Technical Report
RAL-TR-2001-043

CLRC

Initial Studies in the Modelling of a Position Resolving Calorimeter Detector

C Greenough J V Ashby and R F Fowler

29th November 2001

COUNCIL FOR THE CENTRAL LABORATORY OF THE RESEARCH COUNCILS

Enquiries about copyright, reproduction and requests for additional copies of this report should be addressed to:

The Central Laboratory of the Research Councils
Library and Information Services
Rutherford Appleton Laboratory
Chilton
Didcot
Oxfordshire
OX11 0QX
Tel: 01235 445384 Fax: 01235 446403
E-mail library@rl.ac.uk

ISSN 1358-6254

Initial Studies in the Modelling of a Position Resolving Calorimeter Dectector

C Greenough, J V Ashby and R F Fowler

October 2001

Abstract

In this report we describe some initial results in the modelling of a position resolving calorimeter detector. The ideas behind the device are based on those of the *microcalorimeter*. These detectors use the heat generation of an X-ray event to provide information on the event's time and position when assembled into an array.

In this report we consider a larger detector which uses the same thermal ideas of the microcalorimeter but also utilises multiple thermistors to determine the event postition on the detector's surface. This first attempt at modelling makes the basic assumption that the heat transport can be represented through a simple linear diffusion process and through that the times at which the temperature change reaches the edge sensors can be used to determine the position of the event. A simple idealised device has been used to assess the potential modelling these devices to aid their design and development.

The report develops a finite element model of the device and performs a series of numerical experiments. The results of these experiments are compared with a simple analytic model. Two methods of determining the event position are presented: one based on the analytic solution and a second using neural network.

A copy of this report can be found at the Department's web site (*http://www.cse.clrc.ac.uk/*) under page *Group/CSEMSW* or anonymous ftp server *www.inf.rl.ac.uk*.

Mathematical Software Group
Computation Science & Engineering Department
Rutherford Appleton Laboratory
Chilton, DIDCOT
Oxfordshire OX11 0QX

Contents

1 Introduction

There are many applications of detectors in science and engineering research. Two large areas are in detectors for high energy physics experiments and in space telescopes of different types. To date considerable use has been made of CCD devices in these area and there is a continuing activity to improve these types of imaging system and to design new generations of devices.

2 The Physical Model

In this initial modelling activity of position sensive calorimeter detectors we will assume that the heat conduction process is purely one of diffusion. However, it may well be the case that the diffusion process does not adequately represent the heat transport due to phonon and electron generation after an X-ray event. This will be true if the dominant transport mechanism is ballistic. Clearly this assumption may well be inadequate but it serves as a starting point.

This temperature transport process will be governed by the time-dependent heat transport equation [10]:

$$\frac{\partial T}{\partial t} = \frac{H}{C_p} + \frac{1}{\rho C_p} \nabla \cdot (k \nabla T) \tag{1}$$

where H is the heat production per unit mass of any source, C_p is the specific heat, ρ the density and k the thermal conductivity. If there are no heat sources and k is not a function of position (1) becomes:

$$\frac{\partial T}{\partial t} = \frac{k}{\rho C_p} \nabla^2 T \tag{2}$$

However k may well be a function of T.

In general the physical properties of conductors such as gold and silver can be nonlinear over temperature ranges being consider - 1-5^{o}K. The three important material properties are: ρ, the density; k, the thermal condutivity and C_p, the specific heat capacity.

In the liturature there is not a good consensus of k and C_p for gold particularly at low temperatures. The graphs in Figure 1 and Figure 2 show a `log-log` plot of thermal conductivity and specific heat against temperatue for gold as given in [11, 12].

These figures show that although k and C_p are temperature dependent in the region of interest, around 1^oK, their variation is near linear. This coupled with the fact that the X-ray strike will generally only cause a $\sim 0.1^oK$ temperature rise it is thought that taking constant values for ρ, k and C_p would be adequate in these initial studies.

Assuming that k, ρ and C_p are independent of position and time a basic solution of (2) is given by, assuming the initial headt distribution is a delta function:

$$T(\vec{r}, t) = \frac{\alpha}{t} \exp(-r^2/2\beta t) \tag{3}$$

where $\beta = 2k/C_p\rho$ and α is dependent on the initial energy input and the size of the detector [10]. We can use this to assess the numerical experiments in later sections.

3 Temperature Spike Transmission Time

Once the X-ray event has occurred a temperature pulse will spread out through the device. The sensors at the detector edges will measure a rise in temperature as the pulse reaches

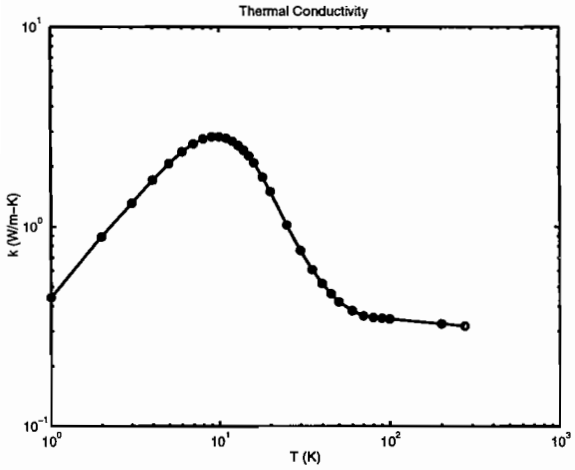

Figure 1: Log-log plot of the thermal conductivity k

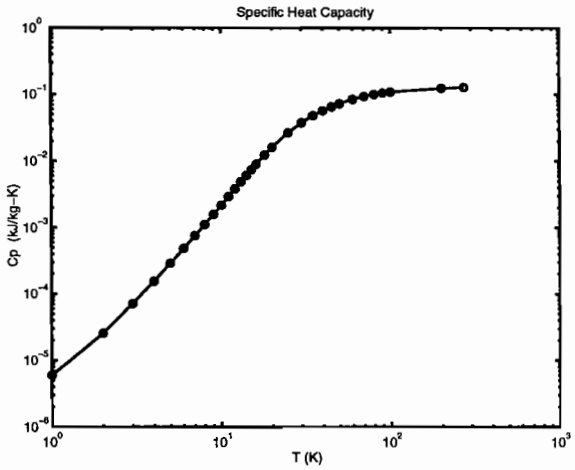

Figure 2: Log-log plot of the specific heat C_p

them. In the simple case, where the material is assumed linear and the event happens on a large (compared with the detector area) plate, the pulse diffuse out from the impact point with circlular symmetry. The position of the event will be directly related to the speed of this diffusion process and hence its arrival time.

The differences in arrival time of the pulse at the sensors will provide sufficient information to determine the event position ($\vec{r}_o$) and time (t_o). For the purposes of this paper we shall take the arrival (t_i) to be that at which the maximum temperature rise is seen by a sensor.

The time taken for the temperature wave to reach the sensor will clearly be dependent on the material properties of the detector and its shape. Two approaches to detemine $\vec{r}_o$ and t_o will be discussed in later sections.

If the arrival time is defined as the arrival time of the peak (i.e. temperature maximum) then we can derive a simple relationship between the arrival time and the distance to the event using the analytic form (3). At the temperatue peak

$$\frac{\partial u}{\partial t} = 0$$

So using (3) we have

$$\frac{\partial u}{\partial t} = -\frac{\alpha}{2\beta t^3}\left(2\beta t - r^2\right)\exp\left(-\frac{r^2}{2\beta t}\right) = 0$$

thus

$$\begin{aligned} 0 &= 2\beta t - r^2 \\ r &= \sqrt{2\beta t} \end{aligned} \tag{4}$$

where r is the distance to the event and t time taken for the temperature peak generated by the event to reach the sensor. This expression will be used later in deriving an analytic expression for the position of an event.

4 The Computational Model

The diffusion equation will be solved computationally using the finite element method. The detector plate is subdivided into a number of finite elements. The number and distribution of these can be varied to match the accuracy requirements of the solution. Associated with each element are a set of discrete points (called nodes) at which the temperature will be approximated. In the simplest case there will be four nodes associated with each quadrilateral element.

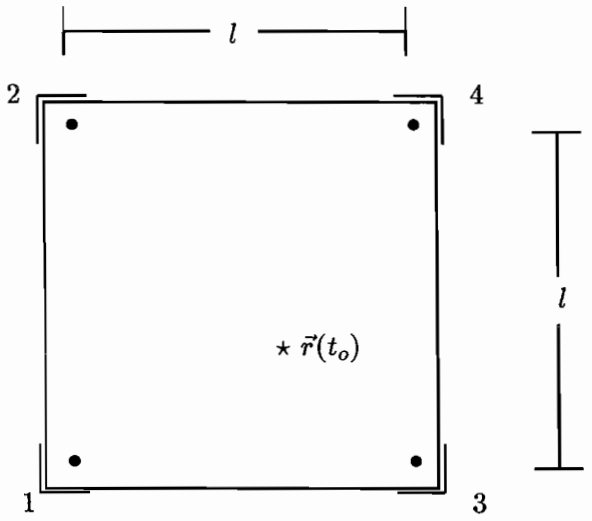

Figure 3: Geometry of test device

Over each of these finite elements the temperature distribution is approximated by a bi-linear polynomial using the nodes associated with the element. A typical finite element mesh is shown in Figure 4. The temperature representation over each element is given by:

$$T_e = \sum_{i=1}^{n} N_i T_i \tag{5}$$

where N_i are the interpolation functions (shape functions) and T_i are the nodal value of the temperatue. The governing equation (1) is solved by using the Galerkin approach. So (1) becomes:

$$\int_{\Omega_e} N_j \left(\frac{1}{\rho C_p} \nabla \cdot k \nabla T_e + \frac{H}{C_p} - \frac{\partial T_e}{\partial t} \right) d\Omega = 0 \tag{6}$$

where N_j are the shape functions. Integration by parts and substitution of (5) into (6) leads to:

$$\int_{\Omega_e} \frac{1}{\rho C_p} \nabla N_j \cdot k \nabla N_i \, d\Omega \; T_i + \int_{\Omega_e} \frac{1}{C_p} N_i H d\Omega - \int_{\Omega_e} N_j N_i \, d\Omega \; \frac{\partial T_i}{\partial t} = 0 \;\; \forall \;\; i,j = 1,n \tag{7}$$

If we assume that k is independent of position and that the heat sources, H, can be represented in the same manner as T:

$$H_e = \sum_{i=1}^{n} N_i H_i \tag{8}$$

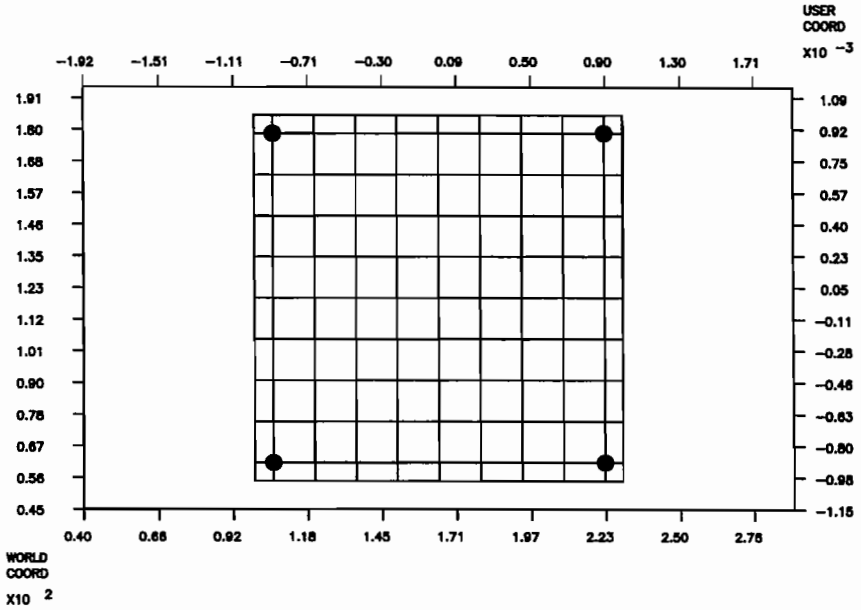

Figure 4: Test device and its finite elemen mesh

we can simplify this single equation for node j and combine all nodes in an element into a single system of ordinary differential equations for the element.

$$\mathbf{K}_e\mathbf{T}_e + \mathbf{M}_e\frac{\partial \mathbf{T_e}}{\partial t} = \mathbf{S}_e\mathbf{H}_e \tag{9}$$

where

$$\begin{aligned}
\mathbf{T}_e &= [T_i, T_j, T_k, T_l] \\
\mathbf{H}_e &= [H_i, H_j, H_k, H_l] \\
K_{ij}^e &= \frac{k}{\rho C_p}\int_{\Omega_e} \nabla N_i \cdot \nabla N_j \, d\Omega \\
M_{ij}^e &= \int_{\Omega_e} N_i \cdot N_j \, d\Omega \\
S_{ij}^e &= \frac{1}{C_p}\int_{\Omega_e} N_i \cdot N_j \, d\Omega
\end{aligned} \tag{10}$$

These elemental equations can be assembled into a system of equations for the whole problem domain. The solution of the time dependent system (9) is performed using an implicit θ method. By this technique two time levels, n and $n+1$, are linked through a weight θ. (9) is written as

$$\begin{aligned}
\mathbf{K}_e\left[\theta\mathbf{T}_e^{n+1} + (1-\theta)\mathbf{T}_e^n\right] + \mathbf{M}_e\left[\theta\frac{\partial \mathbf{T}_e}{\partial t}^{n+1} + (1-\theta)\frac{\partial \mathbf{T}_e}{\partial t}^{n}\right] \\
= \mathbf{S}_e\left[\theta\mathbf{H}_e^{n+1} + (1-\theta)\mathbf{H}_e^n\right]
\end{aligned} \tag{11}$$

where $n+1$ and n indicate that the expression is evaluated at times $t_{n+1} = (n+1)\Delta t$ and $t_n = n\Delta t$. Δt being the step size in time. Finally the time derivatives are approximated by finite differences; a forward difference at n and a backward difference at $n+1$. This leads to

$$\left[\mathbf{K}_e\theta + \frac{1}{\Delta t}\mathbf{M}_e\right]\mathbf{T}_e^{n+1} + \left[(1-\theta)\mathbf{K}_e - \frac{1}{\Delta t}\mathbf{M}_e\right]\mathbf{T}_e^{n} = \mathbf{S}_e\left[\theta\mathbf{H}_e^{n+1} + (1-\theta)\mathbf{H}_e^{n}\right] \qquad (12)$$

where the value of θ can be varied to provide different schemes. Two common values are $\theta = \frac{1}{2}$, the Crank-Nicolson scheme, and $\theta = 1$, the fully implicit scheme.

5 The Computational Experiments

The software developed for these experiments was based on the NAG/SERC Finite Element Library [7] and the basic structure of the program followed the example program Segment 4.1 (Transient Heat Conduction) [8, 9].

For these initial experiments a very simple detector structure has been used - that of a square plate (see Figure 3 and Figure 4). The plate has been subdivided into quadrilateral finite elements. The values of ρ, k and C_p were taken to be: $19500kg/m^3, 2.07kW/m^oK$ and $2.9\text{x}10^{-4}kJ/kg^oK$. The temperature sensors are marked by the cirles in each corner of the device.

The initial background temperature of the device was 1^oK and the event strength set so that an initial temperature rise of around 0.1^oK was observed near the event position. The basic boundary conditions used were $\partial T/\partial n = 0$ over the majority of the boundary save near the sensors where T was fixed to 1^oK. The mesh density and time step size were chosen to produce mesh and time step independent results. In all 121 nodes and 100 elements were used with a time step of $1.0\text{x}10^{-7}$ s (10 μs).

The results of the simulation are best displayed through temperature response graphs at the sensors. Figure 5 show the four response graphs of the model detector for an event occuring at the centre of the plate. These graphs show the temperature rise at the four sensors and in this case are identical as the event was a point equi-distant from the sensors. The identical graphs confirm the symmetry of the problem and to some extent the accuracy of the computational solution. Table 1 gives the basic information on the temperature peak arrival times (t_{peak}) and the size of the peak (T_{peak}) for this case. The second set of graphs in Figure 6 show

Sensor	$T_{peak}(^oK)$	$t_{peak}(s)$
1	1.008	$1.0\text{x}10^{-6}$
2	1.008	$1.0\text{x}10^{-6}$
3	1.008	$1.0\text{x}10^{-6}$
4	1.008	$1.0\text{x}10^{-6}$

Table 1: Temperature peak characteristics for $(0,0)$ event

the sensor response graphs for an event at an arbitary point on the detector surface. As can be clearly seen the response graphs are quite different and their peaks reach the sensors at differring times. Table 2 summarises these results for one arbitrary event position.

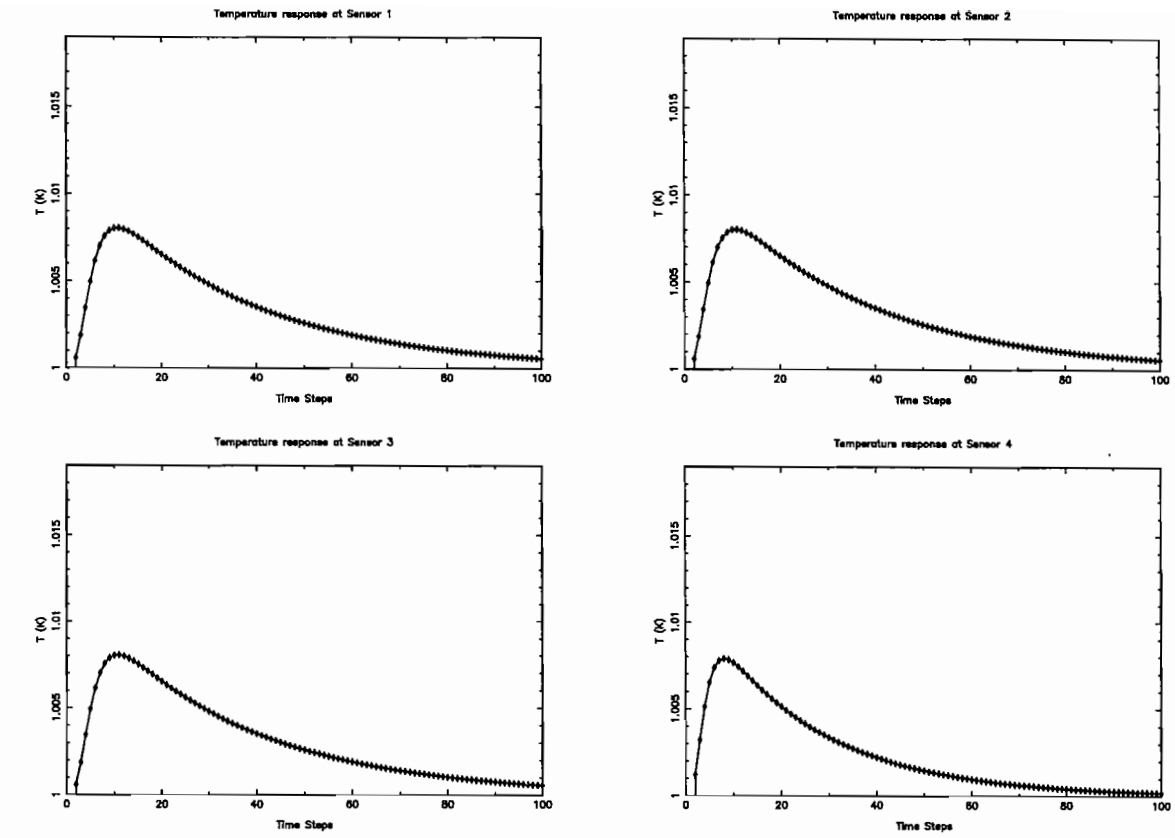

Figure 5: Sensor Response Curve for event at $(0,0)$

The calculation of the position and event time from the four arrival times will be considered in later sections.

These computational results can be compared against the analytic solution (3) which represents the diffusion of a temperature pulse in a semi-infinite plate. The value of α is related to the total heat in the system at any one time. Although one could compare the instaneous temperture change at particular time a more appropriate comparison will be in the temperature response curve at each sensor and the values of T_{peak} and t_{peak} as given in Tables 1 and 2. Using the same values of ρ, k, C_p and r_o Tables 3 and 4 give these values for the same two events: $(0,0)$ and $(0.2, 0.4)$.

As can be seen these are very similar values as one might expect from such a simple device geometry.

The final set of computational results produced are a complete time and temperature

Sensor	$T_{peak}(^oK)$	$t_{peak}(s)$
1	1.0050	$1.8\text{x}10^{-4}$
2	1.0091	$0.8\text{x}10^{-6}$
3	1.0178	$0.5\text{x}10^{-6}$
4	1.0062	$1.3\text{x}10^{-4}$

Table 2: Temperature peak characteristics for $(0.2, 0.4)$ event

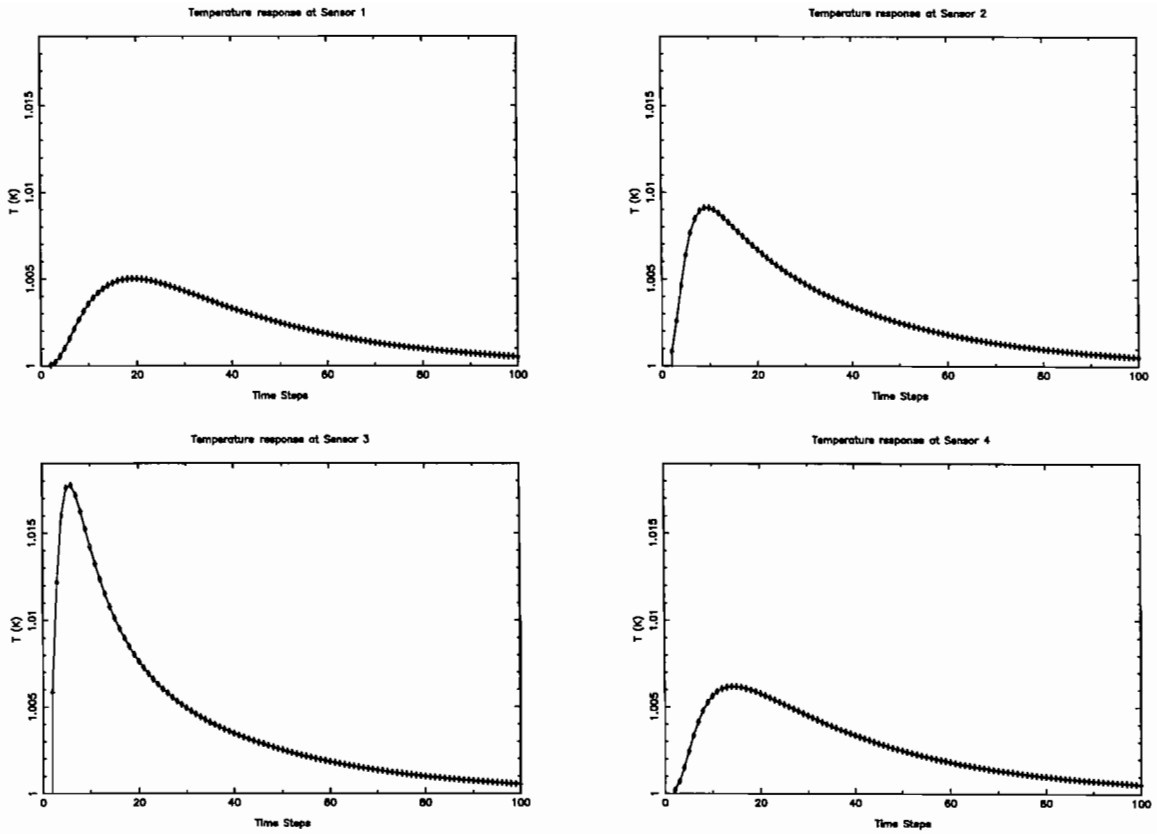

Figure 6: Sensor Response Curve for event at an arbitary point (0.2,0.4)

response field for the device. Figure 7 and Figure 8 show the temperature and time response surfaces at Sensor 1 for any event occuring in the device. It has been generated through a sequence of 361 events space evenly on a regular 19x19 grid. The results from these runs show good symmetry not only at one sensor but also between all sensors reflecting the basic symmetry of the device. The castellation in the time response surface reflects the $1.0\text{x}10^{-7}$ time step being used in the computations.

Sensor	$T_{peak}(^oK)$	$t_{peak}(s)$
1	1.008	$1.1\text{x}10^{-6}$
2	1.008	$1.1\text{x}10^{-6}$
3	1.008	$1.1\text{x}10^{-6}$
4	1.008	$1.1\text{x}10^{-6}$

Table 3: Analytic temperature peak characteristics for $(0.0, 0.0)$ event

Sensor	$T_{peak}(^oK)$	$t_{peak}(s)$
1	1.0046	$2.0\text{x}10^{-6}$
2	1.0090	$1.0\text{x}10^{-6}$
3	1.0178	$0.5\text{x}10^{-6}$
4	1.0061	$1.5\text{x}10^{-6}$

Table 4: Analytic temperature peak characteristics for $(0.2, 0.4)$ event

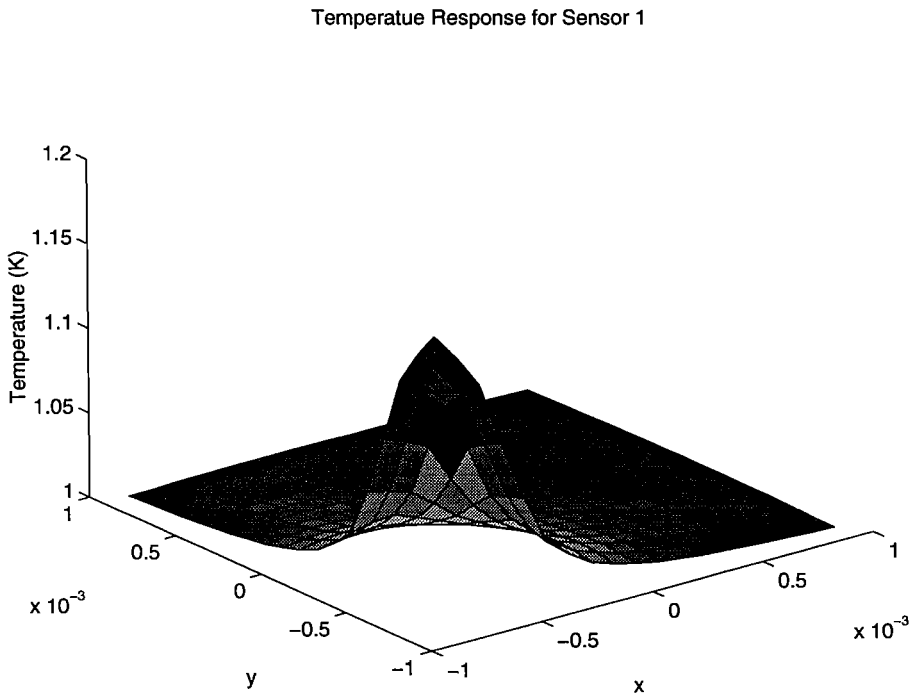

Figure 7: Temperature response surface

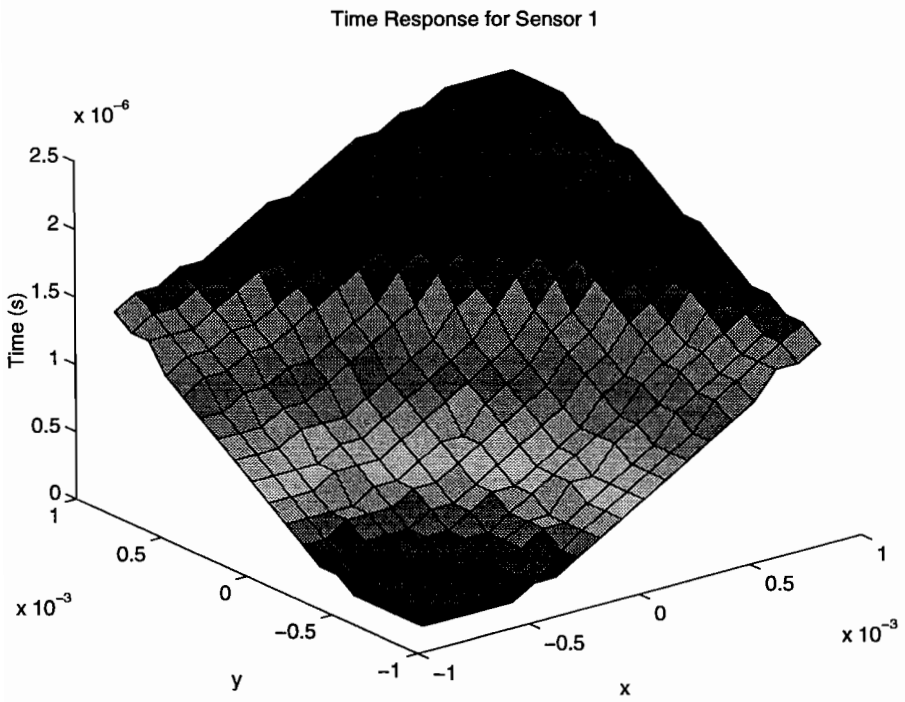

Figure 8: Time response surface

6 Analytic Position Detection

The outputs of the model detector are a set of four times at which the temperature peaks arrive at a particular sensor. This is the raw input to the position calculation. If the basic analytic solution (3) is assumed then this position can be found very simply using the expressions below. Give a set of three arrival times t_1, t_2 and t_3 the position of the event $\vec{r}(t_o)$ satifys the following relations using (4):

$$\begin{aligned} |\vec{r}_1 - \vec{r}_o|^2 &= \beta(t_1 - t_o) \\ |\vec{r}_2 - \vec{r}_o|^2 &= \beta(t_2 - t_o) \\ |\vec{r}_3 - \vec{r}_o|^2 &= \beta(t_3 - t_o) \end{aligned} \tag{13}$$

t_o is eliminated between the three equtions. This leads to:

$$\begin{aligned} |\vec{r}_2 - \vec{r}_o|^2 - |\vec{r}_1 - \vec{r}_o|^2 &= \beta(t_2 - t_1) \\ |\vec{r}_2 - \vec{r}_o|^2 - |\vec{r}_3 - \vec{r}_o|^2 &= \beta(t_2 - t_3) \end{aligned} \tag{14}$$

two equations, in the two unknowns x_o and y_o (using $\vec{r}_k = \{x_k, y_k\}$). Expansion of these leads to:

$$\begin{aligned} \beta(t_2 - t_1) &= x_2^2 + y_2^2 - x_1^2 - y_1^2 + 2[x_o(x_1 - x_2) + y_o(y_1 - y_2)] \\ \beta(t_2 - t_3) &= x_2^2 + y_2^2 - x_3^2 - y_3^2 + 2[x_o(x_3 - x_2) + y_o(y_3 - y_2)] \end{aligned} \tag{15}$$

This can all be express more conviently in a matrix form as:

$$\frac{1}{2}\begin{pmatrix} \beta(t_2 - t_1) - x_2^2 - y_2^2 + x_1^2 + y_1^2 \\ \beta(t_2 - t_3) - x_2^2 - y_2^2 + x_3^2 + y_3^2 \end{pmatrix} = \begin{pmatrix} x_1 - x_2 & y_1 - y_2 \\ x_3 - x_2 & y_3 - y_2 \end{pmatrix}\begin{pmatrix} x_o \\ y_o \end{pmatrix} \tag{16}$$

The positions of the sensors are given by the vectors $\vec{r}_1$, $\vec{r}_2$ and $\vec{r}_3$. For the simple configuration in Figure 3 one set of vectors could be:

$$\vec{r}_1 = l\begin{pmatrix} -1 \\ -1 \end{pmatrix}, \vec{r}_2 = l\begin{pmatrix} -1 \\ 1 \end{pmatrix} \text{ and } \vec{r}_3 = l\begin{pmatrix} 1 \\ 1 \end{pmatrix} \tag{17}$$

where $2l$ is the edge distance between sensors.

$$\frac{1}{2}\begin{pmatrix} \beta(t_2 - t_1) \\ \beta(t_2 - t_3) \end{pmatrix} = l\begin{pmatrix} 0 & -2 \\ 2 & 0 \end{pmatrix}\begin{pmatrix} x_o \\ y_o \end{pmatrix} \tag{18}$$

Clearly

$$x_o = \frac{1}{4l}\beta(t_1 - t_2) \tag{19}$$

$$y_o = \frac{1}{4l}\beta(t_2 - t_3) \tag{20}$$

The time at which the event took place is recovered from one of (13) given x_o and y_o.

7 General Position Detection

In general the device may not operate in as simple a manner as an infinite plane. The shape of the detector will effect the transmission time of the temperature pulse as might variability in the material properties. Although one might think that all is required is the fitting of a surface to the output to characterise the detector's response this surface might not be a simple bivariate polynomial. Given four sensors we may have four outputs (t_1, t_2, t_3 and t_4) to determine the three inputs (x_o, y_o and t_o).

A more general approach of determining the event time and position is through the use of a neural network which is *trained* to simulate the detector's response. The Multilayer Perceptron (or MLP) network is probably the most often considered member of the neural network family. It is able to model both simple and very complex functional relationships which has been proven through a large number of applications [4]. Figure 9 shows the general characterisation of an MLP network. The basic ideas and software used is from the *Neural Network Based System Identification TOOLBOX* of Norgaard [1]. The class of MLP-network

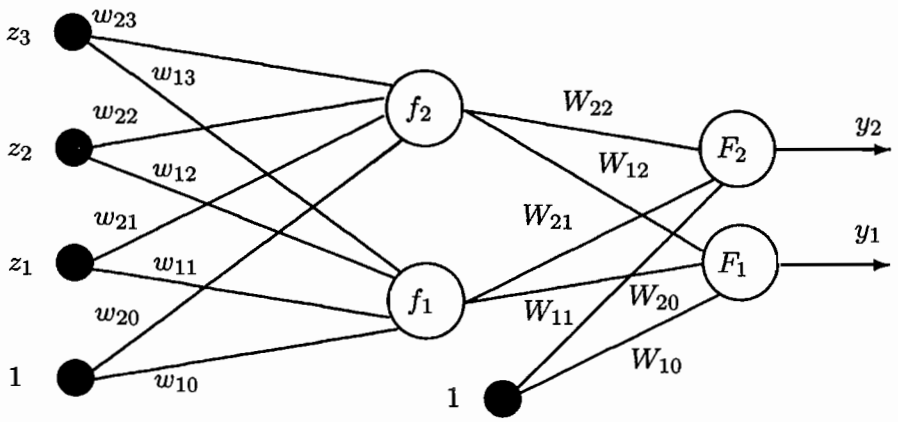

Figure 9: A fully connected two-layer MLP-network

being used is the two-layer model in which there is only one hidden layer. Also the activation functions (f, F) have been chosen to be the hyperbolic tangent and the linear activation.

We will give a brief description (this material draws heavily on [1]) of the use and training of neural network but more detailed explanation can be found in [5]. The Two-layer MLP can be characterised by the following expression:

$$\begin{aligned} \hat{y}_i(\mathbf{w},\mathbf{W}) &= F_i\left(\sum_{j=1}^{q} W_{ij} h_j(w) + W_{i0}\right) \\ &= F_i\left(\sum_{j=0}^{q} W_{ij} f_i\left(\sum_{l=1}^{m} w_{jl} z_l + w_{j0}\right) + W_{i0}\right) \end{aligned} \tag{21}$$

The weights (specificied by the vector θ or by the weight matrices $\mathbf{w}$ and $\mathbf{W}$) are the adjustable parameters of the network and they are determined from a set of *examples* through the process called *training*. The examples, or the training data as they are usually called, are a set of

input, $u(t)$ and corresponding desired outputs, $y(t)$. The objective of training is to determine a set of weights that will produce predictions $\hat{y}(t)$ which are *close* to the true output $y(t)$.

The method by which the weights are determined is through the minimisation of some mean square error criterion. This can be represented by

$$V_N(\theta, Z^N) = \frac{1}{2N} \sum_{t=1}^{N} (y(t) - \hat{y}(t,\theta))^T (y(t) - \hat{y}(t,\theta)) \tag{22}$$

where θ is a vector of weights and N the number of data points in the training set.

The weights are then found as

$$\theta = min V_N(\theta, Z^N) \tag{23}$$

by some kind of iterative minimisation scheme

$$\theta^{i+1} = \theta^i + \mu^i f^i \tag{24}$$

where θ^i specifies the current iterate, f^i is the search direction and μ^i the step size.

A large number of training algorithms exist, each of which is characterised by the way in which the search direction and step size are selected. For these computations the method due to Levenberg and Marquardt is used the details of which are given in [6].

The NNSYSID Toolbox was developed for the MATLAB System [3] however the version of MATLAB available to the authors was only the rather restrictive *Student* version. However the SCILAB System [2] developed at INRIA provides very similar functionality to MATLAB and a basic converter between MATLAB *m-files* and SCILAB macros (*sci-files*), `mfile2sci`. Using this tool and some subsequent small modifications the NNSYSID Toolbox has been converted into SCILAB functions.

7.1 Initial test using analytic form

Some inital tests were carried out on a four-input, three-output two-layer feed-forward network using four hidden units. The first tests were run using a limited training set generated using the analytic solution (3). A very simple problem was investigated. Assuming a 2x2 device with the same physical characteristics as the test device, the analytic solution was used to generate a grid of 36 data points (four sensor time and two positions) which was divided into the *training* data and the *test* data.

Firstly the basic data had been scaled to milimeters and micro-sceonds. Using the basic Levenberg-Marquardt training algorithm provide in the NNSYSID Toolbox a MLP network with four inputs and two outputs, with four hidden units was trained. This took about 200 iterations of the algorithms to reduce the error to around $1\text{x}10^{-8}$. Figure 10 and Figure 11 show the fitting between the predicted and actual values of the two outputs (x, y) for the test data together with plots of the prediction error. In this test only the position has been calculated.

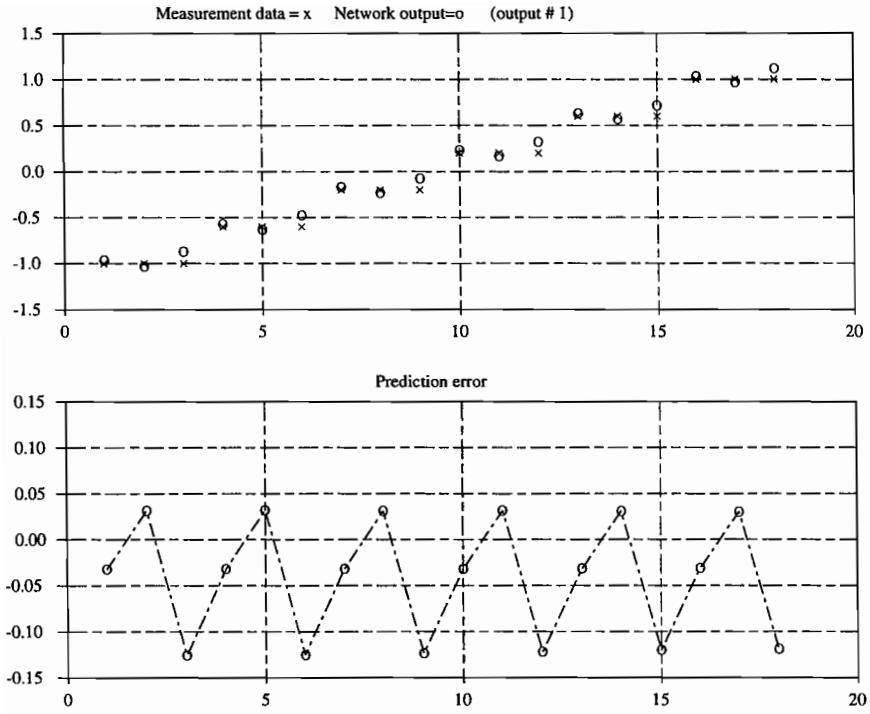

Figure 10: Fitting of *output 1* (x coordinate) to initial test data

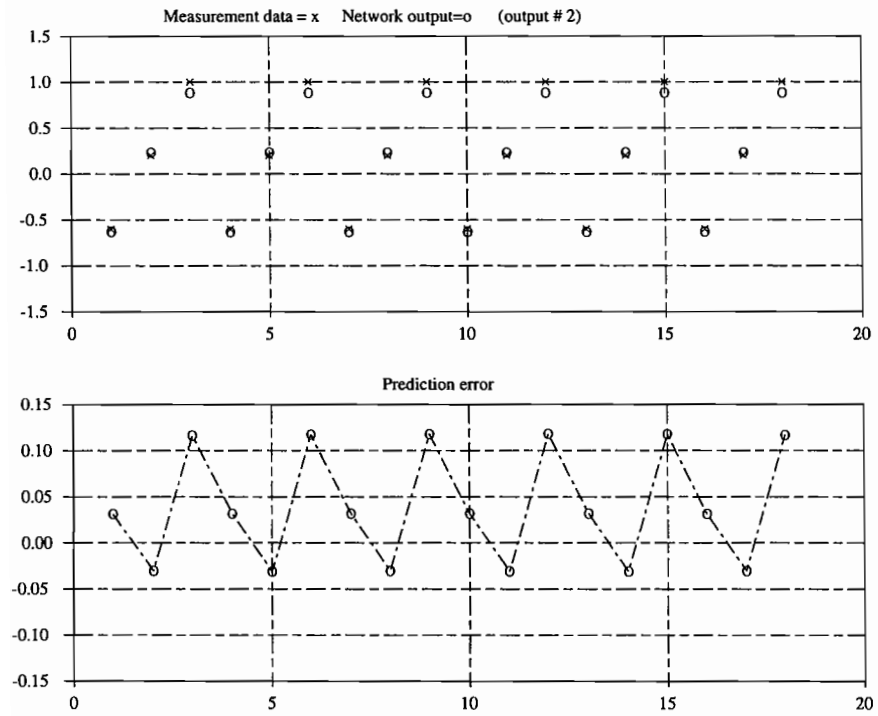

Figure 11: Fitting of *output 2* (y coordinate) to initial test data

7.2 Fitting of computational simulations

The second step in using the neural network concept for position determination is to use the results from the computational model. In general the temperature/time response of the device to an event will not conform to the simple analytic solution. Variations in device shape and material properties will affect its performance.

The computational model was used to generate a grid of data points over the device shown in Figure 4. 361 data points consisting of the event position, (x_i, y_i) and the response times and temperature peaks for each sensor, $(t_i^j, T_i^j), i = 1, 4; j = 1, N$ were generated. The data

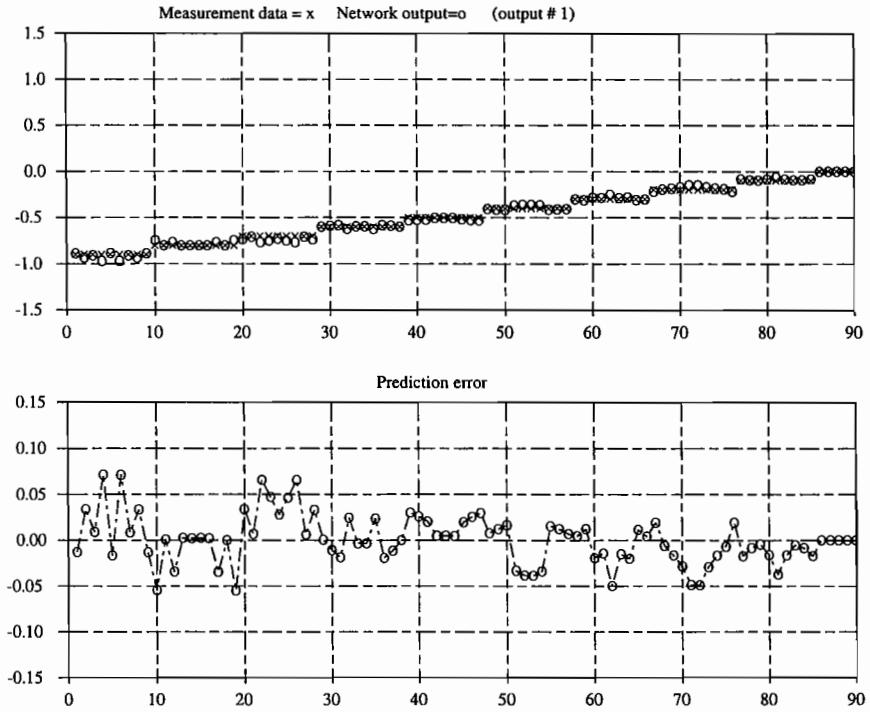

Figure 12: Fitting of *output 1* (x coordinate) to computational data

was separated into the training (181 points) and test (180 points) sets. As with the initial test data the computational data was suitably scaled and using the same basic MLP network 500 iterations were used to train the network. The final error was around $1\text{x}10^{-3}$.

Figures 12 to Figure 15 show the final fit of the network to the test data.

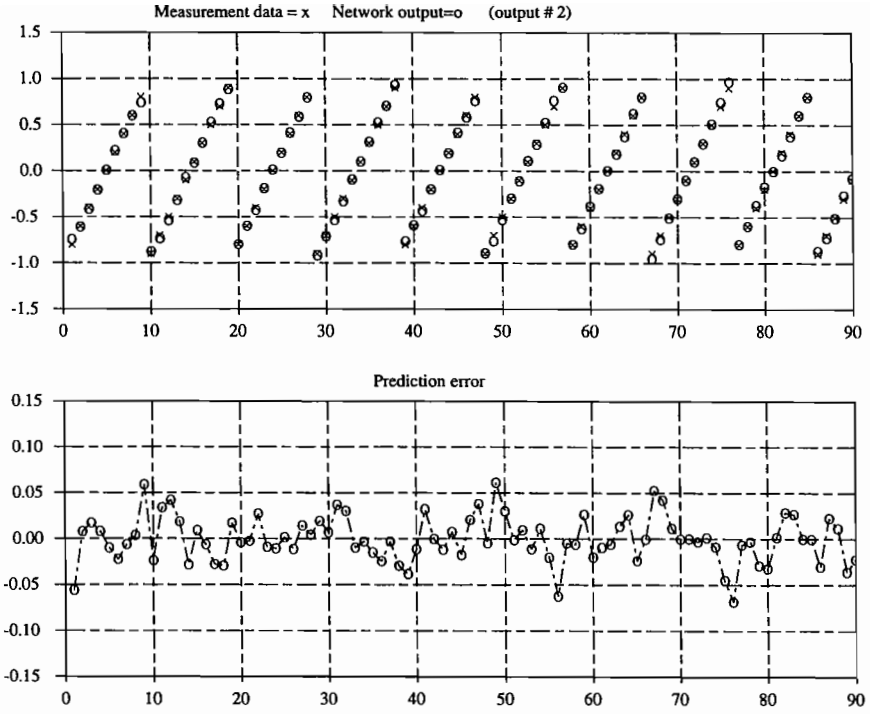

Figure 13: Fitting of *output 2* (y coordinate) to computational data

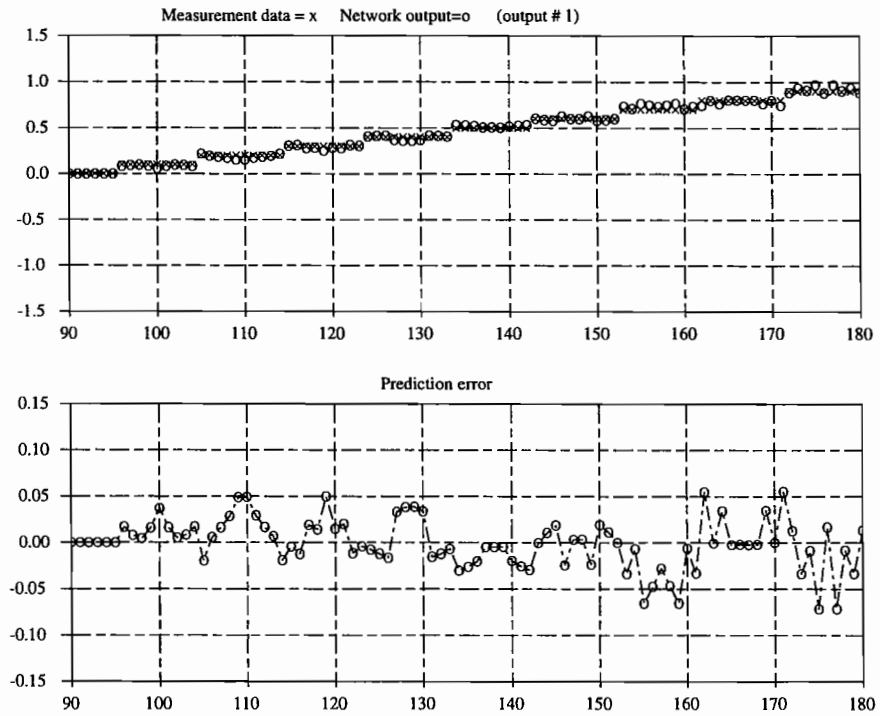

Figure 14: Fitting of *output 1* (x coordinate) to computational data

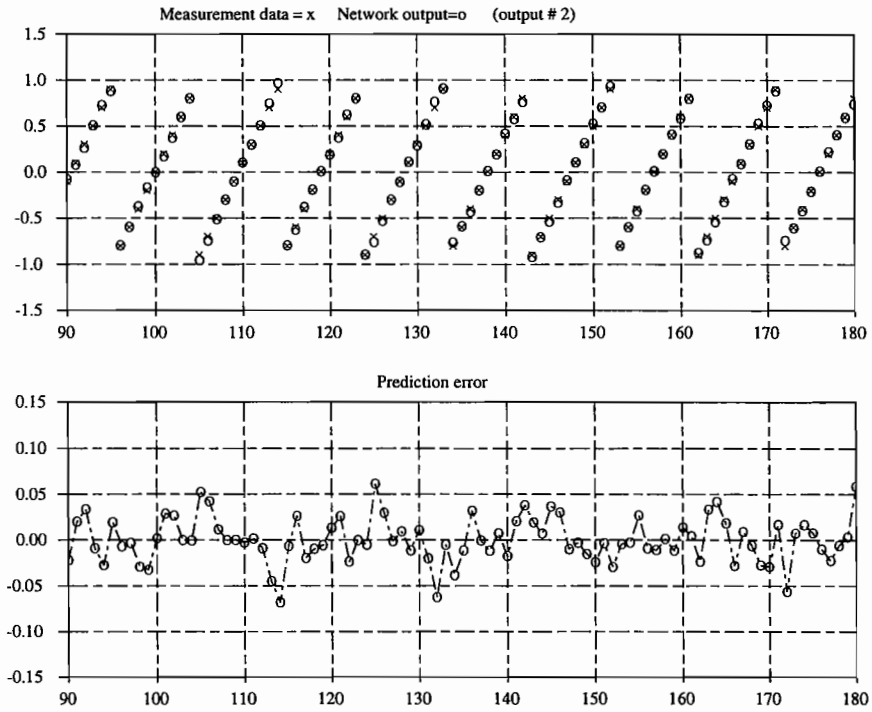

Figure 15: Fitting of *output 2* (y coordinate) to computational data

8 Conclusion

In this report we have described some initial attempts at modelling a position-resolving cryogenic detector. It is clear from the computational results that such a device is viable only if the electronics monitoring the peak arrival times can discriminate at the level of a micro-second. The results also show that given an event that produces a temperature rise of around $0.1^{o}K$ the properties of gold will allow the detection of such an event. The computational results also show a the relaxation time of the device to be in the order of $10\mu s$ which should be sufficient for X-ray applications.

Although this work is based on the assumption that the heat transfer process can be adequately represented by a linear diffusion, comparison will be needed against experimental data to validate this approach.

We have shown that the computational results compare well with the simple analytic solution on an infinite plate and that the position of an event can be recovered via either an analytical expression (using the analytic solution) or an appropriately trained neural network.

Although this has been a good start it is thought that future work should include a more detailed physical model of the detector. The model needs to have a better representation of the heat flow into the constant temperature bath on which the detector sits and also some representation of the heat losses from the device surfaces.

The differences between the analytic and computational models, although quite small, are still significant for the off-centre events. This would indicate a more accurate simluation might be required. Using a fine mesh and a smaller time-step would provide this but there would be a significant increase in the time taken to perform the computations.

References

[1] M Norgaard (1997): "Netural Network Based System Identification ToolBox", Department of Automation, Technical University of Denmark, Technical Report 97-E-851

[2] Scilab Group (1998): "Introduction to SCILAB", Scilab Group, INRIA Meta2 Project, Unite de recherche de Rocquencout, BP 105 - 78153 e Chesnay Cedex

[3] MathWorks (1995): "The Student Edition of MATLAB - Version 4", Prentic Hall, Englewood Cliffs, NJ 07632

[4] H Demuth and M Beale (1987): "Neural Network Toolbox", The MathWorks Inc.

[5] MT Hagan, HB Demuth and M Beale (1995): "Neural Network Design", PWS Publishing Company, Boston, MA

[6] R Fletcher (1987): "Practical Methods of Optimisation", Wiley

[7] C Greenough *etal* (1991): "The Finite Element Library (Vesion 3) - Level 0 Manual", Numerical Algorithm Group Ltd, Oxford

[8] C Greenough *etal* (1991): "The Finite Element Library (Vesion 3) - Level 1 Manual", Numerical Algorithm Group Ltd, Oxford

[9] C Greenough and K Robinson (1982): "Examples in the Use of the Finite Element Library: Inviscid Potential Flow (Diffusion Equation)", Rutherford Laboratory Report, RL-82-068

[10] F Mandl (1978): "Statistical Physics",John Wiley & Son Ltd, UK

[11] RE Bolz and GL Tuve (Eds) (1973) "CRC Handbook of Tables for Applied Engineering Science", CRC Press

[12] JK Shackelford and W Alexander (Eds) (1994) "CRC Materials Science and ENgineering Handbook", CRC Press