas.

Wright, G. 1984. *Behavioural Decision Theory.* Penguin Books.