Road Traffic Noise Prediction: An Artificial Intelligence Approach

Barry L. Doolan and Steven J.B. Carter

School of Engineering, University of Tasmania, Private Bag 65, Hobart, Tasmania 7001

ABSTRACT

Present road traffic noise prediction models, such as TNOISE, use semi-empirical adjustments to account for factors that influence the noise level impacting a receiver. Most adjustments are based on actual sound level measurements, for example of noise attenuation by different ground types, and hence present models perform satisfactorily for the simple situations in which the measurements were made. However, accurate noise prediction in more complex situations is beyond the ability of such models, because determination of a comprehensive set of adjustments is defeated by the numerous possible variations in terrain characteristics, building geometries, and so forth. This paper describes how this problem can be overcome using a neural network approach to road traffic noise prediction. We demonstrate how a simple neural network easily mimics one of the present road traffic noise models, and how neural networks trained on grid-based data can learn to predict road traffic noise in complex situations.

INTRODUCTION

There is clear motivation for seeking to develop better road traffic noise prediction models, and an historical comparison with air emission dispersion models helps to make this point.

In the 1980s, Gaussian spreading disk models emerged to replace hand calculations of air emission dispersion. These included Australia's *Ausplume*, (Lorimer 1986); the U.S. Industrial Source Complex models; and Canada's Regulation 308 models. Their descendents are work horses of the industry and, because they can be driven by meteorological data spanning a year or more, ambient air quality standards are now written in anticipation of such models being used.

At the same time, simple road traffic noise prediction models were developed, such as *Tnoise*, a West Australian model based on the 1988 U.K. Calculation of Road Traffic Noise (the Welsh method); and Canada's *Stamson*. These models are based on reference sound power level values to which semi-empirical adjustments were logarithmically added to account for terrain type and other factors that influence the noise level impacting a receiver. Some countries, such as Canada, require use of these models as part of mandatory noise impact studies to support development applications in areas subject to road traffic noise nuisance.

In the 1990s, more advanced air emission models were developed for application to complex situations. These models still solve the advection-diffusion-decay equation, but are driven by time-varying 3-D wind fields, with dispersion modelled by puff or particle tracking strategies. Diagnostic models such as *Ausmet* produce wind fields from field data, while prognostic models such as CSIRO's TAPM produce the wind fields from algorithms similar to those used in numerical weather prediction. These more advanced models are best used by specialist practitioners, but the models are recognised and valued by environmental authorities.

No comparable advances have been made in road traffic noise prediction modelling, although there have been attempts to develop more sophisticated models. Some of the better known examples are the Environmental Noise Model developed by RTS Technology, and the European *SoundPLAN* model, which has a road traffic noise module.

Attempts to extend the strategy used by the 1980s models have failed, because determination of empirical adjustments to a reference sound power level for complex situations is defeated by the number of possible variations in terrain characteristics, building geometries, and so forth. For example, no general adjustment table is possible for the variation of noise levels around a cluster of buildings.

Similarly, no-one has yet developed a generally applicable model for complex situations, that can predict road traffic noise impact using fundamental sound propagation physics.

At present, an acoustician must rely heavily on experience in situations such as predicting the sound level impacting on an upper level window of a proposed development, due to future traffic levels on intersecting roads that are about to be upgraded. The need to solve this problem is obvious. Many Australian State of the Environment Reports have noted the growing problem of road traffic noise, and SoE (2001) estimated that 70% of environmental noise is due to road traffic.

This paper shows that neural network modelling may offer a way forward. Learning about artificial intelligence methods is a standard part of many engineering degree courses, and there are several good texts on neural networks and other A.I. tools, such as Negnevitsky (2003). However, a literature review failed to find any reports of work to apply neural network methods to the problem of road traffic noise prediction.

The nature of neural networks is described in the next section, and a neural network model able to mimic classical models such as *Tnoise* is then presented. Finally, a model extension able to handle complex situations is demonstrated.

THE NATURE OF NEURAL NETWORKS

(Artificial) neural networks are well understood pattern recognition tools, and have been successfully applied to many engineering problems, facilitated by technical computing software packages such as *Matlab*.

A neural network approach is indicated when there is a relationship between a set of variables, but the nature of the relationship is poorly understood, and perhaps non-linear.

A neural network mimics the operation of a human brain. As shown in Figure 1, a network consists of layers of data processing units called neurons.

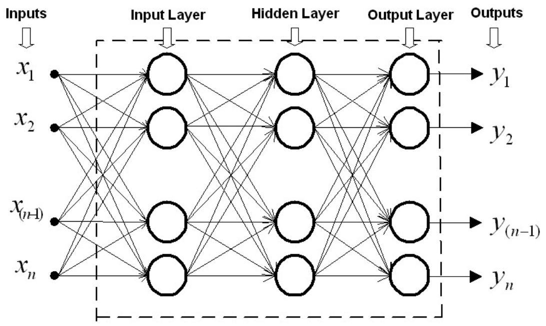

Figure 1 An Artificial Neural Network.

Figure 2 shows the operation of a single neuron. The neuron receives weighted output signals ($w_i\,x_i$) from all the inputs, or from all the neurons in the previous layer. It sums the weighted signals, adds a bias (b) to the result, and produces its own output signal (y) according to a "transfer function", f, which is typically either some form of sigmoidal function, or a simple linear function.

$$I = \sum \omega_i x_i + b \qquad y = f(I)$$

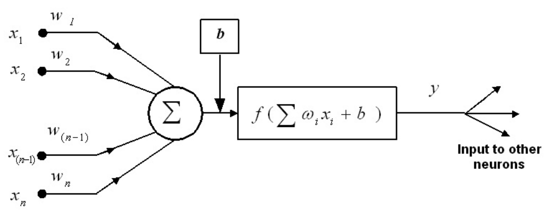

Figure 2 Operation of a neuron.

A neural network is trained on an appropriate set of input data, using a training strategy to adjust the weights and biases of the neurons until the required outputs are achieved.

A properly trained neural network can generalise to predict the output associated with input data that were not part of the training set. This is similar to regression analysis, except that the far greater degree of connectivity enables much more complex patterns to be considered.

A feed-forward backpropagation neural network was selected as appropriate for this modelling exercise. "Feed-forward" refers to the one-way propagation of information from the first (input) layer of neurons to the last (output) layer. "Backpropagation" refers to the network training method.

SIMPLE NOISE LEVEL PREDICTIONS

This section presents a neural network trained to predict the equivalent sound level (L_{eq}) due to road traffic, at distances of 20 to 200 m from the road, and for posted speed limits of 50 to 100 km/h. For this exercise, all other parameters, such as traffic composition, are held constant, and the receiver is assumed to have an uninterrupted view of the road.

Figure 3 shows the equivalent sound level surface mapped out by this range of distance and speed parameters. Sound levels have a logarithmic dependence on both vehicle speed and distance from the road, and Figure 4 shows this by replotting the sound level surface on a log-log scale.

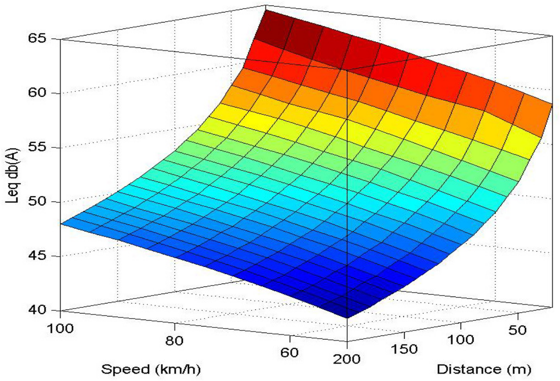

Figure 3 Noise level variation with speed and distance.

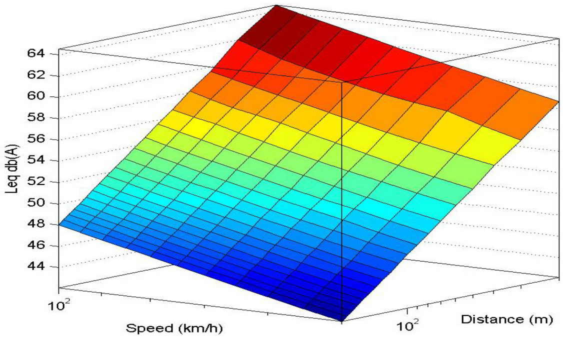

Figure 4 Log-log plot of above noise level variation.

It is tempting to linearise the problem by using logarithmic input variables, since a pattern represented by a plane needs only three points to be defined. However, to demonstrate a neural network's ability to recognise non-linear patterns in data, conventional values are retained.

Neural network architecture

A neural network's architecture refers to the number of neuron layers, the number of neurons in each layer, and the types of transfer functions used by the neurons.

A simple 2-layer neural network is adequate for this task. A multi-layer backpropagation network may handle complex or noisy data better than a simpler network, but this problem is straightforward, and hidden layers of neurons are not needed.

Another rule of thumb is not to specify too many neurons, since networks with a relatively large number of neurons compared to the complexity of the pattern to be discerned are prone to overfitting problems, discussed below. In this case, an input layer with 30 neurons was found to be sufficient.

A tangent sigmoidal transfer function was specified for the input layer neurons. The input variables were normalised by roughly half their maximum values (e.g. speeds were divided by a normalisation factor of 50 km/h) to match the response range of this kind of transfer function.

The number of neurons in the output layer must be equal to the number of output variables, in this case just one, the equivalent sound level. A linear transfer function was prescribed for the single output neuron.

Training and validation data

The Canadian model *Stamson* was used to produce the neural network model training data. *Stamson* predicts L_{eq} values for a specified time period, and Figure 5 shows its command window.

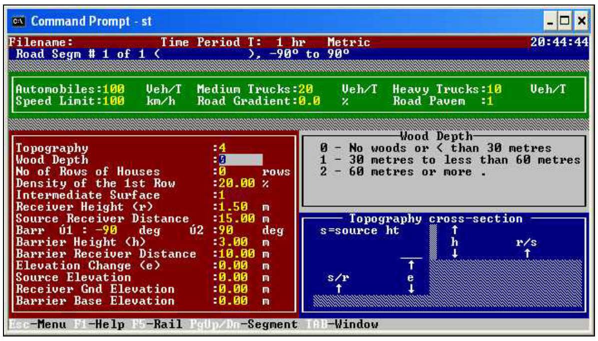

Figure 5 *Stamson.* A typical road noise model.

Since the problem is in non-linear form, a key question is how much data are needed to train a neural network. It is not sufficient that a network be able to correctly produce the outputs corresponding to the training data; it must be able to generalise to new data.

In this case, 39 points were provided, sufficient to define the surface in Figure 3. Table 1 shows data for three speeds, with additional data relating to speeds of 50, 70 and 90 km/h, at receiver set backs from the road of 30, 50, 70, 90, 120, 150, and 200 m.

	60 km/h	80 km/h	100 km/h
20 m	60.16	62.60	64.56
40 m	55.18	57.62	59.58
60 m	52.26	54.70	56.67
100 m	48.59	51.03	53.00
140 m	46.17	48.61	50.58
200 m	43.61	46.05	48.01

Table 1. Training sound level data. L_{eq} (1h) in dB(A)

The neural network training algorithm, *traingdx*, was used, and operated in "batch mode", whereby the model's noise level predictions for all 39 combinations of posted speed limit and receiver distance from the road are compared to the actual (i.e. *Stamson*) values. Based on this comparison, the algorithm adjusts the neuron signal weights and biases across the network, working backward from the output layer ("backpropagation") and using an error gradient descent technique with momentum and an adaptive learning rate (Demuth & Beale 2001). The model then makes a new set of predictions, and the process is repeated.

Each such iteration, known as a training epoch, results in a better set of predictions. Ideally, a neural network training session results in a model that has the ability to generalise its predictions to sets of input values that were not part of the training data set group. To achieve this, it is necessary to stop model training before the phenomenon of overfitting becomes a problem. The neural network overfitting problem is analogous to a high order polynomial fit to a set of training data points producing the required values at those points, but varying wildly and incorrectly between the training points.

A good way to avoid overfitting is to examine the neural network model predictions for a second set of input data. Initially, as the training proceeds, the sum-squared error between predictions and actual values decreases for both the training data set and the second data set. Training is stopped when the sum-squared error for the second data set starts to rise, indicating that overfitting is starting to be a problem.

Neural network performance

Figure 6 shows the equivalent sound level surface mapped out by the neural network over the speed and distance ranges of Table 1. The wrinkles in the prediction surface are minor.

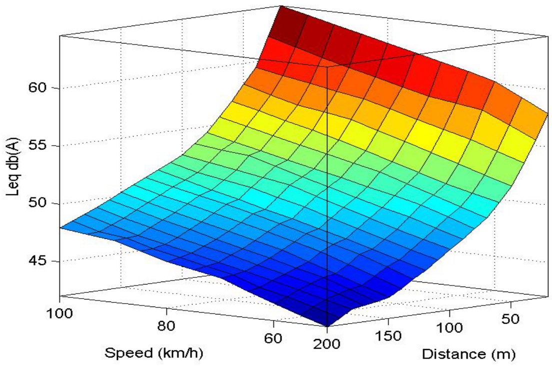

Figure 6 Neural net L_{eq} predictions (compare to Fig 3)

Figure 7 shows a typical slice through the surface in Figure 6, for a speed of 70 km/h. The solid line shows the Stamson sound level curve, and the dashed line shows the neural network predictions.

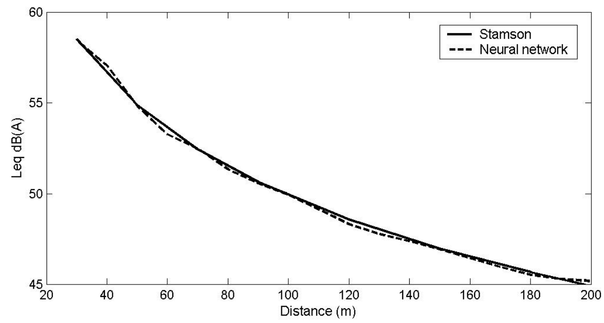

Figure 7 Stamson vs neural net predictions 70 km/h.

Figures 6 and 7 show that a neural network is easily able to mimic a conventional road noise prediction model. The exercise only examined predictions within the range of training data values, but the neural network can also extrapolate patterns in data. We now examine the architecture of neural networks able to discern patterns in grid-based data.

GRID BASED NOISE PREDICTIONS

Neural networks have a proven ability to find patterns in grid-based data, such as images, and this is the key to using a neural network approach to predicting road traffic noise levels in complex situations.

Artificial intelligence tools are based on biomimicry principles, and biomimicry can also guide strategies to apply these tools. In this case, it is common wisdom that people often solve a complex problem by breaking it into components to the extent possible. This suggests that a good modelling strategy is to develop one neural network to predict the dependence of L_{eq} values on speed across a grid, and then modify these baseline values by adjustments predicted by other neural networks. This baseline-plus-adjustments approach is used by conventional models such as *Tnoise,* and it appears to work just as well for grid-based neural networks.

ROAD

	0m	10m	20m	30m	40m	50m	60m	70m	80m	90m	100m	110m	120m	130m	140m	150m
20m	62.6	62.6	62.6	62.6	62.6	62.6	62.6	62.6	62.6	62.6	62.6	62.6	62.6	62.6	62.6	62.6
30m	59.7	59.7	59.7	59.7	59.7	59.7	59.7	59.7	59.7	59.7	59.7	59.7	59.7	59.7	59.7	59.7
40m	57.6	57.6	57.6	57.6	57.6	57.6	57.6	57.6	57.6	57.6	57.6	57.6	57.6	57.6	57.6	57.6
50m	56.0	56.0	56.0	56.0	56.0	56.0	56.0	56.0	56.0	56.0	56.0	56.0	56.0	56.0	56.0	56.0
60m	54.7	54.7	54.7	54.7	54.7	54.7	54.7	54.7	54.7	54.7	54.7	54.7	54.7	54.7	54.7	54.7
70m	53.6	53.6	53.6	53.6	53.6	53.6	53.6	53.6	53.6	53.6	53.6	53.6	53.6	53.6	53.6	53.6
80m	52.6	52.6	52.6	52.6	52.6	52.6	52.6	52.6	52.6	52.6	52.6	52.6	52.6	52.6	52.6	52.6
90m	51.8	51.8	51.8	51.8	51.8	51.8	51.8	51.8	51.8	51.8	51.8	51.8	51.8	51.8	51.8	51.8
100m	51.0	51.0	51.0	51.0	51.0	51.0	51.0	51.0	51.0	51.0	51.0	51.0	51.0	51.0	51.0	51.0

Figure 8. Grid for two-dimensional road traffic noise predictions. L_{eq} values are for a speed of 80 km/h.

Baseline grid

The baseline neural network that predicts L_{eq} values at different speeds and distances from the road has only speed as its input, while the output is an entire grid of L_{eq} values.

Figure 8 shows the simple grid used to establish this modelling approach. The road is assumed to be straight and long, and the grid extends 150 m along the road, and from 20 m to 100 m from the road: road traffic noise predictions are not generally made closer than 15-20 m from a road. The L_{eq} values in dB(A) are given over a 10 m x 10 m square grid, and Figure 8 shows the values for a speed of 80 km/h (with only one decimal place, for clarity).

A 2-layer feed-forward neural network performed quite well for this problem. The best performance was achieved by networks containing 20-25 neurons in the input layer, with tangent-sigmoidal transfer functions. The network output layer contains 144 neurons, with linear transfer functions, corresponding to the 9 row x 16 column prediction grid.

A neural network trained on only four speeds (40, 60, 80, and 100 km/h) was able to generalise satisfactorily. Figure 9 compares the model L_{eq} predictions for a speed of 70 km/h, to the actual L_{eq} values. Since the grid is flat, the L_{eq} contours lie parallel to the road.

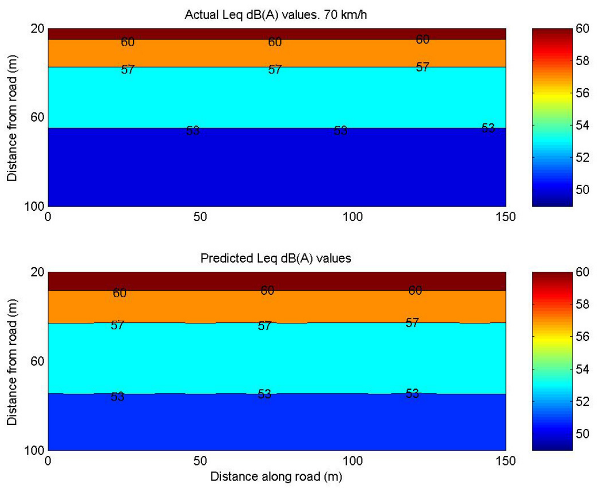

Figure 9 Actual vs predicted L_{eq} grid for 70 km/h.

The predictions shown in Figure 9 are accurate to about 0.5 dB(A), which is a reasonably good performance given that the neural network was trained on only four input speeds.

Adjustment for barrier

A neural network modelling approach has the potential to handle more complex situations than a classical model, such as *Tnoise*, because a neural network can be trained to discern patterns in two-dimensional data.

To demonstrate this, consider the adjustment to the grid of baseline L_{eq} values due to a 70 m long noise barrier situated parallel to the road. Figure 10 shows a barrier 40 m from the road, with a prescribed L_{eq} adjustment pattern.

	0m	10m	20m	30m	40m	50m	60m	70m	80m
20m	0	0	0	0	0	0	0	0	0
30m	0	0	1.0	1.0	1.0	1.0	1.0	0	0
40m	0	2.0	2.5	2.5	2.5	2.5	2.5	2.0	0
50m	0	-7.0	-9.0	-11.0	-11.0	-11.0	-9.0	-7.0	0
60m	0	0	-7.0	-9.0	-11.0	-9.0	-7.0	0	0
70m	0	0	0	-7.0	-9.0	-7.0	0	0	0
80m	0	0	0	0	-7.0	0	0	0	0
90m	0	0	0	0	0	0	0	0	0

Figure 10 L_{eq} adjustments due to a 70 m noise barrier.

In Figure 10, the barrier is denoted by the heavy black line. The L_{eq} adjustments of 2.0 and 2.5 dB in front of the barrier are realistic, since a perfectly reflective barrier will produce a 3 dB increase (i.e. a doubling) in sound levels. The 1.0 dB adjustments are 10 m in front of the barrier.

Behind the barrier, a triangular pattern of L_{eq} adjustments is specified. For example, 10 m behind the barrier, the adjustments vary from –7 to –11 dB. The triangular pattern does not correspond well to the actual adjustments, but simplifies the evaluation of a neural network's performance in this demonstration exercise.

A neural network with the same architecture as that described in the previous section, but with 120-150 neurons in its input layer, was found to be adequate.

The neural network inputs for a given training data set consist of 9 x 16 = 144 values of either one or zero. Seven input values are set to one, denoting the location of the barrier, and the remaining 137 values are set to zero.

The neural network is trained by requiring it to produce the correct pattern of L_{eq} adjustments for a given barrier location. As usual, a key question is how much training data is needed – i.e. how many barrier locations and associated patterns of L_{eq} adjustments are needed – to properly train the neural network. The test is whether the network can generalise to predict the L_{eq} adjustment patterns for barrier locations that were not part of the training data.

There are ten possible locations for a 70 m long barrier on each of the nine rows in the grid, giving a total of 90 possible barrier locations. The set of barrier locations used to train the neural network must contain at least three barriers on each row (i.e. 3 x 9 = 27 training data sets), or some inputs will always be zero. A neural network quickly learns to ignore inputs that are constant, or which are not related to the required output.

However, the training data really needs to contain four barriers on each row, or else many inputs are set to one only once. This is not sufficient for the neural network to accurately recognise how the pattern of ones and zeros in the input values are related to the required pattern of L_{eq} adjustments.

Experimentation confirmed that a neural network could generalise reasonably well if it was trained on a minimum of 35-40 barriers. Figure 11 shows the predictions of the neural network for three barrier locations that were not part of the training data. The contour values are –10, -9, -8, -6, -5, +1, and +2 dB.

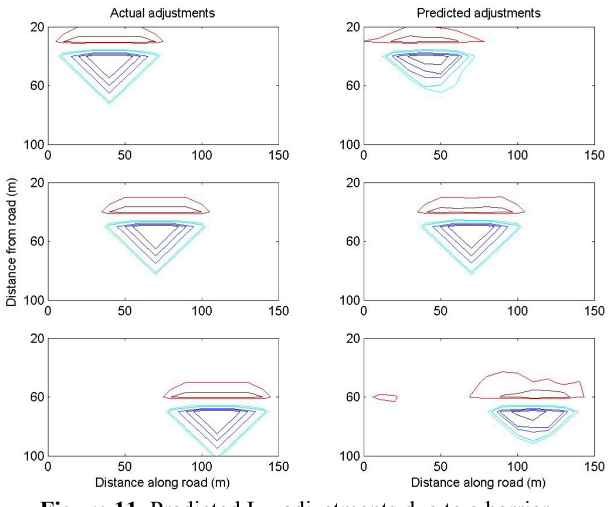

Figure 11 Predicted L_{eq} adjustments due to a barrier.

CONCLUSIONS

This paper has presented an (artificial) neural network approach to predicting road traffic noise, and a simple 2-layer neural network is shown to have the ability to mimic present models, such as *Tnoise* and *Stamson*.

Neural networks are not a universal remedy for all modelling situations. However, it is appropriate to consider applying them to the problem of predicting road traffic noise in complex situations. The present models are underpinned by field measurements to establish how sound levels vary with speed, distance, ground reflectivity, and so forth. These are essentially one-dimensional patterns.

Neural networks can discern patterns in two dimensional (i.e. grid based) field measurements, and this means that they have the potential to handle situations beyond the capability of present road traffic noise prediction models, situations such as the variation of noise levels around buildings in uneven terrain.

We have examined the case for using neural networks to predict road traffic noise in complex situations. The two keys to solving this problem are, first, to move to grid-based input data and predictions; and, second, to continue the strategy of making a set of baseline predictions that are then refined by a set of separate adjustments.

Training a neural network to make baseline L_{eq} predictions as a function of traffic speed, over a (fairly coarse) grid is shown to be a straightforward task. We then show how a neural network can be trained to produce L_{eq} adjustments due to the presence of a noise barrier.

Similar neural networks can be trained to determine adjustments to the baseline L_{eq} for a variety of other factors that influence road traffic noise. Examples include terrain variation, barriers of different lengths and heights, buildings, multiple roads with non-trivial geometries, and so forth. This is the principal advantage of the neural network approach. A neural network's powerful ability to find patterns in data provides a way to extend the empirical baseline-plus-adjustment approach of models such as *Tnoise*, to complex situations.

A key issue in an attempt to develop an effective complex situation modelling capability is how much field data are needed to adequately train the various neural networks. In the barrier adjustment example, the necessary 35 or so training barriers is a large percent of the 90 possible barriers. However, for this adjustment, all the training data can quickly be generated from only one set of field measurements.

For other adjustments, such as the effect of terrain variations, multiple sets of field measurements are needed to train a neural network. However, the required effort is probably comparable to that necessary to underpin the development of models such as *Tnoise*.

REFERENCES

Demuth, H. and Beale, M., 2001, *Neural network toolbox for use with MATLAB*, MathWork Inc, Natick, MA, USA.

Negnevitski, M., 2003, *Artificial Intelligence,* Addison-Wesley.

SoE 2001. *Australia: State of Environment 2001*. Report to the Commonwealth Government by the State of the Environment Advisory Council, CSIRO Publishing.

Lorimer, G. 1986. *The Ausplume Gaussian Plume Dispersion Model.* Victoria Environment Protection Authority Publication 264.