LA-UR- 98-3093

Approved for public release; distribution is unlimited.

Title: PLUTONIUM ISOTOPIC DETERMINATION FROM GAMMA-RAY SPECTRA

CONF-980733--

Author(s): Alexei N. Skourikhine
Richard B. Strittmatter
Andrew Zardecki

Submitted to: 39th Annual Meeting of the Institute of Nuclear Materials Management
Naples, Florida
July 26-30, 1998

Los Alamos
NATIONAL LABORATORY

DISCLAIMER

PLUTONIUM ISOTOPIC DETERMINATION FROM GAMMA RAY SPECTRA

Alexei N. Skourikhine, Richard B. Strittmatter, and Andrew Zardecki
Los Alamos National Laboratory, MS E541, Los Alamos, NM 87545 USA

Abstract

The use of low- and medium-resolution room-temperature detectors for the nondestructive assay of nuclear materials has widespread applications to the safeguarding of nuclear materials. The challenge to using these detectors is the inherent difficulty of the spectral analysis to determine the amount of specific nuclear materials in the measured samples. This is especially true for extracting plutonium isotopic content from low- and medium-resolution spectral lines that are not well resolved. In this paper, neural networks trained by stochastic and singular value decomposition algorithms are applied to retrieve the plutonium isotopic content from a simulated NaI spectra. The simulated sample consists of isotopes ^{238}Pu, ^{239}Pu, ^{240}Pu, ^{241}Pu, ^{242}Pu, and ^{241}Am. It is demonstrated that the neural network optimized by singular value decomposition (SVD) and stochastic training algorithms is capable of estimating plutonium content consistently resulting in an average error much smaller than the error previously reported.

Problem Context

As a result of nuclear reactor operation, plutonium isotopes are naturally produced. The concentration of produced plutonium depends on the irradiation history, the time period during which the plutonium was stored, and whether it was reintroduced (recycled) into a reactor for additional energy production. Determination of concentrations of plutonium isotopes is important for IAEA applications and nonproliferation tasks [1].

In this research, neural networks having feedforward architecture and trained by stochastic [2] and singular value decomposition [3] algorithms that differ from the algorithm used in Ref.[4] are applied to retrieve the plutonium isotopic content from a simulated NaI spectra. The gamma-ray spectral data generated by SYNTH [5], a spectrum synthesizer developed by Pacific Northwest National Laboratory, forms a basis for training and testing the network. SYNTH allows the user to describe the sample geometry and composition and to define the quantities of radionuclides producing radiation, the source-detector distance, the thickness of specified absorbers, the size and composition of the detector, and the electronic setup used to gather data. Based on this input, SYNTH produces an appropriate gamma-ray spectrum. In the simulation we consider 1 g samples with fixed geometry, distance and air absorber. The NaI detector located 8 centimeters away from the sample generates the spectrum consisting of 512 channels, calibrated to 1.5 keV per channel. The synthesized spectrum obtained from SYNTH code is illustrated in Fig. 1.

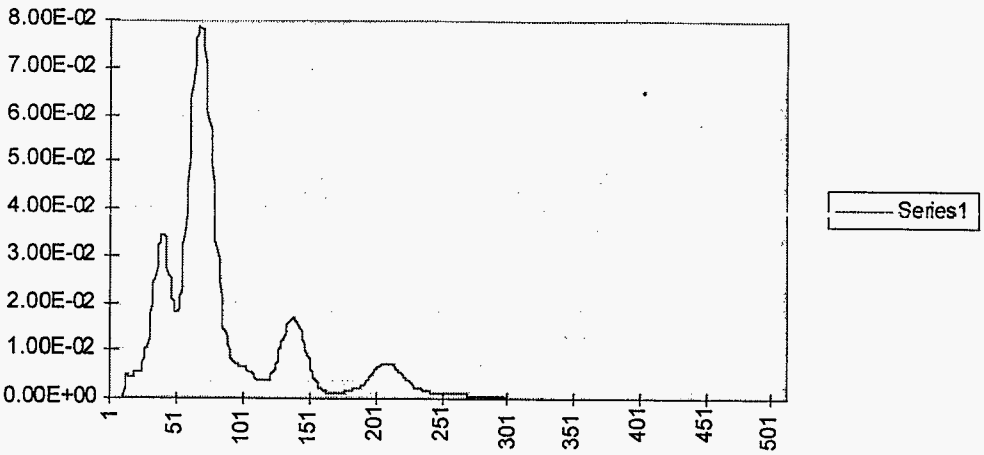

Fig. 1. Scaled synthetic gamma ray spectrum of a plutonium sample.

Data preparation is shown in Fig.2. The inputs to SYNTH describing the source are the concentration of isotopes, such as ^{238}Pu, ^{240}Pu, ^{241}Pu, ^{242}Pu, and ^{241}Am which are expressed as a function of x, ^{239}Pu concentration, for a typical light water reactor [4] (Eq.1-5). The ^{239}Pu concentration used for generating data is in the range of 45-100%.

$$C_{238}(x) = 7.589 \cdot 10^{-4} \cdot x^2 - 0.151x + 7.43, \quad (1)$$

$$C_{240}(x) = -11.21 \cdot 10^{-2} \cdot x^2 + 1.133x - 1.81, \quad (2)$$

$$C_{241}(x) = 5.061 \cdot 10^{-3} \cdot x^2 - 0.999x + 49.65, \quad (3)$$

$$C_{242}(x) = 5.702 \cdot 10^{-3} \cdot x^2 - 1.036x + 47.08, \quad (4)$$

$$^{Am}C_{241}(x) = 2.239 \cdot 10^{-3} \cdot x^2 - 0.480x + 25.50. \quad (5)$$

The neural network is aimed at an estimation of concentrations of Pu isotopes based on gamma-ray spectra obtained by a low-resolution NaI detector (Fig. 3).

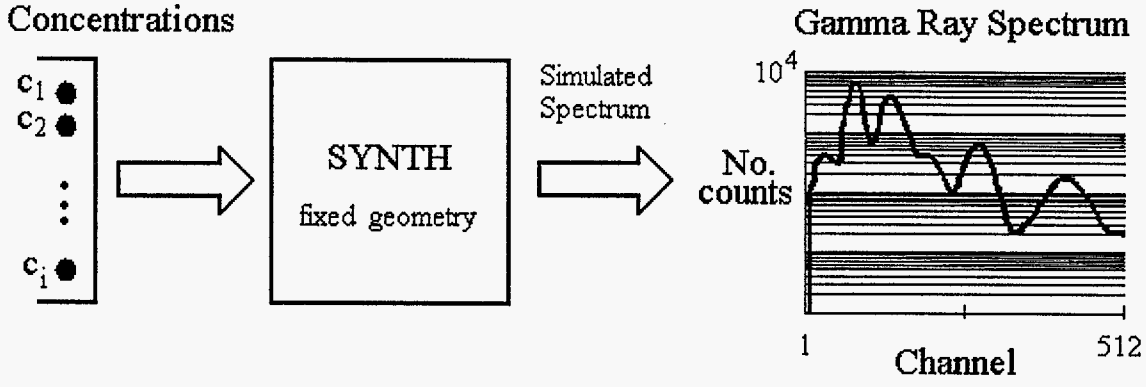

Fig. 2. Forward problem: Isotopes concentrations → Counts per channel.

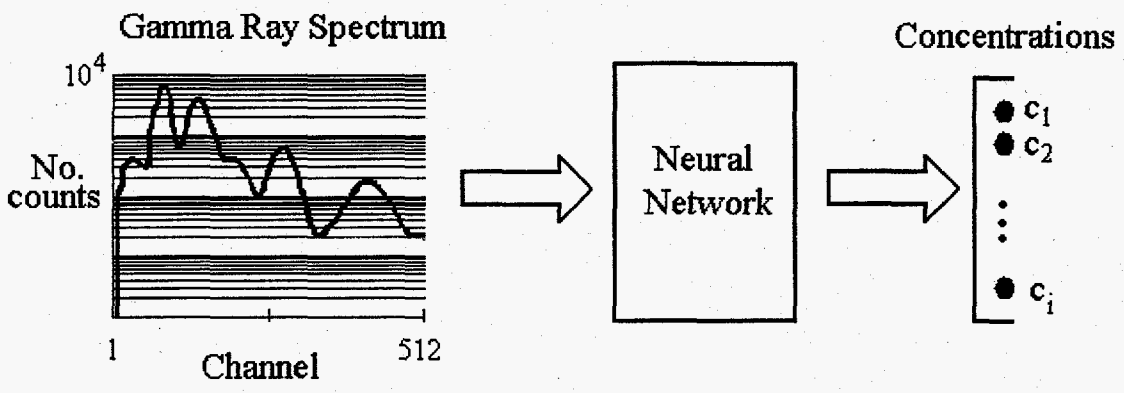

Fig. 3. Inverse problem: Counts per channel → Isotopes concentrations.

Neural Network Algorithms

Learning algorithms for neural networks can be categorized into two classes; those that depend on gradient calculations (such as backpropagation [6]) and those that do not. In this research, we apply two algorithms: Alopex [2], which belongs to the last category, and SVD modification of the cascade-correlation algorithm [7] representing the first class.

The Alopex stochastically training algorithm was developed by Unnikrishnan and Venugopal [2]. The algorithm does not make any assumptions about transfer functions of individual nodes, nor does it explicitly depend on the functional form of the error measure or the structure of the neural network. It minimizes the global error, E, with respect to network weights w, for a given set of training samples. But instead of error gradient, Alopex uses local correlations between changes in individual weights and changes in the global error.

The basic architecture of the model trained by Alopex is shown in Fig. 4. The model allows connections to skip the hidden layer, the outputs of nodes in the input layer connect directly to nodes in the output layer, in addition to usual links between neighboring layers.

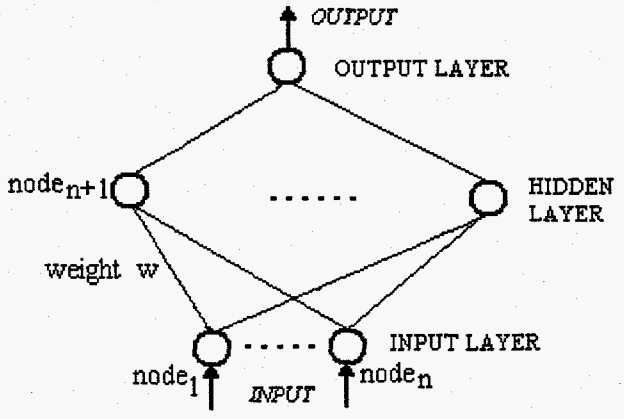

Fig. 4. Schematic of multiple layer network used to solve the problem by the ALOPEX algorithm.

The node's output, x_i, is calculated as:

$$x_i = f\left(\sum_j w_{ij} x_j\right), \tag{6}$$

where x_j is an output from the node j and w_{ij} is the weight connecting nodes j and i.

All the hidden and output nodes use the nonlinear sigmoid function,

$$f(x) = 1/(1+e^{-x}). \tag{7}$$

During the *t-th* iteration, the weight w_{ij} is updated according to the rule,

$$w_{ij}(t) = w_{ij}(t-1) + d_{ij}(t), \tag{8}$$

where $d_{ij}(t)$ is a small positive or negative step of size d with the following probabilities:

$$d_{ij}(t) = \begin{cases} -d & \text{with probability } p_{ij}(\Delta w_{ij}(t), \Delta E(t)) \\ +d & \text{with probability } (1 - p_{ij}(\Delta w_{ij}(t), \Delta E(t))) \end{cases} \tag{9}$$

where p_{ij} is a probability given by the Boltzmann distribution whose parameters depend on $\Delta w_{ij}(t)$ and $\Delta E(t)$, changes in weight w_{ij} and the error measure E over the previous two iterations.

The neural network architecture employed by the second algorithm we apply is shown in Fig. 5. This is cascade architecture that is similar to the architecture used by a cascade-correlation algorithm. In the cascade architecture hidden nodes are added to the network one at a time. For each new hidden node that is added to the network we adjust its incoming and outgoing weights by maximizing its income into the error reduction. The neural network training begins with some number of inputs and outputs dictated by the problem under consideration without any hidden nodes. In a result, outputs are linear superpositions of inputs. We find weights connecting inputs and outputs. Then, if we do not get an acceptably small error we add a new node, find its weights and so on until we get a small error or exceed a predefined number of iterations.

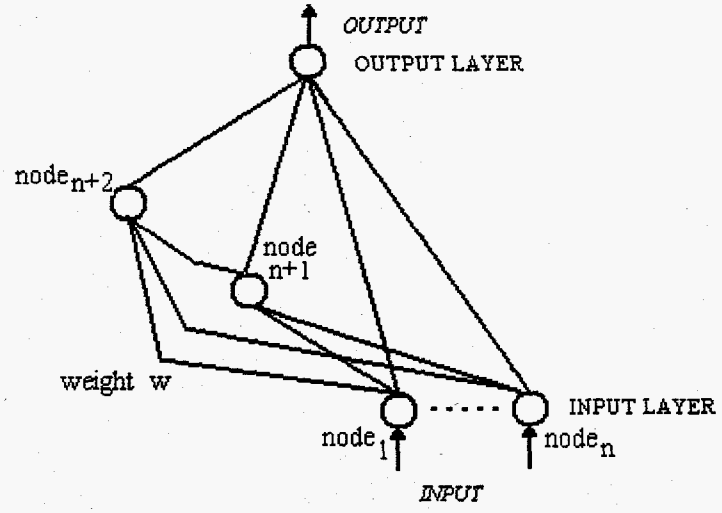

Fig. 5. Schematic of cascade network used to solve the problem by the SVD-based algorithm.

The known cascade-correlation algorithm minimizes error using gradient technique, the quickprop update method [8], that is very similar to the backpropagation. In our case, instead of quickprop we use SVD to find weights within the network. The neural network we consider transforms the input pattern into an output in a nonlinear way. However, the input to each node in the hidden or output layers is a linear combination of outputs of the nodes in the preceding layer. Since we know the target values at the output layer, we can, by applying the inverse activation function, given by Eq. (7), evaluate the linear combinations of outputs from the preceding layers. The inverse activation function of the targets for the output nodes is calculated for each training sample in order to create a vector of desired inputs for the output nodes Y. We then construct the matrix A containing activations of the hidden node whose number of rows is equal to the number of training samples. In the final stage we consider the equation

$$A \bullet X = Y , \tag{10}$$

where X is the node's weights including bias. The vector X can be found via SVD according to the following formula [3]:

$$X = V \bullet W^{-1} \bullet U^{T} \bullet Y , \tag{11}$$

where U and V are column-orthogonal matrixes well-known in SVD methodology. In Eq. (11), W is a diagonal matrix with positive or zero singular values.

The used network trained by ALOPEX has 11 input nodes, 10 hidden nodes, and 3 output nodes. The cascade network trained by SVD evolved into an architecture with 2 hidden nodes. It has 11 inputs and 3 outputs. The data are divided into training and testing sets. We use the first set to train the network, whereas the second one is used in the retrieval stage to test the trained network. The training set contains 20 data cases generated for the ^{239}Pu concentrations between 45% and 95%, in steps of 5%. The testing set contains concentration values of ^{239}Pu ranging from 47.5% to 97.5%, in steps of 5%. The considered data channels are those at 204, 333, 367.5, 634.5, 646.5, 649.5, 651, 652.5, 655.5, 660, 666 keV. The selection of these channels were made based on prior knowledge of their usefulness for the considered problem.

Results of Simulation Experiments

Results of the experiment and its comparison with the results referenced in [4] are shown in Table 1. As a rule, if the process of Alopex training converged, it took about 10000 iterations on the average. However, some trials failed to meet the pre-specified criterion for convergence, especially after increasing the number of output variables. The algorithm of the cascade construction of the neural network based on SVD always finds a solution. Training stops after adding two nodes with sigmoid functions to the network. The results show the superiority of Alopex and cascade training over the multiple-layer architecture trained by backpropagation. Accuracies achieved by SVD based algorithm for ^{238}Pu and ^{241}Pu are 3.36% and 1.98% respectively.

Table 1. Retrieval of ^{239}Pu

Training Algorithm	Mean Relative Error (%)
Backpropagation algorithm [4]	0.949
Alopex algorithm	0.637
SVD algorithm	0.903

Conclusions

Stochastic Alopex and SVD-based neural network training algorithms were applied to plutonium isotopic determination from spectra. Their applicability and usefulness for a given problem are shown. Comparative analysis has shown better performance of the SVD training algorithm, which does not fail to converge to an acceptable result as happens with Alopex. The principal advantage of the cascade network trained by SVD over Alopex is an incremental construction of the network architecture without any initial specification. The network evolves by itself to the architecture that optimizes predefined criteria. This is not the case for Alopex training where we have to specify the network's architecture beforehand. The resulting mean relative errors for ^{239}Pu over a testing set obtained by the SVD training algorithm is 0.903%. Future research will focus on application of the neural networks to increasingly realistic samples accounting for nonlinear absorption effects and variable geometry.

References

[1] T. E. Sampson, "Plutonium isotopic composition by gamma-ray spectroscopy," in *Passive Nondestructive Assay of Nuclear Materials* (D. Reilly, N. Ensslin, and H. Smith, Jr., eds.), pp. 241-271, 1991.

[2] K. P. Unnikrishnan and K. P. Venugopal, "Learning in connectionist networks using the ALOPEX algorithm," in *Proc. of the Intern. Joint Conf. on Neural Networks*, Part I, pp. 926-931, 1992.

[3] G. E. Forsythe, M. A. Malcolm, and C. B. Moler, *Computer methods for mathematical computations*, Prentice-Hall, Inc., 1977.

[4] A. Zardecki, and R. B. Strittmatter, "Chemical and Isotopic Determination from Complex Spectra," *Nucl. Mater. Manage*. XXIV, 81-822, 1995.

[5] W. K. Hensley, A. D. McKinnon, H. S. Miley, M. E. Panisko, and R. M. Savard, "SYNTH: a spectrum synthesizer," in *INMM: 35th Annual Meeting Proceedings*, Naples, Florida, July 17-20, pp. 629-634, 1994.

[6] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Internal Representations by Error Propagation," in *Parallel Distributed Processing*, MIT Press, Vol.1, pp. 318-362, 1986.

[7] S. E. Fahlman and C. Lebiere, "The cascade-correlation learning architecture," *Advances in Neural Information Processing Systems*, **2**, (R. P. Lippmann, J. E. Moody, D. S. Touretzky, eds.), 524-532, San Francisco: Morgan Kaufmann, 1991.

[8] S. E. Fahlman, "An empirical study of learning speed in back-propagation networks," Carnegie Mellon University *Technical Report CMU-CS-88-162*, September 1988.