ELSEVIER

Journal of Nuclear Materials 270 (1999) 74–86

journal of nuclear materials

Artificial neural network models for volatile fission product release during severe accident conditions

William S. Andrews [a,*], Brent J. Lewis [a], David S. Cox [b]

[a] *Department of Chemistry and Chemical Engineering, Royal Military College of Canada, Box 17000 Stn Forces, Kingston, Ontario, Canada K7K 7B4*

[b] *Atomic Energy of Canada Limited Research Company, Chalk River Laboratories, Chalk River, Ontario, Canada K0J 1J0*

Received 30 April 1998; accepted 15 September 1998

Abstract

Artificial neural network (ANN) models have been developed to predict the release of volatile fission products from both Canada deuterium uranium (CANDU) and light water reactor (LWR) fuel under severe accident conditions. The CANDU model was based on data for the release of ^{134}Cs measured during three annealing experiments (Hot Cell Experiments 1 and 2, or HCE-1, HCE-2 and metallurgical cell experiment 1, or MCE-1) at Chalk River Laboratories. These experiments were comprised of a total of 30 separate tests. The ANN established a correlation among 14 separate input variables and predicted the cumulative fractional release for a set of 386 data points drawn from 29 tests to a normalized error, E_n, of 0.104 and an average absolute error, E_{abs}, of 0.064. Predictions for a blind validation set (test HCE2-CM6) had an E_n of 0.064 and an E_{abs} of 0.054. From this 14 variable ANN model, a pruned version utilizing only the 6 most significant variables was trained to provide comparable predictions. An ANN model was also developed for LWR fuel, based on data from the vertical induction (VI) series of tests (VI-2 to VI-5) conducted at Oak Ridge National Laboratory. Predictions for data not used in ANN training had an E_n of 0.045 and an E_{abs} of 0.059. A methodology is presented for deploying the ANN models by providing the algorithms for trained ANNs and the corresponding connection weights. Finally, the performance of the full ANN CANDU model was compared to a fuel oxidation model developed by Lewis et al. and to the US Nuclear Regulatory Commission's CORSOR-M.

1. Introduction

The need to define nuclear power reactor source terms for fission products released during severe accident conditions has been underscored by the accidents at Three Mile Island and Chernobyl. In the United States, tests have been conducted involving the heating, or annealing, of fuel fragments and short segments of light water reactor (LWR) fuel rods under varying environmental conditions, such as steam and hydrogen [1]. Analysis of early annealing experiments performed at the Oak Ridge National Laboratory (ORNL) has provided a correlation of the cumulative release of volatile fission products with temperature and time in a steam environment. This correlation, called CORSOR-M, is used by the United States Nuclear Regulatory Commission for LWR source term prediction [2]. Its applicability to CANDU pressurized heavy water reactor (PHWR) fuel has yet to be established. Further, it only considers two variables (temperature and time) in one environment (steam).

Corresponding annealing experiments have been conducted at the Chalk River Nuclear Laboratories (CRL) on bare fuel fragments and mini-elements (i.e., short-length fuel specimens with Zircaloy cladding) derived from spent CANDU fuel. In fact, the Hot Cell Experiments 1 and 2 (HCE-1 and HCE-2) [3,4] and the Metallurgical Cell Experiment (MCE-1) [5] include a total of 30 separate tests conducted using a wide variety of sample sizes and geometries, thermal conditions and environments, as shown in Table 1. The resulting large

* Corresponding author. Tel.: 1-613 541 6000; fax: 1-613 542 9489; e-mail: andrews_w@rmc.ca

0022-3115/99/$ – see front matter
PII: S0022-3115(98)00765-X

Table 1
Summary of CRL tests used to develop and test ANN models

Test	Pre-fuel test conditions		Sample conditions		Test conditions						Max. cumulative fractional release
	Max linear power (kW/m)	Burnup (MWh/kg)	Zircaloy weight (g)	Fuel weight (g)	Atmosphere	Isothermal temperature (°C)	Time sample >1000°C (s)	Time at isothermal temperature (s)	Time in steam (s)	Time in air (s)	
HCE1-10	54	436	3.89	18.8	Air	1600	12288	11762	0	10630	0.962
HCE1-11	54	436	3.89	18.8	Steam	1600	12285	8107	7026	0	0.881
HCE1-12	54	436	5.92	18.8	Argon/2% H_2	1600	10842	10442	0	0	0.396
HCE1-13	54	436	5.92	18.8	Steam	1600	11092	10892	10892	0	0.913
HCE1-14	54	436	5.92	18.8	Steam	1600	8852	8652	8252	0	0.889
HCE1-15	54	436	5.92	18.8	Air	1611	10895	10895	0	10695	0.962
HCE1-16	54	436	5.92	18.8	Air	1603	11490	11290	0	11490	0.931
HCE1-17	54	436	3.89	18.8	Air	1497	11044	10081	0	9100	0.916
HCE1-18	54	436	3.89	18.8	Steam	1500	8848	8448	6800	0	0.938
HCE2-BM2	46	462	6.701	24.561	Steam	1350	14158	9923	8842	0	0.439
HCE2-BM3	58	540	6.48	22.462	Steam	1500	14751	8573	7000	0	0.755
HCE2-BM4	57	570	6.701	24.32	Steam	1522	13540	9256	9256	0	0.672
HCE2-BM5	46	462	6.59	23.615	Steam	1500	11290	6460	5810	0	0.565
HCE2-CF1	54	457.2	0	0.46	Steam	1635	8461	8167	7467	0	0.799
HCE2-CF2	54	457.2	0	0.566	Steam	1363	8007	7557	7257	0	0.942
HCE2-CF3	54	457.2	0	0.534	Steam	1504	6845	6428	5728	0	0.839
HCE2-CM1	54	457.2	5.265	11.204	Steam	1642	6484	5738	4362	0	0.853
HCE2-CM2	54	457.2	5.154	9.542	Steam	1356	11004	10504	9029	0	0.752
HCE2-CM4	54	457.2	5.486	12.668	Air	1350	13000	8160	0	8160	0.767
HCE2-CM5	54	457.2	8.9	45.761	Steam	1500	16544	10477	7799	0	0.873
HCE2-CM6	54	457.2	5.375	11.515	Steam	1495	11909	7829	7829	0	0.846
HCE2-CM7	54	457.2	6.59	23.99	Steam	1622	22507	22195	10647	0	0.951
MCE1-1	43	257	0	0.256	Air	1700	6941	1523	0	900	0.759
MCE1-2	43	257	0	0.261	Air	1800	10972	1751	0	1300	0.9925
MCE1-3	43	257	0	0.265	Air	1900	9890	2064	0	1200	0.9595
MCE1-4	43	257	0	0.298	Air	2000	11460	1710	0	1100	0.9875
MCE1-5	43	257	0	0.277	Argon/2%H_2	1815	12460	2000	0	0	0.97
MCE1-6	43	257	0	0.33	Argon/2%H_2	2000	13189	1872	0	0	0.8215
MCE1-7	43	257	0	0.22	Air	2090	13253	2115	0	2000	0.9635
MCE1-8	43	257	0	0.191	Air	2080	12378	1728	0	2800	0.976

data base has yet to be analyzed to the point that an overall comprehensive model could predict experimental results with any degree of confidence.

The physical mechanisms involved in the release of volatile fission products under severe accident conditions are felt to be extremely complex. In the US, the FASTGRASS code has been developed to model such phenomena as fission gas bubble nucleation, migration, interlinking and resolution [6], while in Canada, SOURCE-2 is a mechanistic model for CANDU fuel [7]. Both of these codes, however, are computationally intensive and thus do not run in real time. As a result, work continues on simpler, semi-empirical models which are based on the controlling physical phenomena. Examples of this are work by Osborne and Lorenz at ORNL [8] and more recently by Lewis et al. in Canada [9]. This notwithstanding, the comprehensive model of fission product release considering the full spectrum of conditions of the annealing tests and able to run in real time has not appeared yet. This paper will outline a novel approach to empirically modelling the results of 30 CRL tests in one instance and 4 ORNL tests in another with the use of artificial neural networks (ANNs) in order to predict the cumulative fission product release of volatile fission products (specifically ^{134}Cs).

2. Artificial neural networks

Neural network development began in the mid 1940s, but went into a hiatus due to a lack of discernible applications and sufficiently powerful computers. The most widely-used paradigm, back propagation, emerged in the mid 1980s from the work of two psychologists, McClelland and Rumelhart, in their efforts to model the functioning of the brain [10]. Neural networks are composed of simple nodes (neurons) which take inputs, sum them, perform a simple mathematical operation on this sum via a transfer function and pass the result on to other nodes. Before the output arrives at a subsequent node, however, a separate numerical value or weight is applied to it. As a consequence, nodes receive outputs of preceding nodes which have been modified by these weights.

Current practice in back propagation networks is to use three layers of neurons, with interconnections as shown in Fig. 1. (Recent usage also exhibits some degree of connection directly between the input and output nodes, in addition to the connections indicated in Fig. 1.) These layers are usually named input (X), hidden (Y) and output (Z). The input layer contains the values of the variables and parameters considered for correlation. In this application, each node would correspond to a value from the list in Table 2 or Table 3. These inputs are mapped into the range -1.0 to $+1.0$ using the mapping function

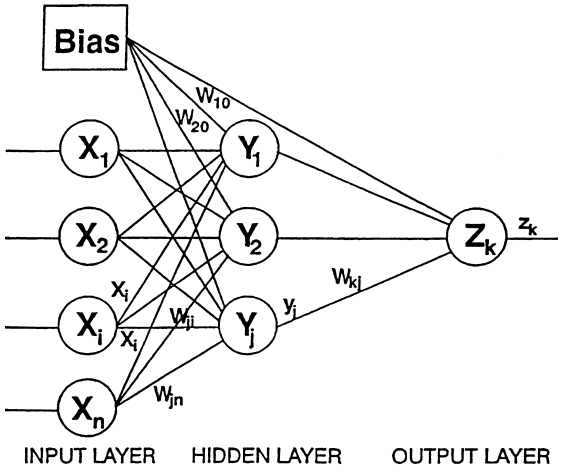

Fig. 1. Architecture of a back propagation artificial neural network. Input variables are listed in Table 2 and connection weights provided in Table 9 for CRL model and in Tables 3 and 10, respectively, for ORNL model.

$$x_i = \frac{2v_i - (M_i + m_i)}{(M_i - m_i)}, \tag{1}$$

where x_i is the scaled or mapped ith input value, corresponding to v_i, the unscaled or raw input value. M_i and m_i are the maximum and minimum values of v_i, respectively. Each input node is connected to each node in the hidden layer. It is also usual to connect a bias node (with a set value of 1.0) to all hidden and output nodes. The bias node serves to offset the origin of the transfer function and tends to cause the network to converge more quickly.

Each connection to a hidden layer node contains the normalized input value leaving the input layer or bias node, modified by a connection weight. Thus, the ith scaled input value, x_i, connected to the jth hidden node has a weight, w_{ji}, applied to it. Consequently, the hidden node receives as an input from the ith node, the value $w_{ji}x_i$. Each node in the hidden and output layers, then, receives and sums the inputs to it. In Fig. 2, I_j is the sum of i inputs, each multiplied by its own connection weight, so that

$$I_j = \sum_{i=0}^{n} w_{ji}x_i. \tag{2}$$

When optimized, this is the equation of a linear regression, with the intercept being the weight associated with the bias node, w_{j0}.

Non-linearity is introduced into the model by the transfer function, which is applied to the summed inputs, I_j. Several different functions are available (sinusoidal, sigmoidal), but the one most widely used is the hyperbolic tangent, which is a smoothed version of the step function from -1 to $+1$. The application of the tanh

Table 2
Variables and parameters used in input space for developing ANN for CANDU fuel, including maximum and minimum values. Nodes are depicted in Fig. 1

Node number (i)	Variable (v_i)	Minimum value (m_i)	Maximum value (M_i)
0	Bias (value fixed at 1)		
1	Time above 1000°C (s)	−3320	22507
2	Fuel temperature (°C)	435	2090
3	Time at temperature (s)	0	22195
4	Time in steam (s)	0	10892
5	Time in air (s)	0	11490
6	Rate of temperature change (°C/s)	−0.2	0.5
7	Weight of Zircaloy (g)	0	8.9
8	Cladding closed (y or n)	0	1
9	Rate of steam flow (mL/min)	0	200
10	Rate of air flow (mL/min)	0	400
11	Rate of Ar/H_2 flow (mL/min)	0	800
12	Peak linear power (kW/m)	43	58
13	Burnup (MWh/kgU)	257	570
14	Weight of fuel (g)	0.191	45.761

Table 3
Variables and parameters used in input space for developing ANN for LWR fuel, including maximum and minimum values

Node number (i)	Variable (v_i)	Minimum value (m_i)	Maximum value (M_i)
0	Bias (value fixed at 1)		
1	Time above 1000°C (s)	−1871	10490
2	Fuel temperature (K)	450	2754
3	Rate of temperature change (°C/s)	−1.79	1.64
4	Rate of steam flow (L/min)	0	1.6
5	Rate of Ar/H_2 flow (L/min)	0	0.4

transfer function yields an output, y_j, from the jth hidden node, such that

$$y_i = \frac{e^{I_j} - e^{-I_j}}{e^{I_j} + e^{-I_j}} = \tanh I_j. \tag{3}$$

This same process is repeated between the hidden layer and the output layer, with the transfer function

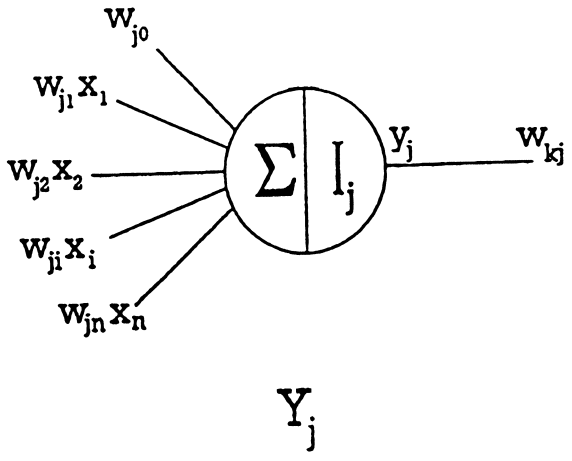

Fig. 2. Architecture of the jth neuron in the hidden layer.

again applied to the summed inputs to produce the output. Thus

$$I_k = \sum_j w_{kj} y_j, \tag{4}$$

where I_k is the sum of the weighted inputs to the kth output node and w_{kj} is the connecting weight between the kth output node and the jth hidden node (or the bias node).The scaled output from the kth output node is given by

$$z = \tanh I_k. \tag{5}$$

This output value z must then be mapped back to provide a real value for the cumulative fractional release f. This process is similar to, but the reverse of, the input mapping. Thus

$$f = \frac{(M-m)z + (Rm - rM)}{R - r}, \tag{6}$$

where M and m are the measured maximum and minimum values of the output variable f, and R, r are the maximum and minimum values of the network output z (here 0.8 and −0.8 respectively).

When the network is initialized, the values of the weights are randomly assigned. The 'knowledge' or

'artificial intelligence' within the network, however, resides in the distribution of these weights, which must be adjusted to be able to produce an output as close as possible to the desired output. This process of adjusting the weights is called supervised learning and is conducted during the training phase of the network development.

To effect training, the network is not only presented with a full array of inputs, but also with known (measured) outputs for each input set. When the initial pass through the network for a given set of data is complete, an output, z, is determined. This value is then compared to the desired output value, d (the scaled measured value of the cumulative fractional release), to arrive at a global error E

$$E = 0.5(d - z)^2. \tag{7}$$

The global error is then propagated backwards through the network and used, in conjunction with the appropriate learning rule (such as gradient descent) to adjust the individual connecting weights. This back and forth iterative process is continued until the global error is minimized. At this point, another set of input data (vector) is introduced into the network and the process is repeated. Generally, the connecting weights are adjusted after an 'epoch' of up to about 500 different inputs (this can be adjusted to facilitate learning, but updates are rarely done after each input set, in order to prevent oscillations in weight values). A description and derivation of the application of gradient descent to back propagation for error minimization is contained in abridged form in Ref. [11] and more fully in Appendix A to this paper.

Once training is complete, the network is tested against data which had not been seen during training. Like the training set, the test set should represent, to the greatest extent possible, the whole range of the input space. Also, like the training set, the test set must have known output values available for comparison with the network predicted values. Finally, the network should be validated by predicting results for a data set representative of a likely application.

Essentially, then, a trained neural network is an n-dimensional correlation, and provides a result similar to a non-linear regression. The link with the physical phenomena it is modelling is through the choice of variables or parameters. Any relationship among these inputs is established by the learning rule itself, and not by any real or postulated physical relationships.

Neural networks have a number of advantages over other types of models or correlations:

1. With the 'knowledge' or 'intelligence' distributed throughout the network, a reasonable response is possible when the input space contains incomplete, noisy or previously unseen values, as well as variables which are not linearly independent.
2. A careful analysis of the weights throughout the network permits the various parameters or variables in the input space to be ranked in order of influence on the output.
3. A trained neural network model operates in real time, making it suitable for being embedded in much more complex computer codes, such as modelling the progression of a severe reactor accident.

3. Experimental

The data base used to construct the CANDU model comprised 9 tests of HCE-1, 13 of HCE-2 and 8 of MCE-1, with the tests conducted between 1350 and 2100°C and in steam, air or argon/2% hydrogen atmospheres. Each annealing test involved placing a fragment or mini-element in an induction furnace, and introducing the appropriate environment (steam, air or argon/2% hydrogen) into the furnace. The release of fission products from the sample was determined by measuring the change in fission product activity by using gamma-ray spectrometry. As well as environment, other factors varied included temperature, time at temperature, heating ramp rate, sample size, amount of Zircaloy cladding and sample burnup. The fission products measured included ^{134}Cs and ^{137}Cs, ^{103}Ru and ^{131}I, although the model reported on in this paper was developed for ^{134}Cs, as the cumulative release values showed minimal randomness and the cesium behaviour was felt to be representative of volatile fission products in general.

Factors such as temperature and the partial pressure of oxygen in the release environment influence the oxidation rate of uranium and Zircaloy (if present) and, by extension, the release rates of fission products escaping as vapors. It is anticipated that the corresponding rates of release of intragranular volatile fission products from the fuel matrix should be roughly equal. When investigating irradiated fuel that has been out of a reactor for upwards to two years, the longer-lived cesium isotopes remain the best representatives of volatile fission products, which can be considered as being composed of isotopes of Xe, Kr, I, Te, Rb and Cs [12–14]. For instance, cumulative release fractions of ^{137}Cs and ^{85}Kr were similar in the HI and VI series of tests at the ORNL [8]. Moreover, in high temperature annealing tests at the Commissariat à l'Energie Atomique, in which spent fuel samples were reirradiated to restore the short-lived inventory, the measured release kinetics for I, Cs, Xe and Te were similar in a steam test after the Zircaloy cladding had been completely oxidized; in a hydrogen atmosphere, the measured kinetics of I, Cs and Ba were also comparable [15]. Consequently, the release kinetics exhibited by ^{134}Cs in a variety of environments and test conditions using discharged high burnup fuel are felt to

be representative of the release kinetics of the noble gases and other volatile products, since these species all have similar solid state diffusivities and exhibit a high partial pressure under most reducing and oxidizing conditions [16].

The VI tests at ORNL used to develop the LWR model involved the annealing of 15 cm sections of high burnup LWR fuel to temperatures of up to 2467°C in either a hydrogen or steam environment. The fuel sections were cut from full length rods and had press-fit Zircaloy end caps applied. Release data for the relatively volatile cesium isotopes (^{134}Cs and ^{137}Cs) were collected using gamma-ray spectroscopy. A more detailed description of the experimental procedures can be found in Ref. [17], as well as Refs. [4] for CRL and [1] and [8] for ORNL experiments.

4. Analysis

The neural net used to model the CRL tests was developed using a commercial software shell, *NeuralWorks Professional II/Plus* by NeuralWare [11] and featured 14 different inputs, as listed in Table 2. Related to Fig. 1, the input space would extend from x_1 to x_{14}. A single hidden layer was used with differing numbers of nodes (from 2 to 15). All the architectures returned comparable results, except the networks with only 2 hidden nodes, which provided poor correlations. The output layer contained a single node and represented the cumulative fractional release of ^{134}Cs.

Much of the effort needed to train a neural network must be invested in creating the data base to provide the input vectors. Each test contained values for temperature and cumulative fractional release measured at intervals of 100 to 300 s. Most tests exhibited a characteristic response of: (i) an initial plateau on the time/fractional release curve (Fig. 3) displaying an initial low release rate, normally due to a release of the grain boundary inventory, with some diffusion within the grains (during the initial temperature ramp); (ii) a fairly steep climb due to an increased release rate as a consequence of diffusive release particularly during fuel oxidation; and (iii) a final plateau, indicating some possible trapping in the fuel porosity with a complete release of the available fission product inventory [9]. As a consequence, cumulative fractional release values (the target output) are over-represented in the plateau areas in the original test data with respect to the individual fractional release values associated with the high release rate portion, e.g., some 16 points correspond to a cumulative fractional release of 0.2 and 13 between 0.7 and 0.8, while only 4 lie between 0.4 and 0.5. A model trained on the raw data could provide good overall predictions by concentrating on the initial and final values, leaving predictions for releases during the high release rate

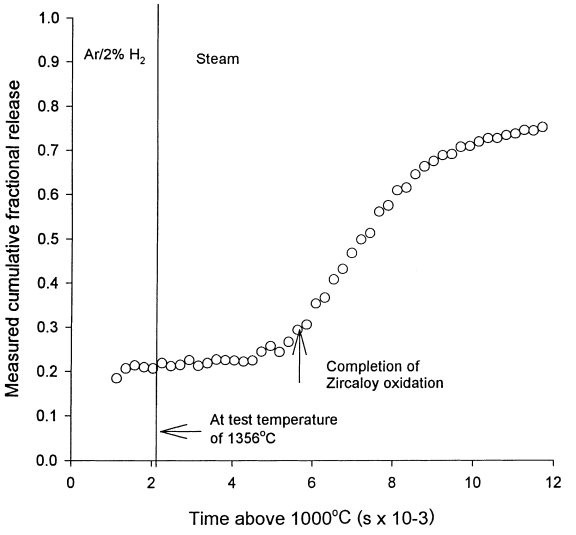

Fig. 3. Typical data distribution of the cumulative fractional release of ^{134}Cs for a CRL annealing test with Zircaloy cladding (HCE2-CM2). Test conditions are described in Table 1. The initial temperature ramp was performed in an Ar/2%H_2 environment.

phase of questionable validity. Further, more tests were conducted at 1600°C than at any other temperature, although the isothermal test temperatures ranged from 1350 to 2100°C. In order for the model to be able to interpolate with any degree of confidence, the input space had to be as balanced as possible. Without this, inadvertent biases would be introduced and trained into the network. In other words, the network would tend to provide better predictions for conditions approximating the preponderance of training data and provide poorer predictions for other areas in the input space. To redress this imbalance, the data available for each test were expanded significantly by interpolation, so that the available number of training vectors was increased from 1371 to 4049. Any inaccuracies introduced by this approach were felt to be well within the actual noise of the data itself. For example, in Fig. 3, the number of cumulative fractional release values between 0.3 and 0.7 was expanded by linear interpolation to provide a more balanced distribution of target (cumulative fractional release) values over the range of the dependent variable. Such an expansion would leave values well within the uncertainty associated with the spectroscopy measurements, as can be seen by inspection of Fig. 3.

The expanded data base was separated into two portions, with 90% of the data provided for the training set and the remaining 10% for the test set. In order to achieve balance, data from some tests were repeated, so that the total input space contained 12,516 vectors. Further, the results from a complete test, HCE2-CM6, were withheld from both the above sets to provide a

validation by exposing the trained network to conditions on which it had not been trained.

ANN training used the extended delta-bar-delta, or edbd, learning rule, which is an enhancement of the basic gradient descent rule [11,17]. Several model development parameters were available for optimization: number of hidden nodes and hidden layers, epoch size (number of training vectors introduced between successive connection weight adjustments), number of training cycles and the initial weight distribution prior to training. Overall, model development was found to be relatively robust, with only slight differences in effectiveness being realized by varying these parameters. For the full CANDU model, 4–5 hidden nodes in a single layer, an epoch size of 125 vectors and 7 complete training cycles (total of 87612 training vectors) provided optimal predictions.

Model effectiveness was gauged in three ways:

1. Network predicted cumulative fractional releases (outputs) were plotted against the corresponding values actually determined by CRL. A perfect correlation would have all points fall along the diagonal with a slope of 1 and an intercept of 0. The corresponding values from the linear regression through the data were then computed, including the slope and the correlation coefficient, r, which is the covariance divided by the product of the sample standard deviations [18].
2. The normalized error, E_n, is the ratio of the average sum-squared error to the average of the squared deviations. This value is felt to be particularly useful for back propagation, as networks learn the average or smoothed target values. The normalized error, then, can be considered as reflecting the proportion of the output variance that is due to error, rather than the network architecture itself.
3. The average absolute error, E_{abs}, is the average difference (in absolute terms) between the measured and predicted values for a test or validation set.

As noted already, networks of differing numbers of hidden nodes had comparable r values for the test set. An example of the scatter plot for a network having 5 hidden nodes in one hidden layer can be seen in Fig. 4. The solid dots show a perfect correlation, which can be compared with the actual linear regression through the points. Most of the 386 test vectors provide points very close or on the regression line. Overall, the slope of 0.899 and intercept of 0.080 are fairly close to the optimal values of 1 and 0, respectively. The r value of 0.946 shows the extremely strong correlation between the actual measured cumulative fractional release values from 29 separate tests and the corresponding values predicted by the ANN model.

The validation of the network involved using the vectors of a complete test, HCE2-CM6, which was not included in the test set. In fact, HCE2-CM6 was felt to be representative of all the tests conducted and of severe accident conditions, as it involved heating a Zircaloy-clad mini-element in argon/2%hydrogen and then steam environments to 1500°C. The results of the validation are depicted in Figs. 5 and 6. For this particular test, the ANN model provides predictions having a good linear correlation (r is 0.995), although it slightly underpredicts in the steam portion of the test and overpredicts in the

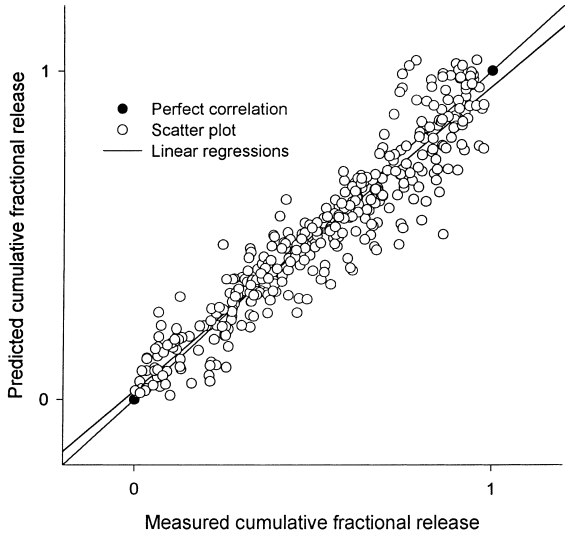

Fig. 4. Scatter plot of 386 data point test set of 14-5-1 ANN model.

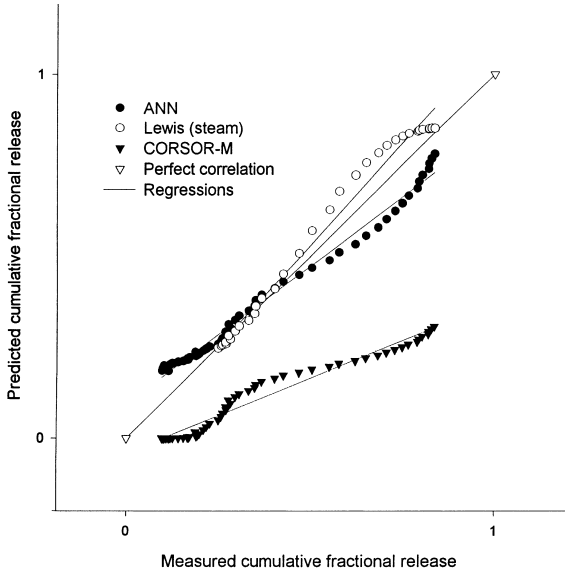

Fig. 5. Scatter plots of models predicting cumulative fractional release for validation set (HCE2-CM6). Statistical comparison in Table 4. The lowest release value of the Lewis model corresponds to the change from an argon to a steam environment.

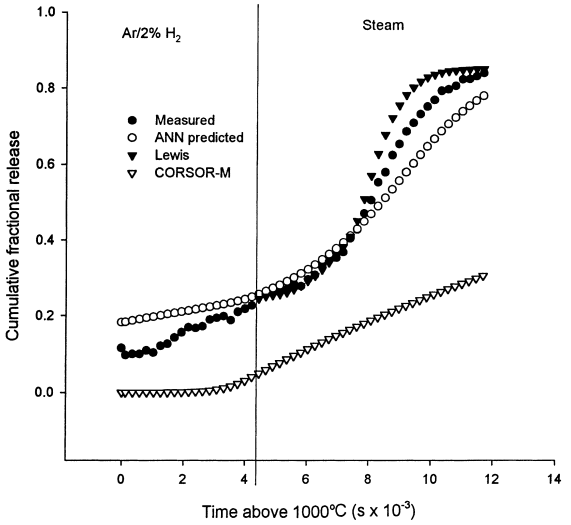

Fig. 6. Comparison of model performance predicting cumulative fractional release for validation set (HCE2-CM6). Statistical comparison is in Table 4. The initial point of the Lewis model corresponds to the change from argon to steam.

argon/hydrogen portion. As shown in Table 4, the E_n value is 0.064, while E_{abs} is 0.054 over the whole test. Returning to Fig. 4, moreover, it can be seen that across the whole test set, some tests are underpredicted or overpredicted, with the vast majority of vectors being predicted very closely.

Fig. 6 shows the measured and predicted cumulative fractional releases plotted against time that the sample is above 1000°C. Two points are of note here. First, the neural network model provides a smoothing of the data, (the discontinuity at about 4000 s reflects the time at which the environment changed from inert to steam). The second point is that the model is able to reproduce the non-linearity of the relationship between fractional release and time, due to varying release rates. In this particular test, though, the model values diverge from the measured at less than 4000 s and beyond about 7500 s.

A sensitivity analysis and an examination of the distribution of connection weights was conducted (the most significant inputs should have the highest connecting weights). The results can be seen in Table 5. The fuel temperature was found to be the predominant influence in predicting the cumulative fractional release of ^{134}Cs, while time was second to temperature but more important than all other factors. The weight of Zircaloy followed time and presumably reflected the presence or absence of a physical barrier (cladding) to the escape of volatile fission products and a chemical effect due to hydrogen production from the steam/Zircaloy reaction. This, in turn, results in a lower oxygen potential that will retard the fuel oxidation (and hence diffusive release) [9]. Once the Zircaloy cladding is oxidized, early in the steam tests, the volatile release kinetics will be similar to those of the bare fuel fragments, (e.g., tests where the Zircaloy weight equals zero in Table 1) since the Zircaloy cladding will no longer chemically influence the release behaviour as a source of hydrogen. Thus, the presence of Zircaloy acts to delay the release kinetics. The hierarchy of the remaining influences is felt to be somewhat ambiguous, with these variables being second order influences, at best. The relatively minor role of closed cladding may also suggest that the cladding on the fuel samples was never really closed, as the end caps were only held loosely by wires. This, plus evidence of double sided oxidation [19] indicates that the cladding may never have provided a significant barrier to the release of cesium. Another inference might be that the fuel porosity, even at relatively high burnups, may not have been sufficient to permit a complete release through the interconnected pore network. Rather, the relative importance of time suggests that much of the inventory was intragranular, with diffusion to the grain boundary

Table 5
Ranking of relative importance of input variables based on sensitivity and weight space analyses

Variable
Temperature
Time sample above 1000°C
Time in air
Time at temperature
Time in steam
Zircaloy weight
Steam flow rate
Cladding open
Fuel weight
Linear power
Burnup
Temperature change rate
Air flow rate
Ar/H_2 gas flow rate

Table 4
Statistical attributes of models applied to validation set (HCE2-CM6). Models are depicted graphically in Figs. 5 and 6

Model	Correlation coefficient (r)	Normalized error (E_n)	Average absolute error E_{abs}
ANN	0.995	0.064	0.054
Lewis (steam)	0.992	0.056	0.038
CORSOR-M	0.979	1.523	0.270

surface necessary before subsequent release through the pore network. In addition, the relatively modest influence of the fuel weight (varying over two orders of magnitude), and thus the surface-to-weight ratio, reinforces the significance of the intragranular inventory.

In addition, there was relatively little range of values in the linear power and burnup of the fuel samples. Not surprisingly, then, these variables had relatively little influence on the cumulative fractional release of cesium from the range of test samples.

Finally, the ANN CANDU model was compared to CORSOR-M and to the Lewis fission product release model [9]. The results can be seen graphically in Figs. 5 and 6 applied to the same validation set used in the ANN development, HCE2-CM6. The models are compared statistically in Table 4. It can be seen that the CORSOR-M model greatly underestimated the release fraction. This may be attributable to the different experimental conditions upon which CORSOR-M was developed, i.e., higher temperatures and larger fuel samples. The performance of the Lewis model is quite good, although it must be borne in mind that the model is fixed at the experimentally measured release fraction at the introduction of steam. In contrast, the ANN model is a pure or blind prediction of the HCE2-CM6 measured releases in both Ar/H_2 and steam environments. It should be noted, however, that the model depicted in Figs. 5 and 6 (having a 14-4-1 architecture) is not the same as the one in Fig. 4 (having a 14-5-1 architecture). In the latter case (Fig. 4), the model provided the best overall predictions across the whole set of 29 tests. The model shown in Figs. 5 and 6 provided a better prediction for the specific validation set chosen, HCE2-CM6, but was slightly less effective at predicting over the test set representing the other 29 tests. Of the two models, it is felt the one with the better general behaviour (Fig. 4) is the more useful.

Based on the sensitivity analysis results, inputs were pruned, beginning with the least significant, and successive models were trained. It was found that only after the removal of the input variable used for the weight of Zircaloy in the sample did significant degradation in convergences appear, with consequential poor predictions of cumulative fractional release. Thus, an ANN model containing the 6 inputs shown in Table 6 was able to approximate the predictions of the full 14-input network. An advantage inherent to the pruned model is that it renders deployment easier, as a reduced number of values for input variables is required. The performance of this network is contained in Table 7.

Table 6
Input variables used in a pruned (6-4-1) ANN model for CANDU fuel based on data from 29 CRL tests

Variables
Time sample above 1000°C (s)
Fuel sample temperature (°C)
Time sample at a given temperature (s)
Time sample in steam (s)
Time sample in air (s)
Weight of Zircaloy (g)

The overall relative closeness of the ANN model predictions to the values measured by CRL indicates that a trained neural network model has been able to establish a good correlation between a number of disparate parameters representative of possible severe reactor accident conditions and the cumulative fractional release of fission product cesium.

Data for the LWR fuel model were taken from the four VI tests (VI-2–VI-5) and divided into training and test sets in a 3:1 ratio. No balancing of the input space was conducted, as data were fairly evenly distributed between steam (VI-2, VI-3) and hydrogen (VI-4, VI-5). The isothermal temperatures were well above most of those used for the CRL CANDU tests and the fuel samples were much larger. A 5-3-1 network was found to give good predictions, with the metrics contained in Table 8. A scatter plot of the test set can be seen at Fig. 7. As with the CANDU ANN model, the LWR ANN model provided better predictions of the cumulative fractional release of cesium under simulated reactor accident conditions than did CORSOR-M or semi-empirical diffusion-based models [20].

5. Ann model deployment

The ANN models were developed from an input space of 14 variables, as shown in Table 2 for CANDU fuel and 5 variables for LWR fuel, as shown in Table 3. In order to use a trained model to predict the cumulative fractional release of cesium under conditions within the range of input values, the appropriate variable values must be scaled according to Eq. (1). The appropriate maximum and minimum values are contained in Tables 2 and 3. The connection weights for the 14-5-1 (bias

Table 7
Statistical attributes of pruned (6-4-1) ANN model for CANDU fuel applied to validation set (HCE2-CM6)

	Correlation coefficient (R)	Normalized error (E_n)	Average absolute error E_{abs}
Test set	0.883	0.226	0.096
Validation set	0.971	0.214	0.095

Table 8
Statistical attributes of ANN model for LWR fuel applied to data from ORNL tests VI-2 to VI-5

	Correlation coefficient (R)	Normalized error (E_n)	Average absolute error (E_{abs})
Test set	0.985	0.045	0.059

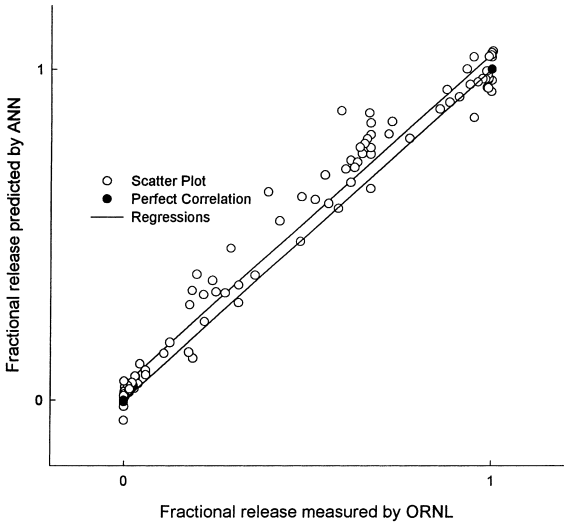

Fig. 7. Scatter plot of test set for ORNL VI data.

node 0, input nodes 1-14, hidden nodes 15-19 and output node 20) CANDU network shown in Fig. 4 are contained in Table 9. The corresponding maximum and minimum values for LWR fuel for a 5-3-1 network are in Table 2, with the hidden nodes being 7–9 and the output 10. The connection weights are in Table 10. The output values from each input node must then be multiplied by the appropriate connection weight and summed at each hidden node (Eq. (2)). A hyperbolic tangent transfer function is then applied at each hidden node to determine the appropriate output (Eq. (3)). This process is repeated at the output node (Eqs. (4) and (5)). The output from the output node must then be scaled back to a real world value, according to Eq. (6), with the respective values of M and m being 0.9925 and −0.117 for the CANDU network and 1.0023 and 0 for LWR. This series of operations can easily be performed by a spreadsheet calculation or written into a short code in a programming language, e.g., Basic. The series of simple calculations allows the computations to be performed in real time.

As with any correlation, as distinct from physically-based models, ANNs can only be used with confidence within the range of the input space on which they were developed. For the models discussed above, the individual variable ranges are given in Tables 2 and 3. Consideration of cases outside these ranges represents

Table 9
Connection weights w_{ji} and w_{kj} for a 14-5-1 trained ANN

Source node	Destination node					
	15	16	17	18	19	20
0	−0.4305	0.2578	0.0018	0.0606	−0.1546	0.4668
1	−1.5001	0.8345	−0.6655	−1.2945	−0.1987	
2	0.0521	−2.0045	−1.0748	−2.2791	−0.9801	
3	−0.2716	−0.5367	−0.0451	0.2944	0.0935 8	
4	−0.5602	−0.6749	−0.2416	−1.9599	−0.0978	
5	−1.8507	0.1387	−0.4572	−0.0618	−0.0943	
6	1.1470	−0.4846	1.1114	2.5243	0.5328	
7	1.2937	−0.4971	0.1895	0.7858	0.2406	
8	−0.6041	0.4703	−0.0633	−0.4438	0.3048	
9	−0.3888	0.4718	−0.1081	−0.7703	−0.0689	
10	−0.8062	−0.0842	0.1111	0.4438	−0.1239	
11	0.6468	−0.1865	−0.0033	0.2545	−0.0193	
12	−0.9383	−0.8495	−0.6413	−0.5259	−0.1234	
13	0.8866	1.0589	0.5407	1.1923	0.3248	
14	0.4028	−0.2117	−0.1219	−0.2636	−0.0983	
15						−0.5391
16						−0.4932
17						−0.0579
18						−0.3824
19						0.2102

Table 10
Connection weights w_{ji} and w_{kj} for a 5-3-1 trained ANN for LWR fuel

Source node	Destination node			
	7	8	9	10
0	−1.4640	0.9716	1.8189	−1.0947
1	−0.6076	1.2991	0.7445	
2	−2.7166	0.9193	0.9614	
3	0.9719	0.5590	−1.7007	
4	−1.1473	−2.0448	−2.7551	
5	1.7188	−0.1419	−0.5720	
6				−1.2394
7				0.5573
8				0.7505

extrapolation, which is of questionable validity and thus strongly discouraged.

The models do, however, reflect a general correlation among 14 input variables developed from data collected during 30 separate and distinct tests. This represents the first comprehensive model involving data from all these tests.

6. Conclusions

Back propagation neural network models with modified delta learning rules have been trained to predict the cumulative fractional release of the volatile fission product cesium from CANDU fuel fragments and mini-elements, as well as LWR elements, under a variety of simulated severe accident conditions within the range of the experimental data on which they were trained. The models were able to reproduce the non-linearities inherent in the relationships between fractional release and time, and provided a smoothing of the data. Finally, the CANDU model was able to predict the general trend of the release kinetics for a validation set which was not used for training, and to predict the cumulative fractional release to within an average absolute error of 0.054.

Acknowledgements

The authors gratefully acknowledge the use of the experimental data collected by the Chalk River Laboratories and Oak Ridge National Laboratory. The experimental work in Canada was funded by Atomic Energy of Canada Limited, Ontario Hydro, Hydro Quebec, and New Brunswick Electric Power Corporation under the CANDU Owners Group agreement. The work performed at the Royal Military College was funded by the Academic Research Program of the Department of National Defence of Canada under allocation 3705-882.

Appendix A. Error minimization in back propagation networks

To effect training, the network is not only presented with a full array of inputs, but also with known (measured) outputs, for each input set or vector. When the initial pass through the network for a given set of data is complete, a set of outputs, z_k, is determined. These values are then compared to d_k, the value corresponding to the desired outputs, but mapped into the same range as z_k, to arrive at a global error E, where

$$E = \frac{1}{2}\sum_k (d_k - z_k)^2. \tag{A.1}$$

This form for the global error, rather than what might appear to be a more logical $\sum(d_k - z_k)$, has two advantages, it removes the sign of the difference (thus preventing two differences from cancelling each other) and the first derivative becomes the basic difference as shown below.

$$\frac{\partial E}{\partial z_k} = -(d_k - z_k). \tag{A.2}$$

The global error is then propagated backwards through the network to adjust the individual connecting weights and the process of determining z_k is repeated using the same input vector $\mathbf{x}$, (i.e., $x_1, x_2, x_3, \ldots x_i, \ldots x_n$). This back and forth iterative process is continued until the global error is minimized. At this point, another set of input data (vector) is introduced into the network and the process is continued. Generally, the connecting weights are adjusted after an 'epoch', which can be a multiple of the complete training set, where the training set is the set of training vectors, although the epoch does not normally exceed 500 vectors. This adjustment involves the average adjustment to each weight during the epoch, so the weight adjustments must be stored during each epoch.

Basic back propagation networks use the 'delta' rule during training, which implements a gradient descent in E into the weight space (w_{ji} and w_{kj} for all i, j and k). The

delta rule can be used if the network has 'semi-linear' activation or transfer functions which are differentiable and non-decreasing. Linear threshold functions are not satisfactory, as their derivatives are infinite at the threshold and zero elsewhere. For the delta rule, the following relationship is used:

$$\Delta w_{kj} \propto -\frac{\partial E}{\partial w_{kj}}, \tag{A.3}$$

where Δw_{kj} is the change made to the connecting weight between the kth output node and the jth hidden node. It is useful to apply the chain rule to the partial derivative above, so that

$$\frac{\partial E}{\partial w_{kj}} = \frac{\partial E}{\partial I_k} \cdot \frac{\partial I_k}{\partial w_{kj}}. \tag{A.4}$$

This product on the right-hand-side reflects, in the first term, the change in the global error as a function of the change in the net input to a particular node or neuron. The second term reflects the effect on the net input of changing a particular connecting weight. The second term, then, is

$$\frac{\partial I_k}{\partial w_{kj}} = \frac{\partial}{\partial w_{kj}} \sum_j w_{kj} y_j = y_j. \tag{A.5}$$

The term δ, or delta, which provides the name of the rule, and which is sometimes called the local error e, is defined as

$$\delta_k = -\frac{\partial E}{\partial I_k}. \tag{A.6}$$

Using this, Eq. (A.4) can be rewritten as

$$-\frac{\partial E}{\partial w_{kj}} = \delta_k y_j. \tag{A.7}$$

Thus, from Eq. (A.3), changes to weights between output and hidden nodes can be determined from

$$\Delta w_{kj} = \eta \delta_k y_j, \tag{A.8}$$

where η is a proportionality constant and is often called the learning coefficient. It can be seen from Eq. (A.8) that the term δ_k or e_k is essential for determining the appropriate weight adjustment. To determine δ_k itself, the chain rule is applied to Eq. (A.6),

$$\delta_k = -\frac{\partial E}{\partial z_k} \cdot \frac{\partial z_k}{\partial I_k}. \tag{A.9}$$

The second term on the right-hand side is the derivative of the hyperbolic tangent transfer function, shown in Eqs. (3) and (5), such that

$$\frac{\partial z_k}{\partial I_k} = (1 + z_k)(1 - z_k). \tag{A.10}$$

The first term of Eq. (A.9) is already expanded in Eq. (A.2). Consequently, for the output layer,

$$\delta_k = (d_k - z_k)(1 + z_k)(1 - z_k). \tag{A.11}$$

The δ is slightly different when entering a hidden node, however, as the error from the output is propagated backwards. Thus

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial w_{ji}}, \tag{A.12}$$

which can be expanded by the chain rule

$$\Delta w_{ji} = -\eta \frac{\partial E}{\partial z_k} \frac{\partial z_k}{\partial I_k} \frac{\partial I_k}{\partial y_j} \frac{\partial y_j}{\partial I_j} \frac{\partial I_j}{\partial w_{ji}}. \tag{A.13}$$

The first partial derivative is already addressed in Eq. (A.2), the second in Eq. (A.10) and the fourth in Eq. (3). The third partial derivative is expanded by

$$\frac{\partial I_k}{\partial y_j} = \frac{\partial}{\partial y_j} \sum_j w_{kj} y_j = \sum_j w_{kj}. \tag{A.14}$$

Finally, the expansion of the fifth partial derivative is analogous to Eq. (A.5)

$$\frac{\partial I_j}{\partial w_{ji}} = \frac{\partial}{\partial w_{ji}} \sum_i w_{ji} x_i = x_i. \tag{A.15}$$

Thus

$$\Delta w_{ji} = \eta (d_k - z_k)(1 + z_k)(1 - z_k) \sum_j w_{kj} (1 + y_j)(1 - y_j) x_i. \tag{A.16}$$

Substituting δ_k from Eq. (A.11) into Eq. (A.16) gives

$$\Delta w_{ji} = \eta \sum_j \delta_k w_{kj} (1 + y_j)(1 - y_j) x_i. \tag{A.17}$$

The local delta or error for the hidden layer is, then,

$$\delta_j = -\frac{\partial E}{\partial I_j} = -\frac{\partial E}{\partial z_k} \frac{\partial z_k}{\partial I_k} \frac{\partial I_k}{\partial y_j} \frac{\partial y_j}{\partial I_j}. \tag{A.18}$$

The partial derivatives are the same as the first four derivatives in Eq. (A.13). The full expansion of this equation, then, is

$$\delta_j = (d_k - z_k)(1 + z_k)(1 - z_k) \sum_j w_{kj} (1 + y_j)(1 - y_j) \tag{A.19}$$

which can be simplified to

$$\delta_j = (1 + y_j)(1 - y_j) \sum_j \delta_k w_{kj}. \tag{A.20}$$

Eq. (A.20) can be substituted back into Eq. (A.17) to give a simpler expression of the adjustments to be made to the weights between the input and hidden layers as a result of supervised learning

$$\Delta w_{ji} = \eta \delta_j x_i. \tag{A.21}$$

Thus, the weights between the input and hidden layers are adjusted by

$$(w_{ji})_{\text{new}} = (w_{ji})_{\text{old}} + \Delta w_{ji}. \quad \text{(A.22)}$$

In the extended delta-bar-delta paradigm, the weight adjustment term, Δw_{ji}, is modified to include a portion (α, the momentum coefficient, where $\alpha < 1$) of the previous weight adjustment $(\Delta w_{ji})_{\text{old}}$, so that

$$(\Delta w_{ji})_{\text{new}} = \eta \delta_j x_i + \alpha (\Delta w_{ji})_{\text{old}}. \quad \text{(A.23)}$$

It is worth reiterating that in the models developed for this paper, there is only one output node, so $k = 1$ for all the derivations above. Also, it can be seen from the equations for the hidden layer weight adjustments (Eqs. (A.17) and (A.21)) how the error at the output layer, δ_k, is propagated backwards through the network and is incorporated in the weight adjustments between the input and hidden layers [10,11].

References

[1] M.F. Osborne, R.A. Lorenz, Results of ORNL VI series fission product release tests, Proceedings of the 20th Water Reactor Safety Information Meeting, Bethesda, Maryland, 21–23 October, 1992.

[2] M.R. Kuhlman, D.J. Lehmicke, R.O. Meyer, CORSOR User's Manual, NUREG/CR- 4173, USNRC March, 1985.

[3] D.S. Cox, Z. Liu, P.H. Elder, C.E.L. Hunt, V.I. Arimescu, Fission-product release kinetics from CANDU and LWR fuel during high-temperature steam oxidation experiments, presented at the IAEA Technical Meeting on Fission Gas Release and Fuel Rod Chemistry Related to Extended Burnup, Pembroke, Ontario, 28 April–1 May, 1992.

[4] D.S. Cox, Z. Liu, R.S. Dickson, P.H. Elder, Fission-product releases during post-irradiation annealing of high burnup CANDU fuel, presented at the Third International Conference on CANDU Fuel, Pembroke, Ontario, 4–8 October, 1992.

[5] B.J. Lewis, F.C. Iglesias, C.E.L. Hunt, D.S. Cox, Nucl. Technol. 99 (1992) 330.

[6] J. Rest, J. Nucl. Mater. 150 (1987) 203.

[7] A.C. Brito, F.C. Iglesias, Y. Liu, M.A. Petrilli, M.J. Richards, R.A. Cribb, P.J. Reid, SOURCE 2.0: A computer program to calculate fission product release from multiple fuel elements for reactor scenarios, Fourth International Conference on CANDU Fuel in Pembroke, Ontario, 1–4 October, 1995.

[8] M.F. Osborne, R.A. Lorenz, Nucl. Safety 33 (3) (1992) 334.

[9] B.J. Lewis, B. Andre, B. Morel, P. Dehaudt, D. Maro, P.L. Purdy, D.S. Cox, F.C. Iglesias, M.F. Osborne, R.A. Lorenz, J. Nucl. Mater. 227 (1995) 83.

[10] D.E. Rummelhart, J.L. McClelland, Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Volume 1: Foundations, MIT, Cambridge, MA, 1986.

[11] NeuralWare, Reference Guide for NeuralWorks Professional II/PLUS and NeuralWorks Explorer, Pittsburgh, PA, 1991.

[12] J.H. Gittus, J.R. Matthews, P.E. Potter, J. Nucl. Mater. 166 (1989) 132.

[13] H. Kleykamp, The chemical state of the fission products in oxide fuels, J. Nucl. Mater. 131 (1985) 221.

[14] D.R. Olander, Fundamental aspects of nuclear reactor fuel elements, Technical Information Center, Office of Public Affairs, Energy Research and Development Administration, US Department of Commerce, 1976.

[15] B.J. Lewis, B. Andre, G. Ducros, D. Maro, Nucl. Technol. 116 (1996) 34.

[16] B.J. Lewis, B.J. Corse, W.T. Thompson, M.H. Kaye, F.C. Iglesias, P. Elder, R. Dickson, Z. Liu, J. Nucl. Mater. 252 (1998) 235.

[17] W.S. Andrews, Ph.D. (Nuclear Engineering) dissertation, Department of Chemistry and Chemical Engineering, Royal Military College of Canada, 1994.

[18] R.L. Mason, R.F. Gunst, J.L. Hess, Statistical Design and Analysis of Experiments with Applications to Engineering and Science, Wiley, New York, 1989.

[19] B.J. Lewis, D.S. Cox, F.C. Iglesias, J. Nucl. Mater. 207 (1993) 228.

[20] A.C. Harnden-Gillis, B.J. Lewis, W.S. Andrews, P.L. Purdy, M.F. Osborne, R.A. Lorenz, Nucl. Technol. 109 (1995) 39.