DEVELOPMENT OF AN EXPERT SYSTEM FOR ULTRASONIC FLAW CLASSIFICATION

Lester W. Schmerr, Jr., Ken E. Christensen, and Stephen
M. Nugen

Center for NDE
Iowa State University
Ames, IA 50011

INTRODUCTION

The complete characterization of a flaw requires information about
the flaw type (crack, void, inclusion, etc.), flaw size, and orientation.
Here we are only concerned with the determination of the flaw type so
that the appropriate sizing algorithms can be chosen. This type of
classification problem using ultrasonic waves is very suitable for employ-
ing the tools and techniques of artificial intelligence [1,2]. Adaptive
learning methods, for example, have in the past been employed to train
a flaw classification module so that it can distinguish between cracks
and volumetric flaws [3]. Some of the limitations of this approach,
however, have been due to the empirical nature of the features used
for classification and the difficulty of understanding and adjusting
the decision-making process when errors occur.

In contrast, we have chosen to develop a classification scheme
in the form of a rule-based expert system where the features used by
the system for classification come from model-based fundamental knowledge,
and where the rules are made explicit and modifiable. In this paper
we will describe the nature of the expert flaw classification system
we are building and demonstrate its use with some ultrasonic data.
As currently constituted, the domain of knowledge of the system is highly
constrained. The flaw classifier is concerned with distinguishing between
single isolated volumetric and crack scatterers. The design of the
system, however, is such that these constraints are not essential.

SYSTEM OVERVIEW

As outlined in Fig. 1, the expert flaw classification system, FLEX
(Flaw Expert), consists of essentially four major components: 1) a
user-interface that allows the visual display and manipulation of ultrasonic
data, 2) a set of tools that allow a user to manipulate and modify the
rules of the system, 3) a module called FEAP (for FEAture Processing),
and 4) a module called FLAP (for FLAw Processing). FEAP and FLAP are
being designed as two separate, cooperating expert systems.

It is the job of FEAP to take the preprocessed ultrasonic data from a given experiment, and determine confidence factors associated with each feature being used by the system. These confidence factors are numbers in the range [-1,1], where -1 indicates complete certainty that a feature is not present, 1 indicates complete certainty that a feature is present, and numbers in between indicate the degree of certainty or uncertainty (see Appendix). Both FEAP and FLAP manipulate these confidence factors according to the conventions and methods developed by Shortliffe and Buchanan for the MYCIN project [4]. FEAP also determines the percentage of the ultrasonic data sets, if there are more than one, in which there is positive evidence (positive confidence factors) for each feature. Currently, FEAP assumes that the data it uses has had non-flaw dependencies removed through the application of the measurement model of Thompson and Gray [5]. This preprocessing is done so that we can rationally evaluate features characteristic of the flaw type only. FEAP uses a combination of fundamental knowledge of flaw scattering properties, and heuristic knowledge based on our familiarity with actual experimental data. For example, a Kirchhoff model of how a flat crack behaves at normal incidence indicates [6] that in the frequency domain we can expect a linearly increasing amplitude with increasing frequency (solid line in Fig. 2a). A more exact numerical model of the scattering process verifies this linear behavior, but indicates

PRE-PROCESSED
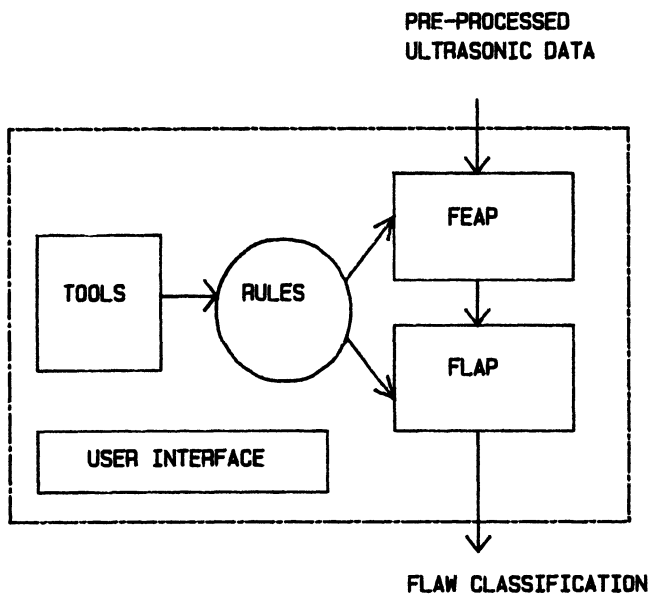ULTRASONIC DATA

FLAW CLASSIFICATION

Fig. 1. Outline of components of FLEX.

that a modulation also exists (dotted line in Fig. 2a). In principle, therefore, we would expect to seek such a characteristic feature over the entire bandwidth of our experimental ultrasonic system. In reality, however, experimental results (see Fig. 2b) show the presence of this feature only up to about the center frequency of the transducer. This type of heuristic knowledge is then factored into our actual search for, and evaluation of, this feature.

Currently, the module FEAP is in an early stage of development. We have outlined a set of decision trees, for extracting all the features and their associated confidence factors, and are now in the process of automating that extraction process. However, because of the modular nature of the system, flaw classification evaluations are still possible by having a human operator replace FEAP and provide the necessary confidence factors to FLAP. Below we will outline a sample application of FLEX to some ultrasonic data which does the feature evaluation in just that manner.
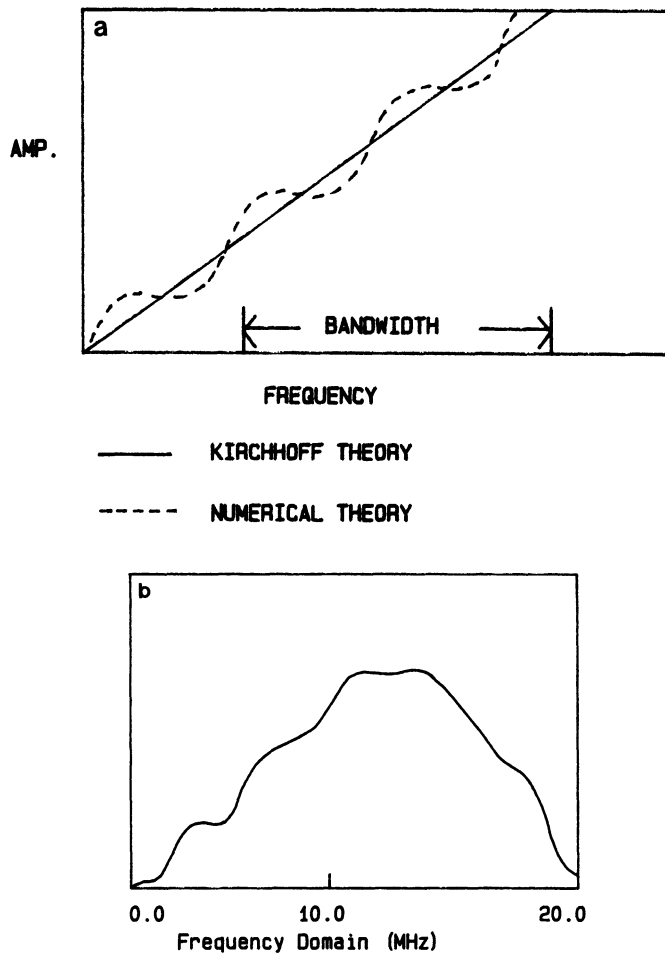

Fig. 2. a) Theoretical response of a crack at normal incidence. b) 400 micron crack in IN100.

Flaw Processing (FLAP)

Once the feature evaluation process is complete, FLAP evaluates the evidence produced by FEAP according to an explicit set of rules and reaches a conclusion as to flaw type. Currently, we are using nine features and a corresponding set of nine rules to perform the classification. These nine features (see Table 1) were chosen because analytical, numerical, and approximate models of the scattering process show that they can be expected to help distinguish between single, isolated volumetric and crack-like scatterers. For more complicated geometries (flaws near a surface, etc.) and flaw types (porosity, microcracking, etc.), these features would have to be modified and/or supplemented by other flaw features. As indicated in Table 1, both time and frequency domain features are used in the evaluation process. Having multiple domains has been found, in our test cases, to be particularly useful for handling noise and other experimental system inaccuracies.

Table 1. Features used in FLEX for classification of an isolated flaw as to flaw type (volumetric or crack) and the corresponding domain in which the feature is defined.

	Feature	Domain
1.	Positive-Leading-Edge-Pulse	Time
2.	Flash-Points	Time
3.	Rayleigh-Wave	Time
4.	Creeping-Wave	Time
5.	Ringing	Time
6.	Linearly-Increasing-Amplitude	Frequency
7.	Plateau-With-Shallow-Nulls	Frequency
8.	Decreasing-Amplitude-With-Deep-Nulls	Frequency
9.	Sharp-Nulls	Frequency

The flaw classification rules used by FLAP exist in two forms: external and internal. The external form of the rules is designed for English-like readability and easy modification by non-programmers. The external rules are in the form of (if... then ...) types of conditional statements where the "if" part of the rule is called the antecedent part of the rule and the "then" part is the consequent. A typical external rule in FLAP is:

```
(Rule 204
(if      ringing is detected in the trailing response of at least 50
         percent of the flaw signals in the time domain)
(then    there is weak belief in the accumulating evidence supporting
         the determination of a volumetric flaw)
```

There are four important pieces of such a rule: 1) the feature it applies to, 2) the percentage of flaw signals where this feature exists, 3) the qualification (e.g., weak belief) on the strength of evidence associated with this feature, and 4) the type of evidence (accumulating or non-accumulating). The percentage value is used as a threshold to decide if the evidence is sufficient to warrant invoking the rule. This gives us, in a simple manner, a way to account for uncontrollable uncertainties in the system, such as noise, and to minimize their influence on the flaw classification process. Similarly, the use of phrases such

as "weak belief" in our rules allows us to control the "weight" of the evidence associated with that rule. Note that, unlike adaptive learning systems, such weights are explicit in their meaning and modifiable in clear English text. The use of accumulating and non-accumulating evidence allows us to merge the existence of strong features (non-accumulating evidence), where the existence of a single data set with this feature is sufficient to add this evidence into our conclusion, and less strong features (accumulating evidence) that are indicative of flaw type, but where an average value over all data sets is taken as the net evidence for this feature. Actually, our feature evidence is categorized into three classes:

(1) Sufficient. Example: The unique response of linearly increasing amplitude in the frequency domain is sufficient evidence to conclude that the flaw type is a crack. This type of evidence is considered to be non-accumulating.

(2) Necessary. Example: We always expect to see a negative leading edge pulse in the time domain for cracks. If positive evidence exists for a positive leading edge pulse, we record negative accumulating evidence for cracks.

(3) Indicative. Example: Ringing, or resonance, in the time domain is indicative of a volumetric flaw. When a net positive evidence is found for this feature, we record positive accumulating evidence for a volumetric flaw.

The external rules of FLAP are automatically translated into internal form as part of the rule modification tools of FLEX. This internal rule form is designed for simple evaluation by the program. For example, the above Rule 204 would translate to:

```
(RULE 204
      (IF (>= (FEAT-PC (QUOTE RINGING)) 50))
      (THEN (SETQ ACC_VOL_EVD
              (CONS (* 0.3 (ZERO-CLIP (FEAT-CF (QUOTE RINGING))))
                    ACC_VOL_EVD))))
```

Once all the external rules are translated into such internal forms, it is the task of FLAP to evaluate these rules and reach a conclusion based on the confidence factors and percentage values provided by FEAP (or an equivalent human operator). This part of FLAP is a simple inference engine whose actions can be summarized as follows:

Rule evaluation consists of two phases: evaluating the rule antecedent and evaluating the rule consequent. Recall the rule antecedent is the conditional or "if" clause. If the conditional is true, then the consequent or "then" clause of the rule is evaluated for side effects, i.e., making an entry into one of the evidence lists supporting the two hypotheses. At the current time, FLAP selects every rule for evaluation without regard to order or weight. A rule is said to "fire" if the antecedent is true.

Another capability of FLAP is its ability to explain the results of its classification. We have chosen to implement this feature of the system in the form of an audit trail which provides a summary of the rule firings and the evidence (or lack of it) which caused the particular conclusions to be reached. An example of such an audit trail for a specific problem will be given in the next section.

The conclusion of the system is in the form of the total evidence (confidence factors) for both volumetric and crack-like flaws). Typically, there may be positive evidence for both types of flaws. If, however, there is a wide enough variation in this evidence, a firm conclusion can be reached.

FLEX is currently being developed on a Symbolics 3670 system using Symbolics extended Common Lisp. The user interface, which employs bit-mapped graphics and a mouse, is necessarily system dependent. The overall architecture, knowledge representation scheme, and inference strategy are not system dependent. Our expectation is that production-oriented implementations will be possible on a variety of 32-bit microcomputers.

AN EXAMPLE FLAW CLASSIFICATION

To see some of the elements of the behavior of this system, we have given in diagram 1 an outline of the application of FLEX to the classification of an artificial 400µm radius crack placed in a sample of IN100. Thirteen different time domain waveforms and corresponding frequency domain results were available from this sample through the use of the multi-viewing transducer system developed at Iowa State by Dr. D. O. Thompson and his co-workers [7]. Each of the waveforms or spectra corresponded to a different "look-angle" at this flaw. In diagram 1, we have followed the behavior of this sytem by examining in detail, for this particular example, one of the nine rules, Rule 202, and its consequences. Diagram 1 shows the external form of this rule and the internal form that this rule is translated into by the Translator. For this particular example, a human operator, using the visual display features of FLEX, examined all thirteen look-angles and provided estimates of the confidence factors associated with each feature. This data was fed into the inference procedure of FLAP and the conclusion shown was drawn. As mentioned previously, there is typically evidence for both flaw types, as we see here. However, we also see that the difference in confidence values is strong enough so that one could, with moderate confidence, conclude that this was a crack.

By invoking the explanation facility, we can see the reason why each rule did or did not fire. In the case of Rule 202, we found flash-points in 100 percent of the look-angles so the threshold of 50 percent was exceeded and the rule fired. The value of 0.68 given is the average confidence factor given for this feature over the thirteen look-angles. This value is multiplied by 0.5 to factor in the weight of the evidence (moderate belief) for this feature (see Appendix). Following this explanation in diagram 1, we see a tabulation of all the non-accumulating and accumulating evidence from all the rules for this example. Rule 202 is seen to provide an entry in the accumulating crack evidence list (ACC-CRK_EVD) as it should. Finally, we note that the total confidence factors given in the conclusion are just the "sum" of all the accumulating or non-accumulating evidence for crack or volumetric, where the "sum" is labelled with an M superscript to indicate it is actually carried out according to the methods developed for the MYCIN expert system [4] (see also the Appendix).

SUMMARY AND CONCLUSIONS

We have shown some of the major elements of FLEX that are currently operational. As mentioned, the automating of the feature extraction portion of the system (FEAP) is a particularly challenging task that we are now undertaking. Extension of this system to do intelligent signal preprocessing and post-classification flaw sizing is also possible because of the very modular nature of the system. However, probably

more important from a practical standpoint, is the ability of the system
to handle different testing needs and classification problems. The design
of the system will also take into account this important necessity.

ACKNOWLEDGEMENTS

APPENDIX

Confidence Factor Notes

1. Confidence Factors (CF's) are not the same as probabilities.
Particularly, a CF of N for conclusion X does not imply a CF of (1-N)
for conclusion not-X. For this reason, CF's are usually calculated
and manipulated with heuristics developed as part of the expert system.

2. The CF for a particular feature evaluation can be regarded
as the difference between the belief that the feature is present and
the belief that the feature is not present. That is,

   CF = MB - MD

MB = Measure of Belief. $0 =< MB =< +1$
MD = Measure of Disbelief. $0 =< MD =< +1$
CF = Confidence Factor $-1 =< CF =< +1$

3. For uncertain judgements, we partition the range of CF $[-1,+1]$
into nine non-overlapping subranges:

   $-1.0 =< CF =< -0.9$ ------> Certain disbelief
   $-0.8 < CF =< -0.6$ ------> Strong disbelief
   $-0.6 < CF =< -0.4$ ------> Moderate disbelief
   $-0.4 < CF < -0.2$ ------> Weak disbelief

   $-0.2 =< CF =< +0.2$ ------> Uncertainty

   $+0.2 < CF < +0.4$ ------> Weak belief
   $+0.4 =< CF < +0.6$ ------> Moderate belief
   $+0.6 =< CF < +0.8$ ------> Strong belief
   $+0.8 =< CF =< +1.0$ ------> Certain belief

These subranges have two purposes:

      a. When explaining a conclusion to an end user, map the CF
values to the applicable partition name.

      b. When getting uncertain judgements from users, map them to
the CF value which is the midpoint of the applicable partition. For
example, if a user's confidence in his/her evaluation of ringing is
"moderate belief", then assign a value of +0.5 to the CF.

4. The $\overset{M}{+}$ operation for summing accumulating evidence (see the
note in Diagram 1) is borrowed from MYCIN [4]. For example, in adding
two positive confidence factors CF1 and CF2, the sum is defined as:

$$\overset{M}{CF1 + CF2} \equiv CF1 + (1-CF1)*CF2$$

Note that this sum is independent of the order of adding CF1 and CF2.

REFERENCES

1. S. M. Weiss, and C. A. Kulikowski, A Practical Guide to Designing Expert Systems, Rowman and Allanheld, Totowa, NJ, 1984.
2. E. B. Hunt, Artificial Intelligence, Academic Press, NY, 1975.
3. M. F. Whalen, L. J. O'Brien, and A. N. Mucciardi, Proceedings of the DARPA/AFML Review of Progress in Quantitative NDE, AFWAL-TR-80-4078, 1980.
4. B. G. Buchanan, and E. H. Shortliffe, Rule-Based Expert Systems, Addison-Wesley, Reading, MA, 1984.
5. R. B. Thompson, and T. A. Gray, J. Acoust. Soc. Am., 74, 1297-1290, 1983.
6. Chien-Ping, Chiou, Unpublished Master's Thesis, Iowa State University, Ames, IA   50011, 1986.
7. D. O. Thompson, and S. J. Wormley, in, Review of Progress in Quantitative NDE, D. O. Thompson and D. E. Chimenti, Eds., (Plenum Press, NY, 1985), pp. 287-296.

Diagram 1. An example flaw classification of an artificial circular crack in IN100.

● EXTERNAL RULES
  ●
  ●
  ●
(RULE 202
(IF   FLASH POINTS ARE DETECTED IN THE LEADING EDGE RESPONSE
        OF AT LEAST 50 PERCENT OF THE FLAW SIGNALS IN THE TIME DOMAIN)
(THEN   THERE IS MODERATE BELIEF IN THE ACCUMULATING EVIDENCE SUPPORTING
        THE DETERMINATION OF A CRACK)
  ●
  ●
  ●   TRANSLATOR
● INTERNAL RULES
  ●
  ●
  ●
(RULE 202 (IF (>= (FEAT-PC (QUOTE FLASH-PTS)) 50)
(THEN (SETQ ACC-CRK-EVD (CONS(* 0.5 (ZERO-CLIP)
(FEAT-CF (QUOTE FLASH-PTS)))) ACC-CRK-EVD))))
  ●
  ●
  ●
● FEATURE EVIDENCE   INFERENCE PROCEDURE
  ●
  ●
  ●   CONCLUSION
● THE CONCLUSION: THE FLAW TYPE IS A CRACK WITH CONFIDENCE 0.51
    (THE CONFIDENCE IN VOLUMETRIC IS 0.18)
  ●
  ●
  ●

● EXPLANATION FACILITY ─────────→ AUDIT TRAIL
   ● |
   ●   ↓
   RULE 202 FIRED. FLASH-PTS DETECTED IN 100.0 PERCENT OF THE
     SIGNALS (50 PERCENT NEEDED), THE RESULTING CONTRIBUTION
     TO ACC-CRK-EVD IS (0.5 * 0.68 = 0.34).
   ●
   ●
   (NAC-CRK-EVD 0.0)
   (ACC-CRK-EVD (0.175 0.108 0.34))
   (NAC-VOL-EVD NIL)
   (ACC-VOL-EVD (0.03 0.075 0.081)
                      M   M
● NOTE: 0.51 = 0.175 + 0.108 + 0.34
                      M   M
         0.18 = 0.030 + 0.075 + 0.081

887