(*AI EDAM*) (1987) **1**(1), 47–57

MODIFIABLE COMBINING FUNCTIONS[1]

PAUL R. COHEN, GLENN SHAFER AND PRAKASH P. SHENOY

Computer and Information Science, University of Massachusetts, Amherst, MA 01003 *and School of Business, University of Kansas, Lawrence, KA* 66045, *U.S.A.*

Modifiable combining functions are a synthesis of two common approaches to combining evidence. They offer many of the advantages of these approaches and avoid some disadvantages. Because they facilitate the acquisition, representation, explanation, and modification of knowledge about combinations of evidence, they are proposed as a tool for knowledge engineers who build systems that reason under uncertainty, not as a normative theory of evidence.

1. Introduction

This paper presents a synthesis of two general approaches to combining evidence. When designing knowledge systems, knowledge engineers typically select one approach over the other, but each has strengths and weaknesses in terms of the ease with which knowledge can be acquired, represented, interpreted, modified, and explained. The synthesis we propose, called *modifiable combining functions* has many of the advantages of both approaches and overcomes some of their disadvantages. The basic idea of modifiable combining functions is to acquire degrees of belief for a subset of all possible combinations of evidence, then infer degrees of belief for other combinations in the set. If, in the course of knowledge engineering, a particular degree of belief is challenged, then it (and others) can be modified by an appropriate method.

A combination of evidence is a list of propositions, each with an associated score. For example, an expert system for diagnosing plant diseases has propositions like this:

((soil texture = heavy, 0.7)
(soil oxygen = low, 0.9))

One interpretation of the scores is that they represent degrees of belief that a proposition is true; for example 0.7 may represent the degree of belief that soil texture is heavy. Alternatively, scores may represent the extent of a phenomenon; for example, 0.9 may represent the extent to which soil is devoid of oxygen.

Combinations of evidence are often found in the premises of inference rules. These rules can take two forms, called *specified* and *derived*, respectively:

specified form:

IF ((soil texture = heavy, 0.7)
(soil oxygen = low, 0.9))
THEN (water damage = yes, 0.8)

derived form:

IF ((soil texture = heavy, x)
(soil oxygen = low, y))
THEN (water damage = yes, f (x, y, k))

These forms suggest two general approaches to combining evidence. The specified form requires that for each combination of scores from the items in the premise, a degree of belief is specified for the conclusion. The derived form requires a function, f, that derives a degree of belief in the conclusion for *any* scores from the items in the premise. The symbol k in the derived form represents the degree of belief that would be assigned to the conclusion if the degree of belief in the premise was 1.0, that is, the degree of belief in the inference rule itself. This quantity is implicit in the specified form.

These forms combine evidence *within* inference rules, but they have counterparts for the cases in which two or more rules draw the same, corroborating conclusion. By analogy with the specified form, degrees of belief can be acquired for each combination of corroborating rules; alternatively, a general function, analogous to f in the derived form,

Received 1 June 1987.

[1] We thank Carole Beal, Sabine Bergler, Tom Gruber, and Adele Howe for their comments on drafts of this paper. This research is funded by a DARPA/RADC Contract F30602-85-C-0014.

0890–0604/87/010047 + 11$02.00/0

can be acquired to calculate degrees of belief for all corroborations.

Both approaches have been used in AI systems. Considering medical expert systems alone, we find knowledge in the specified form in PIP (Pauker *et al.*, 1976), IRIS (Trigoboff, 1978), MDX (Chandrasekaran, 1982), CSRL (Bylander and Mittal, 1986), and MUM (Cohen *et al.*, 1987); while MYCIN (Shortliffe, 1976), INTERNIST/CADUCEUS (Pople, 1977), and CASNET (Weiss *et al.*, 1977) use knowledge in the derived form.

In outline, we describe representations for combining functions that are closely related to the specified and derived forms. We will discuss the tradeoffs between these approaches that motivate the idea of modifiable combining functions. Then, in the context of an example, we will illustrate how modifiable combining functions are generated and modified, focusing especially on interpretations for the numbers that are supplied to and produced by the functions.

2. Forms of combining functions

2.1 TABULAR COMBINING FUNCTIONS

Tabular combining functions are often represented as tables that specify degrees of belief in conclusions for each combination of scores from items of evidence. Figure 1 shows a tabular function that combines two pieces of evidence, E_1 and E_2, for conclusion C. In this case, scores range from -1 to $+1$. A score of zero denotes ignorance. For example, if E_1 is the proposition soil-oxygen = low, then $(E_1, 0)$ means that the value of soil oxygen is unknown, either because it is an unavailable datum or because the data from which it is inferred are ambiguous. Scores of $+1$ and -1 represent the situations in which soil-oxygen is clearly low and clearly not low, respectively. Many of the cells are blank, meaning that the expert does not consider these combinations of evidence relevant—does not expect them to arise during problem solving. From the knowledge engineer's perspective, blank cells and zero cells represent different situations. A blank means that that particular combination of evidence was never considered, but a zero means it was considered and found to be uninformative. From the perspective of an AI program's interpreter, blank and zero *may* both mean that the combination of evidence is uninformative; or a blank may be used to alert the user to incompleteness in the combining function.

In tabular combining functions, scores from items of evidence index degrees of belief in conclusions. The combining function in Figure 1 specifies that when the scores of E_1 and E_2 are 0.75 and 0.5, respectively, the degree of belief in C is 0.25. Since conclusions are often used as evidence for subsequent inferences, the values in the cells of tabular combining functions may themselves be used to index degrees of belief in other tabular functions. Tabular functions increase exponentially in size: A function for N pieces of evidence requires an N-dimensional table, similar to the *signature tables* invented by Samuel (1959).

Some important knowledge about patterns or

Score (E_2) \ Score (E_1)	1.0	0.75	0.50	0.25	0	−0.25	−0.50	−0.75	−1.00
1.0	1.0	1.0	0.75	0.50	0				
0.75	1.0	1.0	0.50	0.25	0				
0.50	0.50	0.25	0	0	0				
0.25	0.25	0	0	0	0				
0	0	0	0	0	0				
−0.25									
−0.50									
−0.75	0	−0.50	−0.75	−0.75	−1	−1	−1	−1	−1
−1.0	0	−0.50	−0.75	−0.75	−1	−1	−1	−1	−1

FIGURE 1

regularities in combinations of evidence is implicit in tabular combining functions. For example, the entire upper-right quadrant of Figure 1 is blank, suggesting that no combination of negative scores for E_1 and positive scores for E_2 is meaningful. Similarly, in the lower-left quadrant we see a *threshold* on the score for E_2: the values in the table are determined by E_1 for all scores for E_2 less than or equal to -0.75. These regularities are easily captured by a *rule-based variant* of tabular combining functions. The two examples we just mentioned can be represented this way:

Upper-right quadrant:

IF score $(E_1) \leq 0$ and
score $(E_2) \geq 0$
Then score $(C) := 0$

Lower-left quadrant:

IF score $(E_1) = 0.5$ or
score $(E_1) = 0.25$ and
score $(E_2) \leq -0.75$
Then score $(C) := -0.75$

IF score $(E_1) = 0.75$ and
score $(E_2) \leq 0.75$
Then score $(C) := -0.5$

IF score $(E_1) = 1.0$ and
score $(E_2) \leq -0.75$
Then score $(C) := 0$

Irrespective of whether the knowledge engineer acquires tables like Figure 1, or rules as above, he or she must take care to maintain important distinctions in the domain. For example, the rule for the upper-right quadrant could be extended to account for the blank cells in the lower-right quadrant, too, by changing its first clause to 'IF score $(E_2) \geq -0.5$'. While this rule describes the table, it obscures what may be an important distinction between positive and negative scores for E_2.

Tabular combining functions and their rule-based variant are ways to represent combinations of evidence given in the specified form, described above. A representation that relies on both specified and derived combinations is discussed next.

2.2 INTERPOLATED COMBINING FUNCTIONS

Three of the four corner cells of Figure 1 represent degrees of belief in the conclusion given categorical scores (either 1 or -1) for E_1 and E_2 (the upper-right cell is blank because nothing is known about it.) They can be arranged in a *categorical table* as shown below. To distinguish categorical tables from the larger ones like Figure 1, we call the latter *full* tables.

bel (E_2) \ bel (E_1)	1	−1
1	1	X
−1	0	−1

The upper-left cell contains the degree of belief in C given that E_1 and E_2 both have scores of 1; conversely, the lower-right cell is the degree of belief in C when both have scores of -1; the 0 in the lower-left cell represents ignorance in C given that score $(E_1) = 1$ and score $(E_2) = -1$. To reiterate, these are the corner cells of the full table in Figure 1. All other, noncorner cells in Figure 1 represent interpolations between the values in this categorical table, interpolations over the range of possible scores for E_1 and E_2. For example, the cells around the centre of Figure 1 tend toward the value 0, since the centre cell represents the case in which the scores for E_1 and E_2 are both zero, that is, completely uninformative. Similarly, in the lower half of the table, we see degrees of belief in C ranging from 0 when score $(E_1) = +1$, to -1 for lower scores for E_1.

The full table in Figure 1 was built by hand, but full tables can also be derived by interpolating functions. Figure 2 shows the derivation of a full table by a Bayesian interpolating function. The categorical corner cells are 1.0, 0.95, 0.25, and 0.0, respectively. All other cells contain intermediate values that reflect uncertainty about the evidence. For example, when the scores for E_1 and E_2 are both 0.75, the degree of belief in the conclusion is 0.79, a value intermediate between the four corner points but nearer to 1.0—its nearest neighbor—in magnitude. This table and its derivation will be explained in Section 5.

To summarize, full tables can be built by hand, by specifying the value in each cell, or by specifying rules that assert the values of subsets of the cells. Alternatively, they can be derived automatically by interpolating from categorical tables. Once the decision has been made to use interpolating functions, full tables are usually not generated and stored. Instead, the values of combinations of evidence are computed as needed. However, the following section suggests that there are advantages to keeping both forms of combining functions.

Categorical table

1.0	0.25
0.95	0

S(R) \ S(E)	1.0	0.875	0.75	0.625	0.50	0.375	0.25	0.125	0
1.0	1.0	0.91	0.81	0.72	0.63	0.53	0.44	0.34	0.25
0.875	0.99	0.90	0.80	0.70	0.61	0.51	0.41	0.32	0.22
0.75	0.99	0.89	0.79	0.69	0.59	0.49	0.39	0.29	0.19
0.625	0.98	0.88	0.78	0.67	0.57	0.47	0.36	0.26	0.16
0.50	0.98	0.87	0.76	0.66	0.55	0.44	0.34	0.23	0.13
0.375	0.97	0.86	0.75	0.64	0.53	0.42	0.31	0.20	0.09
0.25	0.96	0.85	0.74	0.63	0.51	0.40	0.29	0.18	0.06
0.125	0.96	0.84	0.73	0.61	0.49	0.38	0.26	0.15	0.03
0	0.95	0.83	0.71	0.59	0.48	0.36	0.24	0.12	0

FIGURE 2

3. Comparison

Our comparison will focus on the tabular and interpolating forms of combining functions. The strengths of one often correspond to weaknesses in the other. First, tabular combining functions do not infer anything that is not stated by the expert. Most of the cells in a table are blank, meaning that the expert does not consider them to represent meaningful combinations of evidence. In theory, every nonblank cell represents a meaningful combination and every blank cell represents a meaningless one. But in practice, the sheer size of tabular functions means that some meaningful combinations of evidence are simply overlooked during knowledge acquistion. In this sense, tabular functions are brittle: they cannot account for all meaningful situations that will arise during problem solving.

Interpolation is clearly a solution to the brittleness problem. A value for any blank cell can be obtained by interpolation from the corners of a categorical table, or perhaps from its nearest neighbors. The disadvantages of interpolating functions are that, unlike tabular functions, they produce values for *all* combinations of evidence in their domain, meaningful or not. Moreover, no value derived by an interpolating function is guaranteed to reflect an expert's judgment.

Tabular functions are locally modifiable. A knowledge engineer can change the values of individual cells in the table with the assurance that the performance of the system will remain unchanged except in the case of these particular combinations of evidence. This allows a combining function to be tuned in the normal course of knowledge base refinement: when the system presents a conclusion that the expert thinks is wrong, and the source of the error is localized to a particular cell, then that cell can be changed. In contrast, changing an interpolating function may affect the values assigned to *all* combinations of evidence in its domain.

4. Modifiable combining functions

If the knowledge engineer decides to use interpolating functions, then why consider full tables (e.g. Figure 1) at all? Why not simply acquire categorical tables, as above, and design interpolating functions to fill in the intermediate values? Clearly, the two approaches are equivalent if the interpolating functions generate the same values as the expert for any combination of

evidence. But there is no way to test this, other than to acquire an entire table and then compare it with the results of an interpolation function. Consequently, the knowledge engineer can take one of two positions with respect to potential differences between interpolated values and the expert's judgment:

(1) The knowledge engineer can design a function that has desirable properties and assume that, if the expert's judgment is different, it is because the expert's reasoning is inconsistent or otherwise flawed.
(2) The knowledge engineer can design a function that is assumed to reflect expert judgment, but modify it to conform to the expert when deviations become apparent.

The first position is associated with normative models, the second with performance models. In both cases, the knowledge engineer must carefully design interpolation functions given what he knows and can assume about the evidence in a domain. In the latter case, in addition, he must have some mechanism for modifying combining functions.

Modifiable combining functions are a synthesis of tabular and interpolating functions. They are tabular functions that have most of their values derived by interpolation, but which can be modified to conform to an expert's judgment. Knowledge engineers must first acquire a categorical table and any other cells in the full table that the expert can provide. Interpolating functions ideally should fill in cells that the expert and knowledge engineer neglected to specify, with values that are likely to match the expert's judgment, but not fill in cells they intended to leave blank. If these goals are not achieved, the tabular function can be modified by one or more of three mechanisms discussed below.

5. An example

This section illustrates modifiable combining functions for two pieces of evidence from a medical diagnosis problem. Most diagnosis begins with the physician taking a *history*: asking about the patient's chief complaint, age, past medical history, and so on. Our example concerns the diagnosis of angina and two pieces of evidence from the history: the patient's report of an episode of chest pain, and the patient's risk factors for coronary artery disease, which is the cause of angina. We will focus on a single rule:

$$\text{episode} \;\&\; \text{risk factors} \rightarrow \text{angina} \qquad (1)$$

When a physician considers a patient's report of an episode of chest pain, he/she focuses on the question of how closely the description of the episode matches a prototypical episode of angina. The following examples illustrate how descriptions might be scored:

(a) crushing chest pain, induced by exercise, lasting a few minutes, radiating to one or both arms, accompanied by sweating and shortness of breath: score (episode) = 1.0

(b) sharp, fleeting chest pain, induced by sudden movement, not radiating: score (episode) = 0.0

(c) diffuse chest pain, came on after eating, radiating, lasting about 30 seconds: score (episode) = 0.5

The first description has a score of one because it is a classic description of an angina episode. (This does not necessarily mean the episode was angina; there are other causes for such episodes.) The second has a score of zero because it is clearly not a description of an angina episode. The third has only some characteristics of an angina episode and is given an intermedite score.

It is tempting to interpret this score as a probability for the event that the episode was angina-like. We will explore the possibilities of this interpretation in the next section. In this case, however, the scores are assessed as similarities. The physician can also score the patient's degree of risk for coronary artery disease. Again, this score is most naturally interpreted as similarity—the degree to which the patient's characteristics match those of a stereotypical person at risk for coronary artery disease. The following examples illustrate how risk factors might be scored:

(a) 60 year-old male, overweight, smoker, with high blood pressure, and two brothers with coronary artery disease: score (risk-factors) = 1.0.

(b) 30 year-old female, nonsmoker, not overweight, normal blood pressure, no history of heart disease in the family: score (risk-factors) = 0.0.

(c) 45 year-old male, smoker, not overweight, marginally-high blood pressure, uncle had coronary at age 60: score (risk-factors) = 0.5.

We will let $S(\mathrm{E})$ denote the score assigned to the patient's episode and $S(\mathrm{R})$ denote the score assigned to the patient's risk factors.

How should the knowledge engineer go about finding out how the scores $S(\mathrm{E})$ and $S(\mathrm{R})$ are combined to obtain a degree of belief for angina? Degrees of belief resulting from each possible combination of scores could be acquired in the specified form and arranged in a table. Alternatively, the knowledge engineer might design a combining function.

Modifiable combining functions offer an intermediate alternative: the knowledge engineer acquires degrees of belief for some possible combinations, designs a function to interpolate values for the rest, and arranges the results in a table. He then modifies the table if necessary to accord with the expert's judgment. The obvious place to begin this process is with the categorical table, from which a full table can be interpolated. Imagine the following rules, qualified by degrees of belief, are acquired from the expert:

$$\begin{aligned} \text{E \& R} &\rightarrow \text{angina, } 1.0 \\ \text{E \& } \sim\text{R} &\rightarrow \text{angina, } 0.95 \\ \sim\text{E \& R} &\rightarrow \text{angina, } 0.25 \\ \sim\text{E \& } \sim\text{R} &\rightarrow \text{angina, } 0.0 \end{aligned}$$

Here E denotes a typical episode, ~E denotes a completely atypical one, R denotes a patient maximally at risk, and ~R denotes a patient not at risk. These can be arranged in the following categorical table:

$S(R)$ \ $S(E)$	1	0
1	1.0	0.25
0	0.95	0.0

The knowledge engineer needs to design a function from which degrees of belief for angina can be derived for values of $S(E)$ and $S(R)$ other than zero and one. An obvious choice is linear interpolation. We set

$$\begin{aligned} P(A) = {} & 1.0 \times S(E)S(R) \\ & + 0.95 \times S(E)(1 - S(R)) \\ & + 0.25 \times (1 - S(E))S(R) \\ & + 0.0 \times (1 - S(E))(1 - S(R)). \end{aligned} \quad (2)$$

Figure 2 shows the table of values of $P(A)$ generated by this function, letting $S(R)$ and $S(E)$ range through the values 0, 0.125, 0.25, 0.375, 0.5, 0.625, 0.75, 0.875, and 1.0. Examination of this table shows that we are indeed interpolating linearly. To get the first row, say, we interpolate linearly between 1.0 and 0.25. To get the last row, we interpolate linearly between 0.95 and 0. Then, we get the columns by interpolating linearly between their first and last values.

The expert may question the results of our interpolation. In the case where $S(E) = 0.5$ and $S(R) = 0.75$, for example, the expert may prefer a much higher value for $P(A)$ than the value of 0.59 given by Figure 2. He may say that if there is moderate evidence of an episode and strong evidence of risk then the probability of angina should be much higher, say, 0.75.

What should the knowledge engineer do in this case? If he is relying exclusively on interpolating functions then he has 3 options:

1. insist that the expert's judgment is flawed
2. change the categorical table
3. change the interpolating function

The first is reasonable only if the knowledge engineer has strong confidence in the assumptions that underlie his interpolating function. The other two have global effects on all the numbers in a full table, not just the few the expert criticized. Thus, in fixing the immediate problem the knowledge engineer could introduce new ones. Knowledge engineering often extends over a period of months, and the knowledge engineer relies on a kind of monotonicity—adding new knowledge to a system will not make it perform differently on most previous cases.

Changing the categorical table will usually not have pervasive effects on the system's performance, because categorical tables are associated with individual inference rules. For example, the categorical table in Figure 2 really represents four inference rules for inferring angina from E and R, each with a different degree of belief. Changing one or more of these numbers will certainly affect the entire derived full table, but the effect of these changes on system performance will be localized to cases where the rules associated with the categorical table are used. In contrast, changing an interpolation function will change the degrees of belief of all the conclusions derived by that function—potentially every conclusion previously derived by a knowledge system.

If the knowledge engineer *does* decide to change the function, how should he go about it? An obvious change would be to interpolate on a different scale than the probability scale. But there is no substantive change if we merely make a linear change of scale, such as a change from the [0, 1] proability scale of the [−1, 1] MYCIN-like belief scale by the linear transformation $2P - 1$. On the other hand, there are many reasonable non-linear transformations.

We might, for example, use a log-odds scale, corresponding to the transformation $L = \ln(p/(1 - p))$. Interpolation on the log-odds scale ensures that changes in moderate-value scores have disproportionate effects on degrees of belief relative to changes in high-value or low-value scores. For example, Figure 2 shows that when $S(R) = 0.5$, changing $S(E)$ from 0.5 to 0.75 produces a moderate change in degree of belief in angina: it goes from 0.55 to 0.76. Because Figure 2 is generated by linear interpretation, the degree of belief in angina changes by approximately the same amount when $S(E)$ goes from 0.75 to 1.0.

An expert may find this linearity counterintuitive. He may say that by the time the score of $S(\mathrm{E})$ reaches 0.75 he is very sure the patient has angina, and that higher scores for $S(\mathrm{E})$ do not add much to his certainty. The interpolation function we choose should reflect the expert's judgment that increasing $S(\mathrm{E})$ from 0.5 to 0.75 has a large effect on his degree of belief in angina, but increasing $S(\mathrm{E})$ to 1.0 has little further effect.

Interpolation on a log-odds scale accomplishes this. However, it does not work for initial degrees of belief of zero or one, and hence will not work for the categorical table in Figure 2. But if we use the categorical table

$S(\mathrm{R})$ \ $S(\mathrm{E})$	1	0
1	0.99	0.25
0	0.95	0.01

then we obtain the tabular combining function in Figure 3 by interpolation on a log-odds scale. To calculate entries in this table we first apply the transformation $\ln(p/(1-p))$ to the categorical table, obtaining

$S(\mathrm{R})$ \ $S(\mathrm{E})$	1	0
1	4.59	−1.10
0	2.94	−4.59

Then we linearly interpolate by the following formula, which is just (1) with different coefficients:

$$\begin{aligned} L = {} & 4.59 \times S(\mathrm{E})S(\mathrm{R}) - 1.10 \times S(\mathrm{E})(1 - S(\mathrm{R})) \\ & + 2.94 \times (1 - S(\mathrm{E}))S(\mathrm{R}) \\ & - 4.59 \times (1 - S(\mathrm{E}))(1 - S(\mathrm{R})). \end{aligned} \tag{3}$$

Then we find $P(\mathrm{A})$ by reversing the transformation:

$$P(\mathrm{A}) = \exp(L)/(1 + \exp(L)).$$

The resulting table (Figure 3) reflects the expert's judgment that degrees of belief in angina should increase rapidly over moderate scores for $S(\mathrm{E})$ and $S(\mathrm{R})$, but should change little for extreme scores. When $S(\mathrm{R}) = 0.5$, the degree of belief in angina for $S(\mathrm{E}) = 0.75$ is 45% larger than the degree of belief in angina for $S(\mathrm{E}) = 0.5$, but it increases just another 9% when $S(\mathrm{E})$ increases to 1.0.

$S(\mathrm{R})$ \ $S(\mathrm{E})$	0.75	0.5	0.25	0
	0.99	0.979	0.959	0.921	0.851	0.738	0.580	0.404	0.25
0.875	0.987	0.974	0.948	0.897	0.806	0.665	0.486	0.311	0.177
	0.984	0.968	0.933	0.866	0.751	0.583	0.393	0.230	0.122
0.625	0.981	0.959	0.915	0.829	0.686	0.496	0.307	0.166	0.082
	0.977	0.949	0.892	0.783	0.613	0.409	0.232	0.117	0.054
0.375	0.972	0.937	0.864	0.730	0.534	0.328	0.171	0.081	0.036
	0.966	0.922	0.830	0.668	0.454	0.256	0.124	0.055	0.023
0.125	0.958	0.903	0.789	0.601	0.376	0.195	0.088	0.037	0.015
	0.95	0.881	0.742	0.529	0.304	0.145	0.062	0.025	0.01

4.595	−1.10
2.944	−4.595

FIGURE 3

Categorical table

1.0	0.25
0.95	0

S(R) \ S(E)	1.0	0.875	0.75	0.625	0.50	0.375	0.25	0.125	0
1.0	1.0	0.91	0.81	0.72	0.63	0.53	0.44	0.34	0.25
0.875	0.99	0.90	0.80	0.70	0.61	0.51	0.41	0.32	0.22
0.75	0.99	0.89	0.79	0.69	0.59	0.49	0.39	0.29	0.19
0.625	0.98	0.88	0.78	0.67	0.57	0.47	0.36	0.26	0.16
0.50	0.98	0.87	0.76	0.66	0.55	0.44	0.34	0.23	0.13
0.375	0.97	0.86	0.75	0.64	0.53	0.42	0.31	0.20	0.09
0.25	0.96	0.85	0.74	0.63	0.51	0.40	0.29	0.18	0.06
0.125	0.96	0.84	0.73	0.61	0.49	0.38	0.26	0.15	0.03
0	0.95	0.83	0.71	0.59	0.48	0.36	0.24	0.12	0

FIGURE 4

Categorical table

1.0	0.25
0.95	0

S(R) \ S(E)	1.0	0.875	0.75	0.625	0.50	0.375	0.25	0.125	0
1.0	1.0	0.91	0.81	0.75	0.75	0.50	0.50	0.34	0.25
0.875	0.99	0.90	0.80	0.75	0.75	0.50	0.50	0.32	0.22
0.75	0.99	0.89	0.79	0.75	0.75	0.50	0.50	0.29	0.19
0.625	0.98	0.88	0.78	0.75	0.75	0.50	0.50	0.26	0.16
0.50	0.98	0.87	0.76	0.66	0.55	0.44	0.34	0.23	0.13
0.375	0.97	0.86	0.75	0.64	0.53	0.42	0.31	0.20	0.09
0.25	0.96	0.85	0.74	0.63	0.51	0.40	0.29	0.18	0.06
0.125	0.96	0.84	0.73	0.61	0.49	0.38	0.26	0.15	0.03
0	0.95	0.83	0.71	0.59	0.48	0.36	0.24	0.12	0

FIGURE 5

If the knowledge engineer does not rely exclusively on interpolating functions to calculate degrees of belief, then he has another option besides the three listed above: He can simply change the values that the expert says are wrong and store the new values in a tabular form that overrides the derived values. The idea of modifiable combining functions is, in essence, to use simple interpolating functions such as (2) and (3) to derive full tables from categorical tables, then, when the expert criticizes a derived degree of belief, to simply change it. This is shown in Figures 4 and 5. In Figure 4, the expert identifies a block of cells with values that are too low. Figure 5 shows one possible modification.

Another possibility is to use local rather than global interpolating functions. The global methods we have discussed so far require only the four cells of the categorical table to generate a full table, but if degrees of belief for other cells in the full table have been acquired from the expert, they can be used as additional 'anchors' for more local interpolation. For example, the knowledge engineer might complete the table using splines or some other local fitting method. If, as we discussed earlier, individual cells in the table are changed by the expert, then the local fitting method can be repeated, changing only values in the neighborhood of the newly specified cells.

In sum, there are four methods for changing modifiable combining functions to match expert judgments. First, values in the categorical table can be changed. Second, one can change individual cells or blocks of cells in the full table. Third, individual cells can be changed and local interpolation can be repeated. Fourth, and as a last resort, the global interpolating function can be changed.

6. A probabilistic interpretation

Some readers may prefer a more probabilistic analysis than we have just offered. In this section, we consider a probabilistic interpretation for linear interpolation in our example.

Suppose we interpret the scores $S(\mathrm{E})$ and $S(\mathrm{R})$ as probabilities. We intepret $S(\mathrm{E})$ as the probability that the episode was angina-like, and $S(\mathrm{R})$ as the probability that the patient is at risk for coronary artery disease. This is a distortion. $S(\mathrm{E})$ and $S(\mathrm{R})$ are really scores: they measure the similarity between an angina episode and the patient's episode, or the extent to which the patient is at risk. They are not really probabilities. But interpreting them as such does allow us to give a probabilistic interpretation to our linear interpolations.

To see this, consider the rule of total probability:

$$P(\mathrm{A}) = \sum_{i=1 \to n} P(\mathrm{A} \mid \mathrm{B}_i) P(\mathrm{B}_i), \tag{4}$$

where $\mathrm{B}_1, \ldots, \mathrm{B}_n$ is an exhaustive list of mutually exclusive possibilities. For our example, A is the conclusion angina and $\mathrm{B}_1, \ldots, \mathrm{B}_n$ are the events E, $\sim$E, R, and $\sim$R. We can then assess the probabilities of these events: $P(\mathrm{E})$, $P(\sim\mathrm{E})$, $P(\mathrm{R})$, and $P(\sim\mathrm{R})$.

If we assume that E and R are independent, then (2) becomes:

$$\begin{aligned} P(\mathrm{A}) = {} & P(\mathrm{A} \mid \mathrm{E} \,\&\, \mathrm{R}) P(\mathrm{E}) P(\mathrm{R}) \\ & + P(\mathrm{A} \mid \mathrm{E} \,\&\, \sim\mathrm{R}) P(\mathrm{E}) P(\sim\mathrm{R}) \\ & + P(\mathrm{A} \mid \sim\mathrm{E} \,\&\, \mathrm{R}) P(\sim\mathrm{E}) P(\mathrm{R}) \\ & + P(\mathrm{A} \mid \sim\mathrm{E} \,\&\, \sim\mathrm{R}) P(\sim\mathrm{E}) P(\sim R). \end{aligned} \tag{5}$$

This has the same form as (2), since $P(\sim E) = (1 - P(\mathrm{E}))$ and $P(\sim\mathrm{R}) = (1 - P(\mathrm{R}))$. Using (5) gives the same results as (2), except now we are interpreting E and R probabilistically.

The assumption that E and R are independent would be unacceptable if we were to model the patient as a random person from the population and we thought of $P(\mathrm{E})$ as the probability of the event that a random person's episode was genuinely angina-like, and $P(\mathrm{R})$ as the probability that this random person is at risk for coronary artery disease. For this random person, E and R are likely to be correlated, not independent. However, there is another probabilistic interpretation of $P(\mathrm{E})$ and $P(\mathrm{R})$. When we calculate (5) we are constructing a probability distribution for a *particular* patient using ingredients from two different sources. One is our knowledge of the population, which gives us the conditional probabilities $P(\mathrm{A} \mid \mathrm{E} \,\&\, \mathrm{R}) \ldots$. The other is our evaluation of the individual patient's story, which gives us $P(\mathrm{E})$ and $P(\mathrm{R})$. Our uncertainty about E from this story may well be independent of our uncertainty about R.

When the rule of total probability is used to combine two sources of evidence in this way, it is called Jeffrey's rule (Shafer, 1981; Shafer and Tversky, 1985, p. 333).

The difficulty with this thoroughly probabilistic interpretation is its brittleness. How can we give a probabilistic interpretation to modifications of the combining function? As it is now interpreted, this function combines probabilities $P(\mathrm{E})$ and $P(\mathrm{R})$ with transition probabilities $P(\mathrm{A} \mid \mathrm{E} \,\&\, \mathrm{R}) \ldots$.

Can we still interpret $P(\mathrm{A})$ as a probability if we modify the function in the ways described earlier? Changing the categorical table does no violence to the probabilistic interpretation, but it is unclear how to

maintain this interpretation under the other changes we considered.

A tempting modification would be to drop the assumption that the events E and R are independent. We could do this by replacing $P(E)$ with $P(E \mid R)$. However, this is not a change in the combining function: it is a change in inputs. The user is now asked not only how typical an episode is of angina, but also how much the patient's description would suggest a genuine angina-like episode for different levels of risk for coronary artery disease. These data are generally not available to the user of a system. Most knowledge systems are designed to take as data the probabilities of individual events such as E, not the conditional probabilities of one event given another. This approach also decisively replaces the initial 'similarity' semantics we used in assessing $S(E)$ and $S(R)$ with a probability semantics. We will want to make this substitution only if we have more probability-like evidence, that is, extensive knowledge of relevant frequencies. It seems clear that the probabilistic interpretation should be avoided if we are going to assess the episode and the patient's risk factors in terms of similarity.

7. Conclusion

Modifiable combining functions share many of the advantages of tabular and interpolating functions while avoiding some of their disadvantages. The information burden of tabular functions is reduced because the full table is derived by interpolating from the values the expert *can* provide. (One natural basis for the interpolation is the categorical table, but others are possible.) The brittleness of tabular combining functions, especially multidimensional ones, is overcome. Simple interpolating functions can be used, requiring relatively few numbers from the expert. But any value in the derived full table can be overridden by the expert's judgment. Discontinuities can easily be expressed in the rule-based variant of tabular combining functions. When an interpolating function fills in cells that the expert thinks should be blank (meaningless), the cells can be modified accordingly. All modifications to cells are local in the sense that they affect the system's performance for combinations of evidence represented by those cells only. But if global modifications are appropriate, if *all* the values in a modifiable combining function seem wrong to the expert, then the knowledge engineer can first consider modifying the categorical table (or any other set of points used for interpolation) and then consider modifying the interpolation function.

Currently, we are acquiring tabular combining functions for a medical expert system (Cohen *et al.*, 1987) and a plant pathology system. They are represented as rules, as discussed above. We have built interfaces for acquiring and modifying these rules, including a graphic interface for representing them in tabular form. Currently, we do not fill in the values of empty cells by interpolation, so the full promise of modifiable combining functions has yet to be demonstrated.

References

Bylander, T. and Mittal, S. 1986. CSRL: A language for classificatory problem solving and uncertainty handling. *AI Magazine* 7(3), 66–77.

Chandrasakeran, B., Mittal, S., and Smith, J. W. 1982. Reasoning with uncertain knowledge: the MDX approach. *Proceedings of the Congress of American Medical Informatics Association,* San Francisco, pp. 335–339.

Cohen, P., Day, D., Delisio, J., Greenberg, M., Kjeldsen, R., Suthers, D., and Berman, P. 1987. Management of uncertainty in medicine. *International Journal of Approximate Reasoning.* **1**(1). Forthcoming.

Gruber, T. R. and Cohen, P. R. 1987. Design for acquisition: principles of knowledge system design to facilitate knowledge acquisition. *International Journal of Man Machine Studies* **26,** 143–159.

Patil, R. S., Szolovits, P. and Schwartz, W. B. 1981. Causal understanding of patient illness in medical diagnosis. *Proceedings of 7th International Conference on Artificial Intelligence.* British Columbia: William Kaufmann Publishers. pp. 893–899.

Pauker, S., Gorry, G., Kassirer, J. and Schwartz, W. 1976. Toward the simulation of clinical cognition: Taking a present illness by computer. *American Journal of Medicine.* **60,** 981–995.

Pearl, J. 1986. Fusion, propagation, and structuring in Bayesian networks. *Artificial Intelligence* **29,** 24–288.

Pople, H. 1977. The formation of composite hypotheses in diagnostic problem solving—an exercise in synthetic reasoning, *Proceedings of the Fifth International Joint Conference on Artificial Intelligence.* British Columbia: William Kaufmann. pp. 1030–1037.

Samuel, A. L. 1959. Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development* **3,** 210–229.

Shafer, G. 1981. Jeffrey's rule of conditioning. *Philosophy of Science* **48**(3), 337–362.

Shortliffe, E. 1976. *Computer-Based Medical Consultations: MYCIN.* New York: American Elsevier.

Shortliffe, E. H. and Buchanan, B. G. 1975. A model of inexact reasoning in medicine. *Mathematical Biosciences* **23,** 351–379.

Shafer, G. and Tversky, A. 1985 Languages and Designs for Probability Judgment, *Cognitive Science* **9,** 309–339.

Shenoy, P. and Shafer, G. 1986. Propagating belief functions with local computations. *IEEE Expert* **1**(3), 43–52.

Trigoboff, M. 1978. IRIS: A framework for the construction of clinical consultation systems. Doctoral dissertation, Computer Science Dept. Rutgers University.

Weiss, Kulikowski and Safir. 1977. A model-based consultation system for the long-term management of Glaucoma. *Proceedings of the Fifth International Joint Conference on Artificial Intelligence.* British Columbia: William Kaufmann. pp. 826–832.

Paul R. Cohen is Assistant Professor of Computer Science at the University of Massachusetts at Amherst. He received his PhD from Stanford University for work in qualitative approaches to reasoning under uncertainty. He is an Editor of the *Handbook of Artificial Intelligence.*

Glenn Shafer is Professor of Business at the University of Kansas. He received his PhD in Statistics from Princeton University in 1973. He is an author of *A Mathematical Theory of Evidence* and numerous articles on probability theory, the Dempster–Shafer theory and decision theory.

Prakash Shenoy is Associate Professor in the School of Business at the University of Kansas. He received his B.Tech from IIT in Bombay and his MS and PhD in Operations Research from Cornell. He is interested in the management of uncertainty in knowledge-based systems in Operational Research.