Macrocell Path-Loss Prediction Using Artificial Neural Networks

Erik Östlin, Hans-Jürgen Zepernick, and Hajime Suzuki, *Member, IEEE*

IEEE Transactions on Vehicular Technology (Volume: 59, Issue: 6, July 2010)

Hong Cheng
chenghong@kw.ac.kr

Advisor: Prof. Hyukjoon Lee

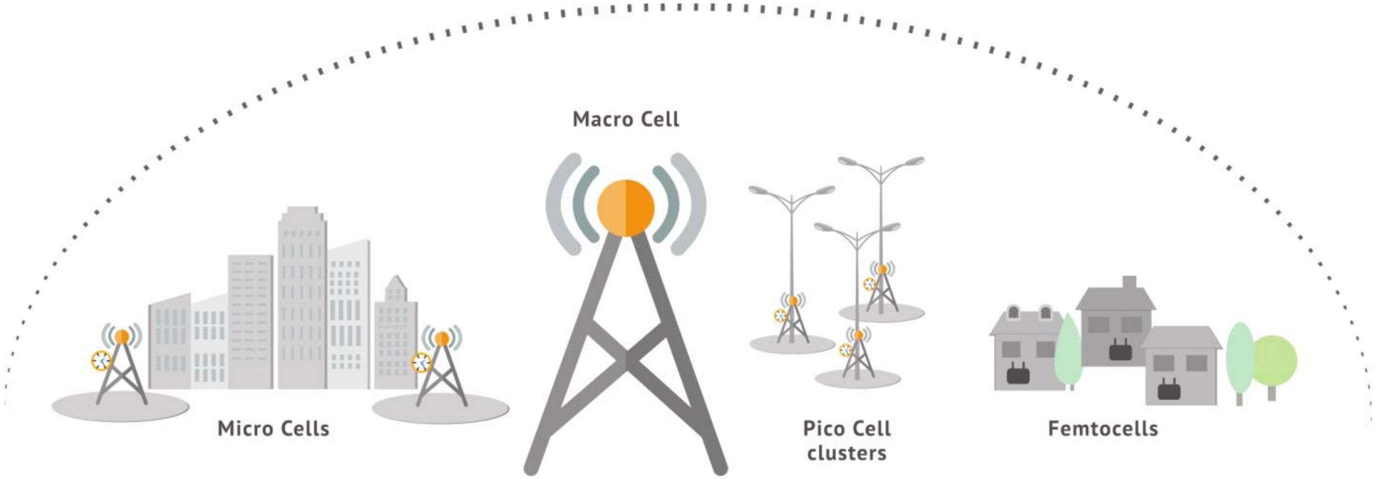
Computer Intelligence Network Laboratory, Dept. of CE, KwangWoon University

CINe Lab

1. INTRODUCTION

- Traditional path loss model
- Multilayered feed-forward networks
- Previous Work
- The work in the paper

Macrocell



- A **macrocell** is a **cell** in a <u>mobile phone network</u> that provides radio coverage served by a <u>high power cellular base station (tower)</u>. Generally, macrocells provide coverage larger than microcell. The antennas for macrocells are mounted on ground-based masts, rooftops and other existing structures, at a height that provides a clear view over the surrounding buildings and terrain. Macrocell base stations have power outputs of typically tens of watts.

Traditional Path Loss Model

Empirical models | such as Okumura–Hata (OH) models [2]

- **computationally efficient** but may **not** be very **accurate**
- since they do not explicitly account for specific propagation phenomena.

Deterministic models | such as geometrical theory of diffraction [3], integral equation [4], and parabolic equation [5]

- depend on the topographic database resolution and accuracy
- very **accurate** but **lack** in computational **efficiency**.

Commonly drawback

- contain numerous neurons
- in each layer is the required training time [10].
- overly complex ANN may lead to data overfitting and generalization problems [11].

Often, in the literature

- unnecessarily complex
- the network size is not even mentioned at all

Previous Work

[13]"Evaluation of the propagation model Recommendation ITU-R P.1546 for mobile services in rural Australia," *IEEE Trans. Veh. Technol.*, vol. 57, no. 1, pp. 38–51, Jan. 2008.

Propagation measurements

- utilizing the **IS-95 [12] pilot signal** of a commercial CDMA mobile telephone network in **rural** Western Australia
- to evaluate and propose modifications to the **Recommendation ITU-R P.1546** [13].

General finding

- a noncomplex ANN is enough
- using an efficient training method, such as the **L**evenberg–**M**arquardt algorithm
- the required number of iterations may be reduced from approximately **150 000 to 10**

reducing the training time from many **hours to** a few **seconds**.

A common phenomenon caused by overtraining

- feed-forward networks with several hidden layers and numerous neurons may lead to **inferior** generalization properties

- **Therefore**, the impact on prediction accuracy and generalization properties due to additional hidden layers and neurons was also investigated.

Works In The Paper

Different-sized ANNs are trained

- By using different backpropagation training algorithms, such as gradient descent and Levenberg–Marquardt.

Different training data-selection strategies

- To investigate what ANN complexity is needed to achieve high prediction accuracy while maintaining good generalization properties

The goal is to obtain an ANN

- not overly complex but still generalizes well
- accurate enough

Compared with [13] and [14]

- this paper presents prediction results for an **extended** number of measurement **routes**, which greatly aided the evaluation of different ANNs' generalization properties.
- **ES**(Early stopping) and **BR**(Bayesian regularization) were incorporated
- the created ANN models also include **land usage and vegetation** input parameters
- all prediction results in this paper can directly be compared with our previously work[13]

2. ARTIFICIAL NEURAL NETWORK

- The motivation
- Neuron model
- Feed-forward network
- ANN training
- Generalization

The Motivation

find the most suitable feed-forward structure

- The importance of the **feed-forward network size** was investigated regarding prediction accuracy and generalization properties.
- **The training algorithm selection** was investigated with regard to training time, i.e., maintaining the prediction accuracy and generalization properties while minimizing the training time.
- The neuron model, feed-forward network, training, and generalization methods are introduced and described.

Neuron Model

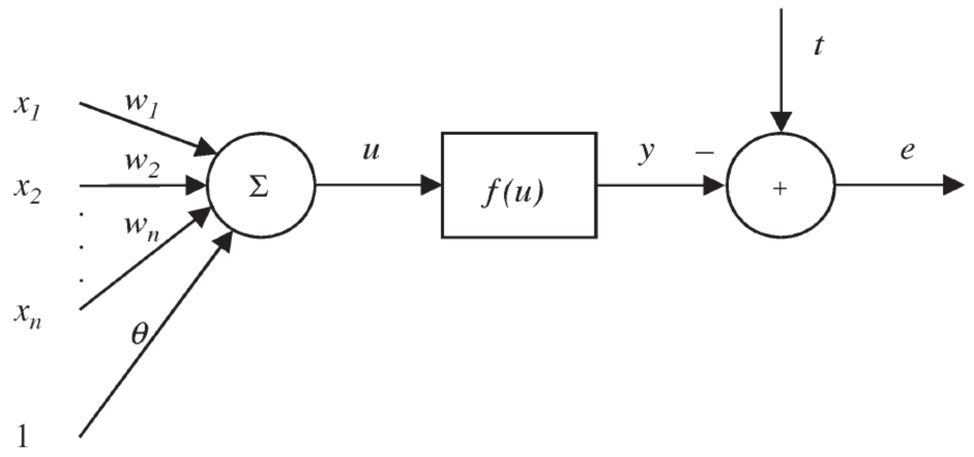


Fig. 1. Neuron model.

Input signal $$\mathbf{x} = [\, x_1 \quad x_2 \quad \cdots \quad x_n \quad 1 \,]^T \tag{1}$$

Output value $$u = \mathbf{w}^T \mathbf{x} \tag{2}$$

Neuron weights $$\mathbf{w} = [\, w_1 \quad w_2 \quad \cdots \quad w_n \quad \theta \,]^T. \tag{3}$$

- θ is an **additional scalar bias** parameter <u>to provide the possibility </u>to shift the activation function $f(\cdot)$

Neuron Model

- For this paper, **the activation function** has been chosen to be the commonly used **hyperbolic tangent sigmoid transfer function** [15], which is defined as

$$f(u) = \frac{1 - \exp^u}{1 + \exp^u} \qquad (4)$$

- It compresses the output into the range ***-1*** to ***1***.

- the neuron **output error** ***e*** is calculated by subtracting the sigmoid output ***f(u)*** from the **target value** ***t*** as

$$e = t - y. \qquad (5)$$

Feed-forward Network

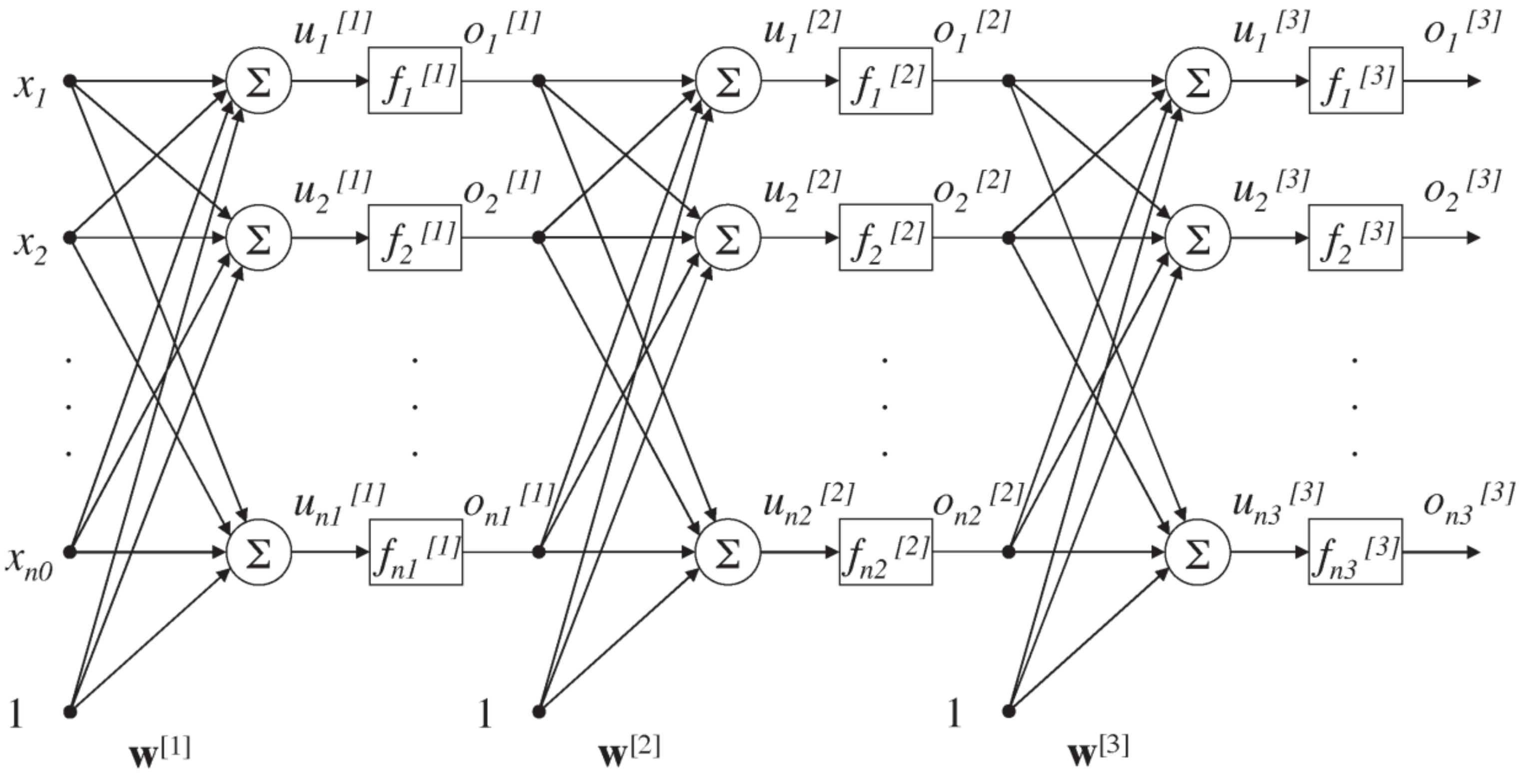


Fig. 2. Three-layer feed-forward ANN.

$$ANN_{n_0-n_1-\cdots-n_k}$$

$$n_1, n_2, n_3$$

$$w^{[1]}, w^{[2]}, w^{[3]}$$

$f_i^{[j]}$ The ith activation function in the jth hidden layer

$o_i^{[j]}$ The output from the ith neuron in the jth layer

ANN Training

The training set should be representative of the problem the ANN is designed to solve.

- The ***Q*** input–output training pairs are randomly chosen from the measurement data and are defined as

$$\{\mathbf{p}_1, t_1\}, \{\mathbf{p}_2, t_2\}, \ldots, \{\mathbf{p}_Q, t_Q\} \qquad (6)$$

  - $\boldsymbol{p}_Q$ is an input vector $\boldsymbol{t}_Q$ is the corresponding output

  - The ANN weights ***w*** are found by minimizing the **mean square errors**

$$\mathrm{MSE} = \frac{1}{Q}\sum_{q=1}^{Q} e(q)^2. \qquad (7)$$

Backpropagation and Gradient Descent

- The standard implementation of backpropagation learning updates (i.e., ANN weights and biases) is to base the updates on the direction of the steepest negative gradient descent.

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \alpha_k \mathbf{g}_k \quad (8)$$

w_k the current weights and biases at the k_{th} iteration.

α_k the learning rate (constant),

g_k the current gradient vector.

Levenberg–Marquardt Algorithm

- This algorithm is designed for training moderate-sized feed-forward neural networks, with less than 100 weights, and least squares problems that are approximately linear [11].
- In **Newton algorithm**, the weight update can be expressed as

$$\mathbf{w}_{k+1} = \mathbf{w}_k - \mathbf{A}_k^{-1}\mathbf{g}_k \tag{9}$$

A_k is the **Hessian matrix** [20] (second-order derivatives) of the performance index consisting of current weights and biases.

drawback complex and computationally expensive to calculate

Approach quasi-Newton method

- Based on the local gradient, an approximative Hessian matrix at each algorithm iteration

Levenberg–Marquardt Algorithm

Hessian matrix

In mathematics, the **Hessian matrix** or **Hessian** is a square matrix of second-order partial derivatives of a scalar-valued function, or scalar field. It describes the local curvature of a function of many variables. The Hessian matrix was developed in the 19th century by the German mathematician Ludwig Otto Hesse and later named after him. Hesse originally used the term "functional determinants".

Specifically, suppose $f : \mathbb{R}^n \to \mathbb{R}$ is a function taking as input a vector $\mathbf{x} \in \mathbb{R}^n$ and outputting a scalar $f(\mathbf{x}) \in \mathbb{R}$; if all second partial derivatives of f exist and are continuous over the domain of the function, then the Hessian matrix $\mathbf{H}$ of f is a square $n \times n$ matrix, usually defined and arranged as follows:

$$\mathbf{H} = \begin{bmatrix} \dfrac{\partial^2 f}{\partial x_1^2} & \dfrac{\partial^2 f}{\partial x_1\,\partial x_2} & \cdots & \dfrac{\partial^2 f}{\partial x_1\,\partial x_n} \\[2.2ex] \dfrac{\partial^2 f}{\partial x_2\,\partial x_1} & \dfrac{\partial^2 f}{\partial x_2^2} & \cdots & \dfrac{\partial^2 f}{\partial x_2\,\partial x_n} \\[2.2ex] \vdots & \vdots & \ddots & \vdots \\[2.2ex] \dfrac{\partial^2 f}{\partial x_n\,\partial x_1} & \dfrac{\partial^2 f}{\partial x_n\,\partial x_2} & \cdots & \dfrac{\partial^2 f}{\partial x_n^2} \end{bmatrix}.$$

Levenberg–Marquardt Algorithm

- Approaches the second-order behavior without the need to calculate or approximate the Hessian matrix
- The Hessian matrix can be approximated using the **Jacobian matrix** J as

$$\mathbf{H} = \mathbf{J}^T \mathbf{J}. \qquad (10)$$

- The gradient g is then expressed as

$$\mathbf{g} = \mathbf{J}^T \mathbf{e} \qquad (11)$$

- e is a vector that contains the network errors.

- The Jacobian matrix contains only **first-order derivatives**

ANN Training

Levenberg–Marquardt Algorithm

Newton Method

- In numerical analysis, Newton's method is a method for finding successively better approximations to the roots (or zeroes) of a real-valued function.

The Newton–Raphson method in one variable is implemented as follows:

The method starts with a function f defined over the real numbers x, the function's derivative f', and an initial guess x_0 for a root of the function f. If the function satisfies the assumptions made in the derivation of the formula and the initial guess is close, then a better approximation x_1 is

$$x_1 = x_0 - \frac{f(x_0)}{f'(x_0)}.$$

Geometrically, $(x_1, 0)$ is the intersection of the x-axis and the tangent of the graph of f at $(x_0, f(x_0))$.

The process is repeated as

$$x_{n+1} = x_n - \frac{f(x_n)}{f'(x_n)}$$

$$\cos(x) - x^3 = 0$$

$$f(x) = \cos(x) - x^3$$

$$f'(x) = -\sin(x) - 3x^2$$

$$-1 \le \cos(x) \le 1$$

$$-1 \le x^3 \le 1,\ -1 \le x \le 1$$

$$x_0 = 0.5$$

$$\begin{aligned}
x_1 &= x_0 - \frac{f(x_0)}{f'(x_0)} &= 0.5 - \frac{\cos(0.5) - 0.5^3}{-\sin(0.5) - 3\times 0.5^2} &= 1.112141637097 \\
x_2 &= x_1 - \frac{f(x_1)}{f'(x_1)} &= \vdots &= \underline{0}.909672693736 \\
x_3 &= \vdots &= \vdots &= \underline{0.86}7263818209 \\
x_4 &= \vdots &= \vdots &= \underline{0.86547}7135298 \\
x_5 &= \vdots &= \vdots &= \underline{0.8654740331}11 \\
x_6 &= \vdots &= \vdots &= \underline{0.865474033102}
\end{aligned}$$

Levenberg–Marquardt Algorithm

Jacobian matrix J

In vector calculus, the **Jacobian matrix** (/dʒɨˈkoʊbiən/, /jɨˈkoʊbiən/) is the matrix of all first-order partial derivatives of a vector-valued function. When the matrix is a square matrix, both the matrix and its determinant are referred to as the **Jacobian** in literature.[1]

Suppose $\mathbf{f}:\mathbb{R}^n \rightarrow \mathbb{R}^m$ is a function which takes as input the vector $\mathbf{x} \in \mathbb{R}^n$ and produces as output the vector $\mathbf{f}(\mathbf{x}) \in \mathbb{R}^m$. Then the Jacobian matrix $\mathbf{J}$ of $\mathbf{f}$ is an $m \times n$ matrix, usually defined and arranged as follows:

$$\mathbf{J} = \frac{d\mathbf{f}}{d\mathbf{x}} = \begin{bmatrix} \dfrac{\partial \mathbf{f}}{\partial x_1} & \cdots & \dfrac{\partial \mathbf{f}}{\partial x_n} \end{bmatrix} = \begin{bmatrix} \dfrac{\partial f_1}{\partial x_1} & \cdots & \dfrac{\partial f_1}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \dfrac{\partial f_m}{\partial x_1} & \cdots & \dfrac{\partial f_m}{\partial x_n} \end{bmatrix}$$

or, component-wise:

$$\mathbf{J}_{ij} = \frac{\partial f_i}{\partial x_j}.$$

Levenberg–Marquardt Algorithm

- The Newton-like weight update

$$\mathbf{w}_{k+1} = \mathbf{w}_k - [\mathbf{J}^T\mathbf{J} + \mu\mathbf{I}]^{-1}\mathbf{J}^T\mathbf{e} \qquad (12)$$

- The parameter I denotes the identity matrix.
- μ is an adaptive parameter
  decreased if the performance function is decreasing
  increased if the performance function is increasing.
- For large μ values, the Levenberg–Marquardt algorithm becomes the gradient descent algorithm (small step size).
- When μ is zero, the weight update method approximates the Newton method, which is faster and more accurate near an error minima [11].

Improved generalization properties

- **ES(Early stopping)** was incorporated with the gradient descent algorithm to serve as the benchmark method.
- When trying to minimize the required training time while maintaining the ANN's generalization characteristics, both **ES** and **BR(Bayesian regularization)** were used with the LM algorithm [11],[23].
- To support the understanding of how to improve the generalization properties of ANN-based path-loss prediction models, the **ES** and **BR** methodologies are explained respectively.

ES Early stopping

- The measurement data are divided into three subsets : **training validation evaluation**
- The **training set** is used for computing the gradient and the ANN weights.
- The errors obtained from the **validation set** are monitored during the training.
- When the network is starting to **overfit** the data from the training set, the errors obtained from the validation set usually start to increase.
- When the validation error has increased for a specified number of iterations, the training stops and the weights and biases at the minimum of the validation error are returned.

Generalization

BR Bayesian regularization

- The measurement data are divided into two subsets : **training evaluation**
- All training data can directly be used in the training process, which is preferable if only small amounts of training data are available.

- ***drawback***

  require more iterations than **ES** to converge [11].

3. MEASUREMENT PROCEDURE

- Measurements Parameters
- Digital Elevation Map (DEM)
- Compare with previous work

Measurements Parameters

Mobile radio wave propagation

- carrier frequency ***881.52*** MHz
- CDMA pilot scanner

Measurement tool

- a Global Positioning System receiver
- an omni-directional receiving antenna
- which, during the measurements, were placed on the roof of a car at a height of approximately ***1.7*** m above ground.
- Without considering the influence of the roof of the car, the antenna’s half-power beamwidth was measured to be close to ***80°***.

Digital Elevation Map (DEM)

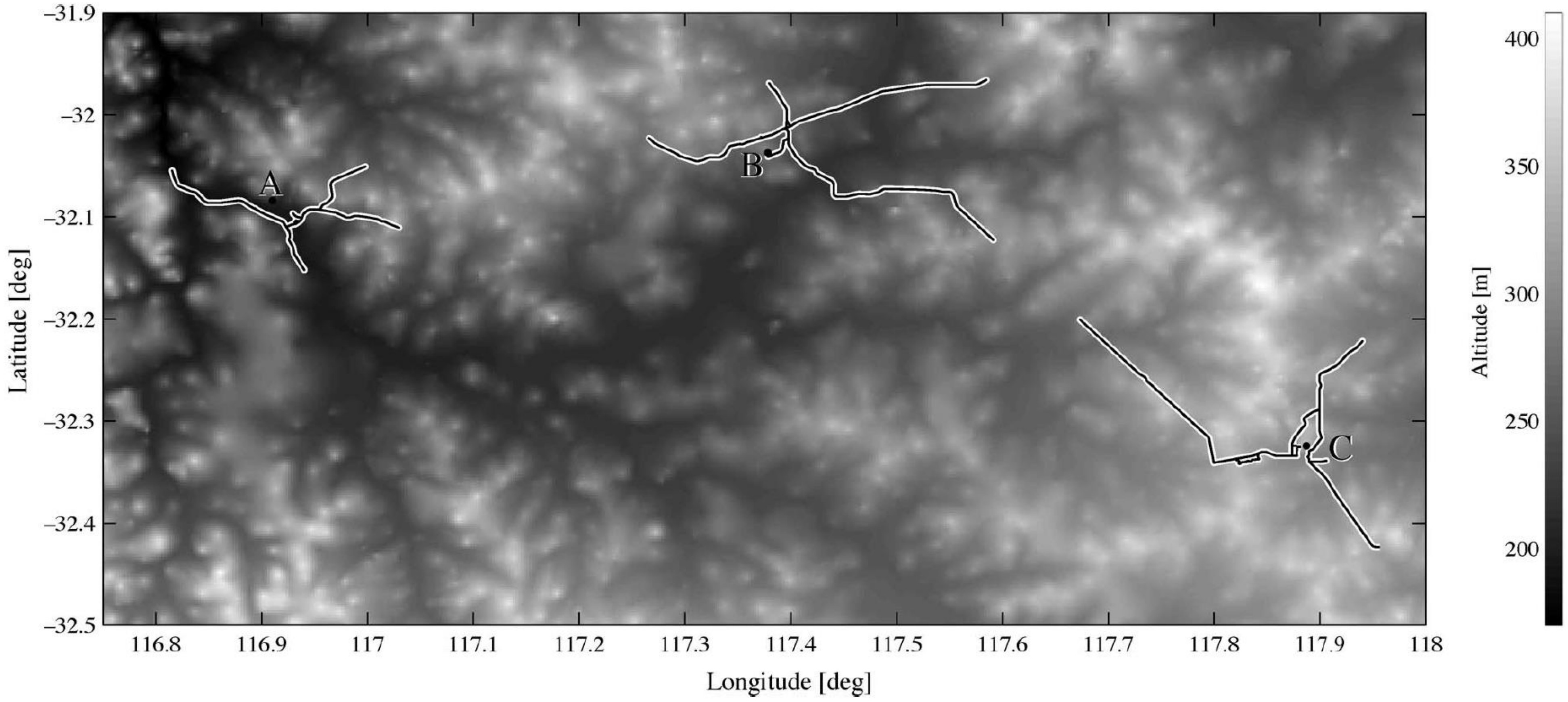


Fig. 3. DEM, with 9 s resolution, used for the ANN path-loss models. The measurement data originate from 19 routes and three BSs with omni-directional antenna characteristics in rural Western Australia [13]. From left to right, the BSs Beverley, Quairading, and Corrigin are marked with a dot and A, B, and C, respectively.

Digital Elevation Map (DEM)

- At the time of the measurements, the heights for the three BSs (A, B, and C) were ***30.5***, ***45***, and ***35*** m, respectively.

- The maximum antenna separation distance was ***24*** km, and data originating from antenna separation distances less than ***1*** km were not analyzed

- The correlation peak of the received pilot signal was sampled at ***600 Hz***.

- The roughness of the terrain in the three regions was estimated by the standard deviation of the terrain heights along the terrain path profile for every measurement position. The averaged value for each BS is given as follows: Beverley (A): ***10.9*** m, Quairading (B): ***14.1*** m, and Corrigin (C): ***16.5*** m.

- In the measurements originating from BS B, the presence of a large built-up area including a huge grainstorage facility (***>15 m*** height) has a noticeable impact on the measurements.

Compare with previous work

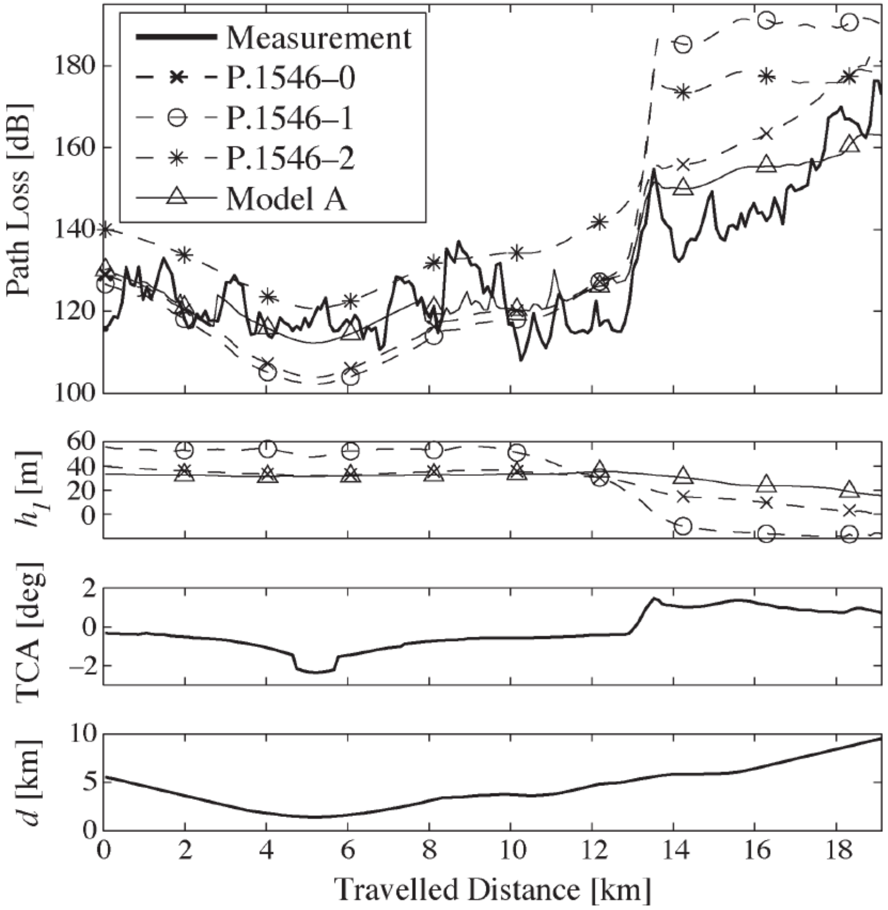


Fig. 4. P.1546 and Model A path-loss predictions for measurement 2003 A1 around the Beverley BS [13], where the transmitting/base antenna height is negative for some parts of the route (after 13 km traveled distance).

Paper [13] that compares different versions of the Recommendation ITU-R P.1546 model [26]–[28](0,1,2), the local average received powers are computed by averaging signal measurements over a measurement track of ***300*** wavelengths, which corresponds to approximately ***100*** m.

4. TRAINING AND PREDICTION PARAMETERS

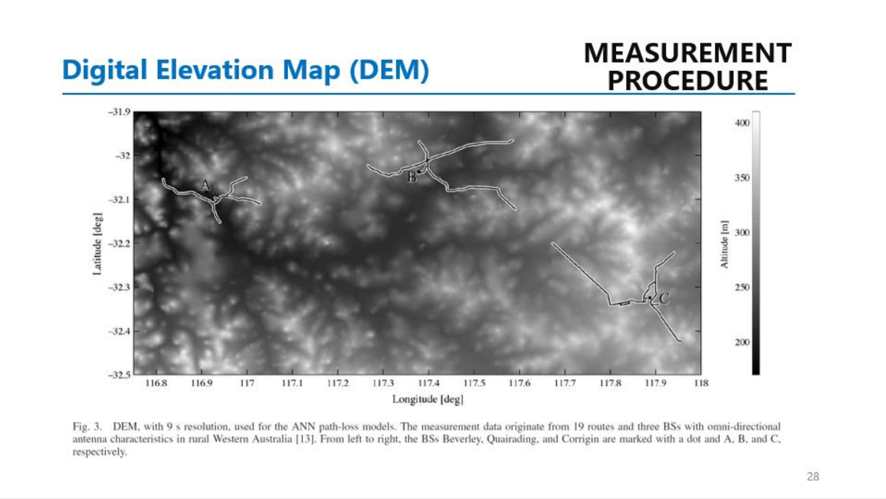
- Input Parameters
  - The terrain parameters
  - Transmitting/Base Antenna Height h1
  - TCA terrain clearance angle
  - Land Usage and Vegetation Information

Input Parameters

- The ANN inputs are chosen to be the antenna-separation distance, the transmitting/base antenna height [26], the terrain clearance angle (TCA) [26], the terrain usage, the vegetation type (VT), and the vegetation density near the receiving antenna [13].

The terrain parameters

- It is derived using a DEM with a ***9*** s resolution grid (approximately ***250*** m) supplied by Geoscience Australia (see Fig. 3).



- For an efficient ANN implementation, all input parameters are normalized to fall in the range between ***-1*** and ***1***.

Input Parameters

Transmitting/Base Antenna Height h_1

- The definition of h_1 is based on the definition of effective antenna height as in the Okumura model [29] and calculated as in the Recommendation ITU-R P.1546 [26].

TCA terrain clearance angle

- A TCA correction enables obstacles close to the receiver site to be taken into account to increase the prediction accuracy .

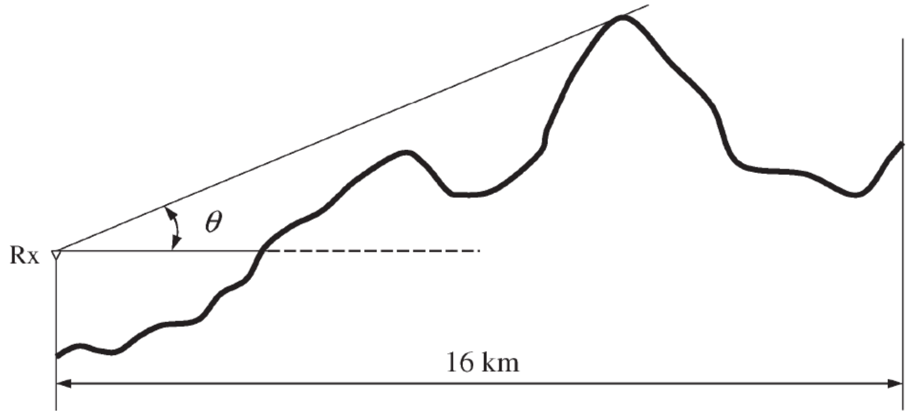


Fig. 5. Definition of θ ($d \geq 16$ km).

- The TCA is based on the angles relative to the horizontal between a line connecting the transmitting and receiving antennas and a line at the receiving antenna.

Input Parameters

Transmitting/Base Antenna Height h_1

- The h_1 definition that is used in the P.1546 models based on the definition of effective antenna height, as in the Okumura model [4].
- Effective antenna height h_{eff} is calculated as the base station (BS) height in meters over the average level of the ground between distances d of 3 and 15 km in the direction of the mobile antenna (see Fig. 2).

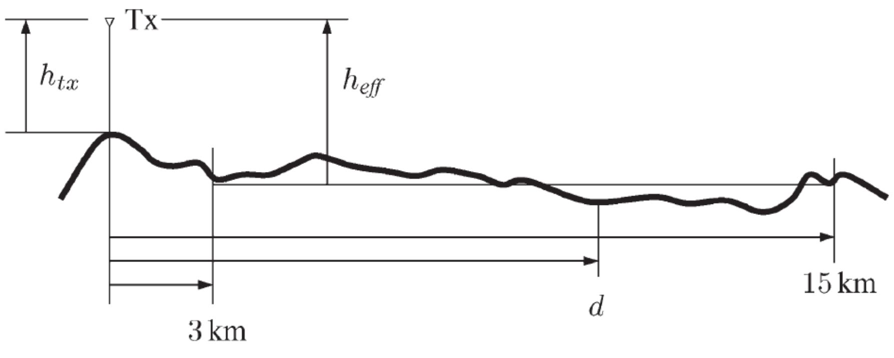
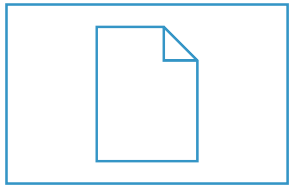


Fig. 2. Definition of effective antenna height h_{eff}.

- The h_1 definition that is used in the model A1



$$h_1 = \begin{cases} h_{tx} + \frac{(h_e - h_{tx})d}{15}, & \text{for } \ 1 \text{ km} \leq d < 12 \text{ km}, \\ h_{tx} + \frac{(h_t - h_{tx})d}{15}, & \text{for } 12 \text{ km} \leq d < 15 \text{ km}, \\ h_{eff}, & \text{for } 15 \text{ km} \leq d \leq 1000 \text{ km}. \end{cases} \quad (12)$$

Input Parameters

Land Usage and Vegetation Information

- The measured field strength is typically attenuated due to buildings near the BS. Information regarding land usage was obtained by inspecting satellite photos and notes made during the measurement campaigns.
- 1) **VT**:
  - The data set contains information on two levels, namely, land use data and VT
  - these VTs were divided into three groups corresponding to woodland, shrublands, and no vegetation.
- 2) **Vegetation density**:
  - The perennial crown density information is given on an accurate 25 m grid and is represented by a scalar value ranging from 0% to 100% corresponding to no vegetation and dense vegetation, respectively.
  - The crown density metric corresponds to the estimated percentage of land covered by the tree crown [30].
  - The vegetation density near the receiver (VDN) corresponds to the average vegetation density within a 100 m × 100 m area surrounding the receiver.

5. STATISTICAL ANALYSIS METRICS

- Parameter to evaluate
  - The first-order statistics
  - the correlation factor
  - the average total hit rate error AHRE
  - The prediction error
    - The maximum prediction error
    - the mean prediction error
    - the error standard deviation

Parameter to evaluate

STATISTICAL ANALYSIS METRICS

- The first-order statistics, the correlation factor, and the average total hit rate error (AHRE , in percentage) have been used to evaluate the results.
- Predicted and measured values are denoted p_i and m_i, respectively, and are given on a logarithmic (in decibels) scale.
- The prediction error ϵ_i is expressed as

$$\epsilon_i = p_i - m_i, \qquad i = 1, 2, \ldots, N \qquad (13)$$

- N is the number of samples
- The maximum prediction error, the mean prediction error, and the error standard deviation are denoted $\epsilon_{\max}[\mathrm{dB}]$, $\bar{\epsilon}[\mathrm{dB}]$, and $\sigma_\epsilon[\mathrm{dB}]$, respectively.

- The correlation coefficient is denoted r
- It provides a measure of the degree of linear relationship between measured m_i and predicted p_i values.

Parameter to evaluate

STATISTICAL ANALYSIS METRICS

- This single value metric (AHRE) is derived from the total hit rate (THR) curve [31].
- The location-specific THR curve is used as a direct indication of the quality of the prediction model.

- Given a path-loss threshold L_T , if both predicted and measured pathloss values are greater than, less than, or equal to L_T, the prediction is regarded as correct irrespective of the deviation of the predicted from the measured value. The AHRE is the mean deviation from 100% THR and is expressed as

$$\mathrm{AHRE} = \frac{1}{N_{L_T}} \sum_{L_T = L_{T,\min}}^{L_{T,\max}} 100\% - \mathrm{THR}(L_T) \qquad (14)$$

- N_{LT} is the number of THR points.
- The thresholds $L_{T,min}$ and $L_{T,mx}$ are chosen so that the AHRE may be interpreted as the area between the THR curve and *100%*.
- The method is useful in assessing the validity of a model where coverage is simply determined by a threshold value. A small AHRE value indicates a good fit between predicted and estimated values.

6. TRAINING AND EVALUATION

- ❑ Same Cell—Different Route
- ❑ Different Cell(A and C)
- ❑ All Cells
- ❑ The work in the paper

Tabel1 TRAINING AND EVALUATION

TABLE I
STATISTICS FOR MODEL A AND THE OH RURAL MODEL [13]

	ϵ_{max} [dB]		$\bar{\epsilon}$ [dB]		σ_ϵ [dB]		r_ϵ		AHRE [%]	
Meas.	Model A	OH	Model A	OH	Model A	OH	Model A	OH	Model A	OH
2003_A1	17.9	45.5	1.7	-12.4	7.7	11.4	0.89	0.75	9.5	18.4
2003_A2	18.2	34.2	-0.7	-8.5	8.5	9.5	0.67	0.49	13.8	16.5
2004_A3	17.4	24.0	3.4	-5.4	7.7	8.4	0.47	0.26	22.8	21.9
2004_A4	19.7	22.3	3.6	-5.1	8.1	9.2	0.39	0.13	23.8	23.0
2004_A5	27.3	31.6	4.7	-8.3	11.5	8.8	0.61	0.45	19.4	20.1
2003_B1	15.3	22.9	-2.2	-7.9	5.8	7.3	0.77	0.59	13.9	24.0
2003_B2	23.5	42.7	6.8	-4.1	9.3	10.3	0.82	0.78	12.1	9.6
2003_B3	31.2	49.2	-0.5	-10.1	10.2	12.0	0.75	0.62	10.7	14.4
2004_B4	29.2	37.0	-4.7	-13.6	6.8	9.2	0.80	0.59	12.0	24.5
2004_B5	25.3	33.3	1.9	-10.2	6.8	6.4	0.74	0.71	13.2	24.2
2004_B6	19.0	18.4	8.4	-5.5	5.8	5.9	0.59	0.58	26.9	22.4
2004_B7	18.7	26.4	-4.6	-13.6	6.9	6.7	0.58	0.59	19.6	30.2
2004_B8	22.0	41.4	2.8	-8.2	8.4	9.0	0.82	0.77	10.6	12.2
2004_B9	22.1	33.0	1.3	-8.1	9.5	9.7	0.71	0.58	14.1	16.5
2003_C1	31.1	45.4	-0.3	-15.8	8.3	9.8	0.87	0.84	8.2	17.4
2004_C2	24.5	30.4	4.9	-7.2	8.4	8.5	0.81	0.75	13.6	13.5
2004_C3	21.1	32.0	4.4	-4.9	8.0	8.6	0.76	0.68	12.4	11.2
2004_C4	27.6	32.7	4.7	-2.6	9.4	8.2	0.67	0.65	14.6	13.5
2004_C5	22.0	40.6	1.5	-12.8	8.2	8.7	0.86	0.80	9.2	15.9
Average	22.8	33.8	2.0	-8.7	8.2	8.8	0.71	0.61	14.8	18.4

Same Cell—Different Route

TABLE II
STATISTICAL ANALYSIS SUMMARY: SAME CELL—DIFFERENT ROUTE

Data	Metric	ANN_{6-1}	ANN_{6-3-1}	$ANN_{6-7-3-1}$
TR	ϵ_{max} [dB]	27.0	31.9	32.5
	$\bar{\epsilon}$ [dB]	0.4	0.3	-0.1
	σ_ϵ [dB]	7.0	7.0	6.7
	r_ϵ	0.81	0.84	0.82
	AHRE [%]	10.2	9.2	10.3
EV	ϵ_{max} [dB]	25.6	23.1	22.2
	$\bar{\epsilon}$ [dB]	-4.2	-3.5	-2.8
	σ_ϵ [dB]	7.7	7.4	7.3
	r_ϵ	0.85	0.86	0.87
	AHRE [%]	9.5	8.8	8.5

- Table II presents the numerical results for different-sized ANNs
  - training (TR) data
  - evaluation (EV) data
- For the TR data, increasing the number of hidden layers and neurons slightly decreases the prediction accuracy for some of the statistical analysis metrics
- For the EV data, it slightly increases the prediction accuracy for all statistical analysis metrics.

- The TR and EV data are taken from different measurement routes
- Hence, some deviations due to specific topographical features shown in the analysis of the TR data are not present in the analysis of the EV data.

Different Cell(A and C) TRAINING AND EVALUATION

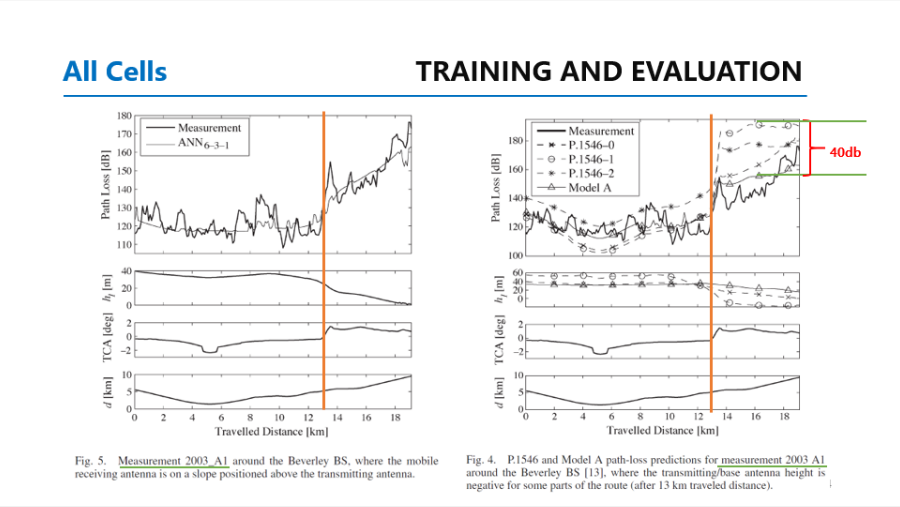
TABLE III
STATISTICAL ANALYSIS SUMMARY: DIFFERENT CELL

Data	Metric	ANN_{6-1}	ANN_{6-3-1}	$\text{ANN}_{6-7-3-1}$
EV	ϵ_{max} [dB]	35.8	32.2	29.4
	$\bar{\epsilon}$ [dB]	-5.1	-4.9	-3.3
	σ_ϵ [dB]	9.3	8.3	8.1
	r_ϵ	0.87	0.87	0.86
	AHRE [%]	11.3	10.4	9.5

- It shows the prediction results for the different-sized ANNs using EV data from measurement 2003_A1

- It can be seen that all statistical analysis metrics, except for the correlation factor r, slightly improve when extending the ANN
- for this scenario, the prediction accuracy somewhat increases with the additional number of hidden layers and neurons.
- It can be concluded that the ANN models generalize quite well when being evaluated using measurement data that originate from another BS and a different region.

All Cells TRAINING AND EVALUATION



- For parts of the route, after approximately ***13*** km traveled distance, the mobile receiving antenna is on a slope above the BS.

- This particular scenario causes some severe overestimation (***+40*** dB; see Fig. 4 after ***13*** km traveled distance) of the path-loss when using the Recommendation ITU-R P.1546 model due to how the transmitting/base antenna height is defined and incorporated in the model [13].

- Using an ANN model that has been trained with measurement data obtained from this specific scenario, it can be seen that the predicted path-loss accuracy is very good, even for the region after ***13*** km traveled distance.

All Cells

TRAINING AND EVALUATION

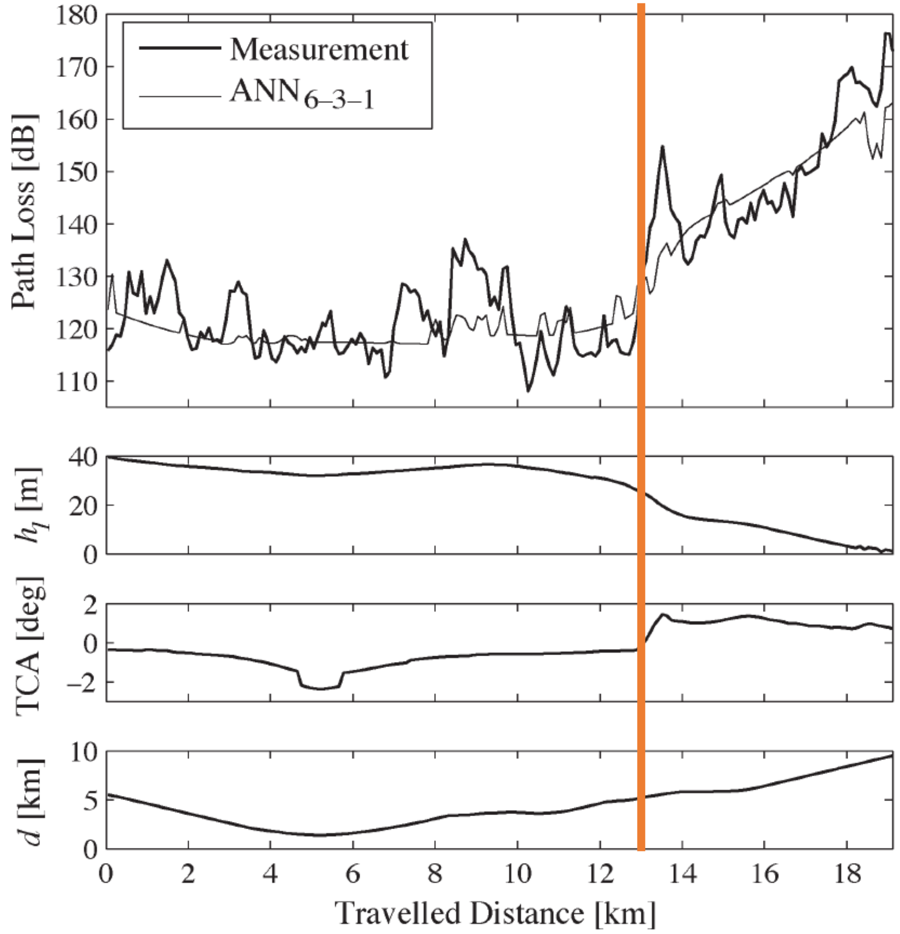


Fig. 5. Measurement 2003_A1 around the Beverley BS, where the mobile receiving antenna is on a slope positioned above the transmitting antenna.

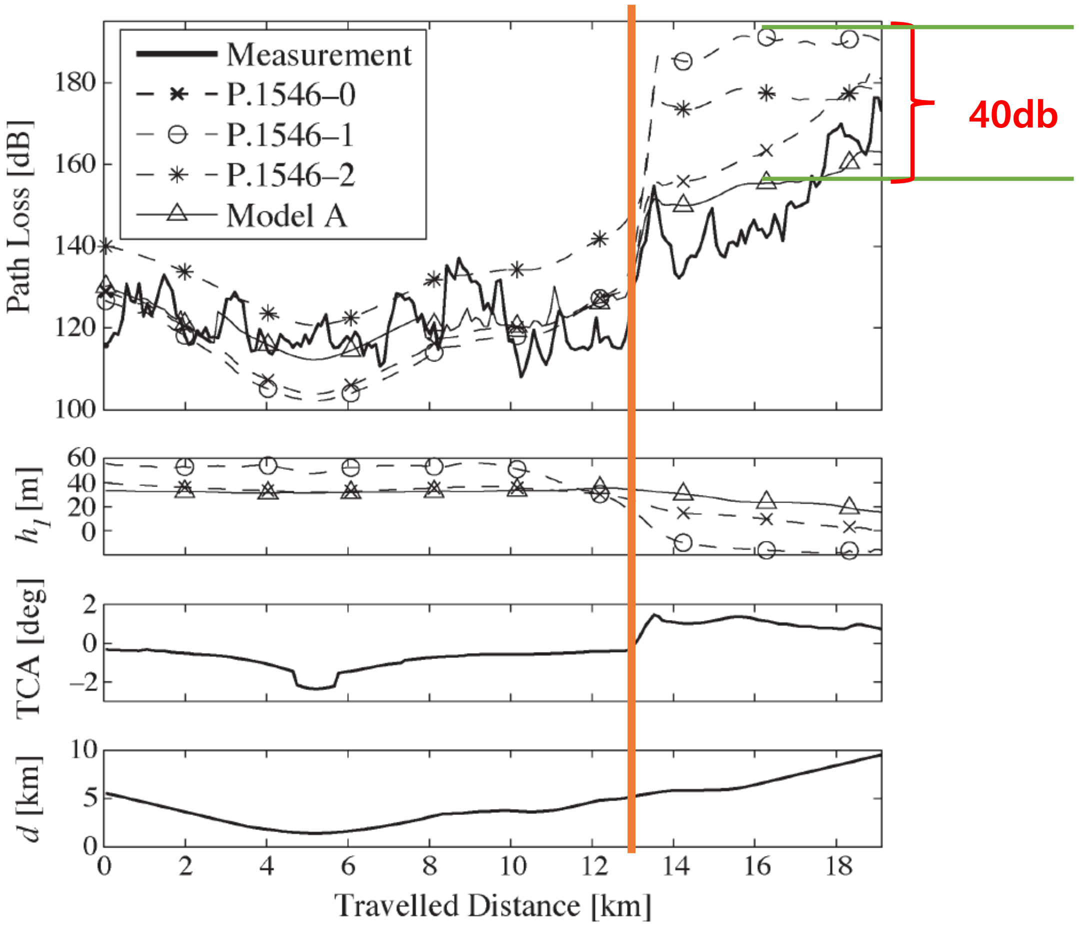


Fig. 4. P.1546 and Model A path-loss predictions for measurement 2003 A1 around the Beverley BS [13], where the transmitting/base antenna height is negative for some parts of the route (after 13 km traveled distance).

All Cells

TRAINING AND EVALUATION

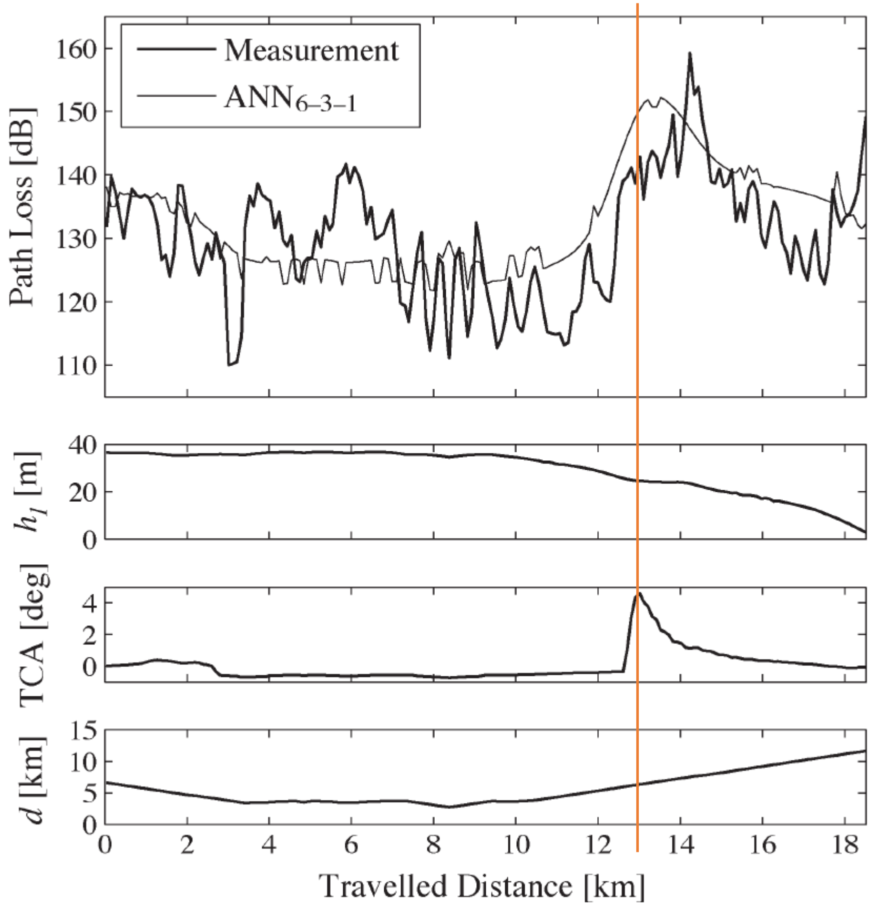


Fig. 6. Measurement 2004_A5 around the Beverley BS, where the transmitting/base antenna height is close to zero for parts of the route and a distinct peak in the TCA at 13 km traveled distance.

- At approximately ***13*** km traveled distance, a sharp peak in the TCA can be seen.
- For this specific scenario, the P.1546 model produces a correction overshoot (***20–40*** dB), which can result in unrealistic path-loss predictions [13].
- Now, when using the ANN prediction model, the path-loss correction due to TCA is smoother and does not significantly overshoot.

All Cells TRAINING AND EVALUATION

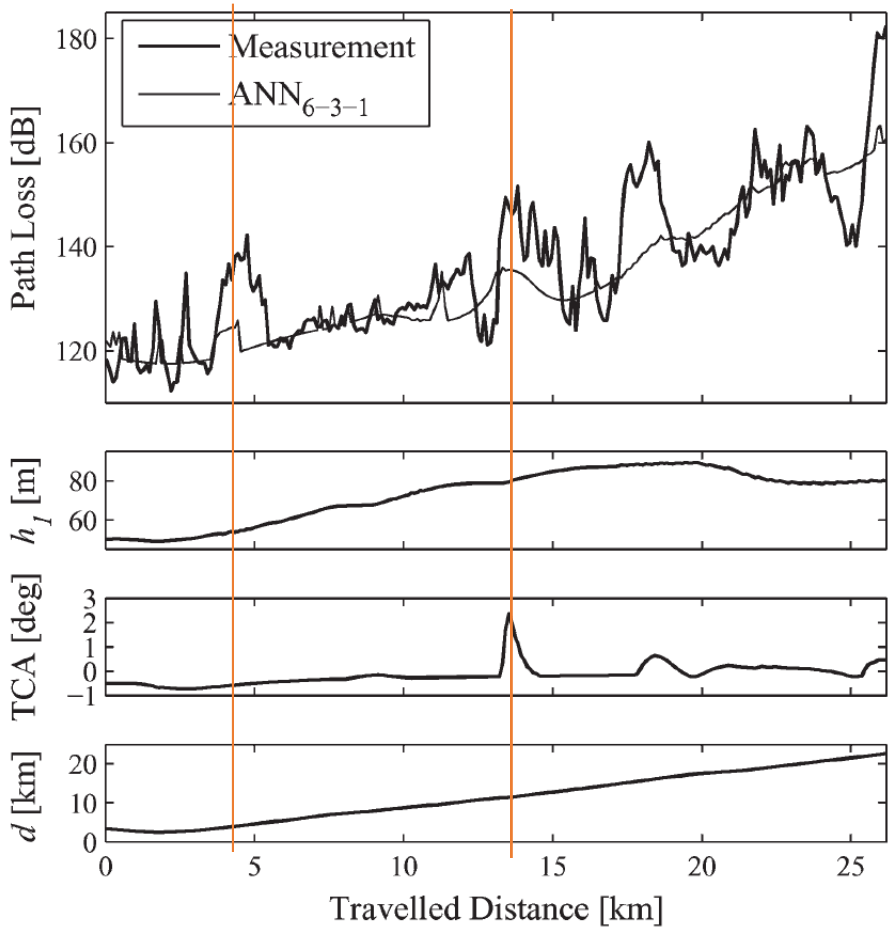


Fig. 7. Measurement 2004_B8 around the Quairading BS with a distinct peak in the TCA at 13 km traveled distance.

- At around ***4.5*** km traveled distance, there is a huge grain-storage facility (***>15*** m height) that affects the measurements.
- Furthermore, at a traveled distance of approximately ***13*** km, a distinct TCA peak is present.
- In the path loss prediction, it can be seen that the ANN does not create an overshoot due to this peak, i.e., the response is relatively smooth.

All Cells

TRAINING AND EVALUATION

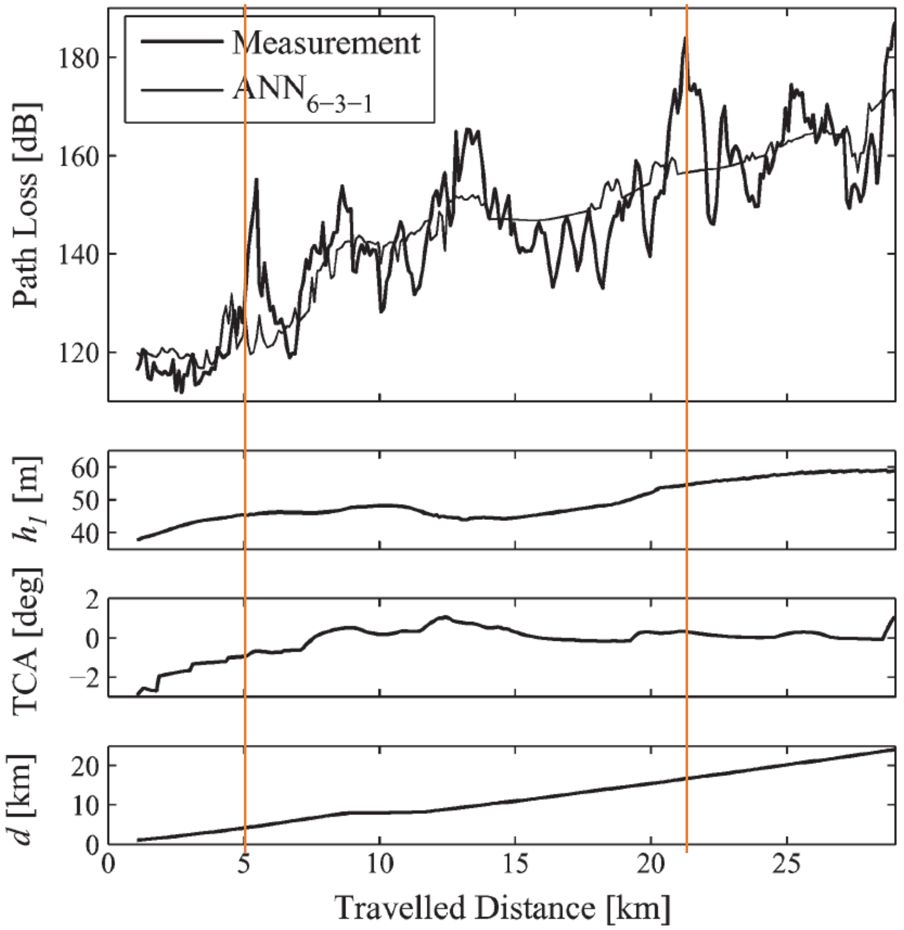
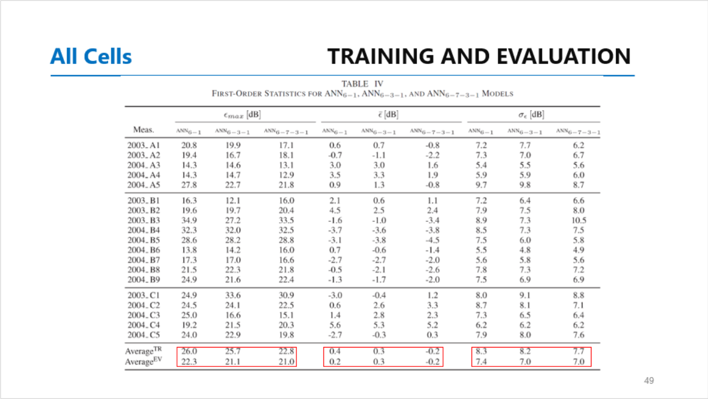
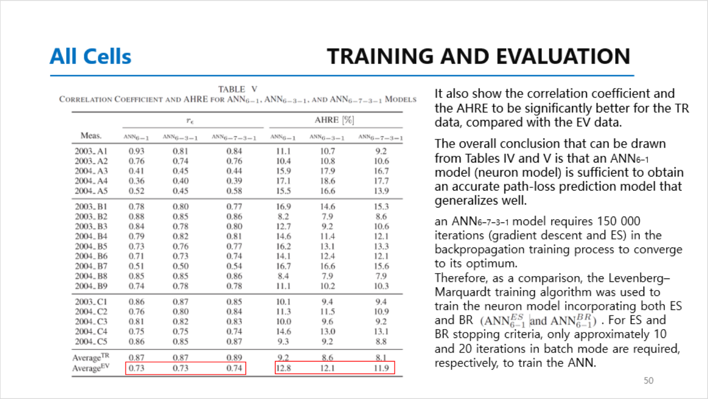


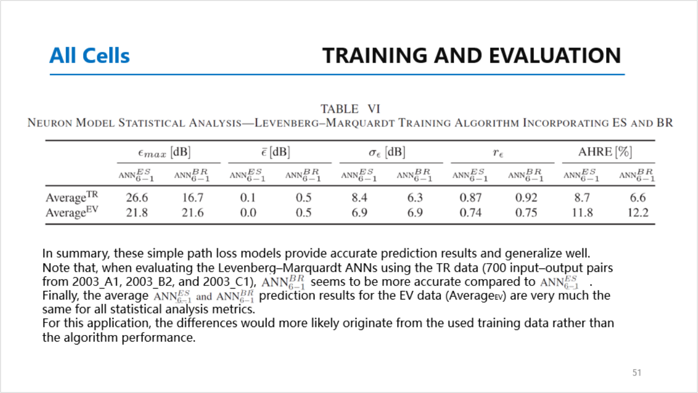
Fig. 8. Measurement 2003_C1 around Corrigin BS spanning over 1–24 km to the BS.

- The overall predictions provide good agreement with the measurements without introducing any major abnormalities.
- Note that, at approximately ***5*** and ***22*** km traveled distances, there are distinct increases in the measured path loss.
- These peaks are due to features that are not described by the topographical data and, hence, not reflected by the input parameters.

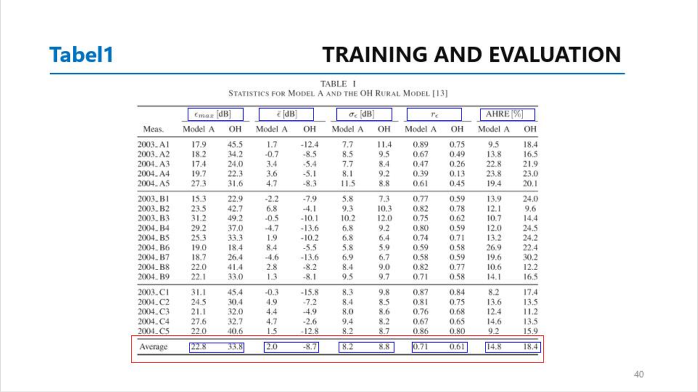
All Cells TRAINING AND EVALUATION







- Tables IV–VI show the statistical analysis results for the evaluated ANN models and the ***19*** measurement routes.
- The numerical values given in Tables IV and V correspond to EV data.
- For comparison, the two bottom rows in all tables provide the average prediction results for TR and EV data ($Average_{TR}$ and $Average_{EV}$).



All Cells

TRAINING AND EVALUATION

TABLE IV
FIRST-ORDER STATISTICS FOR ANN_{6-1}, ANN_{6-3-1}, AND $ANN_{6-7-3-1}$ MODELS

Meas.	ϵ_{max} [dB]			$\bar{\epsilon}$ [dB]			σ_ϵ [dB]		
	ANN_{6-1}	ANN_{6-3-1}	$ANN_{6-7-3-1}$	ANN_{6-1}	ANN_{6-3-1}	$ANN_{6-7-3-1}$	ANN_{6-1}	ANN_{6-3-1}	$ANN_{6-7-3-1}$
2003_A1	20.8	19.9	17.1	0.6	0.7	-0.8	7.2	7.7	6.2
2003_A2	19.4	16.7	18.1	-0.7	-1.1	-2.2	7.3	7.0	6.7
2004_A3	14.3	14.6	13.1	3.0	3.0	1.6	5.4	5.5	5.6
2004_A4	14.3	14.7	12.9	3.5	3.3	1.9	5.9	5.9	6.0
2004_A5	27.8	22.7	21.8	0.9	1.3	-0.8	9.7	9.8	8.7
2003_B1	16.3	12.1	16.0	2.1	0.6	1.1	7.2	6.4	6.6
2003_B2	19.6	19.7	20.4	4.5	2.5	2.4	7.9	7.5	8.0
2003_B3	34.9	27.2	33.5	-1.6	-1.0	-3.4	8.9	7.3	10.5
2004_B4	32.3	32.0	32.5	-3.7	-3.6	-3.8	8.5	7.3	7.5
2004_B5	28.6	28.2	28.8	-3.1	-3.8	-4.5	7.5	6.0	5.8
2004_B6	13.8	14.2	16.0	0.7	-0.6	-1.4	5.5	4.8	4.9
2004_B7	17.3	17.0	16.6	-2.7	-2.7	-2.0	5.6	5.8	5.6
2004_B8	21.5	22.3	21.8	-0.5	-2.1	-2.6	7.8	7.3	7.2
2004_B9	24.9	21.6	22.4	-1.3	-1.7	-2.0	7.5	6.9	6.9
2003_C1	24.9	33.6	30.9	-3.0	-0.4	1.2	8.0	9.1	8.8
2004_C2	24.5	24.1	22.5	0.6	2.6	3.3	8.7	8.1	7.1
2004_C3	25.0	16.6	15.1	1.4	2.8	2.3	7.3	6.5	6.4
2004_C4	19.2	21.5	20.3	5.6	5.3	5.2	6.2	6.2	6.2
2004_C5	24.0	22.9	19.8	-2.7	-0.3	0.3	7.9	8.0	7.6
Average^{TR}	26.0	25.7	22.8	0.4	0.3	-0.2	8.3	8.2	7.7
Average^{EV}	22.3	21.1	21.0	0.2	0.3	-0.2	7.4	7.0	7.0

All Cells

TRAINING AND EVALUATION

TABLE V
CORRELATION COEFFICIENT AND AHRE FOR ANN_{6-1}, ANN_{6-3-1}, AND $ANN_{6-7-3-1}$ MODELS

Meas.	r_ϵ			AHRE [%]		
	ANN_{6-1}	ANN_{6-3-1}	$ANN_{6-7-3-1}$	ANN_{6-1}	ANN_{6-3-1}	$ANN_{6-7-3-1}$
2003_A1	0.93	0.81	0.84	11.1	10.7	9.2
2003_A2	0.76	0.74	0.76	10.4	10.8	10.6
2004_A3	0.41	0.45	0.44	15.9	17.9	16.7
2004_A4	0.36	0.40	0.39	17.1	18.6	17.7
2004_A5	0.52	0.45	0.58	15.5	16.6	13.9
2003_B1	0.78	0.80	0.77	16.9	14.6	15.3
2003_B2	0.88	0.85	0.86	8.2	7.9	8.6
2003_B3	0.84	0.78	0.80	12.7	9.2	10.6
2004_B4	0.79	0.82	0.81	14.6	11.4	12.1
2004_B5	0.73	0.76	0.77	16.2	13.1	13.3
2004_B6	0.71	0.73	0.74	14.1	12.4	12.1
2004_B7	0.51	0.50	0.54	16.7	16.6	15.6
2004_B8	0.85	0.85	0.86	8.4	7.9	7.9
2004_B9	0.74	0.78	0.78	11.1	10.2	10.3
2003_C1	0.86	0.87	0.85	10.1	9.4	9.4
2004_C2	0.76	0.80	0.84	11.3	11.5	10.9
2004_C3	0.81	0.82	0.83	10.0	9.6	9.2
2004_C4	0.75	0.75	0.74	14.6	13.0	13.1
2004_C5	0.86	0.85	0.87	9.3	9.2	8.8
$Average^{TR}$	0.87	0.87	0.89	9.2	8.6	8.1
$Average^{EV}$	0.73	0.73	0.74	12.8	12.1	11.9

It also show the correlation coefficient and the AHRE to be significantly better for the TR data, compared with the EV data.

The overall conclusion that can be drawn from Tables IV and V is that an ANN_{6-1} model (neuron model) is sufficient to obtain an accurate path-loss prediction model that generalizes well.

an $ANN_{6-7-3-1}$ model requires 150 000 iterations (gradient descent and ES) in the backpropagation training process to converge to its optimum.
Therefore, as a comparison, the Levenberg–Marquardt training algorithm was used to train the neuron model incorporating both ES and BR (ANN_{6-1}^{ES} and ANN_{6-1}^{BR}) . For ES and BR stopping criteria, only approximately 10 and 20 iterations in batch mode are required, respectively, to train the ANN.

All Cells

TRAINING AND EVALUATION

TABLE VI

NEURON MODEL STATISTICAL ANALYSIS—LEVENBERG–MARQUARDT TRAINING ALGORITHM INCORPORATING ES AND BR

	ϵ_{max} [dB]		$\bar{\epsilon}$ [dB]		σ_ϵ [dB]		r_ϵ		AHRE [%]	
	ANN_{6-1}^{ES}	ANN_{6-1}^{BR}	ANN_{6-1}^{ES}	ANN_{6-1}^{BR}	ANN_{6-1}^{ES}	ANN_{6-1}^{BR}	ANN_{6-1}^{ES}	ANN_{6-1}^{BR}	ANN_{6-1}^{ES}	ANN_{6-1}^{BR}
$\mathrm{Average}^{\mathrm{TR}}$	26.6	16.7	0.1	0.5	8.4	6.3	0.87	0.92	8.7	6.6
$\mathrm{Average}^{\mathrm{EV}}$	21.8	21.6	0.0	0.5	6.9	6.9	0.74	0.75	11.8	12.2

In summary, these simple path loss models provide accurate prediction results and generalize well. Note that, when evaluating the Levenberg–Marquardt ANNs using the TR data (700 input–output pairs from 2003_A1, 2003_B2, and 2003_C1), ANN_{6-1}^{BR} seems to be more accurate compared to ANN_{6-1}^{ES} . Finally, the average ANN_{6-1}^{ES} and ANN_{6-1}^{BR} prediction results for the EV data ($\mathrm{Average}_{\mathrm{EV}}$) are very much the same for all statistical analysis metrics.
For this application, the differences would more likely originate from the used training data rather than the algorithm performance.