Expert System Technology for Nondestructive Waste Assay

J.C. Determan, G.K. Becker

Idaho National Engineering and Environmental Laboratory
P.O. Box 1625
Idaho Falls, Idaho 83415
jcd@inel.gov

Abstract

A system combining genetic algorithms and a fuzzy-rule induction routine has been developed. Two prototype expert systems, one derived analytically, and one derived empirically by the fuzzy-rule induction system, were developed to evaluate the utility of expert system technology relative to waste nondestructive assay (NDA) data review. Technical review of waste NDA measurement data, though warranted with respect to present day waste NDA system capabilities, is labor intensive. Hence it is desirable to have an automated system to perform technical review. It has been shown that both the analytically and empirically derived expert systems produce reasonable results, but that the automatic system can produce fuzzy rules more efficiently and accurately than the analytical method. A visual explanation facility for fuzzy expert systems has also been developed.

Content areas: AI systems, expert systems, fuzzy reasoning, inductive learning, applications.

Tracking number: I-34

Introduction

Management of U. S. Department of Energy (DOE) defense generated containerized transuranic (TRU) waste requires determination of the entrained TRU material quantity and associated parameters. Nondestructive assay (NDA) techniques are the most common and efficient means to determine the TRU material quantity. Quality assurance objectives (QAOs) for NDA techniques used to characterize TRU waste destined for the Waste Isolation Pilot Plant are delineated in the National TRU Program Transuranic Waste Characterization Quality Assurance Program Plan (QAPP) (DOE 1996).

Technically justifying compliance with applicable requirements and QAOs for the variety of TRU waste forms in the DOE inventory can be a complex process. Some waste form configurations manifest NDA system response complexities that diminish the ability to clearly establish compliance. Such complexities require that a technical review be performed at the data generation level for each assay to ensure operational boundaries are maintained relative to QAPP requirements.

Technical review of waste NDA measurement data, though warranted with respect to present day waste NDA system capabilities, is labor intensive. Hence it is desirable to have an automated system to perform the technical review. Therefore, an evaluation of methods to represent knowledge, reason, and make decisions was undertaken. The automated system must be capable of providing a comprehensive waste assay data assessment, and must be reproducible, auditable and compatible with the overall throughput requirements of the waste characterization process. Expert system technology is under consideration to perform this task.

Analytically Derived Expert System

The prototype analytically derived expert system consists of a module of fuzzy rules characterizing the quality of waste assay TRU mass estimates, as measured by the Stored Waste Examination Pilot Plant (SWEPP) passive-active neutron assay system (SAS). The expert system was built using the package FuzzyCLIPS (Orchard 1995), a fuzzy variant of the expert system shell CLIPS (Giarratano 1994). The input to the expert system consists of various SAS neutron measures for a given waste assay. These are processed into figures of merit and passed to the expert system module. The expert system operates on these inputs to arrive at a set of confidence values for the particular assay. The expert system output is comparable to the confidence a domain expert would assign to mass estimates resulting from the assay in terms of compliance with the National TRU Program QAOs.

SAS Fundamentals

The analytically derived expert system design is predicated on the operating principles of the neutron detection, acquisition and data reduction technique implemented in the SAS. The means by which the SAS detects the presence of TRU materials, processes detected signals and reduces the information to a mass estimate defines the attainable functional performance and resulting assay validity. Waste form configuration also influences the

mass estimation routine. Hence, the SAS neutron measures input to the expert system must embed base

The SAS has two primary modes of operation: passive neutron detection and active neutron interrogation. The SAS is currently calibrated to assay the quantity of ^{240}Pu in the passive measurement mode, and ^{239}Pu in the active mode. Selection of either the ^{240}Pu or ^{239}Pu mass as the most appropriate measure to apply to the waste container assay is based on criteria coded in the SAS algorithm.

Physically, the SAS neutron detection system consists of an aluminum structure supporting a top, bottom and four side walls, which contain neutron detection assemblies. The SAS chamber accommodates a 55 gal waste drum.

All six sides of the chamber contain two types of detection assemblies: bare and shielded. The shielded assemblies are preferentially sensitive to fast neutrons, while the bare detection assemblies are sensitive to both slow and fast neutrons. This dual response system allows a degree of neutron energy discrimination. The combined response of the bare and shielded assemblies is termed the system response.

Neutrons detected in the passive mode are the result of TRU spontaneous fission and (α,n) neutron processes. The spontaneous fission neutrons are time-correlated and distinguished from the (α,n) neutrons, which are not time-correlated, through a data acquisition technique known as coincidence event counting. In this type of counting, a coincident event is recorded when two or more neutrons are detected by the system within a specified time window. The coincidence signal is converted to the desired Pu mass using appropriate conversion factors.

In the SAS active mode, the shielded detector assemblies are used to detect neutrons produced by stimulated fission resulting from thermal neutron interrogation. The interrogation neutron source is a Zetatron 14 MeV neutron generator located inside the cavity. The signal of interest is taken from a time gated count of the shielded detectors from 700 μsec to 2700 μsec following each interrogating neutron burst. To account for the neutron background that may be included in this measure, another count window is opened from 5.7 to 15.7 msec after each neutron burst. The net induced fission neutron signal used for quantifying the Pu mass is arrived at by taking the difference of these signals. The active mode also employs the response of two additional neutron counters: the cavity flux monitor and the barrel flux monitor. These monitors are used to acquire information regarding neutron moderation and absorption processes occurring in the waste matrix.

Analytically Derived System Input Variables

A number of signals from the SAS signal processing modules provide output carrying the response of the passive and active detection assemblies and active flux monitors. These signals are combined into figures of merit that characterize the system response, and become the expert system input variables. Table 1 tabulates the input variables for each of the three SAS mass estimation modes. The selected variables are not all inclusive, but contain sufficient SAS neutron response information to assess expert system utility for assay data review.

Input Variable	Derivation
Passive System Mode	
Passive System Total Rate	System Total Counts/ (10 kHz clock ticks/10000)
System Total/ Gated Event Ratio	System Total Counts/ Long Gate System Events
Early Barrel/ Cavity Count Ratio	Barrel Flux Monitor/ Flux Monitor
Passive Shielded Mode	
Passive Shielded Total Rate	Shielded Total Counts/ (10 kHz clock ticks/10000)
Shielded Total/ Gated Event Ratio	Shielded Total Counts/ Short Gate Shielded Events
Early Barrel/ Cavity Count Ratio	Barrel Flux Monitor/ Flux Monitor
Active Mode	
Active/Passive Shielded Rate Ratio	(Shielded Counts/4)/ Passive Shielded Total Rate
Early/ Late Count Ratio	Shielded Counts/ Shielded Background Counts
Early Barrel/ Cavity Count Ratio	Barrel Flux Monitor/ Flux Monitor

Table 1. Expert system input variables.

Input Variable Membership Function Definitions

The figures of merit discussed in the previous section comprise the input variables to the expert system. The fuzzy variables *Low*, *Medium* and *High* are assigned by the domain expert to span the range of possible values for each input variable by defining appropriate membership functions. The membership function definitions are based on known SAS responses associated with the inventory of waste forms to which the SAS will be applied. Figure 1 illustrates a typical set of membership functions.

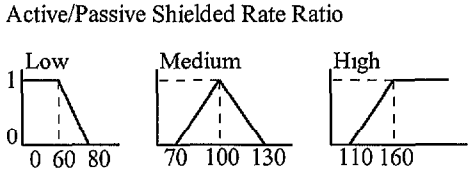

Figure 1. Typical membership function definitions.

Analytically Derived System Fuzzy Rules

To form the set of fuzzy rules, a table is constructed listing all possible combinations of the three input variables (i.e., *Low*, *Medium* and *High)* for each SAS measurement mode. The domain expert then makes a judgment for each indicator combination for each measurement mode as to the level of confidence that should be attributed to the assay estimate for a particular combination. Examples of some active mode confidence rules are provided in

Table 2. Similar rule sets exist for the passive system and passive shielded SAS response modes.

Each of the confidence consequents is also defined by membership function specifications. The confidence membership functions required calibration and are discussed in the Test Results section.

Rule ID	Antecedents			Consequent
	Active/ Passive Shielded Rate Ratio	Early/ Late Count Ratio	Early Barrel/ Cavity Count Ratio	Active Mode Confidence
A3	High	High	High	High
A4	High	Medium	Low	Low
A12	Medium	High	High	Medium
A22	Low	Medium	Low	Low

Table 2. Active mode confidence rules.

Explanation Facility

Decisions in a fuzzy expert system are not arrived at via a simple causal chain: Producing an explanation for a given decision is more difficult in a fuzzy expert system than in a crisp expert system. In a fuzzy expert system, every fact that supports a given conclusion contributes in some degree to the final decision, and an explanation becomes an attempt to represent the influence of each fact on the final decision. Thus, an inverted tree structure was chosen to represent the decision process. Individual facts are shown in the top level of the structure (i.e., input variable names and their associated values). The next level displays the names of the rules that were fired to arrive at a given decision, with lines of varying thickness connecting the facts to the rules. The match strength between an individual fact and its associated rule antecedent is proportional to the interconnecting line thickness. The bottom level of the structure is a single node representing the decision, in this case an assay confidence value. The line thickness for the line connecting a rule to the final decision represents the overall match strength of the rule. The rule match strength is the minimum match strength of the facts that support the rule, hence the rule strength line thickness is the minimum line thickness of the lines connecting facts to the given rule.

Empirically Derived Expert System

Eliciting expert knowledge through the formulation of fuzzy rules was resource-intensive; therefore, methods of automatic fuzzy rule generation from data were examined (Yager and Filev 1994) (Chiu 1994). One method of fuzzy rule generation consists of rule estimation by subtractive clustering, followed by gradient descent optimization. This process is repeated many times under the control of a genetic algorithm in order to find an optimal rule set. The rule set derived in this manner is referred to as the empirically derived expert system.

The automatic rule generation process is an instance of supervised learning; therefore, correctly classified training and test data must be provided. The input data is some set of figures of merit that a domain expert has deemed significant. In this instance, the same figures of merit used in the analytically derived expert system were used in the empirically derived expert system. The training and test data was classified by a panel of domain experts.

Automatic Rule Generation Procedure

In subtractive clustering, the data are viewed as input/output vectors in a hyperspace, the dimension of which is the sum of the number of input and output elements. A cluster of data points in this hyperspace represents an approximate relationship from input to output, a relationship that can be represented as a fuzzy rule. The number of clusters to be found is not directly specified; instead, a parameter between zero and one called the cluster radius is specified. The number of clusters found increases as the cluster radius decreases.

In general, for n input and m output elements in each datum, the generated fuzzy rules take the following form:

if (x_1 matches A_1) and ... (x_n matches A_n)
then (y_1 is B_1) and ... (y_m is B_m), where

x_1 ... x_n are normalized rule input values,
y_1 ... y_m are rule output values,
A_1 ... A_n are exponential membership functions, and
B_1 ... B_m are symmetric membership functions.

The use of symmetric output functions allows for a simpler solution, in that only the centroids of the output membership functions need be determined. The input membership functions are expressed as follows:

$$A_j(x_j) = \exp(-.5((x_j-x_j^*)/\sigma_j)^2), j=1..n, \text{ where}$$

x_j^* and σ_j are derived as discussed below.

For simplicity, the remainder of the discussion will focus on the more specific case of one output element in each rule, or m = 1. This simplification is employed because the assay confidence assignment is the only output element.

The result of the calculation $A_j(x_j)$ is referred to as the match strength of the input x_j with the associated membership function A_j. The match strength of a rule with multiple inputs is the product of the component match strengths. This differs from the analytically derived expert system, in which the rule match strength is the minimum of the input match strengths. The rule strength is given the symbol μ and is expressed as:

$$\mu(x_j) = \exp(-.5 \Sigma ((x_j-x_j^*)/\sigma_j)^2), j=1...n.$$

Using a standard method of defuzzification (the center of

gravity algorithm), the output of the system is found from:

$y = \Sigma_{i=1}(\mu_i y_i^*)/\Sigma_{i=1}\mu_i$, i=1...l, where

y_i^* is a vector of output membership function centroids.

The membership function parameters x_j^* and y_i^* are initialized from the cluster center coordinates, and the parameters σ_j are initially derived from the cluster radius, where there is one vector x_j^* and σ_j, for each rule in the system, resulting in the matrices x_{ij}^* and σ_{ij}. The y_i^* are centroids of the symmetric output membership functions B. The rules as initialized with data found in the clustering phase are a first approximation.

Gradient descent optimization is performed on the parameters x_{ij}^*, y_i^*, and σ_{ij} to fit them to the training data. This process is analogous to back-propagation training of neural networks. The following equation is the basic equation for back-propagation in neural networks and gradient descent optimization of fuzzy rule parameters:

$$z_{new} = z_{old} - \alpha\, \delta E/\delta z,$$
$$E = 0.5\, e_k^2$$

where z could be an interconnection weight in a neural network, or any of the parameters σ_{ij}, x_{ij}^* or y_i^*. The term, e_k, is the error between the predicted and assigned confidence value of a given assay, and α is the learning rate. The learning rate is a value between zero and one, where low values accelerate convergence. Convergence is achieved when the root-mean-square (RMS) error of the system with respect to the training data becomes relatively constant (i.e., the difference in successive iteration-averaged RMS values is below a specified criterion).

The back-propagation method described in (Yager and Filev 1994) employed incremental back-propagation. Other enhancements to back-propagation commonly used in the training of neural networks (Hassoun 1995) were employed in the current application, including a variable learning rate and the inclusion of a momentum term. The learning rate was initialized to a user specified value, and decreased in finite steps as the convergence criteria was approached, resulting in an initially elevated learning rate, to accelerate convergence, and a reduced learning rate as convergence progressed, improving the effectiveness of the training process. The momentum term is intended to accelerate convergence, and involves a user specified factor between zero and one. Higher values of this factor increase the rate of convergence, but may induce numerical instabilities. A momentum factor set to 0.1 resulted in stable calculations that quickly converged.

Genetic Algorithm Search for Rule Generation Parameters

The clustering and optimization procedures just discussed require the specification of a cluster radius, a learning rate, and a convergence criteria. The performance of the generated system is highly dependent on the values of these rule generation parameters. System generation, given a set of rule generation parameters, can take anywhere from several seconds to several minutes, depending on the number of rule inputs and the values of these rule generation parameters. Because more than one hundred trials were typically required to achieve optimal performance, finding an optimal set of rule generation parameters was labor-intensive and the decision was made to automate the search. Genetic algorithms (Goldberg 1989) were selected to control the procedure. The C++ genetic algorithm class library developed at the Massachusetts Institute of Technology called GAlib, version 2.4.2, was acquired and employed in this effort. The application of genetic algorithms in fuzzy-system design has been studied by a number of authors (Heider and Drabe 1997) (Takagi and Lee 1993) (Karr 1991).

To use GAlib, an initializer is defined to create a population of potential solutions over the space of valid rule generation parameters. Methods for performing genetic crossover and mutation are either selected from a set of standard methods, or defined for a particular problem. For example, the standard one-point crossover method was employed in the current application. A specialized mutator was defined that varied the value of a gene by a small delta. The set of three rule generation parameters was encoded as three real-valued genes in a chromosome. Finally, an objective function was defined that scored a particular chromosome based on performance of the system with respect to a specified test set. The objective function first trained a set of rules using the rule generation parameters within the current chromosome, and then scored the derived set of rules with respect to the test data. To accelerate the search, if a particular chromosome recurred, training and testing were not repeated, rather, the score for that chromosome was retrieved from memory.

The genetic algorithm employed in the current application assumed a constant population. The children replaced the parents in succeeding generations, except that elitism was employed. Selection of individuals for mating was performed by the standard roulette-wheel method, in which the most fit individuals have the highest probability of being selected for reproduction.

A few parameters were required by the genetic algorithm. The population size was chosen to be six individuals, as this supplied a reasonable initial diversity in the population. The calculation was terminated by specifying a maximum number of generations. Two hundred generations was typically ample to provide thorough coverage of the search space. Two probabilities were also required: the probability of cross-over, and the probability of mutation. Crossover was given a probability of 0.9, as crossover is the primary means of expanding the search space. Mutation was given a probability of 0.1, as some mutation was needed to thoroughly explore the search space, but too much mutation could result in the loss of a good solution from the population. A mutation probability of 0.1 may appear high, but it is appropriate to the number of unique individuals generated during the search. Six individuals per generation multiplied by

200 generations yields a maximum of 1200 individuals. But many duplications occurred, resulting in at most a few hundred unique individuals in the search space. Were the mutation probability to be much lower than 0.1, so few mutations would occur as to be insignificant.

Test Results

The results of testing both systems are presented in this section. To test the performance of the prototype expert systems, a set of SAS generated waste container assays with known validity confidence ratings was generated. Calibration of the analytically derived expert confidence membership functions was also required.

Test Set Normalization

A panel of three waste NDA experts was assembled and given a set of SAS generated waste assays to assign validity confidence ratings, on a scale from zero to ten. They rated their confidence in each of the three measurement modes of the system, for each assay. The set contained 99 assays evenly selected from the waste types, graphite, combustibles, and glass. Disagreement existed between the experts as to the proper classification of each assay, due partly to differing internal scales of judgment used by each expert, and partly to the considerable uncertainty inherent in the interpretation of NDA results. A normalization procedure was adopted to reduce the scaling bias of each expert. Assays with normalized scores that still did not agree between the experts were removed from the test set.

The score sets for each confidence value were normalized as follows. The mean and standard deviation for each expert's set of scores, and for all the scores combined were calculated. Each expert's set of scores was adjusted via calculated scaling factors to have the mean and standard deviation of the total population.

Where there was good agreement for each confidence value between all three experts the normalized confidence scores were considered reliable. Assays for which disagreement occurred were removed, yielding a total of 67 assays in the test set. These assays were used for calibration of the analytically derived expert system, for training of the automated rule generation system, and for testing the performance of both systems. Sixteen assays were used for calibration and training, and 51 were used for testing. The calibration and training sets were selected to thoroughly represent the spectrum of confidence values.

Analytically Derived Expert System Calibration

During initial design of the expert system, the confidence membership functions were arbitrarily set to triangular functions of width 0.33 and centered on 0.17, 0.5, and 0.83. These arbitrary definitions did not produce results consistent with the test data, hence some of the data were segregated and used to calibrate the confidence scale. Table 3 compares the calibrated and uncalibrated system responses, indicating an initial poor level of agreement to test data.

Calibration was performed as follows: the equations used by FuzzyCLIPS to obtain a defuzzified output were set equal to the normalized and averaged test scores for a selected set of 16 assays. The confidence level outputs produced by FuzzyCLIPS are determined by the center of gravity defuzzification algorithm. One equation of the following form was derived for each of the 16 calibration data points:

$\Sigma C_i x_i^*$ = scored confidence value,

where each x_i^* is the centroid coordinate of a confidence membership function (*Low*, *Medium* or *High*), and C_i is the area of the same membership function, normalized by the sum of the areas of all contributing membership functions. Multiple linear regression was used to find values for the three confidence membership function centroids that best fit the calibration data points.

% Correct Classifications	Active Mode Confidence	Pass. System Confidence	Pass. Shd. Confidence
Uncalibrated	57%	22%	66%
Calibrated	78%	61%	76%

Table 3. Comparison of uncalibrated and calibrated analytically derived expert system performances.

Table 4 lists the centroid values found for each measurement mode confidence value. The *Medium* centroid value for passive confidence were not reasonable (the centroid of *Medium* exceeded the centroid of *High*). This is due to the fact that few data points among the training and test data employed either the passive system or passive shielded *Medium* membership functions. Thus, reasonable calibration data was not available. By the same token, the current results are not greatly affected.

Confidence Value	Low	Medium	High
Active	.28	.53	.76
Passive System	.36	.80	.75
Passive Shielded	.26	.69	.87

Table 4. Calibrated centroid values.

System Performance Comparison

Table 5 compares results of the prototype expert systems. The table indicates that the empirically derived expert system performed as well as or better than the analytically derived expert system for all modes of operation.

% Correct Classifications	Active Mode Confidence	Pass. System Confidence	Pass. Shd. Confidence
Analytical	78%	61%	76%
Empirical	78%	90%	88%

Table 5. Prototype expert system performances.

The design of both expert systems is at a prototype stage, intended to demonstrate the utility of the expert system approach to waste NDA data review. Only neutron measurement data was initially available to develop and assess the performance of the expert systems. Gamma spectroscopy data is also required to arrive at the final assay. Therefore, expert system development was limited by not having all the data used to determine an assay value. Despite this limitation, the results obtained indicate that even prototypes can achieve a high rate of correct classification when applied to actual waste assay data.

The analytically derived expert system required 123 rules, as implemented. The empirically derived expert system partitioned the input space into a total of 18 rules. Hence the empirically derived expert system was simpler in structure than the analytically derived expert system.

With respect to performance of the genetic algorithm a few comments should be made. Initial attempts to find an optimal rule set with a manual search required one day's effort for each confidence value, and produced rule sets that performed at about the 60% correct level. Initial application of the genetic algorithm located rule sets that performed at about the 70% correct level, and required no more than four hours execution time for each confidence value. Improvements in the genetic algorithm search procedure resulted in the performance levels reported in Table 5, such that it can be confidently stated that the application of genetic algorithms resulted in the more efficient location of higher accuracy rule sets.

Acknowledgments

This document was prepared for the U.S. Department of Energy Assistant Secretary for Environmental Management Under DOE Idaho Operations Office Contract DE-AC07-94ID13223.

Conclusions

Two expert systems were developed to operate on neutron assay data acquired using the SAS waste NDA system. Both systems performed well, especially considering that only partial waste NDA data was available for expert system processing (i.e. neutron data only). System performance was benchmarked against domain expert assessments of the compliance quality of the test and training data. The empirically derived expert system was more accurate than the analytically derived expert system for the passive system and passive shielded confidence assessments, and as good as the analytically derived expert system for the active mode confidence assessment.

An important lesson learned during development is that an empirically derived expert system is less labor intensive to develop than an analytically derived expert system. This was due to time consuming work required to define analytically derived expert system fuzzy rules, fuzzy membership functions for the input variables and calibration of the analytically derived expert system output. Therefore, where knowledge is available in the form of data, future development will employ the empirical method of rule formation. As it is expected that analytical rule formation will be required in some instances, methods for integrating empirically and analytically derived rules into a common expert system architecture will be developed.

Initial requirements on the function of the expert system have for the most part been demonstrated. These requirements include representation of domain expert NDA knowledge, and reasoning with supplied data and represented knowledge. A technique to interpret the empirically derived rules is planned as part of the future expert system development activities. At the present time there is ample indication that the expert system technique can be refined to accommodate the balance of NDA data (i.e., gamma measurements), needed to make a comprehensive assessment of waste NDA data quality in accordance with the National TRU Program requirements.

References

U. S. Department of Energy Carlsbad Area Office 1996. Transuranic Waste Characterization Quality Assurance Program Plan, Interim Change Version, Technical Report, CAO-94-1010, U. S. DOE Carlsbad Area Office 1996.

Orchard, R. A. 1995. FuzzyCLIPS Version 6.04 User's Guide, Technical Report, ERB-1045, National Research Council Canada.

Giarratano, J. C. 1994. CLIPS User's Guide, Technical Report, JSC-25013, Lyndon B. Johnson Space Center, National Aeronautics and Space Administration.

Yager, R. R.; Filev, D. P. 1994. Generation of Fuzzy Rules by Mountain Clustering, *Journal of Intelligent and Fuzzy Systems*, 2: 209 - 219.

Chiu, S. L. 1994. Fuzzy Model Identification based on Cluster Estimation, *Journal of Intelligent and Fuzzy Systems*, 2: 267 - 278.

Hassoun, M. H. 1995. *Fundamentals of Artificial Neural Networks*, Cambridge, Massachusetts: The M.I.T. Press.

Heider, H.; Drabe, T. 1997, A Cascaded Genetic Algorithm for Improving Fuzzy-System Design, *International Journal of Approximate Reasoning*, 17:351 - 368.

Takagi, H.; Lee, M. A. 1993, "Neural Networks and Genetic Algorithm Approaches for Auto-Design of Fuzzy Systems," In *Artificial Intelligence Conference -- Fuzzy Logic in Artificial Intelligence*, 68-79. Linz, Austria: Springer Verlag .

Karr, C. L. 1991. Applying genetics to fuzzy logic. *AI Expert*, 6:3, 38-43.