Transactions on Information and Communications Technologies vol 19, © 1997 WIT Press, www.witpress.com, ISSN 1743-3517

Sensitivity Analyses in Knowledge Engineering

Albert K.W. Wu
Department of Computing, Hong Kong Polytechnic University
Kowloon, Hong Kong
Tel: (+852) 2766 7290; Fax: (+852) 2774 0842
Email: csalbert@comp.polyu.edu.hk

Abstract

In this paper a powerful tool called sensitivity analysis in the context of knowledge engineering is discussed. Subtleties related to the popular certainty factor calculus employed by most shells are highlighted. With reference to practical examples, we also show how sensitivity analyses can help validate a knowledge base.

Keywords: Sensitivity analysis, Expert Systems, Knowledge Base, Knowledge-based Systems

1 Introduction

The ultimate product of the process of knowledge engineering is to produce knowledge-based systems - systems exhibiting intelligent behavior by modelling the empirical associations and heuristic relationships that experts have acquired over time. During the process, knowledge of the expert(s) is acquired and some model is established for the building of the knowledge-based system or knowledge base (KB).

Most knowledge-oriented intelligent tasks exercising judgement carry some degree of uncertainty. This is also the nature of heuristic-based reasoning. As such we can always classify the knowledge acquired of a KB into two types:

- "Objective" knowledge that refers to the characterization of the inference of the expert such as what causes that, etc.;
- "Subjective" knowledge that refers to the judgemental inputs (eg. weights, pair-wise comparison, beliefs, strength of preferences, etc.) required to derive the conclusion (and its relative significance) from the objective knowledge.

Transactions on Information and Communications Technologies vol 19, © 1997 WIT Press, www.witpress.com, ISSN 1743-3517

It is also the interaction and manipulation of the above two types of knowledge in the inferencing or reasoning for conclusion. In order to ensure the adequacy of the KB, it is therefore essential to check for the accuracy, consistency and compliance of the system performance with the specified requirements and needs. So called the process of verification and validation.[7,12,13] However, even if a system has gone through successful verification and validation, it does not mean that the system is adequate. Verified and validated systems can still exhibit unsatisfactory functional performance caused by many other factors such as poor hardware/software, poor explanation and interface design, etc. As it is common that the behavior of the resulting systems cannot always be predicted in advance, it is therefore important to have more relevant tools for detecting errors and predicting performance.

In this paper, sensitivity analysis as a powerful tool in aiding validating and understanding the performance of knowledge-bases is presented. First we will discuss on the foundations of the technique. It is then followed by its application in knowledge engineering.

2 Sensitivity Analysis

Sensitivity analysis is essentially an extension and development of a rather old idea, which was known in the theory of differential equations under the name of a “correctly set” problem[10] - the kind of problem admitting a solution y_o not only for an isolated set of parameters λ_o, but also in at least a sufficiently small neighbourhood of λ_o.

Traditionally, the objective of the theory of partial differential equations was limited to the determination of representative solution y_o thus making the study of correctly set problems remain essentially qualitative. In sensitivity analyses however, a quantitative aspect is added by asking how fast (and/or how much) the reference solution y_o varies when one or more parameters of the set λ_o are given slightly different values. As this question remains legitimate for problems not necessarily associated with partial differential equations, the scope of sensitivity study therefore appears to be larger than the scope of the theory of correctly set problems.

In a very broad term, the concept of sensitivity analysis can be visualized as follows (Fig.1):

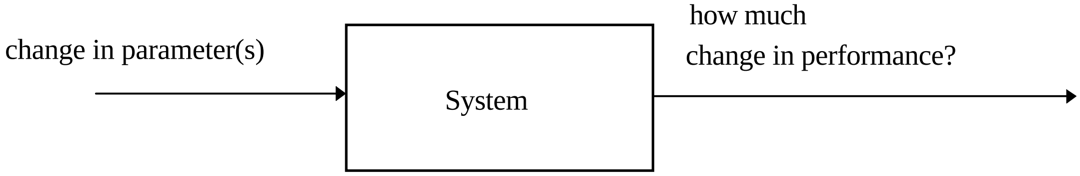

Fig.1: Concept of Sensitivity Analysis

Transactions on Information and Communications Technologies vol 19, © 1997 WIT Press, www.witpress.com, ISSN 1743-3517

And the sensitivity (function) can be broadly defined as:

$$S(P,V) = \frac{\partial P}{\partial V}$$

where ∂P is the change of system performance (or some observed behaviour under certain perspective) ∂V is the change of some system parameter value.

The problem of sensitivity is always critical in the stability design of engineering systems and the the technique has also been used to validate various system models.[6] With sensitivity analyses, one can be shown how a model or system changes with variations or perturbations in its parameters eg. which parameters of the model have the most effect on model behavior, and thus suggest a simplified treatment of parameters that are not important. Recently sensitivity analyses have been applied to models in social studies,[5] physiological systems,[1,2,9] and more recently expert systems.[3] Of course, sensitivity is a concept that can be meaningfully defined only by considering specific systems and their particular purposes for existence.

In summary, depending on the purpose the results of sensitivity analyses can be utilized

- to point out important parameters of the model by revealing how fast (and deep) it affects system behaviour, conversely for indicating unimportant ones;
- to guide the formulation of the model by pointing out important/unimportant elements that should or could be treated fully/simply,
- to warn of strange or unrealistic model behavior by showing how the system changes when perturbed,
- to suggest the accuracy to which the parameters are to be calculated for system stability,
- to suggest new experiments or guide future data collection efforts,
- to suggest plausible adjustment to the numerical values for the parameters, etc., and finally
- to validate a model.

2.1 The Relative Sensitivity

In general the power of sensitivity analyses rests on the sensitivity functions found for revealing the system's behaviour to parameters' changes. Given the performance of a system can be viewed from differet perspectives such as cost, state-behaviour, etc., correspondingly a number of sensitivity functions such as cost sensitiviy, state sensitivity, etc. can also be defined. Indeed there can have many types of sensitivity functions[6]. As long as parameter variations are small and "regular," that is, the sensitivity function is assumed a continuously differentiable function of its arguments at a nominal value of the performance, many classical methods based on conventional differential calculus can be borrowed.

For comparing the effects of different parameters, the relative sensitivity function can be used. The relative sensitivity (R) due to a parameter β with respect to a system (function) F is defined as

$$R(F,\beta) = \frac{\text{percent change in F}}{\text{percent change in }\beta} = \frac{\partial F / F}{\partial\beta / \beta} = \left[\frac{\partial F}{\partial \beta}\right]_{OP} * F_o/\beta_o \qquad (1)$$

where F_o is the function and β_o is the value of the parameter β evaluated at the operating point (OP). The presence of OP provides the point of reference which the sensitivity is evaluated. Relative sensitivity functions are ideal for comparing the different effects of parameters, because they are dimensionless, normalized functions.

3 Sensitivity Analyses for Knowledge Bases

There are a number of parameters governing the performance of a KB. These parameters include (Fig.2):

- the knowledge representation scheme (KR),
- the inference mechanism (IM),
- the system architecture (SA),
- the explanation facilities (EP),
- the user-interface facilities (UI),
- the objective knowledge acquired (OK),
- the subjective knowledge acquired (SK),
- the hardware (HW),
- the software (SW),
- etc.

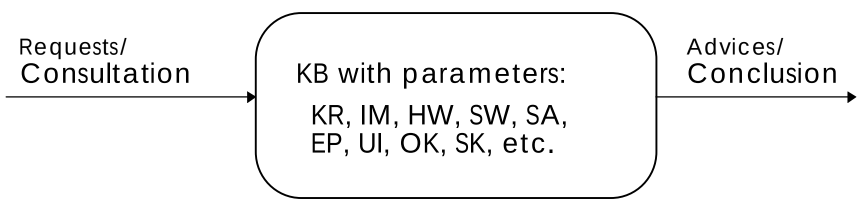

Fig.2: Parameters Affecting KB Performances

Much have been done on validating the performance of KBs with respect to the various aspects of hardware, software, etc. There is also a growing trend on validating the knowledge of the KBs. However, nearly most of them are not counting on the subjective knowledge aspect in their validation process or models, see [11,12,15] for example. The issue also seems a lot less aspired in most current investigations although it's very important to the validity of advices or conclusion provided of the KB. As sensitivity analysis has been employed and proved successful in many areas, it is believed that the technique is also viable for KB validation. Below, before we demonstrate how sensitivity analysis contributes and yields enlightening result in this direction, the way how uncertainty is handled in KBs is discussed first.

3.1 Uncertainty Handling in Knowledge Bases

As previously mentioned, subjective knowledge of a KB refers to the judgemental inputs (weights, pair-wise comparison, beliefs, strength of preferences, etc.) required to derive a conclusion (and its relative significance). And it is usually reflected as uncertainties of the knowledge captured in the KB.

Currently there are a number of uncertainty handling scheme developed for KBs with the most notable being the certainty factors (CFs) derived from the work of MYCIN.[14] Given its popularity, we'll use CF as the example to demonstrate the application of sensitivity analyses to KBs. It should be stressed that the discussion is also applicable to other uncertainty handling models.

3.2 Certainty Factors (CFs) and Its Combinations

In essence, the CF formalism is based loosely on Bayes' analysis and varies slightly in its implementation from system to system. It is a very straightforward scheme for the specification and manipulation of uncertainties in KBs with operations as follows: associate a certainty value or CF value to a piece of evidence or conclusion. And this value is an indication of the expert judgemental weight on the item, be it premise or conclusion of a rule. In general, most CFs are expressed in a scale of -1 (most certain that it is false) to +1 (most certain that it is true) or a scale of -100 to 100%.

Assume we are using a ± 1 scale, the following states some common calculation related to combining CFs.

<u>For conjunctive premises</u>. For the AND of two (or more) premises, the certainty of the rule will be the minimum of the certainty factors in the premises. This is a direct borrowing from the fuzzy calculus.[17] For instance, suppose we are given the rule:

If Pl and P2 then C

If a system consultation reveals that P1 is associated with a certainty (CF1) of 0.3 and that P2 (CF2) is 0.7, then C will be true with a certainty of 0.3, the minimum of 0.3 and 0.7, i.e. Minimum(CF1,CF2). The certainty factors of the premises will be determined during the consultation based on user's statement of certainty and by calculation on other rules. For simplicity, in this paper we will just write them next to the premise as follows:

If P1 (0.5) and P2 (0.6) then C

Transactions on Information and Communications Technologies vol 19, © 1997 WIT Press, www.witpress.com, ISSN 1743-3517

For combining certainties of the premise and the conclusion. Certainty factors of premises and conclusions get multiplied together to get the certainty of the rule. For example, if P1 is associated with a certainty of 0.5, and a certainty of 0.8 is affiliated with the conclusion, then the certainty of the rule would be 0.4. That is,

Given: If P1 (0.5) then C (0.8)
Then: the resultant certainty of the deduction is 0.5*0.8=0.4

For combining certainty factors for two rules. If rule-1 is true (with CF1) and rule-2 is true (with CF2) with the same conclusion, then the certainty of the conclusion is calculated according to the following equation:

$$CF_{final} = CF1 + (1 - CF1)* CF2 \quad (2.a)$$

or

$$CF_{final} = CF2 + (1 - CF2)* CF1 \quad (2.b)$$

as CF1 and CF2 should be symmetrical in the derivation of the final certainty factor.

For disjunctive premises. Most MYCIN derived shells treat rules with ORs of two (or more) premises as two (or more) rules, for example,

Given the rule: If P1 or P2 then C

This rule will be broken into

rule-1: If P1 then C
rule-2: If P2 then C

If both P1 and P2 are true, the certainty of C would be derived according to Eq. (2.a or 2.b). Thus if P1 were true with certainty 0.5 and P2 were true with certainty 0.6, then C would be true with certainty 0.8. Assume each premise of a disjunctive rule has a certainty, say CF1 and CF2 for P1 and P2 respectively, and a certainty value is also associated with the conclusion, say CF3. The formula for calculating the final certainty is

$$CF_{final} = CF1*CF3+(1-CF1*CF3)*(CF2*CF3) \quad (3.a)$$

or

$$CF_{final} = CF2*CF3+(1-CF2*CF3)*(CF1*CF3) \quad (3.b)$$

based on Eqs (2.a and 2.b).

However, some shells derive the certainty for a disjunctive rule by using the maximum of the premises' certainty factors, for example,

Given the rule: If P1 (0.5) or P2 (0.6) then C

These shells would assign a certainty of 0.6 to C, the maximum of 0.5 and 0.6.

Other cases. It should be noted that we have omitted the situations where CFs are negative. In cases where all the CFs are negative, then the treatment is just the same as for positive with sign reversed. Since the way for asserting affirmity is just the same for falsity in theory. For mixed cases of having both positive and negative CFs, a number of factors have to be taken into account, eg. the setting of thresholds on meaningful CF values, the way absolute values of CFs are compared,[8] etc. Different systems also have different ways of handling the situations. Due to space reason and in order not to divert too much from the theme, these cases are not presented. Interested readers may consult [3,8] for more on the issue. Nonetheless, our discussion is also applicable to them.

3.3 Sensitivity Analyses for Certainty Factors

Assume we have a rule of the following form:

If (P1 (CF1) and P2 (CF2)) or P3 (CF3) then C (CF4) (R1)

Below we first show how the final CF, CF_{final}, is dervied. Then how the sensitivity (functions) are calculated and used to obtain the relative sensitivities.

In the AND clause, from the rule given, only one of the premises' certainties (CF1 or CF2) will have an effect on the final certainty. That is, in the AND clause consisting of only CF1 and CF2, if CF1 > CF2 then the sensitivity function $S(CF_{final},CF1) = 0$. And if CF1 = CF2, then $S(CF_{final},CF1) = S(CF_{final},CF2)$.

For simplicity, let us assume that CF1< CF2 and that changes are small enough so that this inequality is not violated. The final certainty factor becomes

$$CF_{final} = CF1*CF4+(1-CF1*CF4)*(CF3*CF4) \quad (4.a)$$

or

$$CF_{final} = CF3*CF4+(1-CF3*CF4)*(CF1*CF4) \quad (4.b)$$

based on Eqs (3.a and 3.b).

Now the sensitivities can be calculated as:

$$S(CF_{final},CF1) = \frac{\partial CF_{final}}{\partial CF1} = CF4*(1- CF3*CF4) \quad (5)$$

$$S(CF_{final},CF2) = \frac{\partial CF_{final}}{\partial CF2} = 0 \quad (6)$$

$$S(CF_{final},CF3) = \frac{\partial CF_{final}}{\partial CF3} = CF4*(1- CF1*CF4) \quad (7)$$

$$S(F_{final},CF4) = \frac{\partial CF_{final}}{\partial CF4} = CF1 + CF3 - 2*CF1*CF3*CF4 \quad (8)$$

In order to obtain the relative sensitivity functions we multiply the above by the nominal values (referring to Eq.1). Thus we have

$$R(CF_{final},CF1) = [CF4*(1- CF3*CF4)]_{OP}* [CF1/CF_{final}]_{OP} \quad (9)$$

$$R(CF_{final},CF2) = 0 \quad (10)$$

$$R(CF_{final},CF3) = [CF4*(1- CF1*CF4)]_{OP}* [CF3/CF_{final}]_{OP} \quad (11)$$

$$R(CF_{final},CF4) = [CF1+CF3 - 2*CF1*CF3*CF4]_{OP}* [CF4/CF_{final}]_{OP} \quad (12)$$

To try out for effect, we can state some values to the CFs. Suppose CF2=CF3=CF4=0.8 and CF1=0.6<CF2. By Eq.(4.a), the nominal value of the final certainty becomes

Transactions on Information and Communications Technologies vol 19, © 1997 WIT Press, www.witpress.com, ISSN 1743-3517

CF_{final} =0.6*0.8+(1-0.6*0.8)*0.8*0.8
= 0.48+0.52*0.64 = 0.813

The sensitivity values for different CFs are

$S(CF_{final},CF1)$ = 0.8* (1-0.8*0.8) = 0.8 * 0.36 = 0.288
$S(CF_{final},CF2)$ = 0
$S(CF_{final},CF3)$ = 0.8*(1-0.6*0.8) = 0.8*0.52= 0.416
$S(CF_{final},CF4)$ = 0.6 + 0.8 - 2*0.6*0.8*0.8 = 1.4 - 2*0.384 =1.4-0.768 = 0.632

with their relative sensitivity values at the operating point of (CF1, CF2, CF3, CF4)=(0.6, 0.8, 0.8, 0.8) being

$R(CF_{final},CF1)$ = 0.288*0.6/0.813=0.213
$R(CF_{final},CF2)$ = 0
$R(CF_{final},CF3)$ = 0.416*0.8/0.813=0.409
$R(CF_{final},CF4)$ = 0.632*0.8/0.813=0.622

The above means that the induced changes in CF_{final} by CF4 are almost three times as significant as those by CF1 and 30% more significant than those by CF3.

One very interesting thing we want to know is how the final certainty would change for premises of having different certainty factors. Suppose we are using the same rule R1:

If (P1 (CF1) and P2 (CF2)) or P3 (CF3) then C (CF4)

And let CF1= 0.5, CF2 = 0.6, CF3 = 0.7 and CF4 = 0.8. Then we have the final CF being

CF_{final} = 0.74

and the relative sensitivities for the various factors at the operating point of (0.5, 0.6, 0.7, 0.8) are

$R(CF_{final},CF1)$ = 0.253
$R(CF_{final},CF2)$ = 0
$R(CF_{final},CF3)$ = 0.457
$R(CF_{final},CF4)$ = 0.696

This means that the premises with bigger certainty factors are more significant (except in the AND clauses). For example, changing the certainty of premises 1 would only have about 40% the effect of changing the certainty of premise 3 and about one third of the effect due to changes in CF4.

For those tools that use the maximum function for disjunctive certainty factors, their sensitivity analyses are relatively simpler. Let CF1= 0.5, CF2 = 0.6, CF3 = 0.7, and CF4 = 0.8. Then

CF_{final} = CF3*CF4= 0.56

and the sensitivities become

$R(CF_{final},CF1)$ = 0
$R(CF_{final},CF2)$ = 0

$R(CF_{final},CF3) = CF4=0.8$
$R(CF_{final},CF4) = CF3=0.7$

Of course this way of treating premises in the OR clauses causes sensitivities to be 0 and produces more abrupt, nonlinear behavior. It also makes only one of the premises' certainty factors has significance in the final conclusion.

By carefully examining the different conditions for values of CF1 to CF3 we can come up with some very interesting result as shown in Table-1 (with s meaning some value in the table).

CF1	CF2	CF3	CF4	CF_{final}	$S(F_{final},CF1)$	$S(F_{final},CF2)$	$S(F_{final},CF3)$	$S(F_{final},CF4)$
0	0	0	not 1	0	s	s	s	s
1	0	0	not 1	0	0	s	s	s
0	1	0	not 1	0	s	0	s	s
1	1	0	not 1	CF4	1	1	0	1
0	0	1	not 1	CF4	0	0	1	1
1	0	1	not 1	CF4	0	s	1	1
0	1	1	not 1	CF4	s	0	1	1
1	1	1	not 1	s	s	s	s	s
0	0	0	1	0	s	s	s	s
1	0	0	1	0	0	s	s	s
0	1	0	1	0	s	0	s	s
1	1	0	1	1	1	1	0	1
0	0	1	1	1	0	0	1	1
1	0	1	1	1	0	s	1	1
0	1	1	1	1	s	0	1	1
1	1	1	1	1	s	s	s	s

TABLE-1: Checking for Sensitivities based on Some Absolutely Certain and Uncertain Premises

As can be observed we have omitted many cases in the table. For example, we have not taken the uninteresting cases of having CF4= 0 for it immediately drives CF_{final} to 0 for all inputs and also makes all the sensitivities to zero except itself. It should also be noted that the results are not necessarily based on Eqs. (5 - 12) because those equations are supposed to be evaluated at the "normal" operating points. And the values of CFs equaling 0 and 1 are usually considered as boundary cases. Further, the table reveals some very odd nonlinearities that result from certainty factor calculations. It also alerts that changing a value for some certainty factor may produce no effect on the output (as 0 in the table entry) for reasons that may be unrelated to itself.

4 Some Practical Considerations

In the last section, we illustrate how sensitivity analysis is used to validate a model/scheme, namely the certainty factor (CF) model, for handling subjective knowledge in KBs. This section elaborates some practical considerations on using the model.

In practice, the assignment of uncertainties is usually done in an ad hoc manner by the domain expert.[8,14] After all the certainty factors are assigned, the knowledge engineer then tries to adjust the certainty factors to look for changes in output certainty. However, as previous mentioned, this method may not work for many situations. For instance, if there is a rule with many conjunctive premises, *only* one of their certainty factors will affect the certainty of the output. The same also applies to those shells assuming a maximum function for disjunctive premises, letting only one of the premises' certainty factors have effect on the output certainty.

Based on our experience of implementing knowledge-based systems in engineering applications, there are also many instances where, quite surprisingly, the sensitivity to some particular certainty factors are quite independent of its CF values. This usually happens when we are building systems, say for some engineering projects, that a particular engineering step or action is of absolute certainty. Translated into CF, this firm belief turns to CF=1 (for absolutely true) or CF= -1 (for absolutely false). However, such would inadvertently drives sensitivities of certainty factors of other premises to zero.

Another situation that would make the insensitivity phenomenon happen is when the certainty of a knowledge item, be it the premise or conclusion is not obtained in the conventional knowledge-then-certainty manner. For instance, in our construction of an expert system for construction site preparation,[16] we found that it is more natural for the expert to *use the certainty to determine the knowledge value*. Say, if a certainty of 90% (i.e. 0.9) is required, the task should require certain days for its completion. If 70% then other days and so on. An approach characterized by first stating the certainty then value. That is, not practising the knowledge-then-certainty style. Moreover, according to the domain expert, in the stating of certainty, only discrete values will be assumed. Say, a 70% certainty is usually assumed for determining the time for a task. Other certainty levels usually taken are 90% and 50% where the former is for conditions of high certainty and the latter for low degree of certainty. And these three levels of certainty, ie. 90%, 70% and 50% are then set as the general norms for stating the certainty of a decision/advice. Thus, in designing our system for handling uncertainty, instead of a single value three values related to the three levels of certainty for each task have to be acquired from the expert. In order that the system can produce consistent conclusions, a weighted certainty formula for the derivation of certainty is devised where the final CF is calculated as

$$CF_{final} = \frac{(T_{90} \times 0.9) + (T_{70} \times 0.7) + (T_{50} \times 0.5)}{(T_{90} + T_{70} + T_{50})}$$

where T_{90} : the total days of work with 90% certainty
T_{70} : the total days of work with 70% certainty
T_{50} : the total days of work with 50% certainty

Thus in here, the certainties for the knowledge items do not depend on individual certainty factors. Rather it is dependent on the knowledge item (value), here is the number of days for a task. For example, suppose the estimated days for the three levels of certainty for the tasks Setting-out, Hoarding, Cut and Fill are as shown in Fig.3.

Setting Out	Hoarding	Cut	Fill
3 days (90%)	10 (90%)	9 (90%)	5 (90%)
2 days (70%)	10 (70%)	6 (70%)	5 (70%)
1 days (50%)	6 (50%)	6 (50%)	4 (50%)

Fig.3: Estimated Task Work-days with Three Levels of Certainty

The sensitivities for 90%, 70% and 50% certainties are then obtained based on the item values of 3:2:1, 5:5:3, 3:2:2 and 5:5:4 for the tasks of Setting-out, Hoarding, Cut and Fill respectively. In other words, if the value is higher, its sensitivity will be higher. This is also consistent with real life situations as validated by our domain expert. More on the related expert system can be found in [16].

5 Discussions

Sensitivity analyses can also be applied to knowledge bases in other ways. For instance, we can run a test case, look at the outputs, and then during a consultation change a value for some attribute until the output change is observed. This change in behavior would then be elicited for further tuning of the knowledge-base. All values of all attributes would have to be given, along with the change in the target value that caused the change in output. Normally, such technique should be used on large complex systems, but for such systems the analysis would be too complex to be described here given the limited space. Nonetheless, we can illustrate the technique below with a simple example.

Suppose we have the following rules for a classification type knowledge-base.[4]

Domain Table:	Rules:
A=[a1, a2, a3]	If A = a1 AND b=b1 then O=x1
B=[b1, b2, b3]	If B = b1 AND C=c1 then O=x2
C=[c1, c2, c3]	If C = c1 AND A=a1 then O=x1

D=[d1, d2, d3]	If D = d1 AND A=a1 then O=x1
E=[e1, e2, e3]	If C = c2 AND A=a3 then O=x2
O=[x1, x2, x3]	If A=a2 AND B = b2 then O=x2

First let's set each attribute equal to its first listed domain value as shown. These values produced the final output of {x1, x2}. Next we performed a sensitivity analysis on this knowledge base. In separate runs we assigned each of the legal values to the individual attributes. We found that changing the values of {A, B, C} may cause changes in the output. However, changing the value of E did not change the output. Thus the recommendation was to delete this attribute from the knowledge base to turn it into a simpler one. Nonetheless, the problem with this technique is that there is no systematic way to evaluate the results of the analysis. Moreover, the operation is also relatively ad hoc.

6 Conclusion

In this paper, we have demonstrated that sensitivity analysis can be a powerful tool for the validation of knowledge-based systems. Our analysis also reveals that

1. Certainty factors obtained by calculation during system consultation may render certain knowledge items irrelevant to the final conclusion. It is therefore that certainty factors directly obtained from the domain expert should be pursued as they are always more relevant.
2. In many cases, clauses with larger certainty factors have more significance than clauses with smaller certainty factors.
3. In many practical cases, for several reasons, the final certainty is often completely insensitive to many premises' certainties.

Besides telling when and where insensitivity of attributes occurs, the technique also aid in the fine-tuning of knowledge-bases.

References

1. Bahill, A. T. *Bioengineering: Biomedical, Medical and Clinical Engineering*, Prentice-Hall, Englewood Cliffs, NJ, 1981.
2. Bahill, A. T., Latimer, J. R. and Troost, R. T. Sensitivity analysis of linear homomorphic model for human movement. *IEEE Trans. Syst. Man Cybern.*, 1980, **10,** 924-929.
3. Buchanan, B.G. and Shortliffe, E.H. *Rule-Based Expert Systems*, Addison-Wesley, Reading, MA, 1984.
4. Clancey, W. J. Heuristic Classification, *Artificial Intelligence*, 1985, **27** (3), 289-350.
5. Ford, A. and Gardiner, P.C. A new measure of sensitivity for social system simulation models, *IEEE Trans. Syst. Man Cybern.*, 1979, **19,** 105-114.
6. Frank, P.M. *Introduction to System Sensitivity Theory*, Academic Press, NY, 1978.

7. Gonzalez, A.J. and Dankel, D.D. *The Engineering of Knowledge-based Systems*, Prentice Hall, Englewood Cliffs, NJ, 1993.
8. Kanal, L. N. and Lemmer, J. F. (eds.) *Uncertainty in Artificial Intelligence*. North-Holland, Amsterdam, 1986.
9. Lehman, S. and Stark, L. Simulation of linear and nonlinear eye movement models: sensitivity analyses and enumeration studies of time optimal control, *J of Cybernatics Information Science*, 1979, **4**, 21-43.
10. Leimanis, E. and Minorsky, N. Dynamics and nonlinear mechanics, *Surveys of Applied Mathematics*, Wiley, 1958.
11. Nguyen, T.A. et al., Checking an Expert Systems Knowledge Base for Consistency and Completeness, in *Proc. of the 9th IJCAI*, 1985, 375 - 378.
12. Nguyen, T.A. et al., Knowledge Base Verification, *AI Magazine*, 1987, **8**(2) (Summer), 69-75.
13. O'Keefe, R.M., Balci, O. and E.P. Smith. Verifying and Validating Expert System Performance, *IEEE Expert*, 1989, **2(**4) (Winter), 81-90.
14. Shortliffe, E. H. *Computer-Based Medical Consultations: MYCIN*. Elsevier, NY, 1976.
15. Suwa, M., A. C. Scoff, and E. H. Shortliffe. An Approach to Verifying Completeness and Consistency in a Rule-Based Expert System, *AI Magazine*, 1982, **3**(3) (Fall), 16-21.
16. Wu, A. K. W. A Knowledge-based Application in Engineering. (Paper submitted for publication in *Advances in Engineering Software*)
17. Zadeh, L. A. Fuzzy sets as a basis for a theory of possibility. *Fuzzy Sets and Systems*, 1978, **1**, 3-28.