NEURAL NETWORK-BASED METAMODELLING APPROACH FOR ESTIMATION OF AIR POLLUTANT PROFILES

HERMAN WAHID

DOCTOR OF PHILOSOPHY

UNIVERSITY OF TECHNOLOGY, SYDNEY

2013

NEURAL NETWORK-BASED METAMODELLING APPROACH FOR ESTIMATION OF AIR POLLUTANT PROFILES

By

HERMAN WAHID

A thesis submitted in fulfilment of
the requirements for the degree of
Doctor of Philosophy

Faculty of Engineering
University of Technology, Sydney
February 2013

CERTIFICATE OF AUTHORSHIP/ ORIGINALITY

I certify that the work in this thesis has not previously been submitted for a similar degree nor has it been submitted as part of requirements for any other degree except as fully acknowledged within the text.

I also certify that the thesis has been written by me. Any help that I have received in my research work and the preparation of the thesis itself has been acknowledged. In addition, I certify that all information sources and literature used are referenced in the thesis.

February 2013

Production Note:
Signature removed prior to publication.

.....

Herman Wahid

This thesis is especially dedicated to my dearest father, mother, wife and family for their love, blessing and encouragement ...

ACKNOWLEDGMENT

First of all, I am greatly indebted to God in His mercy and blessing for making this project successful. This research project was carried out in the Faculty of Engineering, University of Technology Sydney between August 2009 and September 2012. I gratefully acknowledge the financial support for this research by the Malaysia Government and Universiti Teknologi Malaysia (UTM) Scholarship.

I would like to express my deepest gratitude and appreciation to my honourable supervisor, Associate Professor Dr. Quang Ha for his continuous guidance, encouragement, committed support, timely reviews and invaluable advice throughout the duration of this research project.

I would like to thank two atmospheric scientists, Dr. Hiep Duc from Office of Environment and Heritage New South Wales, and Dr. Merched Azzi from CSIRO Energy Technology New South Wales, for their assistance with air quality modelling related theories and their constructive comments with respect to this research project. Also thanks to Dr. Hiep Duc and his division team members for guiding and providing me with the valuable knowledge of air quality measurement methodologies for the collection of the research dataset.

My gratitude is also extended to Dr. Guenter A Plum for editing my thesis draft. Dr. Plum is an Australian academic thesis editor and proofreader, and has more than ten years experience in academic thesis editing, academic thesis proofreading and academic dissertation editing and proofing with clients in Australia, Asia, and around the world. Dr. Plum has worked in universities in Australia and Hong Kong as well as in the computer industry. Dr. Plum's helpful suggestions and attention to detail have been integral to a timely completion of my thesis.

I would also like to extend my thanks and best wishes to all students in my research group, especially to Vahid Vakiloroaya and Raja M Taufika Raja Ismail for some

technical supports during my study. Their presence and companionship made my study experience most comfortable and enjoyable.

Last but not least, I would like to express my sincere appreciation and gratitude to my parents, Wahid Moktar and Mazenah Sidik, for their patience, encouragement and support throughout the years of my study. I would like to dedicate this research to my beloved wife, Nazariah Jadon. Her constant empathy and consistent moral support have eased the tough times during my research.

ABSTRACT

The air quality system is a system characterised by non-linear, complex relationships. Among existing air pollutants, the ozone (O_3), known as a secondary pollutant gas, involves the most complex chemical reactions in its formation, whereby a number of factors can affect its concentration level. To assess the ozone concentration in a region, a measurement method can be implemented, albeit only at certain points in the region. Thus, a more complicated task is to define the spatial distribution of the ozone level across the region, in which the deterministic air quality model is often used by the authority. Nevertheless, simulation by using a deterministic model typically needs high computational requirements due to the nonlinear nature of chemical reactions involved in the model formulation, which is also subject to uncertainties. In the context of ozone as an air pollutant, the determination of the background ozone level (BOL), independent from human activities, is also important as it could represent one of reliable references to human health risk assessment. The concept of BOL may be easily understood, but practically, it is hard to distinguish between natural and anthropogenic effects. Apart from existing approaches to the BOL determination, a new quantisation method is presented in this work, by evaluating the relationship of ozone versus nitric oxide (O_3-NO) to estimate the BOL value, mainly by using night-time and early morning measurement data collected at the monitoring stations.

In this thesis, to deal with the challenging problem of air pollutant profile estimation, a metamodel approach is suggested to adequately approximate intrinsically non-linear and complex input-output relationships with significantly less computation. The intrinsic characteristics of the underlying physics are not assumed to be known, while the system's input and output behaviours remain essential. A considerable number of metamodels approach have been proposed in the literature, e.g. splines, neural networks, kriging and support vector machine. Here, the radial basis function neural network (RBFNN) is concerned as it is known to offer good estimation performance on accuracy, robustness, versatility, sample size, efficiency, and

simplicity as compared to other stochastic approaches. The development requirements are that the proposed metamodels should be capable of estimating the ozone profiles and its background level temporally and spatially with reasonably good accuracies, subject to satisfying some statistical criteria.

Academic contributions of this thesis include in a number of performance enhancements of the RBFNN algorithms. Generally, three difficulties involved in the network training, selection of radial basis centres, selection of the basis function variance (i.e. spread parameter), and training of network weights. The selection of those parameters is very crucial, as they directly affect the number of hidden neurons used and also the network overall performance. In this research, some improvements of the typical RBFNN algorithm (i.e. orthogonal least squares) are achieved. First, an adaptively-tuned spread parameter and a pruning algorithm to optimise the network's size are proposed. Next, a new approach for training the RBFNN is presented, which involves the forward selection method for selecting the radial basis centres. Also, a method for training the network output weights is developed, including some suggestions for estimation of the best possible values of the network parameters by considering the cross-validation approach. For applications, results show that the combination of the proposed paradigm could offer a sub-optimal solution of metamodelling development in the generic sense (by avoiding the iteration process) for a faster computation, which is essential in air pollutant profile estimation.

PUBLICATIONS

Journal Articles

1. Hiep Duc, Merched Azzi, Herman Wahid, Q.P. Ha, "Background ozone level in the Sydney basin: Assessment and trend analysis", *Journal of Climatology*, in Press, doi: 10.1002/joc.3595.
2. H. Wahid, Q.P. Ha, H. Duc, "New sampling scheme for neural network-based metamodelling with application to air pollutant estimation," *Gerontechnology*, vol. 11(2), 2012, pp. 336, doi:10.4017/gt.2012.11.02.325.00.
3. H. Wahid, Q.P. Ha, H. Duc, M. Azzi, "Meta-modelling approach for estimating the spatial distribution of air pollutant levels", *Journal of Applied Soft Computing* (revised and re-submitted).
4. Q.P. Ha, H. Wahid, H. Duc, M. Azzi, "Radial basis function neural network-based metamodelling for ozone spatial estimation," *IEEE Transactions on Neural Network and Learning Systems* (submitted).

Conference Proceedings

1. H. Wahid, Q.P. Ha, H. Duc, "New sampling scheme for neural network-based metamodelling with application to air pollutant estimation," *Proc. 8th World Conference of Gerontechnology & 29th International Symposium for Automation and Robotics in Construction (ISG-ISARC 2012)*, Eindhoven, Netherlands, June 26-29, 2012.
2. Herman Wahid, Q.P. Ha, Hiep Duc, Merched Azzi, "Estimation of background ozone temporal profiles using neural networks," *Proc. 3rd IEEE Int. Conf. on Intelligence Computing and Intelligent Systems (ICIS 2011)*, Guangzhou, China, 18-20 Nov 2011, pp. 292-297.
3. Herman Wahid, Q.P. Ha, Hiep Nguyen Duc, "Computational intelligence estimation of natural background ozone level and its distribution for air quality modelling and emission control," *Pro. 28th International Symposium on*

Automation and Robotics in Construction (ISARC 2011), Seoul, Korea, 29 Jun-2 Jul 2011, pp. 551-557.

4. Herman Wahid, Quang P. Ha, and Hiep Nguyen Duc, "A Metamodel for Background Ozone Level Using Radial Basis Function Neural Networks," *Pro. 11th International Symposium on Control, Automation, Robotics and Vision (ICARCV 2010)*, Singapore, 7-10 Dec 2010, pp. 958-963.
5. H. Wahid, Q. P. Ha, and H. Duc, "Adaptive Neural Network Metamodel for Short-term Prediction of Natural Ozone Level in an Urban Area," *Proc. Int. Conf. on Computing and Communication Technologies (2010 IEEE-RIVF)*, Hanoi, Vietnam, 1-4 Nov 2010, pp. 250-253.

Related publications

1. Herman Wahid, Hiep Nguyen Duc, and Quang P. Ha, "Radial Basis Function Neural Network Metamodelling for 2-D Resistivity Mapping," *Proc. 27th International Symposium on Automation and Robotics in Construction (ISARC 2010),* Bratislava, Slovakia, 25-27 Jun 2010, pp. 364-373.
2. H. Wahid, Q.P. Ha, and M.S. Mohamed Ali, "Optimally-Tuned Cascaded PID Control using Radial Basis Function Neural Network Metamodeling," *Proc. of the 3rd International Workshop on Artificial Intelligence in Science and Technology (AISAT'09)*, Hobart, Australia, 23-24 November 2009, paper S01.2.

TABLE OF CONTENTS

LIST OF TABLES

Table **Page**

LIST OF FIGURES

Figures **Page**

LIST OF ABBREVIATIONS

σ	-	RBF spread parameter
σ_c	-	Isotropic spread parameter for RBF centres selection
λ	-	Regularisation parameter
AQM	-	Air quality management
BOL	-	Background ozone level
CGS	-	Classical Gram-Schmidt
d_2	-	Index of agreement
DAQM	-	Deterministic air quality model
DOE	-	Design of experiments
EPA	-	Environment Protection Authority (in Australia)
FFD	-	Full factorial design
FS	-	Forward selection
GCV	-	Generalised cross-validation
GLS	-	Generalised least squares
GMR	-	Greater Metropolitan Region
LHD	-	Latin hypercube design
LOO-CV	-	Leave one out cross-validation
LS	-	Least squares
MAE	-	Mean absolute errors
MSE	-	Mean squared errors
NAAQS	-	National Ambient Air Quality Standards
NEPM	-	National Environment Protection Measures
NO	-	Nitric oxide
NO_2	-	Nitrogen dioxide
NO_x	-	Nitrogen oxides
O_3	-	Ozone
OLS	-	Orthogonal least squares
PM	-	Particulate matter
ppb	-	parts per billion

pphm	-	parts per hundred million
R^2	-	Determination coefficient
RBFNN	-	Radial basis function neural network
RMSE	-	Root mean square errors
SSE	-	Sum of squared errors
TEMP	-	Ambient temperature
US-EPA	-	U.S. Environmental Protection Agency
VOCs	-	Volatile organic compounds
WCD	-	Weighted clustering design
WDR	-	Wind direction
WLS	-	Weighted least squares
WSP	-	Wind speed

Chapter 1
INTRODUCTION

1.1. General introduction

The issue of air quality continues as the main topic being debated and researched among policy makers and the public. Air pollution is a concern to many people as it directly influences the quality of human health, including respiratory problems, heart and lung diseases, and may even occasion premature death. Children are at greater risk as they are generally more active outdoors and their lungs are still developing, whilst elderly people are also sensitive to some types of air pollution. For example, in Australia, about two million Australians suffer from asthma, and hundreds of thousands of others are affected by respiratory disorders in which poor air quality is presumed to be the most important factor (as reported in DoEH, 2004).

The change in the level of air quality arises from various emission sources, mainly industrial emissions such as from factories and power plants; mobile transportation such as cars, buses, trucks, planes and ships; and also biogenic sources such as bushfires, vegetation and windblown dust. These are illustrated in Fig. 1.1 (a–c). The quantity of pollutants released to the atmosphere and their removal could be affected by factors such as source strengths, sunlight, geography, moisture, clouds, rain, and weather patterns, locally and regionally.

In New South Wales (NSW), Australia, the air quality assessment was being carried out by the Department of Environment and Climate Change NSW (DECC), and recently the responsibility has been taken over by the Office of Environment and Heritage NSW (OEH). In a special project, the “25-year Air Quality Management

Plan – Action for Air", it was noted that some measures of air quality have been conducted in NSW since 1994; the study area mainly covered the urban area of the greater Sydney, Newcastle and Wollongong regions, known collectively as the Greater Metropolitan Region (GMR) (DECC, 2007a). The study involved air quality measurements at monitoring stations and computer simulations using airshed models, and the assessment of the pollutants' precursor emission rates. From the ten year comprehensive emissions inventory from 1994 to 2004, the six most critical agents have been identified as sulphur dioxide (SO_2), nitrogen oxide (NO_2), carbon monoxide (CO), ground level ozone (O_3), lead (Pb), and fine particles less than 10 micrometres (PM_{10}). Several significant impacts of these air pollutants on human health quality are summarised in Table 1.1 (DoEH, 2004). Of the six key air pollutants included under the National Environment Protection Measure for Ambient Air Quality (NEPM), only two remain as significant issues in NSW, the most important being surface ozone and to a lesser extent, fine particles.

(a) (b) (c)

Fig. 1.1 Air pollutant sources: (a) industrial emissions, (b) motor vehicles, and (c) natural sources.

At the present time, the effort is mainly focused on reducing the surface ozone level where it exceeds the national standard (i.e. Air NEPM standard). Ozone (O_3) is a secondary pollutant gas that is naturally produced in the earth's atmosphere, a product of the chemical reaction between nitrogen oxides (NO_x, $NO_x = NO + NO_2$) and volatile organic compounds (VOCs), with the existence of solar radiation (from sunlight), and also influenced by other factors, such as meteorological and topographical. The stratosphere ozone is very useful as it could shield humans from the harmful influences of the sun's ultraviolet rays. However, exposure to the tropospheric ozone (also known as surface ozone or ground level ozone) may be harmful rather than beneficial to living organisms because it can damage living tissues and break down certain materials.

Table 1.1 Six basic pollutants found in Australia and their characteristics.

POLLUTANT	DESCRIPTION	SOURCES	EFFECTS
Carbon monoxide (CO)	Poisonous gas	Produced when fuels containing carbon do not fully combust. Mainly produced by motor vehicles.	Can affect mental function, alertness, and worsen cardiovascular diseases. Harmful even in low concentrations.
Nitrogen dioxide (NO_2)	Highly reactive gas. Plays a major role in the formation of photochemical smog.	Sourced from motor vehicles and industries (i.e. power plants).	Increases respiratory illnesses during short term exposure. Lowers the resistance to respiratory infections during long term exposure.
Ozone (O_3)	Highly reactive gas. Produced in the stratospheric ozone layer. Main chemical in photochemical smog.	Formed as a chemical reaction when sunlight reacts with compounds from motor vehicles, refineries, and vegetation.	It could significantly decrease lung function, increase respiratory symptoms, aggravate asthma, affect vegetation and building materials.
Sulphur dioxide (SO_2)	Reactive gas	Main sources: power plants, refineries and smelters.	Irritates eyes, nose and throat. Aggravates asthma and bronchitis. Can cause lung damage.
Lead (Pb)	Metal	Produced mainly by vehicles using leaded petrol.	Can cause damage to nervous system, kidneys and reproductive organs.
Fine particles	In two size ranges: PM_{10} (inhalable particles) and $PM_{2.5}$ (respirable particles)	Examples: Residues from motor vehicles, domestic wood heaters and bushfires.	It is thought to increase respiratory symptoms, aggravate asthma, and cause premature death.

1.2. Problem statement

As major cities and their surrounding suburbs around the world swell with people, motor vehicles and industries, the number of cities with poor environmental quality continues to grow. There is an urgent need to address these issues by better understanding the connections between air pollution formation, human health, and emission control or urban management. The concentration of air pollutants can be attributed to many factors such as specific individual sources, source emission density, topography, and the state of the atmosphere, hence their formations generally involve very complex chemical reactions. Many possible tools have been used by the policy makers to manage air quality either by using direct measurement or simulation software. However, there is an increasing demand for better solutions involving faster simulations and more reliable results to allow effective decisions to

be made relating to air quality management. Due to air pollution trending upwards and exceeding standards, regulators in many countries are focusing on the ozone pollutant problem.

Basically, ozone concentration could be easily measured by special instruments which are typically located at the monitoring stations, or in mobile measurement stations (see e.g. Elkamel et al., 2001). However, analysis at the fixed monitoring stations can only be assessed at the location of interest, which may limit the value of the information for the policy maker. To overcome this issue, one way is the use of mobile measurement stations that can be moved to other locations after some interval of time, rather than increasing the number of fixed monitoring sites to avoid expensive investment in instrumentation. However, this is generally difficult to be implemented, quite time-consuming and possibly inaccessible at most rural locations. To further extrapolate the results, a spatial distribution approach is another useful method to tackle this problem. Typically, deterministic dispersion models are used to handle the spatial estimation task (see e.g. Seigneur, 2001; Phillips & Finkelstein, 2006; Monteiro et al., 2007), however, they need a high level of expertise in their development, require longer time in execution, and the reliability of the outputs is also questionable.

Hence, to reduce the computation burden for the spatial estimation task, more appropriate and reliable statistical techniques could be implemented. Several works have appeared in the literature related to this approach, for example, Duc et al. (2000) used a Kriging approach to study the spatial correlation of some pollutants over a long-distance network in Sydney, Australia. In this work, a metamodel approach will be proposed incorporating the simulation output of a photochemical dispersion model, namely “The Air Pollution Model and Chemical Transport Model (TAPM-CTM), in which the approach could reduce the computational cost by avoiding the modelling complexity and to improve the reliability of the approximation.

A more difficult task than the ozone level estimation is the determination of the “background ozone level (BOL)” over a region, especially in urban areas. For instance, BOL can be defined as the level of ozone occurring in the troposphere

which is naturally-formed and free from anthropogenic influences (Duc & Azzi, 2009). Accurate determination of the background ozone level requires a clean environment to be free from these anthropogenic influences, which is difficult to achieve in practice. Generally, the determination may be done in two ways; by measurement and modelling. The former method (typically incorporated in quantisation analysis) can only be implemented at remote sites (Donev et al., 2002; Oltmans et al., 2008), which are normally located in pristine area. Thus, this method is not possible for BOL determination in urban areas. A more generic technique is to use an air quality model (AQM) (e.g. US-EPA, 2006a), although most recent AQMs still having a high level of uncertainty in the prediction of the BOL because their estimation process is much influenced by the correctness of the biogenic emission data as input to the model. For this reasons, a neural network-based metamodel using ambient air quality data incorporating some quantisation approaches will be utilised in this work for BOL determination to simplify the solution and improve the reliability of the estimation.

In terms of the methodology, metamodelling (also known as the 'surrogate model') is the technique for determination of simpler models from the complex models that involve less computation but adequately represent a good approximation for the non-linear system behaviour. The exact, inner working of the simulation code is not assumed to be known, while the input-output behaviour is important. Substantial results from the existing works illustrate that using metamodels to locate an optimum solution is often sufficiently accurate in many applications requiring prediction, optimisation, verification and validation (Tunali & Batmaz, 2003). A number of metamodelling techniques exist such as polynomial regression, neural networks, Multivariate Adaptive Regression Splines (MARS), and Kriging. Nevertheless, there is no conclusion about which model is definitely superior to the others. However, insights have been gained through a number of recent studies, whereby Kriging and Radial Basis Function (RBF) models are intensively investigated (Fang et al., 2005). In general, Kriging models are more accurate for nonlinear problems, however, they are difficult to use due to the global optimisation process applied to identify the maximum likelihood estimators. On the other hand, polynomial models are easy to construct, but are less accurate. The RBF model, especially the multi-quadric and

Gaussian RBF, can interpolate sample points and is easy to construct, which results in a trade-off between Kriging and polynomials.

To accomplish the modelling and estimation process for the air quality issues, a metamodelling approach based on a radial basis function neural network (RBFNN) will be used throughout the work. To overview, the RBFNN is a special type of feed-forward neural network architecture which consists of an input layer, a hidden layer and an output layer. Several forms of radial basis functions are used in RBFNN, with Gaussian being probably the most popular because of its attractive mathematical properties of universal and best approximation, and its hill-like shape is easy to control with the variance parameter. In the RBFNN, three difficulties are involved in the training algorithm; the selection of the radial basis centres, the selection of the basis function radius (spread), and the training of network weights. These problems will be addressed in this work, which includes several improvements in the typical algorithm, a new algorithm for training the RBFNN and a new sampling method for a neural network based metamodel.

1.3. Research objectives

This research aims to provide a comprehensive analysis of: (1) to determine the background ozone level and its complicated relationship with other air pollutant factors such as nitrogen oxides (NO_x), volatile organic compounds (VOCs), meteorological conditions and also terrain; (2) to develop a metamodel for the accurate prediction of ozone concentration and background ozone temporal and spatial distribution, under various perspectives and scenarios; and (3) to introduce several improvements in the neural network-based metamodel which includes the new training algorithm of RBFNN and the new sampling scheme for the input-output data set. It is expected that the findings from this research can be used in the larger scale quantification and prediction of the emission sources, and some interpretations may be used as part of the reference to the policy maker for better air quality control and management. These research aims are elaborated further as follows:

- ***Determination of background ozone level:*** In this work, a non-photochemical condition background ozone level will be considered, which will be derived from the ambient measurement data during night time and early morning time (e.g. from 7.00 pm to 7.00 am the next morning). It excludes the photochemical process that would occur during daytime as if only natural sources were present.

- ***Metamodel development for ozone and background ozone:*** In this research, a neural network-based metamodel will be developed as a statistical approximation technique for the prediction of ozone and background ozone levels. RBFNN will be assigned as a system model, whereby the error between the metamodel output and the target output will be minimised. The idea is to design a network model for each measuring station and from this information, to construct a more generic model for application over the region of interest.

- ***Improvements of the RBFNN metamodel processes:*** The RBFNN involves some difficulties in its training algorithm, which corresponds to its modelling performance based on some statistical performance indexes. This research will attempt to develop a new algorithm for the training process, which is expected to offer better performances than other techniques in some respects, by comparing with the actual value (i.e. ambient measured data for air quality), as well as with other available training methods.

- ***Trend analysis and correlation:*** Statistical investigation will be conducted to reveal the ozone background trend, and subsequently to interpret the implication of this trend in setting the ozone goal target for emission reduction.

1.4. Significance of the work and contributions

The significance of the work and its expected contributions generally can be classed into two main groups: first, the contributions in respect to the application of the methodology in the air quality studies; and second, the improvement of the learning

algorithms in the scope of a neural network-based metamodel, the details being described as follows:

A. Application of the methodology in the atmospheric studies

i. Prediction of the air pollutant using metamodel

The future prediction of the air pollutant's temporal and spatial distribution (either short-term or long-term) is essential because its trend could initiate an authority for correcting or setting the right air management policy. Generally, air quality models (AQM) are used to deal with this nonlinear and complicated task. However, because of their complexity, their execution is quite time consuming, and may take several days or a week depending on the model used and the scale of the region under consideration. A metamodel approach featuring radial basis function neural network (RBFNN) is suggested to overcome this difficulty in which the function approximation is developed using input-output relationships, thus possibly avoiding the expensive computation of a complex chemical reaction in the AQM, especially when dealing with spatial estimation. Apart from inexpensive computation, RBFNN may provides more reliable results of the estimation and may offers better predictions of pollutant concentrations than those using the deterministic model, when compared to the measurement data collected at monitoring stations.

ii. Determination of background ozone level

The concept of background ozone is easily grasped but the challenging problem is how to define and distinguish what remains as natural and anthropogenic effects, which requires a 'clean' environment. However, a 'clean environment' before anthropogenic changes is practically hard to find and determine when man has already changed the settings. The best available solution for this predicament is to measure the ozone concentration at pristine sites, combined with some statistical quantisation methods. Unfortunately, this approach cannot be implemented in highly urbanised areas such as in Sydney, Australia. Thus, in this work, a new generic approach will be introduced based on the ambient measurement of night-time

data for ozone and nitrogen oxides, which approach may be used to determine the background ozone in any location. As the quantisation process differs from other authors' suggestions, this definition is specially labelled as 'night-time background ozone level'.

iii. Correlation analysis of the metamodel prediction

Analytical investigation of the metamodel prediction of the air pollutants are envisaged could refine the atmospheric interpretation of the relationship of ozone and their precursors (i.e. oxides of nitrogen and volatile organic compounds) and of the interaction of major sources contributing to ozone and other air pollutants. The integration strategy of the metamodel and the deterministic air quality model such as TAPM-CTM may increase the trustworthiness of the air quality predictions and their future trends, to assist the authority in formulating suitable policies for air quality control.

B. Improvement of the metamodel process and its algorithms

i. Improvement of the radial basis function neural network (RBFNN) learning algorithm

Generally, the training processes of the RBFNN involve three difficulties; to find the best centres from its trial dataset, to set the appropriate values of the radius from its centres (i.e. the variance of the basis function), and to train the network's weights between the inner and output layers. All these properties have inter-relationships with each other, thus to achieve the best training performance, a set of algorithms must consider these three factors. This work attempts to develop a new training algorithm for RBFNN that involves several elements as follows: to suggest some improvements in the typical algorithm which includes the adaptively-adjusted spread parameter based on steepest descent technique, and also the pruning algorithm; to introduce a supervised training algorithm for the selection of the basis centres based on a forward selection strategy by incorporating the regularised and the weighted least squares theory; and to suggest several approaches for optimally tuning the radial basis parameters including the least weighting factors, the regularisation parameter and the isotropic spread parameter.

ii. Improvement in the metamodel process

In the development of the metamodel, three important stages are involved: preparing the data and choosing the modelling approach; parameter estimation and training; and model testing and validation. In neural network training, the preparation of a trial dataset is crucial as it will directly influence the metamodel performance. Thus, the appropriate data sampling strategy is necessary especially when dealing with a large dataset. In this work, a new sampling strategy will be introduced, based on the distance measure and the clustering process. For instance, the proposed strategy uses a distance weight function to measure the normalised distance for all the input-output data points, and followed by clustering to *n* numbers of sampling frequency by using *k*-means theory. The proposed strategy is benchmarked with some available techniques such as *n*-level Full Factorial Design and Latin Hypercube Design, and the results show that in certain conditions, it outperforms the rest in terms of several criteria, which are the performances indexes, the network size and the computation time.

1.5. Thesis structure

This thesis consists of nine chapters. Chapter 1 provides the background of the air pollution impacts and the importance of the air quality estimations and predictions. Next, the problem statement, the objectives of the thesis, and the significance of the work and contributions, are outlined.

In Chapter 2, the literature review related to this research is presented. It begins with an overview of the available air quality modelling techniques in atmospheric studies, followed by the description of air pollutant measurements and predictions, including an explanation of the background ozone level theories. Next, this chapter describes the metamodelling approach which includes the variation of techniques in the literature and the sampling strategies. This is followed by a description of the radial basis function neural network metamodel including the various learning strategies and types of basis function.

Chapter 3 presents the process flow for constructing the metamodel specifically using the neural network approach, which consists of the data preparation, the model training, and the model testing and validation. A proposed data sampling scheme will be described which features the weighted and clustering design (WCD) method. The reliability of the method is identified by comparing several performance criteria with the available methods.

Chapter 4 describes the radial basis function neural network paradigm and a typical training algorithm, followed by some proposed improvements which include a method for an adaptively-tuned spread parameter and a pruning algorithm to optimise the number of hidden neurons in the network. The significant improvements in its performance are evaluated using some test functions.

Chapter 5 focuses on the proposed new training algorithm for the radial basis function neural network, featuring the generalisation network with regularised forward selection and weighted least-squares (GRFSWLS) for the basis centre selections, a method to train the network output weights, and some suggestions on the determination of the radial basis function neural network parameters.

In Chapter 6, an overview of the applied domain will be described, followed by an explanation of some tools and measurement methods for the dataset collection in this work. The data collection involves several methods, including the measurement of the air pollutant at monitoring stations by using some special instruments, the collection of the emission rates dataset of the pollutant sources from the Emission Data Management System (EDMS), and the extraction of some input-output dataset from the air quality model.

Chapter 7 covers the proposed approach in the determination of the background ozone level in which the methodology is generic for use in any considered location. Several quantisation methods regarding this approach are explained which are the non-photochemical background ozone level concept and a technique to determine the suitable time range for the night-time background ozone level. The effectiveness of each strategy will be validated and compared with other available approaches.

Chapter 8 discusses the metamodel application in the estimation of ozone and the background ozone level, temporally and spatially. Each part begins with the description of the data pre-processing which includes the selection of the suitable input-output variables and the data sampling, and followed by the metamodel function estimation, and model validation and testing. An analysis of its outputs performance and the future trend analysis related to the atmospheric field will be discussed.

Finally, Chapter 9 presents the summary of the results drawn from this work, conclusion reached and the recommendations for future research. The last section comprises a bibliography and also appendices containing the program codes and relevant supporting documents.

Chapter 2
LITERATURE REVIEW

2.1. Introduction

The primary goal of all air pollution control programs is to protect human health and the environment from adverse effects of air pollutants. Several guidelines and standards to achieve the air quality management (AQM) programs' goal appear in several documents (e.g. EU, 1999, 2000, 2002; WHO, 2005; US-EPA, 2006b). Air quality may be classified easily by qualitative interpretation, for example 'poor' when pollutants cause (say) a decrease in visibility, and 'good' when the sky appears clear. However, qualitative assessments cannot be used to support regulatory programs designed to protect the environment. Therefore, a quantitative air quality assessment needs to be conducted, having in general three different approaches, namely, air quality monitoring, emissions inventory and assessment, and air quality modelling. Each has its usefulness, in terms of temporal and spatial aspects to the policy maker for the understanding of the nature of air pollution due to various sources in the urban setting.

This chapter reviews the methodological approaches undertaken in the literature to the present time to provide a systematic assessment of AQM, an overview of the main air quality issue that will be addressed in this research which is ozone and background ozone levels, surveys on the variations of air quality models used in the literature in air quality study, a review of the metamodel approach which is the proposed alternative way for air quality estimation, and finally, a review of the radial basis function neural network technique as the metamodel's approximation function.

2.2. Overview of air quality assessment

2.2.1. Three approaches in air quality assessment

2.2.1.1. Air quality monitoring

One effective approach to assess air quality is through the development of an Ambient Air Monitoring Program. In general, air quality samples are collected for one or more of the following purposes (Godish, 2004; US-EPA, 2011a):

- To judge compliance with ambient air quality standards;
- To observe pollution trends (i.e. short-term and long-term) throughout the region (i.e. urban and non-urban areas);
- To support the Air Quality Index (AQI) program;
- To support emissions reduction programs;
- To determine the effectiveness of emission control programs;
- To support research efforts designed to determine potential associations between pollutant levels and adverse health and environmental effects.

Due to the vastness of the atmosphere, it is not possible to evaluate each of the individual pollutants in the program. At the present time, the 'pollutant criteria' (or the six/or seven key pollutants) are chosen in most of the air quality monitoring programs worldwide (e.g. EU 1999, 2000, 2002; DECC, 2007a; US-EPA, 2011b). In general, pollutant concentrations are collected in the monitoring stations (e.g. Fig. 2.1) whether in or on a sampling medium or in automated continuous systems, where they are drawn through a sensing device and concentrations are measured in real time.

For example, in the United States, all air quality monitoring activities must use methodologies approved by the US-EPA as reference methods, namely as Federal Reference Methods (FRMs). A comparison of approved measurement methods for the criteria pollutants between United States and Australia are summarised in Table 2.1. The details of each measurement method will not be described in this thesis (except for ozone and oxides of nitrogen in Chapter 6).

Fig. 2.1 An example of air quality monitoring station located in Sydney, Australia. (Photo courtesy of Department of Environment and Climate Change, DECC).

Table 2.1 A comparison of air quality reference methods used in U.S. and Australia monitoring stations.

Pollutant	Reference method	
	FRMs in U.S. (US-EPA, 2011c)	**Australia (DECC, 2007c)**
Sulphur dioxide	Spectrophotometry	fluorescent spectrophotometry
Nitrogen dioxide	Gas-phase chemiluminescence	chemiluminescence
Carbon monoxide	Non-dispersive infrared photometry	infrared spectrometry
Ozone	Chemiluminescence	ultraviolet spectroscopy
Total non-methane hydrocarbons	Gas chromatography – FID	N/A
Fine particles – PM_{10}	Performance-approved product	tapered element oscillating microbalance (TEOM)
Fine particles – $PM_{2.5}$	Performance-approved product	
Lead	Total Suspended Particulates (TSP)	Total Suspended Particulates (TSP)

Monitoring network requirements

For success in satisfying the purposes of the monitoring program, the network should be designed to meet one of the following basic monitoring objectives:

1. To determine the highest concentrations expected to occur in the area covered by the network;
2. to determine the representative concentrations in areas of high population density;
3. to determine the impact on ambient pollution levels of significant sources or source categories; and
4. to determine the general background concentration levels.

These four objectives indicate the nature of the samples that the monitoring network will collect which must be representative of the spatial area being studied. Thus, in establishing the monitoring station sites, the spatial scales are typically used by the authority to make this decision. Basically, spatial scales are estimates of the sizes of areas around monitoring locations that experience similar pollutant concentrations. Spatial scale categories are: (1) microscale, ranges from a few metres to 100 m; (2) middle scale, ranges from 100 m to 0.5 km; (3) neighbourhood scale, ranges from 0.5 to 4.0 km; (4) urban scale, ranges from 4 to 50 km; and (5) regional scale, ranges from tens to hundreds of kilometres.

Basic references for spatial scale determination are given as follows: spatial scales for the highest concentration or source impact are micro-, middle, neighbourhood, and, less frequently, urban scales; spatial scales for high population densities are middle, neighbourhood, and urban; neighbourhood or regional scales are appropriate for background levels; and urban and regional scales are also appropriate for determination of pollutant transport in remote areas (Godish, 2004).

Averaging periods

The determination of the averaging time is dependent on the sampling durations required to collect samples, and the intended use of the data. For example, one-hour averages are used for short-term evaluation, while 24-hour averages are appropriate for long-term trends. Data from real-time monitoring instruments are able to provide hourly average concentrations or concentrations reflective of the needs of air quality standards (e.g. NAAQS, a standard used in United States; and NEPM, a standard used in Australia). For pollutants such as O_3, where peak levels occur for a limited time period, one-hour and eight-hour averaging times are employed.

2.2.1.2. Emission inventory and assessment

An air emissions inventory is a detailed listing of pollutants discharged into the atmosphere by each source type during a given time period at a specific location. A complete inventory typically contains emission sources that correspond to all the regulated pollutants in air quality standards. Emission inventories are required in the air quality management process for the following reasons:

1. to help determine significant sources of air pollutants;
2. to establish emission trends over time;
3. to formulate emissions control policy;
4. to comply with permitted requirements;
5. to compile national annual emission inventories; and
6. to provide a database for ambient air quality modelling.

Several methods for calculating the emission inventories are available, which may include, but are not limited to: continuous monitoring to measure actual emissions; using stack sampling procedures for gases; extrapolating the results from short-term source emissions tests; and combining published emission factors with known activity levels. An emission factor may be used to estimate emissions when actual emission data is not available. In most cases, these factors are simply averages of all available data of acceptable quality, and are generally assumed to be representative of long-term averages for all facilities in the source category (US-EPA, 2011d).

Typically, the inventory includes emissions derived from biogenic (i.e. natural) and anthropogenic (i.e. human) sources as outlined below (DECC, 2007b):

- Biogenic (e.g. bushfires, trees and windborne dust);
- Commercial businesses (e.g. quarries, service stations and smash repairers);
- Domestic activities (e.g. house painting, lawn mowing and wood heaters);
- Industrial premises (e.g. oil refineries, power stations and steelworks);
- Off-road mobile (e.g. aircraft, railways and recreational boats);
- On-road mobile (e.g. buses, cars and trucks).

The pollutant emission sources can be categorised into three types: (1) criteria pollutant emissions (e.g. carbon monoxide (CO), lead, oxides of nitrogen (NO_x), PM_{10}, $PM_{2.5}$, sulphur dioxide (SO_2) and volatile organic compounds (VOCs)); (2) metal air toxics (e.g. antimony, arsenic, beryllium, chromium and nickel); and (3) organic air toxics (e.g. benzene, formaldehyde, polycyclic aromatic hydrocarbons (PAHs), toluene and xylenes).

Some guides for the preparation of emission inventories, which include purpose, process, methodology and the application of emission inventory investigations, appear in several documents, such as for example in Europe (as in EEA, 2009), in the U.S. (as in US-EPA, 2012), in Australia (as in DECC, 2007c), and in New Zealand (as in ME, 2001).

2.2.1.3. Air quality modelling

Due to the limitation of the resources and practical implementation, an alternative approach other than air quality monitoring is necessary to approximately estimate the distribution of the pollutants, temporally and spatially. Air quality (AQ) models (also known as air dispersion models) are tools that are capable of addressing the limitations in extent to which their use provides a relatively inexpensive and reliable means of determining compliance with air quality standards and the thus the extent of emissions reduction necessary to achieve the required standards. They are widely used by regulatory authorities as surveillance tools to assess the effect of emissions on ambient air quality.

Generally, AQ models are mathematical descriptions of the atmospheric transport, diffusion, and chemical reactions of pollutants' sources (Duc & Azzi, 2009). They consist of one or more mathematical formulae that include parameters that affect concentrations of pollutants at various distances downwind of emission sources. Typically, they operate on sets of input data that characterise the emissions, meteorology, and topography of a region and produce outputs that describe that basin's air quality.

AQ models can be classified in several ways, based for example, on short-term or long term models; according to chemical reactions; according to the type of coordinate system used; or whether the model is simple or advanced (Godish, 2004). Short-term models are used to calculate concentrations of pollutants over a few hours or days, which can be employed to predict worst case episode conditions and are used by regulatory agencies as a basis for control strategies. Long-term models are designed to predict seasonal or annual average concentrations, which may prove

useful in studying atmospheric deposition as well as potentially adverse health effects associated with pollutant exposures.

Models can be described as simple or advanced based on assumptions used and the degree of sophistication with which important variables are treated. Advanced models have been developed for photochemical air pollution, dispersion in complex terrain, and long-range transport. Simpler models like the Gaussian are widely used to predict the impact of emissions of relatively non-reactive gas substances such as SO_2 and CO, as well as particulate matter on air quality downwind of point sources. The background of the Gaussian model will be described further in this review as its theory will be utilised in this work (see Chapter 8).

2.2.2. Air quality standards and air quality index

Air quality standards

In the United States, from the requirement of the *Clean Air Act 1990*, the US-EPA has to set National Ambient Air Quality Standards (NAAQS) for pollutants, which are considered harmful to public health and the environment. The *Clean Air Act 1990* identifies two types of national ambient air quality standards; *primary* and *secondary standards* (US-EPA, 2011e). *Primary standards* provide public health protection, including protecting the health of 'sensitive' populations such as asthmatics, children, and the elderly. *Secondary standards* provide public welfare protection, including protection against decreased visibility and damage to animals, crops, vegetation, and buildings. US-EPA has set National Ambient Air Quality Standards for six principal pollutants, which are called "criteria pollutants".

In Europe, the European Union air quality management organisations use Air Quality Limit Values (AQLVs) as their standard, while in Australia they use the National Environment and Protection Measures (NEPM), regulated in 1998, as the national air quality standards. In 2003 the NEPM was amended to include advisory reporting standards for particles as $PM_{2.5}$ (NEPC, 2003). A comparison of three air quality standards for six criteria pollutants is shown in Table 2.2. Therein, to normalise the unit of measures for the standards, parts per million (ppm) by volume

and micrograms per cubic metre of air ($\mu g/m^3$) are used. In general, there are some similarities on the set standards, with the EU standards showing the tightest concentration limits for most of the criteria pollutants.

Table 2.2 Table comparing U.S., EU and Australia air quality standards for criteria pollutants.

Pollutant		Averaging period	Maximum concentration		
			U.S.	EU	Australia
Carbon monoxide		1 hour	35.00 ppm	NA	NA
		8 hours	9.000 ppm	8.700 ppm	9.000 ppm
Nitrogen dioxide		1 hour	0.100 ppm	0.098 ppm	0.120 ppm
		1 year	0.053 ppm	0.020 ppm	0.030 ppm
Ozone		1 hour	0.120 ppm	NA	0.100 ppm
		8 hours	0.075 ppm	0.061 ppm	0.080 ppm
Sulphur dioxide		1 hour	0.075 ppm	0.133 ppm	0.200 ppm
		1 day	NA	0.048 ppm	0.080 ppm
		1 year	NA	NA	0.020 ppm
Lead		3 months	0.15 $\mu g/m^3$	NA	NA
		1 year	NA	0.50 $\mu g/m^3$	0.50 $\mu g/m^3$
PM	PM_{10}	1 day	150 $\mu g/m^3$	50 $\mu g/m^3$	50 $\mu g/m^3$
		1 year	NA	40 $\mu g/m^3$	NA
	$PM_{2.5}$	1 day	35 $\mu g/m^3$	NA	25 $\mu g/m^3$
		1 year	15 $\mu g/m^3$	25 $\mu g/m^3$	8 $\mu g/m^3$

* NA = Not available; 1 ppm = 1000 ppb.

Air quality index (AQI)

AQI values are derived from air quality data readings, which allows for more meaningful comparison of pollutants affecting air quality. The index is derived using the following formula (i.e. based on the practice in Australia):

$$AQI_{pollutant} = \frac{Pollutant\ \ data\ \ reading}{Pollutant\ \ standard} \times 100. \qquad (2.1)$$

In general, data readings are translated on to a linear scale based on relevant air quality standards to derive the AQI values for the *hourly AQI* and *daily AQI*. The maximum of individual pollutant indexes at a monitoring station is then taken as the overall index for that station. An index is then assigned to one of six colour-coded air quality categories as depicted in the diagram in Fig. 2.2. Therein, an AQI of 100 corresponds to the relevant air quality standard for criteria pollutants. Hence, if the AQI is reported as Poor, Very Poor or Hazardous, it indicates that the determining pollutant levels have reached or exceeded the relevant standard or goal.

Excellent 0 – 33	Good 34 – 66	Fair 67 – 99	Poor 100 – 149	Very poor 149 – 199	Hazardous ≥ 200

Fig. 2.2 Air quality categories based on the AQI values.

2.2.3. Air quality model review

The development of the mathematical model and prediction tools for air pollution started in the early 1970s and further research and application has growing extensively in the last two decades. From the beginning of the 21st century, more sophisticated approaches have been developed owing to advances in computer simulation. Some of the early surveys for the performance of air quality models were undertaken by Tesche (1983), Roth et al. (1989), Blanchard (1999) and Vardoulakis et al. (2003). As has been discussed in section (2.2.1.3), the air quality models can be classified in a few categories. In this review, we categorise the models in a more generic way based on their execution strategies, falling into two classes; deterministic models, and statistical models.

2.2.3.1. Deterministic model

This type of model is based on the physic laws and generally its structure is quite complex, thus its development requires detailed knowledge on a large number of chemical reactions and its mathematical equation. Several approaches appeared in the literature, which include the Eulerian (i.e. grid) model, Lagrangian (i.e. trajectory) model, and also the Gaussian based model.

Eulerian model

The Eulerian (grid) type model simulates the atmosphere for a certain region by dividing it into thousands of individual grid cells that are typically a few kilometres wide. The transport, diffusion, transformation, and deposition of pollutant emissions in each cell are described by a set of mathematical expressions in a fixed coordinate system, which normally involve partial differential equations. The model calculates the concentration of pollutants in each cell by considering the air dispersion effects in each cell, the combination of the pollutants upward and downward in the layers and the volume of emissions from pollutant sources in each cell. Therefore, due to

the huge number of interactions involved, this model is normally expensive to be maintained and run. Typical domain dimensions for different scale models are given in Table 2.3 (Srivastava & Rao, 2011).

Table 2.3 Typical domain dimensions for different scale models.

Model	Typical domain scale	Typical resolution
Micro scale	200×200×100 (m)	5 m
Mesoscale (urban)	100×100×5 (km)	2 km
Regional	1000×1000×10 (km)	36 km
Synoptic (continental)	3000×3000×20 (km)	80 km
Global	6500×6500×20 (km)	4°×5°

The U.S. Environmental Protection Agency (US-EPA), in EPA guidelines (US-EPA, 1986) has suggested using the Urban Airshed Model (UAM) for ozone studies over urban areas. It is a grid-based model that uses a fixed Cartesian reference system within which to describe atmospheric dynamics. Some examples of applications of this model are reported by Whitten et al. (1986), and Chang and Rudy (1989). The Comprehensive Air-quality Model with extensions (CAMx) is a photochemical grid model developed in the late 1990s to treat urban and regional scale air quality problems using the one-atmosphere concept (Morris et al., 2000). The model is an ideal platform for extension to treat a variety of air quality issues including ozone, particulate matter (PM), visibility, acid deposition and air toxins.

In 1998, the US-EPA released the first version of the Community Scaled Air Quality (CMAQ) model, and it has been used extensively to evaluate potential air quality policy management decisions in the U.S. (US-EPA, 2001; Byun & Schere, 2006). This model is also known as 'Models-3' as it consists of three models – a meteorological model, an emission model, and a chemical transport model – very useful for long-term trend analysis and reporting. Another popular model is GEOS-CHEM which has been developed by a group of researchers at Harvard University. This model is a global 3-D model of atmospheric composition driven by assimilated meteorological observations from the Goddard Earth Observing System (GEOS) of the NASA Global Modeling and Assimilation Office. A number of researchers have applied this model in many atmospheric studies (e.g. Fusco & Logan, 2003; Liu et al. 2006; Nassar et al. 2009; Protonotariou et al. 2010).

In Australia, the development of air quality models is mostly carried out by the Commonwealth Scientific and Industrial Research Organisation (CSIRO). The air pollution model and chemical transport model (TAPM-CTM) are commonly used gridded models in Australia for the air quality regulatory program (e.g. Luharrr & Hurley, 2003). It is a complex model featuring three-dimensional and prognostic meteorological elements incorporated by the chemical transport mechanisms.

Lagrangian model

The Lagrangian model approach is based on the calculation of wind trajectories and on the transportation of air parcels along these trajectories. In the source oriented models, the trajectories are calculated forward in time from the release of a pollutant-containing air parcel by a source until it reaches a receptor site. In a receptor oriented model, the trajectories are calculated backward in time from the arrival of an air parcel at a receptor of interest. Numerical treatment of both backward and forward trajectories is the same, and the choice of use of either method depends on the specifics of the case. The major disadvantage of the method is the assumption that wind speed and direction are constant throughout the Physical Boundary Layer. However, as compared to the Eulerian models, the Lagrangian model can save computational cost as they perform computations of chemical and photochemical reactions on a smaller number of moving cells instead of at each fixed grid cell of the Eulerian model. Versions of EMEP (European Monitoring and Evaluation Programme) are examples of the Lagrangian model.

Gaussian models

Gaussian models are among the oldest and most preferred of the models which have been used in the U.S. since the mid–1960s. They depend on the availability of realistic physical data of wind and diffusion. A large number and variety of current air transport models rely on the basic Gaussian equation. This approach can be especially suitable for non-reactive pollutants.

The Gaussian model assumes that a plume travelling downwind will gradually expand and disperse. For example, Fig. 2.3 shows a stack emitting pollutants that are carried downwind (in the x direction). As the plume travels further downwind it

expands in both the *y* (crosswind) and *z* (vertical) direction. Due to dispersion in the *y* and *z* direction, the plume always has its highest concentration in the centre of the plume and the lowest concentrations at the edges of the plume. The Gaussian model assumes that these concentrations can be described by a normal distribution (i.e. using the Gaussian function), which is given as a bell-shaped curve. The size of the plume is characterised by the standard deviation of the concentrations in the plume.

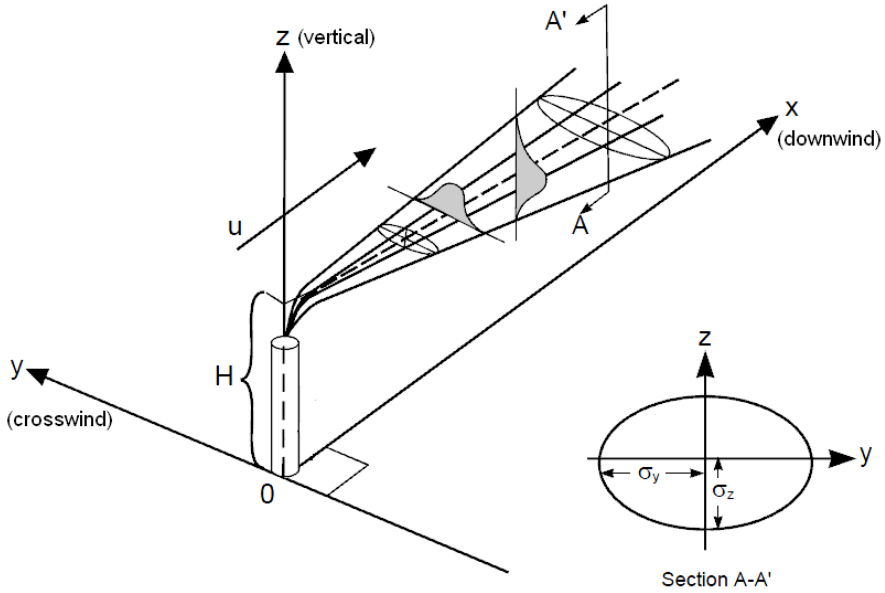


Fig. 2.3 Schematic representation of Gaussian plume model.

Under stable atmospheric conditions or unlimited vertical mixing, ground-level concentrations can be calculated from equation (2.2) (Turner, 1970; Pasquill, 1976):

$$C(x,y,z,H) = \frac{Q}{2\pi\sigma_y\sigma_z u} e^{-\frac{1}{2}\left(\frac{y}{\sigma_y}\right)^2} \left\{ e^{-\frac{1}{2}\left(\frac{z-H}{\sigma_z}\right)^2} + e^{-\frac{1}{2}\left(\frac{z+H}{\sigma_z}\right)^2} \right\}, \qquad (2.2)$$

where: C = ground level pollutant concentration (g/m^3),

Q = emission rate (g/s),

σ_y = standard deviation of pollutant concentration in y direction,

σ_z = standard deviation of pollutant concentration in z direction,

u = wind speed (m/s),

y = distance in crosswind direction (m),

z = distance in vertical direction (m), and

H = effective stack height (m).

The values of σ_y and σ_z depend on how far the plume has travelled in the x direction. In other words, as the plume travels further downwind, the plume will grow in width and height. Values of σ_y and σ_z have been determined empirically by plume studies under previous environmental conditions. The values depend on many variables, and especially on the stability of the atmosphere. Stability classes are based on solar radiation, surface wind speed, and cloud cover, which are normally rated from A to F, where A is the least stable and F is the most stable of environments (as given in Table 2.4). An A stability would result in a plume that is widely dispersed in the y and z direction, thus resulting in lower average concentrations at any given distance. For instance, sigma values can be determined roughly from the Pasquill's dispersion coefficient graphs, which include crosswind and vertical graph (Turner, 1970).

Table 2.4 Atmospheric Stability classes (Turner, 1970).

Wind speed at 10 m (m/s)	Day			Night	
	Strong	Moderate	Slight	> 4/8 Cloud	< 3/8 Cloud
< 2	A	A-B	B	E	F
2 – 3	A-B	B	C	D	E
3 – 5	B	B-C	C	D	D
> 6	C	D	D	D	D

Some examples of air pollution dispersion models (i.e. Gaussian dispersion models) in current use are listed as follows:

- ADMS 4: Developed in the United Kingdom, (http://www.cerc.co.uk/environmental-software/ADMS-model.html)
- AUSPLUME: Developed in Australia, (http://www.epa.vic.gov.au/air/epa/ausplume-pub391.asp)
- CALPUFF: Developed in the United States, (http://www.src.com/calpuff/calpuff1.htm)
- RIMPUFF: Developed in Denmark, (http://www.risoe.dtu.dk/business_relations/products_services/software/vea_dispersion_models/rimpuff.aspx?sc_lang=en)

2.2.3.2. Statistical / Empirical model

Various statistical models (also known as a 'black box model' or 'empirical model') may also be used for air pollution modelling, which have the advantage of simplicity in implementation. This model is based on establishing a relationship between historically observed air quality and the corresponding emissions. There are two main statistical models that are specially developed in air quality modelling: the linear roll back model and the receptor model.

The linear rollback model is the simplest and easiest to use, which was originated with *U.S. Clean Air Act 1990*. The linear relationship is given as follows (CHNPWA, 1993):

$$C_i = B + \sum_i k_i E_i \,, \tag{2.3}$$

where: C_i = future concentration at point *i*,

B = background level of pollutant,

E_i = emission for point *i*, and

k_i = proportionally factor for point *i*.

In the model, it is assumed that emissions outside the region of interest and natural sources, even inside the region, are usually included in this background term. The constant of proportionality k_i, is determined over a historical time period when other values (i.e. C_i, B and E_i) are known. A flaw with this approach is that these equations are valid only for the prevailing conditions of sources and emission levels. If another source is introduced, all the proportionality factors may have to be recalculated. As such, these models may be acceptable for screening air control strategies.

The receptor-oriented model is the apportionment of the contribution of each source, or group of sources, to the measured concentrations without considering the dispersion pattern of the pollutants. The starting point of the model is the observed ambient concentrations at receptors and it aims to apportion the observed concentrations among various source types based on the known chemical fractions of source emissions. Mathematically, the receptor model can be generally expressed

in terms of the contribution from *n* independent sources to *p* chemical species in *m* samples as follows:

$$C_{ik} = \sum_{j=1}^{n} a_{ij} f_{jk}\,, \tag{2.4}$$

where C_{ik} is the measured concentration of the k^{th} species in the i^{th} sample, a_{ik} is the concentration from the j^{th} source contributing to the i^{th} sample, and f_{jk} is the k^{th} species fraction from the j^{th} source (Srivastava & Rao, 2011).

Other statistical approaches are also used, perhaps the most typical being regression analysis. For example, Abdul-Wahab et al. (1996) presented the functional relationship between ozone level and various independent variables by using a stepwise multiple regression modelling procedure. The inputs and outputs of the analysis have been derived from the ambient measurement of air quality data to construct a linear input-output model from the dataset.

In 1986, US-EPA used an Empirical Kinetic Modelling Approach (EKMA) to estimate the percentage of precursor reductions needed to reach the NAAQS requirement of ozone (US-EPA, 1986). It is a one-dimensional model which requires a VOC/NO_x ratio as the input, derived at an upwind location during the peak ozone concentrations. Duc et al. (2000) used a Kriging approach to study the spatial correlation of SO_2, NO, NO_2 and O_3 over a long-distance network in Sydney, Australia. This method used a variogram model, which considers the non-isotropic interpolation of the measurement from each monitoring station. They found that within a 30 km radius, this method showed a reasonable correlation for some air pollutants, but not likely for ozone due to the non-linearity and complicacy of its formation.

Soft computing based on artificial intelligence (AI) can serve as an alternative in environmental science studies. In climate control, Trabelsi et al. (2007) implemented a fuzzy clustering technique to model air temperature and humidity inside a greenhouse to increase crop production. More recently, Fazel Zarandi et al. (2012) used the type-2 fuzzy logic theory to construct a model for the prediction of carbon

monoxide in Tehran, Iran. The application of fuzzy logic approach has also appeared in ozone studies (see e.g. Gomez et al., 2001; Heo & Kim, 2004; Mintz et al., 2005).

In air quality research, neural networks have been successfully applied to model some air quality predictions, mainly in forecasting air pollutant concentrations. Wang et al. (2003b) used combined adaptive radial basis function networks with statistical characteristics to predict daily maximum ozone concentrations at selected areas. The approach presented good results with some limitation on the prediction in the rural areas. (See other examples of pollutants' prediction using neural networks in Boznar et al., 1993; Sousa et al. 2005; Coman et al., 2008; Zainuddin & Pauline, 2011).

Furthermore, a neural network approach is also capable of expressing the source-receptor relationship. Carnevale et al. (2009) used a neural network and a neuro-fuzzy model to estimate the non-linear source-receptor relationship for ozone and PM_{10} concentrations. They utilised input-outputs from a deterministic model to develop the source-receptor models. The models produced accurate estimations as compared to the deterministic model, however they did not show any validation results with the actual sites' measurement data, and thus the real accuracy became questionable. Pfeiffer et al. (2009) used diffusive sampling measurements and a neural network to calculate the average spatial distribution of NO_2 pollutant in Cyprus. A pre-processing step is executed by embedding the measured pollutant level from the diffusive sampler, the wind information, the local emission sources and the population density. However, large numbers of the diffusive samplers are required to generate the correct spatial map of the pollutant (e.g. they used 270 samplers at 270 sites).

2.3. Review of ozone and background ozone level

2.3.1. Ozone and its determination

Definition

Ozone (O_3) is a gas that is naturally produced and present in the earth's atmosphere. Most of the ozone resides in the upper part of the atmosphere about 15 kms to 50 kms from the earth's surface, called the stratosphere, which contains 90% of the ozone layer. This type of ozone is very useful to earth's inhabitants preventing harmful effects from the ultraviolet rays of the sun. The remaining 10% is called tropospheric ozone (also known as surface ozone), located between the earth's surfaces and the stratosphere layer. Tropospheric ozone arises from several sources; a little contribution from the migration of stratospheric ozone, chemical reactions of the natural gases and mostly from reactions with human made pollutant gases. Some of the surface ozone is beneficial where it could help to remove pollutants from the atmosphere. In contrast, the excess of ozone exposure may yield harmful effects rather than benefits to the living organisms because of damage to living tissue, it may break down certain materials and might also contribute to the warming of the earth's surface. The scientists call this the 'bad ozone' as it directly affects humans, animals and plants (Fahey, 2006).

Ozone is a secondary pollutant that is formed from the photochemical reaction of nitrogen oxides, NO_x gas (i.e. NO_x is a combination of nitric oxide, NO gas, and nitrogen dioxide, NO_2 gas), and volatile organic compounds (VOCs) in the presence of solar radiation. Mathematically it is represented in the following reaction:

$$NO_x + VOCs \xrightarrow{\text{solar radiation}} O_3 . \qquad (2.5)$$

The emission may arise from natural (biogenic) sources and man-made sources. The NO_x is produced naturally from biomass burning, lightning and fertilised soils, while it is artificially produced mainly from the combustion of fossil fuels, the major sources including motor vehicles and electricity generation stations. The VOCs in the atmosphere are produced by emissions from motor vehicle exhausts, the chemical and petroleum industries, and the use of solvents.

Determinations

It has been reported that surface ozone is the most important index substance of photochemical smog and one of the key pollutants that lower air quality (Seinfeld, 1989). Thus, measurement, prediction and assessment of the ozone level are

important to the implementation of a long-term plan for improving community health.

The process of ozone formation is complex and its investigation involves an understanding of the highly non-linear photochemical reactions, the sources of ozone precursor emissions, and the meteorological conditions conducive to ozone formation. The rate of production of ozone depends largely on the temperature and the ratio of the precursor pollutants (VOC:NO_x). It has been determined that ozone concentrations are typically elevated during the warmer months, for example during summer and the beginning of autumn seasons (i.e., as reported for the Sydney basin).

Some of the works focus on the temporal prediction of the future ozone concentrations and their trends. The assessment may be accomplished by using air quality models with either deterministic or statistical approaches. Several works on ozone prediction have been presented in the previous section (2.2.3). A more difficult task is to estimate the spatial distribution in the region of interest, where the results may offer a significant indication to the authority to manage the air quality, e.g. by considering the precursor emission reductions of ozone. It was suggested that to essentially reduce ozone concentrations in many urban and suburban areas, the control of NO_x emissions will probably be necessary in addition to controlling the emission of VOCs. Many of the air quality models have determined that ozone is predicted to decrease in response to NO_x reductions in most urban locations.

2.3.2. Background ozone level and its determination

Definitions

The background ozone level is referred to as the ozone level that is biogenically formed, i.e. from natural processes free from anthropogenic influences (Duc & Azzi, 2009), and which occur in the troposphere layer. Unfortunately, the accurate determination of the background ozone level is demanding as it requires a clean environment and free from these anthropogenic influences. The background ozone level is mostly from tropospheric natural sources but may also be of stratospheric

origin transported down to the surface.

Another definition by US-EPA uses the Policy-Relevant Background (PRB) as the reference to the background ozone concentration (US-EPA, 2006a). The PRB referred to the concentration that would occur in the United States in the absence of anthropogenic emissions in continental North America (defined here as the United States, Canada, and Mexico). PRB concentrations include contributions from natural sources everywhere in the world and from anthropogenic sources outside these three countries. The estimated PRB ozone concentrations are shown to be dependent on the season, altitude and total surface ozone. In the 1996 ozone review, the EPA used 4 pphm as the eight-hour daily maximum background ozone level in its health risk assessment evaluations.

Using several techniques, Altshuller and Lefohn (1996) determined that the current ozone background at inland sites in the United States and Canada for the daylight 7-hours (9.00 am to 3.59 pm) seasonal (April to October) average concentrations, usually lie within the range of 35±10 ppb. For coastal sites located in the northern hemisphere, the corresponding ozone concentrations are lower, occurring within the range of 30±5 ppb. These ranges suggest that the background ozone is somewhat dependent on a number of conditions such as the nature of upwind flow and terrain conditions, including deposition with respect to forest or agricultural areas.

Determinations

There have been many efforts to try to define the background ozone level. Clean pristine sites in rural or bushland areas, typically at the boundary of the evaluated region, have been cited as good sites to measure the background ozone. For example, Oltmans et al. (2008) measured ozone levels, which they indicated and used as background ozone, at some remote sites in California (such as Trinidad) where the airflow patterns from the Pacific Ocean are almost free of contamination from continental North America. Their measurement shows that the monthly background ozone fluctuates around 35 parts per billion (ppb) for the six-year period (2002–2007) with a maximum about 50 ppb and a minimum about 25 ppb (see also e.g. Matveev et al., 2002; Saitanis, 2003). That apart, upon reviewing the ozone

trend in Europe by measuring ozone at pristine sites, Monks (2003) has concluded that there is strong evidence that background ozone concentrations are increasing in western and northern Europe.

Basically, direct measurement of ozone does not accurately indicate the background level, thus some quantisation approaches have to be incorporated. According to Parish et al. (2009), background ozone is used to qualitatively describe ozone (O_3) mixing ratios measured at a given site in the absence of strong local effects. To quantify the background ozone for the North American west coast, they used marine air ozone as measured at coastal sites in which continental influences are removed by examining wind data, trajectory and tracers. In another method, to quantify the 'baseline ozone' across the North American continent, Chan and Vet (2010) used ozone data measured at non-urban sites, principal component analysis to group sites forming specific geographic regions for baseline ozone, and backward air parcel trajectory clustering to result in six trajectory clusters of air parcels for each site. The baseline trajectory cluster was chosen as the one having the lowest 95th percentile ozone among the six clusters. According to them, the 95th value is predominately associated with long-range transport for remote locations, while lower percentile values are affected by dry deposition and NO scavenging.

Recently, various software packages have been used for air quality models to estimate the background ozone level. For example, Fiore et al. (2003) used a three-dimensional global model with chemistry (GEOS-CHEM) to assess the background ozone level in the United States. Their study has been included in the US-EPA report (US-EPA, 2006a). The estimated policy-relevant background (PRB) ozone concentrations are shown to be dependent on season and altitude with an estimated range of 30 to 50 ppb for typical summertime BOL of the total surface ozone concentration. However, the estimation performance by modelling is much dependent on the level of uncertainty of the software and the availability of accurate biogenic emission data.

2.4. Review of metamodelling techniques

Metamodelling has been a major research field since the last decade, where the objectives are to obtain a simpler model from a complex model, to approximate the non-linear system behaviour, and to reduce the cost, time, and amount of effort required during a simulation. Some reviews for performance comparison of various metamodelling techniques have been presented (see e.g., Jin et al., 2001; Fang et al., 2005). Substantial results from the existing works illustrate that using metamodels to locate an optimum solution is often sufficiently accurate in many applications requiring prediction, optimisation and validation (Tunali & Batmaz, 2003).

2.4.1. Types of metamodel

Metamodelling evolved from the classical Design of Experiments (DOE) theory, in which polynomial functions are used as response surfaces, or metamodels. This review will briefly survey some of the metamodelling techniques available recently, beginning with traditional response surface methodology (i.e. polynomial regression), followed by some alternative approaches.

Polynomial regression (PR) metamodel

Polynomial regression (PR) is also known as response surface methodology (RSM), however, since the 'response surface' term was also referred to as 'metamodels' in the earliest stages, the term polynomial regression is more often used. The PR method has been used since before the computer era, thus it is mathematically simple and typically consists of first to second order regression functions. It has been used effectively for over thirty years as metamodels with several applications shown by a number of researchers (e.g. Unal, et al., 1996; Chen et al., 1996; Simpson, et al., 1997; Xie et al., 2008) in designing complex engineering systems. For example, the second order PR can be expressed as:

$$\hat{y} = \beta_o + \sum_{i=1}^{N} \beta_i x_i + \sum_{i=1}^{N} \beta_{ii} x_i^2 + \sum_{i=1}^{N} \sum_{j=1}^{N} \beta_{ij} x_i x_j \,, \qquad (2.6)$$

where $\hat{y}$ is the approximation function at x, N is the number of design variables, and β_o, β_i, β_{ii} and β_{ij} are the regression coefficients determined by the linear least square regression analysis.

Due to its simplicity in implementation, and ease of understanding, PR is still used quite often in industry and academia. However, it has a drawback when applying it to model with highly nonlinear behaviours. To overcome this limitation, higher-order polynomials can be used, but some instability conditions may arise (Barton, 1992).

Kriging metamodel (KG)

The Kriging metamodel was invented by Georges Matheron in 1971 in the area of geostatistics and named after the South African geologist, Danie Krige (Gano et al., 2006). Kriging is one of the global metamodels which has been widely used since the beginning of the computer era, and in general has been an accurate metamodel (e.g., Jin et al., 2001; Wang & Shan, 2007). It was first used in metamodelling by Currin et al. (1988) who termed 'design and analysis of computer experiments' (DACE). Kriging can be grouped into three classes; ordinary, universal and detrended Kriging. Detailed explanations about each category can be found in many sources (e.g., in Olea, 1999; Martin & Simpson, 2003).

Kriging is an interpolative approximation method based on an exponentially weighted sum of the sample data. A Kriging model postulates a combination of a polynomial model and a stochastic process of the form (Simpson et al., 1998):

$$\hat{y} = \sum_{j=1}^{N} \beta_j f_j(x) + Z(x), \tag{2.7}$$

where $\hat{y}$ is the unknown function of interest, the summation function represents the polynomial function (i.e. similar as in polynomial regression) and $Z(x)$ the realisation of a stochastic process with mean zero variance σ^2 and non-zero covariance. The covariance matrix of $Z(x)$ is given by:

$$Cov\left[Z(x_i), Z(x_j)\right] = \sigma^2 R(x_i, x_j), \tag{2.8}$$

where σ^2 is the process variance and R is the correlation function. Kriging models are quite flexible as a variety of correlation functions can be chosen for building the model, however, the Gaussian correlation function proposed in (Sacks et al., 1989) is the most frequently used.

The main drawback of the Kriging is that model construction can be very time-consuming especially when dealing with the large sample data in order to determine the maximum likelihood estimates of the θ parameters used to fit the model for the k-dimensional optimisation problem. Moreover, the correlation matrix can become singular if multiple sample points are spaced close to one another in particular designs. Fitting problems have been observed with some full-factorial designs and central composite designs when using kriging models (Meckesheimer, et al., 2000).

Spline metamodel

The spline approach is based on piecewise polynomial basis functions. If the continuity restrictions are applied to adjacent pieces, the piecewise polynomials are called 'splines'. The 'univariate' spline metamodel can be formed as:

$$f(x) = \sum c_j B_j(x), \tag{2.9}$$

where the B_j are the quadratic or cubic piecewise polynomial basis functions and c_j is the coefficient of the expansion. For the univariate case, the domain is divided into intervals $[t_1, t_2)$, $[t_2, t_3)$...$[t_{n-1}, t_n)$ whose endpoints are called 'knots' (Barton, 1998). Two sets of spline basis functions are commonly used; the truncated power function basis and the B-spline basis (deBoor 1978).

Recently, the 'multivariate' spline has been an active area of research. Typically the approximation uses a full factorial experiment design to estimate the spline coefficients of the metamodel, thus it requires expensive simulation. Some alternative models have been proposed, for example Friedman (1991) presented the Multivariate Adaptive Regression Spline (MARS), which uses a stepwise procedure to recursively partition the simulation input parameter space. The univariate product degree and the knot sequences are determined in a stepwise fashion based on a generalised cross validation (GCV) score method by Eubank (1988).

Neural networks metamodel

Neural networks are networks of numerical processors, whose inputs and outputs are linked according to specific topologies. Thus, a neural network metamodel contains a combination of linear or nonlinear functions (embedded in the inner layer) of the argument vector x. Neural networks can be thought of as flexible parallel computing devices for producing responses that are complex functions of multivariate input information (Barton, 1998). They can approximate arbitrary smooth functions and can be fitted using noisy response values.

Multi-layer perceptron (MLP) feed-forward network is typically used for function approximation due to its flexibility to approximate smooth functions with arbitrarily well, by providing sufficient nodes and layers. Radial basis function network is another popular class of feed-forward networks that incorporate radial basis functions as nodal functions and are capable of universal approximation with one hidden layer (Park & Sandberg, 1991). In contrast to the MLP network, the RBF network can be trained more rapidly (Moody & Darken, 1989). The RBF network approach will be described comprehensively in this thesis which includes some proposed training schemes to improve its performance, and will be applied in this work.

2.4.2. Trade-offs between metamodels

The trade-off between accuracy and computational expense as well as between local and global information must be considered when developing a simulation metamodel. Hence, current research into metamodelling has focused on improving its accuracy and computational speed. For example, the support vector regression (SVR) was introduced in Clarke, Griebsch and Simpson (2005) for improving the simulation time and accuracy, while the radial basis function was extended in Mullur & Messac (2006) to add more flexibility in its estimation. Several comparative studies of the performance between metamodels have been carried out by some authors, and their suggestions will be summarised here.

Jin et al. (2001) compared the performance of polynomial regression (PR), radial basis function (RBF), Kriging, and multivariate adaptive regression splines (MARS), under multiple modelling criteria. They found that RBF gave the most accurate results, was relatively efficient, and was the most robust. The RBF was found to perform relatively better for small and scarce sample sets. Furthermore, Jin et al. (2001) discovered that the Kriging metamodel performed quite well for large sample sets but degraded with a decreasing number of sample points. They also found that Kriging was particularly sensitive to noisy problems and performed poorly, and took significantly longer to calculate than other tested metamodels.

Clarke et al. (2005) introduced Support Vector Regression (SVR) and presented comparative studies to PR, RBF, Kriging, and MARS. They found that SVR and Kriging performed similarly in terms of accuracy and robustness with SVR being slightly better. Contrary to Jin et al. (2001), Clarke et al. (2005) found that RBF gave the worst results for accuracy and was the second least robust. Clarke et al. (2005) referred to the study of performance by Jin et al. (2001), but noted that SVR's efficiency was comparable to MARS which is worse than RBF and PR, but significantly better than Kriging. It is still unclear, however, what are the fundamental reasons for the SVR out-performing others. Clarke et al. (2005) were investigating the SVRs due to the contradictory nature of their results compared with other authors.

Wang et al. (2006) compared RBF, Gaussian processes (GP) (i.e. Kriging is a type of GP), MARS, PR, and adaptive weighted least squares (AWLS). They found similar results to those shown by Jin et al. (2001) in which RBF and GP (i.e. Kriging) both produced accurate results, RBF was noted to be slightly better overall but GP performed better for noisy problems. From the standpoint of efficiency, GP were found to take significantly longer to calculate than the other four methods.

Sathyanarayanamurty and Chinnam (2009) used two test functions to compare Kriging, RBF network, and Support Vector Machines (SVMs). The metamodels are then used for sensitivity analysis, using a Fourier Amplitude Sensitivity Test (FAST) and Sobol. In terms of accuracy, in the case of the sinusoidal test problem, RBF performed the best followed by Kriging and SVM for all the different sample sizes.

In another test problem, Kriging performed better than RBF and SVM for all the different sample sizes, and RBF performed better than SVM for the small dataset. However, SVM performed better than RBF for both medium and large datasets. Overall, they concluded that the Kriging technique is the preferred method for building metamodels in the context of Probabilistic Engineering Design. However, SVM and RBF methods might also prove to be effective when proper care is exercised in building the models to avoid issues of over-fitting.

2.4.3. Sampling techniques

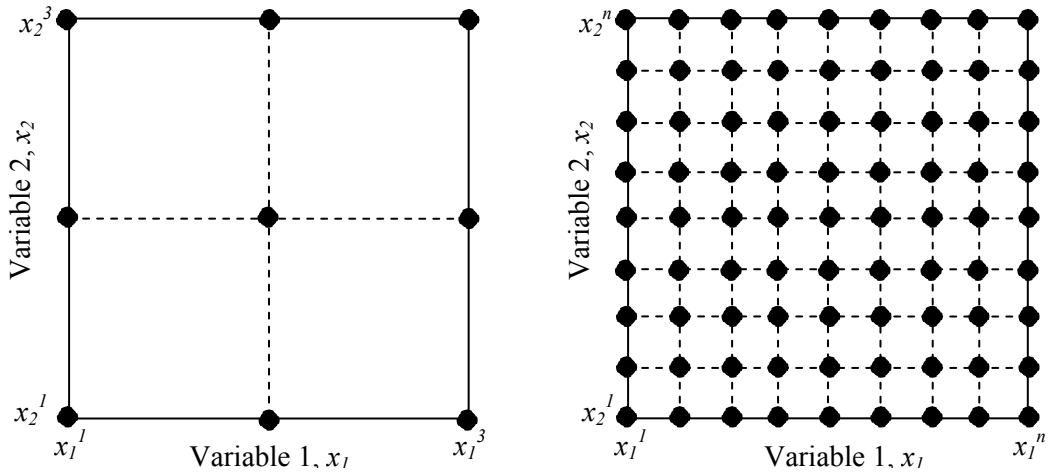
Before executing the function approximation using a metamodel, it is important to select the design points in the domain which is generally termed as sampling, experimental design, or design of experiment (DOE). The aim of any sampling method is to effectively cover the design space and to gather the information of design space characteristics. These sets of independent design variable values from the data points are utilised to produce the values of dependent variables (i.e. response) in which this practice is known as computer experiments. Various sampling classes appeared in the literature such as the full factorial design technique, stratified random sampling and Latin Hypercube sampling (Forrester et al., 2008).

Full factorial design

The primitive experimental design involves the selection of few data points located at the bounds of the design space, and is called *full factorial array.* This is a physical trials method in which the effectiveness of using these points is very poor.

When the computer era began, the experimentation became less costly and the space filling experimental designs started to be used. Full factorial design (FFD) is the simplest approach for this in which it is the most general and standard DOE used over the years for function approximation (Buragohain & Mahanta, 2008). In the approach, the bounds of all the design variables are firstly identified and then they are discretised into equal intervals within the design space. For example, for n-level FFD, a total of n points are selected for each design variable v, all equally spaced over the range. It means the number of design points will be n^v, for which the

topology is illustrated in Fig. 2.4. This approach is also known as *rectangular grid point* sampling.



(a) 3 level FFD (i.e. $n^v=3^2=9$ points) **(b)** n level FFD (i.e. $n^v=n^2$ points)

Fig. 2.4 Full factorial design for two-dimensional problem.

Plain Monte Carlo

Another space filling method called Plain Monte Carlo sampling involves using a random number generator to select the points to reduce the number of points in the trial set. While computationally becoming efficient, Monte Carlo sampling provides no robustness in finding a space filling set of points (Swiler et al., 2006).

Latin Hypercube Design

Latin Hypercube Design (LHD) is a more sophisticated sampling scheme and is continuously being researched, having been invented by McKay et al. (1979). Instead of equally spaced points in the allowable design space, the points are effectively scattered, spanning the whole domain.

For the selection of *n* number of sample points, the range of each design variable is divided into the same number of non-overlapping regions based on the type of probability distribution function (PDF) specified, which can be either normal or uniform PDF. One segment is randomly chosen from each region to form each trial point, but note that this method offers no guarantee that the points will be a balanced set. An example of the scheme is shown in Fig. 2.5. A number of researchers

extended the McKay work such as optimal LHD (Park, 1991), inherited LHD (Wang, 2003), and hybrid LHD (Abdellatif et al., 2010).

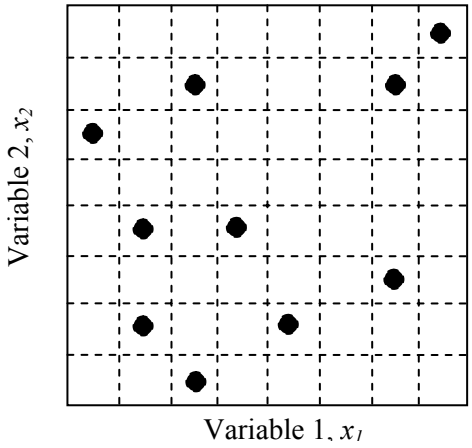


Fig. 2.5 An example of Latin hypercube design for two-dimensional problem with the number of sample points being 10.

2.5. Review of radial basis function neural networks

Over the last few years, the radial basis function neural networks (RBFNNs) have been explored and are being successfully applied across many problem domains that cover engineering, medicine, environment control and geology. They are capable of modelling extremely complex functions with large numbers of input and output variables. As well, the RBFNN paradigms are global, thus a single neural network could be developed to model the entire simulation response surface. This differs from polynomial regression metamodelling, where the regression surface is fitted to a locality, i.e. a subset of the response surface (Ma et al., 2008).

2.5.1. RBFNN general function and its architecture

The radial basis functions were first used to design Artificial Neural Networks in 1988 by Broomhead and Lowe (1988). The RBFNN is motivated by the locally tuned response observed in biological neurons. The main architecture of the RBFNN consists of three layers: (i) an input layer; (ii) a single hidden layer; and (iii) an output layer; as depicted in Fig. 2.6. The outputs of the network implement the weighted sum from the hidden neuron. The input of the network is typically nonlinear, whereas the output from the network is linear. Each of the hidden neuron

implements a radial activated function, which is associated with its radial basis centre. RBFNNs have their origins in a method for performing exact interpolation of a set of data points in multi-dimensional space (Powell, 1987) in which the dimension of the hidden nodes is equal to the pattern of the inputs.

The general output of the RBFNN with l inputs, k hidden units, and m outputs which respond to the input vector $\mathrm{x}^{(i)} \in \Re^n$ (i.e. n is the number of training examples) is mathematically represented as:

$$\hat{y}_j = f(x) = \sum_{k=1}^{N} w_{jk}\phi(x, c_k), \qquad j = 1, 2, ...m, \tag{2.10}$$

where j is the output index, $\phi(.)$ is a basis function, w_{jk} are weights in the output layer, N is the number of neurons (and centres) in the hidden layer in which generally $N << n$, and $c_k \in \Re^n$ are the RBF centres in the input vector space.

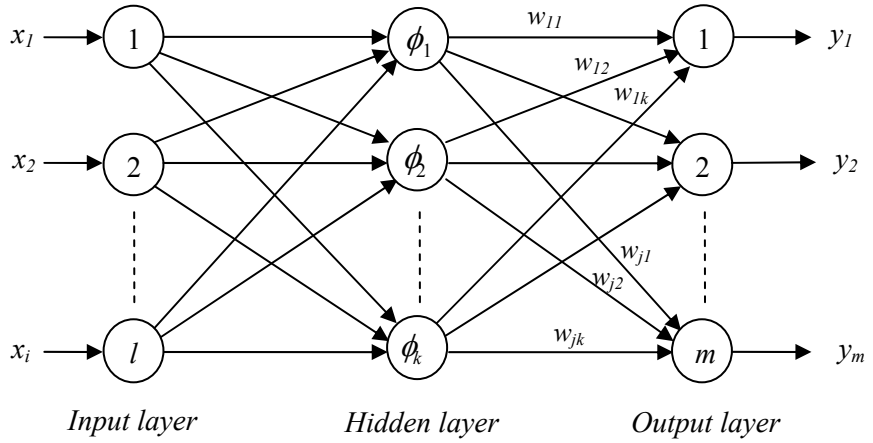


Fig. 2.6 A fundamental architecture of radial basis function neural network.

Radial basis functions ϕ are functions which take the form as follows:

$$\phi(x, c) = \phi(\|x - c\|; \sigma), \tag{2.11}$$

where $\|.\|$ is a vector norm, and σ is a width (also known as scale, or spread) parameter. The value of the radial function depends only on the distance of the point x from the centre point of the function c. The distance function $r = \|x - c\|$ is usually defined as the Euclidean norm (Haykin, 1994). There are three common

types of radial basis functions; the Gaussian, the multiquadratic, and the thin plate spline, which will be discussed in the next section.

2.5.2. Type of basis functions

The selection of radial basis function type is much dependent on the addressed problem in which each has unique characteristics that may make it more suitable for some problems. For example, Harpham and Dawson (2006) suggest that a thin-plate spline function is mostly used in time series modelling, whereas a Gaussian function is preferred with pattern classification problems.

Gaussian function

Gaussian is probably the most popular function because it has attractive mathematical properties of universal and best approximation, and is more flexible to be adjusted in terms of function position and shape. The Gaussian function is said to be locally responsive as the value of the function decays quickly to zero as it moves further from the centre, c (Orr, 1996). The behaviour of one dimensional Gaussian function with width parameter, $\sigma = 1$ and the centre, $c = 0$ is shown in Fig. 2.7. The approximations using Gaussian functions are highly dependent on the value of σ, which needs to be determined. The Gaussian function with a distance function $r = \|x - c\|$ is given in the following equation:

$$\phi(r) = \exp\left(-\frac{r^2}{\sigma^2}\right). \tag{2.12}$$

Multiquadratic function

Multiquadratics are globally responsive in which their value does not decay to zero as the distance from the centre c, increases (Orr, 1996). Its shape is shown in comparison with the Gaussian function in Fig. 2.7, with the same c and σ parameters. Franke (1982) found that approximations using multiquadratic functions are less sensitive to the value of the width parameter σ. The multiquadratic function is mathematically represented as follows:

$$\phi(r) = \sqrt{r^2 + \sigma^2}\,. \tag{2.13}$$

The inverse multiquadratic function was also tested by some authors, which is given by the equation:

$$\phi(r) = \frac{1}{\sqrt{r^2 + \sigma^2}}. \tag{2.14}$$

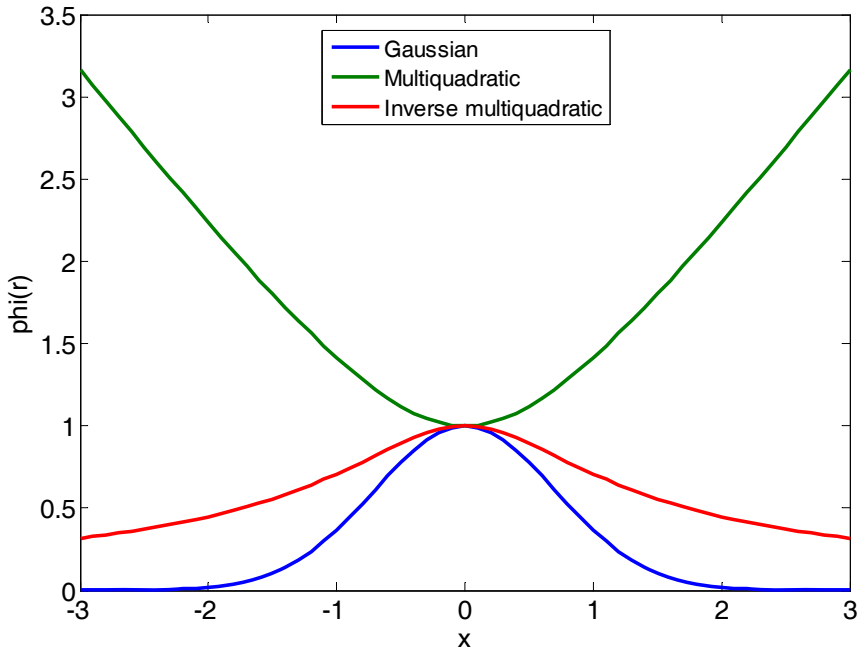


Fig. 2.7 A comparison of one-dimensional Gaussian, multiquadratic and inverse multiquadratic functions with $c = 0$ and $\sigma = 1$.

Thin plate spline function

The thin plate spline is the two-dimensional analogue of the cubic spline in one dimension. It is the fundamental solution to the biharmonic equation $(\nabla^2)^2 = 0$, which physically represents a surface which passes through a given set of data points with the minimum amount of 'bending energy'; a highly desirable property for smooth surfaces. It has the form as follows:

$$\phi(r) = r^2 \log(r). \tag{2.15}$$

The thin plate spline function is a commonly used function for two dimensional regressions due mainly to its physical interpretation. Another advantage is that there is no width parameter which needs to be found.

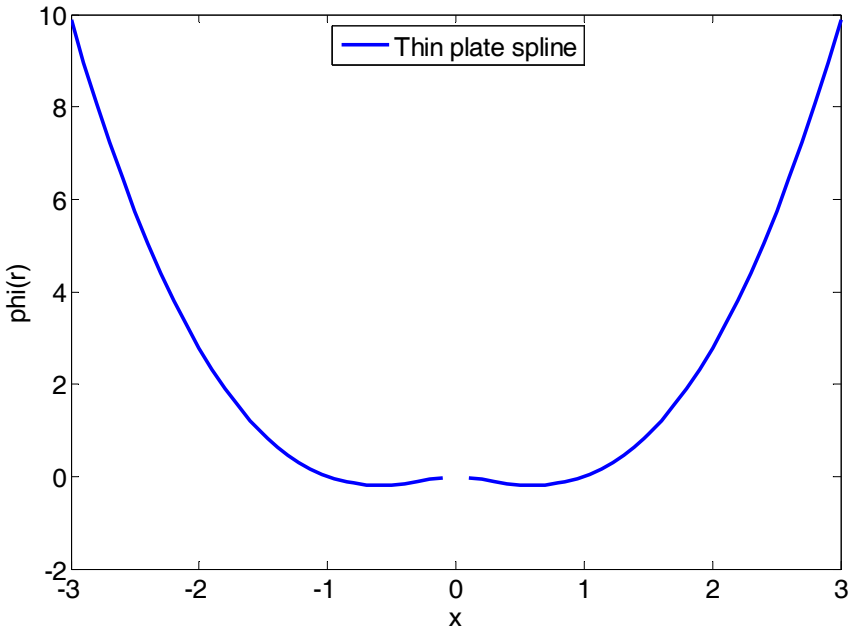


Fig. 2.8 A one-dimensional thin plate spline function with $c = 0$.

2.5.3. Learning strategies

2.5.3.1. General review

One solution to train the network is by exact interpolation (e.g. Powell, 1987). The approach is relatively easier than others; however in practice it is definitely not efficient. First, the interpolating function has to pass through every data point, which is usually noisy, thus leading to over-fitting and thereby poor generalisation. Second, for a large data set, as the number of hidden neurons is equal to the number of data points, the mapping function can be very expensive to be computed. Broomhead and Lowe (1988) showed that the exact interpolation is not a good strategy because of its poor generalisation. Considering this problem, Broomhead and Lowe removed the restriction of the exact interpolation function and designed a two-layer network structure where RBFs are used as computation units in the hidden layer. This model gives a smoother fit to the data using a reduced number of basis functions which depends on the complexity of the mapping function.

Poggio and Girosi (1988) considered a traditional regularisation technique to train the RBF network. The regularisation solution was able to decrease the high computational complexity required for finding an exact solution when the sample

size is very large. In the approach, the number of hidden units is fixed *a priori*, and a subset of the input data points is used as the centre of the hidden units. Each independent variable of the problem is associated with a network input unit. The target function is measured using a cost function, and the gradient descent algorithm is used to train suitable values for the weights of the underlying function. However, output weights need to be properly initialised before the gradient algorithm is run.

Moody and Darken (1989) compared two types of training algorithms, a fully supervised method and a hybrid method combining the supervised method with a self-organising method. They concluded that the hybrid algorithm worked much better and executed faster than the fully supervised one. In the hybrid algorithm, they used the k-means clustering algorithm to estimate the centres of the basis functions. The width values of those basis functions are computed using 'P nearest-neighbour' heuristics and the output weights are estimated by the Least Mean Squares algorithm.

Tao (1993) remarked two potential problems associated with the k-means clustering algorithm. One is that there is still an element of chance in finding the right hidden unit centres. As well, clustering does not guarantee good results in the case of function approximation because clustering is probability oriented and two input patterns close to each other do not necessarily have similar outputs. To overcome these issues, Chen et al. (1991) developed a systematic and popular method of forward selection called Orthogonal Least Square (OLS) to choose the RBF hidden unit centres. A more detailed description of the algorithm will appear in Chapter 4.

After those earlier inventions of the RBFNN algorithm, a number of authors tried to introduce various approaches to improve the network performance or to speed up the computation, including the use of computational intelligence. For example, Chng et al. (1996) have extended the OLS algorithm by Chen et al. (1991) for model selection to include an adaptive procedure to modify the selected node's parameters. The results showed that adaptive OLS could find better subset models than OLS; however, it requires an iteration process by means of gradient descent. Wang et al. (2002) presented a δ-nearest-neighbour cluster algorithm, which combined the *k*-nearest neighbour algorithm with fuzzy c-mean algorithm for the selection of RBF

centres. Their simulation results confirmed that the δ-nearest-neighbour algorithm is an applicable and effective algorithm for forward networks.

Sarimveis et al. (2004) proposed a method to produce a dynamic RBFNN model based on a specially designed genetic algorithm (GA), which is used to auto-configure the structure of the networks and obtain the model parameters. The use of GA in the training algorithm has also appeared in Chang et al., (2009) in which they used the combination of the OLS algorithm, the GA and the LSE method to (i) directly identify the structure of the RBFNN, (ii) search the position of the centres and the width of RBFs, and (iii) identify the linear weights of the output layer. They claimed that the time-consuming search problem is effectively solved and the resulting networks may provide better performance.

From the above reviews, it can be summarised that for a Gaussian function based RBFNN, the training algorithm should involve the three main parameters to be learned, in order to minimise a suitable cost function. These parameters are listed as follows:

1. **Centres of basis function:** The centres of basis functions c_k are determined as part of training process, rather than being constrained to be located at each input data point. The number of basis functions *N*, is typically much less than the number of input data *n*.
2. **Width of basis function:** The training process may adapt the width parameter σ for each of the basis functions, instead of assigning the same width parameter σ to all the basis functions.
3. **The output weights:** The weights between the hidden layer and the output layer w_{ik} are relatively easy to be determined as compared to other parameters. Some of the other approaches include a gradient descent algorithm, regression method, and singular value decomposition (SVD).

In terms of the training strategies, they can be grouped into two classes; fully-supervised training, and two-stage training. For the latter, the first stage involves the determination of radial basis centres and widths, and the following stage is to

determine the weights to the output units. The weights selection in two-stage training can be divided into unsupervised and supervised training.

2.5.3.2. Fully-supervised training

Supervised training may lead to optimal estimation of the centres and widths, however, there are a number of drawbacks with this scheme. First, this approach is typically involved with the gradient descent method, which is a non-linear optimisation technique and thus is computationally expensive. Due to this problem, this method has not attracted many researchers to explore it further. Another drawback is that the learning rates η should be selected carefully to avoid local minima and to achieve acceptable convergence rate and error.

By using gradient descent, the required parameters are iteratively updated to fulfil a certain cost function. For an illustration, consider the sum-of squares error cost function given by:

$$E = \sum_n E_n \tag{2.16}$$

in which:

$$E_n = \frac{1}{2}\sum_j \left\{ t_j^{(n)} - y_j(x^{(n)}) \right\}^2, \tag{2.17}$$

where $t_n^{(i)}$ is the target value of output unit i when the network is presented with input vector x_n. If Gaussian basis functions are used to minimise the cost function in (2.17), the updated equations as appeared in (Ghosh et al., 1992) are given as follows:

$$\Delta w_{jk} = -\eta_1 \frac{\partial E}{\partial w_{ik}} = \eta_1 \left(t_j^{(n)} - y_j(x^{(n)}) \right) \phi_k(x^{(n)}), \tag{2.18}$$

$$\Delta c_k = -\eta_2 \frac{\partial E}{\partial c_k} = \eta_2 \phi_k(x^{(n)}) \frac{\left\| x^{(n)} - c_k \right\|}{\sigma_k^{\,2}} \sum_j \left(y_j(t_j^{(n)}) - x^{(n)} \right) w_{jk}, \tag{2.19}$$

$$\Delta \sigma_k = -\eta_3 \frac{\partial E}{\partial \sigma_k} = \eta_3 \phi_k(x^{(n)}) \frac{\left\| x^{(n)} - c_k \right\|^2}{\sigma_k^{\,3}} \sum_j \left(y_j(t_j^{(n)}) - x^{(n)} \right) w_{jk}, \tag{2.20}$$

where η_1 , η_2 , and η_3 are the learning rates. The simultaneous updating of the three sets of parameters in the above equations may be suitable for non-stationary environments or online settings (Howlett & Jain, 2001).

2.5.3.3. Two-stage training

If the static maps are considered rather than non-stationary environments, the decoupling process, namely the two-stage training procedures, may offer a very attractive alternative, which is involves:

(i) First, the use of unsupervised methods to determine the basis function centres and widths. These are particularly useful in situations where labelled data is in short supply, but there is plenty of unlabelled data (i.e. inputs without target outputs).

(ii) Second, the determination of output layer weights that are connected to all the basis functions using centres and widths from step (i). It can be a supervised or unsupervised technique.

Both sub-problems allow for very efficient batch mode solutions. Furthermore, for many situations, this technique leads to little loss in the quality of the final solution as compared to the optimal solution. In fact, given finite training data and computational resources, it often provide better solutions than those obtained by attempting to simultaneously determine all three sets of parameters.

Stage 1: *Unsupervised training of basis function centres and widths*

A few existing approaches are described as follows:

(i) **Fixed centres selected at random:** This is the simplest and quickest approach in which centres c_k are selected as fixed *N* points randomly from *n* data points. This choice is sensitive to how representative are the selected data points of the overall population. One approach is to use the equal and fixed widths at an appropriate size for the distribution of data points in which for a large training data set this method provides reasonable results. It was suggested that the widths parameter be given as (Howlett & Jain, 2001):

$$\sigma_k = \frac{d_m}{\sqrt{2N}}, \tag{2.21}$$

where d_m is the maximum distance between chosen centres.

(ii) **Clustering algorithm:** Using clustering techniques provides an improved approach which more accurately reflects the distribution of the data points. A variety of clustering techniques can be used. Supposed we need to partition N data points x^n into K clusters and find the corresponding cluster centres. The K -mean algorithm seeks to partition the data points to K subsets S_j by minimising the sum of squares clustering function:

$$J = \sum_{j=1}^{K} \sum_{n \in S_j} \left\| x^n - \mu_j \right\|^2, \tag{2.22}$$

where μ_j is the geometric mean of the data points in the subset S_j, given by:

$$\mu_j = \frac{1}{N_j} \sum_{n \in S_j} x^n . \tag{2.23}$$

One method for finding these clusters is by using the batch version. First, the data points are randomly assigned to K subsets. The centres for each of the subsets are then computed. The data points are then reassigned to the cluster whose centre is nearest. The procedure is repeated till there are no further changes in the grouping.

(iii) **Width determination:** The basis function widths can either be chosen to be the same for all nodes or they can be different from each other. For the first case, the common width can be set at some multiple of average distance between the basis centres. This multiple represents the smoothness of the function, small widths leading to less smooth functions. In the second case, each centre's width is set to a multiple typically one and one-half times to twice the average distance to the L nearest neighbours (Howlett & Jain, 2001). The widths can also be adjusted optimally using equation (2.20), but it is not a popular one due to its computational cost.

Stage 2: *Training the output weights*

The output weights can be obtained by either supervised or unsupervised methods.

(i) **Unsupervised method:** Once the centre and widths are determined, they are kept fixed for the second stage of training during which the hidden output weights are learned. Since the second stage involves just a single layer of weight w_{ik}, they can easily be found analytically by solving a set of linear equations. Usually, this can be done quickly without the need for iterative weight updates such as gradient descent learning. Generally, the desired network output D is always written in matrix form as $D = \Phi^T W$, where Φ is the radial basis function matrix and W is the output weight matrix. Using a least square solution, the weight matrix W can be found by using the pseudo inverse of matrix Φ as given in the following equation:

$$W = \left(\Phi\Phi^T\right)^{-1}\Phi D \quad \text{or} \quad W = A^{-1}\Phi D, \tag{2.24}$$

where $A = \Phi\Phi^T$ is called the *variance matrix*. Thus, in practice, to avoid the possible problems due to ill-conditioning of the matrix Φ, singular value decomposition (SVD) is usually considered to solve the equation, rather than a regression method.

(ii) **Supervised method:** The weights of the output layer can also be trained by a supervised learning method like the back-propagation (BP) algorithm. One example is to use the learning equation as in equation (2.18).

2.5.4. Regularised and generalised RBFNN

Regularised RBFNN

Regularisation is a powerful technique to stabilise certain solutions, by adding a penalty functional to the original cost function so as to bias the solution towards more desirable solutions, e.g. smoothness constraints on the input-output mapping, and make an ill-posed problem into a well-posed one. Regularisation theory was first introduced by Tikhonov (1973). A variety of penalties are studied in Friedman (1994). For RBFNNs, a popular choice for the modified cost function is the summed squared error (*SSE*). Consider for a single output unit, *SSE* is given by:

$$SSE = \sum_{i=1}^{n}\left\{t^{(i)} - y(x^{(i)})\right\}^2 + \lambda\sum_{k=1}^{N} w_k^{\,2}, \tag{2.25}$$

where n is the number of training samples, N is the number of hidden units, λ is the regularisation parameter and w_k is the regularisation weight. This function is associated with the ridge regression in which the network function becomes smoother when the λ is increased. Therefore, the optimal weight matrix in equation (2.24) now becomes:

$$\mathrm{W} = \left(\Phi\Phi^{\mathrm{T}} + \lambda \mathrm{I}_{\mathrm{N}}\right)^{-1}\Phi\mathrm{D}\,, \tag{2.26}$$

where I_{N} is the identity matrix and A now becomes $\Phi\Phi^{\mathrm{T}} + \lambda \mathrm{I}_{\mathrm{N}}$. The estimated regularisation parameter λ can be chosen by the generalised cross-validation (GCV), which involves an iterative formula such as given in the following equation, where P is the projection matrix (Orr, 1996):

$$\hat{\lambda} = \frac{\mathrm{D}^{\mathrm{T}}\mathrm{P}^2 \mathit{trace}\left(\mathrm{A}^{-1} - \hat{\lambda}\mathrm{A}^{-2}\right)}{\mathrm{W}\mathrm{A}^{-1}\mathrm{W}^{\mathrm{T}}\mathit{trace}(\mathrm{P})}. \tag{2.27}$$

Generalised RBFNN

Generalisation is a simplified version of the regularisation strategy for RBFNN. The solution computed by the regularisation network can be said to be an optimal solution. However, the generalisation will not achieve the optimality of the regularisation but would solve the ill-conditioned matrix in the solution. Besides, it does not require expensive computation (i.e. iteration process) because its penalty values (i.e. the bias weights) can be learned quickly using regression at the same time for the determination of the output layer weights.

The framework of the *generalised radial-basis function neural network (GRBFNN)* is shown in Fig. 2.9. In this network, a bias (i.e. data-independent variable) is applied to the output unit. In order to do that, one of the linear weights in the output layer of the network is set equal to a bias and the associated radial basis function is treated as a constant equal to +1 (Haykin, 1994). The bias is added to compensate for the difference between the average value over the data set of basis functions and the corresponding values of the targets (Howlett & Jain, 2001). By applying this bias, the RBFNN mapping formula in equation (2.10) becomes

$$\hat{y}_j = f(x) = \sum_{k=1}^{N} w_{jk}\phi(x, c_k) + w_{j0}\phi_0 = \sum_{k=1}^{N} w_{jk}\phi(x, c_k) + b_j \quad , \tag{2.28}$$

where ϕ_0 is the bias constant (i.e. =1), and w_{j0} is the bias weight.

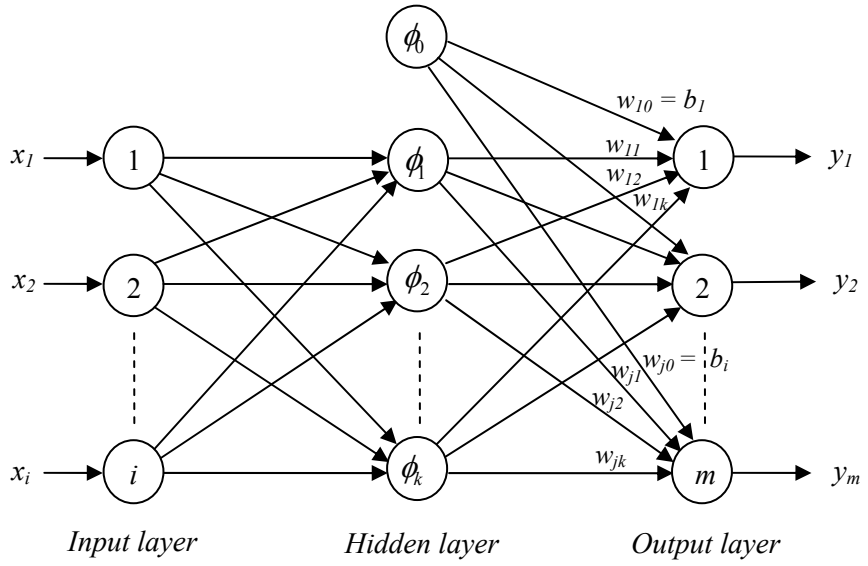


Fig. 2.9 A generalised radial basis function neural network.

2.6. Chapter conclusion

This chapter has reviewed some of approaches that are commonly being practised by the regulation authorities for managing air quality. There are three important strategies involving the air quality monitoring; the pollutants' emission inventory and assessment, and air quality modelling. The former is the most effective and accurate in the assessment process, however, its results can only be used in the region of interest. Therefore, the modelling approach is used because it can add benefit in many ways, such as through its ability to estimate the pollutants' spatial distribution, and can be used in the emission reduction program. In this work, the first and the second methods will be utilised for the purpose of data set collection, whereby the statistical modelling approach which features radial basis function neural network metamodelling, will be focussed in the third method.

The next reviews focus on the air quality issue that will be addressed in this research which is ozone and background ozone level (BOL). Several definitions and

determination approaches have been surveyed. In general, the measurement methods with some quantisation methods have been used widely in the determination of BOL; however, the approach is nearly impossible to be implemented in highly urbanised areas. For this reason, we will propose an applicable method to overcome this limitation.

This chapter also discussed the available methods in metamodelling such as Polynomial Regression, Kriging, Splines, Support Vector Regression and Neural Networks, and the trade-offs of those metamodels are then comparatively reviewed. Generally, some researchers suggested that the radial basis function approach has some unique advantages as compared with other. In the last review, the fundamental theory of radial basis function neural network (RBFNN) has been explained, and some available learning strategies were also reviewed thoroughly.

Chapter 3
BUILDING A NEURAL NETWORK-BASED METAMODEL

3.1. Introduction

This chapter will deal with the development of a neural network as the metamodel that will implemented throughout this research. The reason behind the selection of a neural network among the other techniques (as discussed in Chapter 2) is that it is much more flexible in functional form and is therefore better suited for complex or nonlinear functions that are not easily approximated by low order polynomials. The radial basis function is chosen as the neural network framework due to its advantages such as simplicity, robustness, rapid computation and also reasonable performance (Moody & Darken, 1989; Jin et al., 2001; Wang et al., 2006).

In the last section of this chapter, a new method for the experimental design (i.e. a part of modelling process) will be described. This method is specifically efficient for a neural network-based metamodel and may be implemented in other metamodelling approaches as well as other kinds of statistical modelling. Some numerical analysis using non-linear complex functions will be presented to evaluate its performance.

3.2. The metamodelling process

The fundamental objective of a metamodel is to attempt to learn the mapping of output $y = f(x)$ that exists in a black box (i.e. typically a physical system, or computer experiment to be estimated) that converts the input vector **x** into a scalar

output y. The generic methodology is to collect a set of output values $y = y^{(1)}, y^{(2)}, \ldots, y^{(n)}$ that result from a set of input vectors $\mathrm{x} = \mathrm{x}^{(1)}, \mathrm{x}^{(2)}, \ldots, \mathrm{x}^{(n)}$ and find the best approximation $\hat{y}$ for the black box mapping $f(x)$, based on these known observations (see illustration in Fig. 3.1).

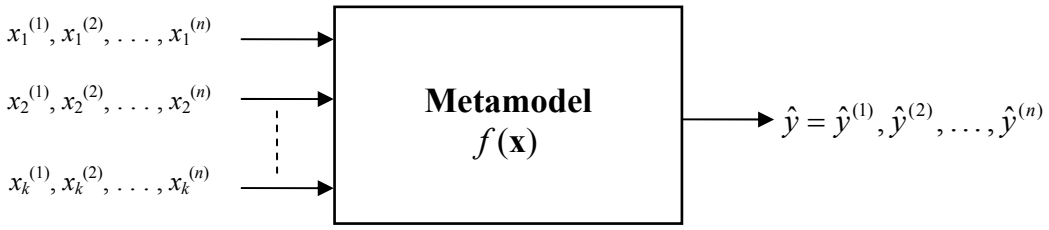


Fig. 3.1 A metamodel map between k input variables and an output, with *n* number of patterns.

Metamodelling can be defined as a process of building a model of a model (Meckesheimer et al., 2002). This process involves the choice of an experimental design (i.e. sampling process), a metamodel type and its functional form of fitting, and a validation strategy to assess the accuracy of the metamodel. As the metamodel class has been specified as a neural network, the metamodel construction now involves three key stages, which include data preparation and sampling, the training process, and also validation and testing (evaluation).

3.2.1. Data preparation and sampling

Initially, before proceeding with other processes in neural network-based metamodelling, a proper preparation of the training data set is essential as it will directly influence the complexity of the developed model as well as the output performance. The choice of relevant design variables and the selection of the most appropriate data points have to be done systematically rather than as a random process. Here, three suggested steps will be described which include the determination of variable importance, the data set division, and the data set sampling.

3.2.1.1. Variable importance

In metamodelling, the cost or performance metric of a process is defined by a *k*-vector of design variables $\mathrm{x} \in \mathrm{D} \subset \Re^{\mathrm{k}}$ in which we shall refer to D as the *design*

space or *design domain*. The problems of characterising degrees of importance for a large number of input (or design) variables is a common issue since the elimination of unimportant inputs leads to a simplification of the problem and often a more accurate model or solution. The importance may be easily determined if the input-output of the system is known from the previous experience or from the source-receptor relationship, but there are usually not easy in most cases.

Several comparative studies have appeared in the literature in this field. For example, Sung (1998) compared three different methods for ranking input importance: sensitivity analysis (SA), fuzzy curves (FC), and change of mean square error (COM); and analysed their effectiveness. They found that the FC method is valuable in building networks of high accuracy, followed by COM and SA. However, FC may be incomplete or even unavailable in many situations. Olden et al. (2004) demonstrated nine methods for interpreting the variable importance, and they concluded that the connection weights (CW) method outperformed the rest, while Garson's important measure equation (introduced in Garson, 1991) showed the worst performance. See also other available methods such as the regression model (Yi & Prybutok, 1996), and principle component analysis PCA (Lu et al., 2004).

Here, two possible methods will be highlighted as follows:

(i) **Connection weights method (CW):** The connection weight method (Olden et al., 2004) sums the product of the weight of the connection from the input neuron to the hidden neurons with the weight of the connection from the hidden neurons to the output neurons for all input parameters. The larger the sum of connection weights, the greater the importance of the parameter that is associated with this input neuron. The relative importance of the input parameter *i* is determined through the following formula:

$$\mathrm{Imp}(i) = \sum_{x=1}^{n} \left(CW_{ih(x)} CW_{ho(x)} \right), \tag{3.1}$$

where $\mathrm{Imp}(i)$ is the relative importance of parameter i, n is the total number of hidden nodes, x is the index number of hidden node, $CW_{ih(x)}$ is the connectivity weight between input parameter i and hidden node x, and $CW_{ho(x)}$

is the connectivity weight between hidden node x and the output node. In the case of RBFNN, the complexity of the function is reduced as the connectivity weights between input and hidden layer are always equal to unity.

(ii) **Change of *MSE* (COM):** This method evaluates the variable significance by measuring the change of mean square error (*MSE*) when that input is deleted from the neural network (Sung, 1998). The *MSE* is defined as:

$$MSE = \sum_{k=1}^{K} \sum_{p=1}^{P} \left(t_{kp} - o_{kp} \right)^2 / P, \tag{3.2}$$

where t_{kp} and o_{kp} are the desired output and the calculated output, respectively, of the k^{th} output neuron for the p^{th} pattern; K is the total number of output nodes and P is the total number of patterns. In the COM method, we retrain the neural network with N-1 inputs each time after an input is deleted and observe the change, where N is the number of original input parameters. Based on the net change in *MSE*, we can rank the importance of input variables in several different ways based on different arguments. For example, we can rank the inputs whose deletion causes the largest change in *MSE* as the most important since the error is most sensitive to these inputs.

3.2.1.2. Data sampling

Once the significant variables have been confirmed, the data set needs to be sampled, especially when dealing with a large data set. Sampling of design space is a separate process which must be done effectively. Several possible techniques have been discussed in section (2.4.3) in Chapter 2. A new sampling strategy will also be introduced in the last section of this chapter.

3.2.1.3. Data set classification

The data set is normally partitioned into three subsets: training set, validation set, and test set. The selection of a number in each set is an issue for which no generic methodology is available. However, the common choices in the literature are the combination of 70%, 20% and 10%, or the combination of 50%, 25% and 25%, for training, validation and testing sub-sets, respectively. In the multilayer perceptron (MLP) network, the validation process is embedded in the training process, and the

division of the data set for training and validation is normally performed automatically at random. However, in the standard RBFNN algorithm only two processes exist; training and testing, from which the validation process is omitted. The purpose of each subset is described as follows:

Training set: It is used for computing the gradient and updating the network weights and biases.

Validation set: The validation process is part of the optimisation strategy. The error of the validation set is monitored during the training process. The validation error normally decreases during the initial phase of training, as does the training set error. However, when the network begins to overfit the data, the error on the validation set typically begins to rise, and the training should be stopped.

Test set: This data is not primarily used during training stage, and is used to confirm the actual predictive performance of the network.

3.2.2. Model training

The model training requires a set of input-output vectors or training pairs which is obtained in the previous stage. Later this data is collectively used to train the network. The overview of the basic learning strategies using radial basis function neural network (RBFNN) has been discussed in section (2.5). More details, explanation and elaboration of the RBFNN algorithm, will be covered comprehensively in Chapters 4 and 5.

In the training loop process, it is necessary to consider a cross-validation method to validate the correctness of the current trained network, and probably call an early halt before it reaches the prescribed goal. The most frequently used cross-validation method is the leave-*k*-out cross-validation method (Martin & Simpson, 2003; Simpson et al., 2004; Kleijen, 2005), which may provide a reasonable assessment of the validity of the metamodel. The basic procedure involves leaving *k* points out of the sample set, then rebuilding the metamodel with *n*-*k* points, where *n* is the number of sample points. The new metamodel results are compared to the exact function evaluations from the *k* points which were left out, using a root mean squares error (*RMSE*) function. This step is repeated, and the RMSE from all steps

are averaged to give the final validity measurement. The value k is suggested to be either $k = 0.1n$ or $k = \sqrt{n}$ for kriging and $k = 1$ is suggested for the polynomial regression (PR) and radial basis function (RBF) (Meckesheimer et al., 2002). Leave-k-out cross-validation is represented in the following equation:

$$RMSE_{CV} = \sqrt{\frac{1}{k}\sum_{j=1}^{k}\left(\hat{y}_j - y_j\right)^2}\,. \tag{3.3}$$

3.2.3. Model testing (validation)

After network training, it is essential to test the model by evaluating it using a set of inputs and comparing it with corresponding outputs. The validation method evaluates the accuracy of the metamodel in order to ensure that the metamodel reflects the actual network. This practice is crucial to knowing to what extent the approximation of the network can be trusted. Hence, some statistical criteria are required, normally referred to as *performance indexes*. However, these performance measures are only useful for development and comparative purposes, not for in-the-loop processing of validity (as does cross-validation).

In this work, four indexes will be used to measure the residual errors, including the root mean square error (*RMSE*), the mean absolute error (*MAE*) and the determination coefficient (R^2), given respectively as

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(O_i - P_i)^2}\,, \tag{3.4}$$

$$MAE = \frac{1}{n}\sum_{i=1}^{n}\left|O_i - P_i\right|, \tag{3.5}$$

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(O_i - P_i)^2}{\sum_{i=1}^{n}(O_i - \overline{O})^2}, \tag{3.6}$$

where P_i and O_i are the predicted (estimated) and observed (actual) values, and $\overline{O}$ represents the observation mean. A higher accuracy of the metamodel is shown by smaller values of *RMSE* and *MAE*, and larger value of R^2. In addition, we also

investigate the index of agreement d_2, a measure expressing the degree to which predictions are error-free (Gadner & Dorling, 2000):

$$d_2 = 1 - \frac{\sum_{i=1}^{n} \left| P_i - O_i \right|^2}{\sum_{i=1}^{n} \left(\left| P_i - \overline{O} \right| - \left| O_i - \overline{O} \right| \right)^2}, \tag{3.7}$$

Other indexes may also be used such as the relative average absolute error (*RAAE*) and relative maximum absolute error (*RMAE*) (Jin et al., 2001).

3.3. Metamodel management

The management of the metamodel process is essential to search for the best possible or optimal solution of metamodelling. The management strategy should aim to effectively use the data sampling process, and to include a validation method to assess the fitted metamodel, in which it is typically implemented in the iteration loop.

One approach is by using a *sequential modelling process*, as outlined in the flowchart given in Fig. 3.2. Here, we assume that the pre-modelling process, such as the selection of a suitable metamodel as a function approximation and the variable importance, has been confirmed, providing a readily available dataset to be used in the modelling, namely a *design space, S*. One starts with a data set, $S\{X,Y\}$ consisting of N input-output pairs (x_i, y_i), for $i = 1,\ldots,N$ where y_i is the model output response at the design input sample point x_i, and N is the total number of disciplinary model samples. To begin the modelling process, an initial dataset should be first constructed by the space filling experimental design (i.e. DOE), where the sampling point number is M from the complete numbers of data set N. Next, the sampled data S_M is grouped uniformly or randomly into q group, thus we have $S_M = \{S_1, S_2, \ldots, S_q\}$. Each group of the sampled dataset is added in every iteration and used to fit the metamodel, until it meets the accuracy goal in the validation process. By using this strategy, the best possible solution is restricted to the sampled

dataset S_M in which it is most likely that there are some other good potential points that may be chosen, which were excluded in S_M.

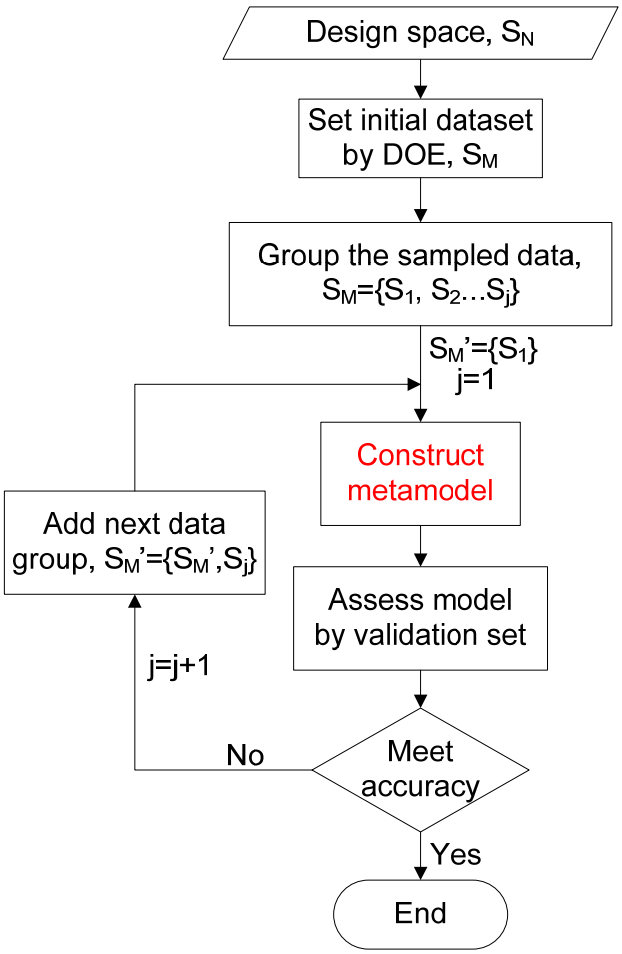


Fig. 3.2 A metamodelling flowchart by sequential sampling process.

In this work, we proposed a more reliable strategy by utilising the maximum potential of the sampling dataset to fit the metamodel. The process flow is illustrated in Fig. 3.3 in which the specific metamodel (i.e. RBFNN) is included in the diagram. The design space S_N is sampled from several possible groups of the dataset using DOE, by increasing the number of data points for each group, e.g. $S_1 = 0.05S_N$, $S_2 = 0.10S_N$, $S_3 = 0.15S_N$ and $S_j = nS_N$, where j is the maximum number of the sampled group and $n < 1$. Each dataset will be evaluated to build the metamodel in every iteration from the minimum to the maximum number of sampled data points, and the process ended when it meets the required performance in the validation stage. By doing this, a possible best solution may be found with a minimal number of sampled data points, which could reduce the simulation time as well as to avoid

the use of exhaustive sampled data that may overfit the neural network construction. The construction of neural network model involves the tuning process of the model parameters such as the selection of basis centres and the training of the output weights. A cross validation (CV) method can be used in the loop to improve the generalisation of the metamodel, which may stop the model training earlier.

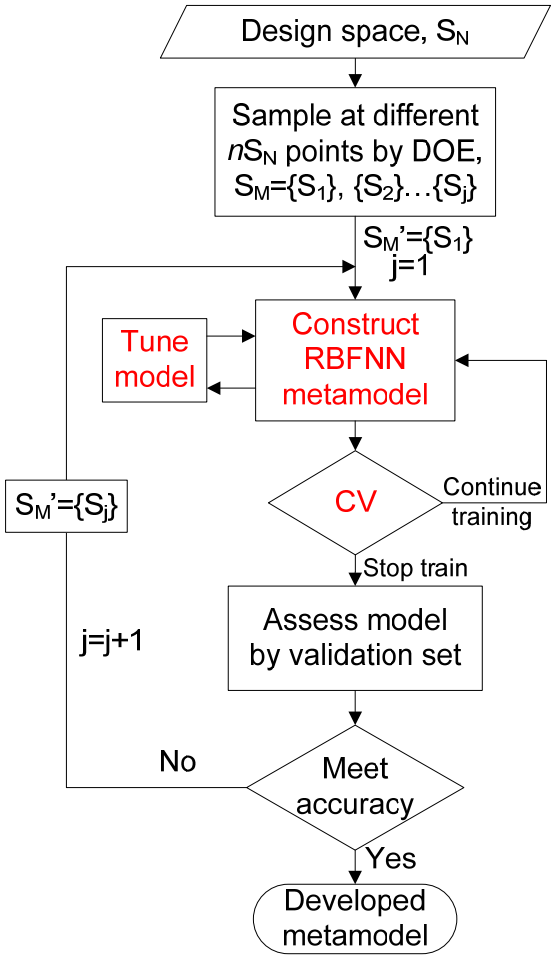


Fig. 3.3 The proposed metamodelling flowchart.

3.4. A new sampling scheme for a NN based metamodel

3.4.1. Introduction

In this work, a new method for the DOE is introduced, which uses a distance weight function to measure the normalised distance for all the input-output data points, and followed by clustering it to *n* numbers of sampling points by using a k-means algorithm, which will be referred to as the weighted clustering design (WCD) scheme. A radial basis function neural network (RBFNN) metamodel approach is

then used to evaluate the performance of the proposed DOE using a nonlinear and high-dimensional mathematical function. A comparison study is also included to analyse the performance of the introduced sampling technique with other available methods such as the *n*-level full fractional design (FFD) method and Latin hypercube design (LHD) method. The results show that the proposed method produces an improved performance in the estimation as compared to one without the implementation of DOE, and in many cases, it outperforms the network developed from other sampling designs of the same size.

3.4.2. Methodology

In the neural network based metamodelling, the dataset is normally divided into two groups, one for the training (trial) and another one for the testing. If we have a set of input-output training datasets denoted by x and y, a mapping solution is given as follows:

$$\left\{x^{(i)} \rightarrow y^{(i)} = f(x^{(i)}) \middle| \, i = 1,2,..., m\right\}, \tag{3.8}$$

where m is the maximum number of data points. For the case of one design objective and n number of input design variables, the inputs and outputs are given as in the following equations,

$$X = \left\{x_1^{(1)}, x_1^{(2)}, ..., x_1^{(i)}; ....; x_j^{(1)}, x_j^{(2)}, .., x_j^{(i)}) \middle| \, i = 1,..,m, \, j = 1,..,n\right\}$$

and

$$Y = \left\{y^{(1)}, y^{(2)}, ..., y^{(i)} \middle| \, i = 1,2,..., m\right\}. \tag{3.9}$$

For a three-dimensional problem, the distribution of four data points is illustrated in Fig. 3.4. Each data point has its own unique weight by measuring the distance weight factors from a common reference point c. By using the Euclidean distance measure, the distance between point p^1 and c is mathematically written as:

$$d(p^1, c) = \left[\left(p^1(x_1) - c(x_1)\right)^2 + \left(p^1(x_2) - c(x_2)\right) + \left(p^1(x_3) - c(x_3)\right)^2 \right]^{1/2}, \tag{3.10}$$

or generally, the weight for all data points of the n-dimensional problem is given as follows:

$$d(p^i, c) = \left[\sum_{j=1}^{n} \left(p^i(x_j) - c(x_j) \right)^2 \right]^{1/2}. \tag{3.11}$$

The weights could represent the distinct patterns between each data point, and some neighbouring points may have a similar weight that could be clustered as a group and one point taken as a candidate.

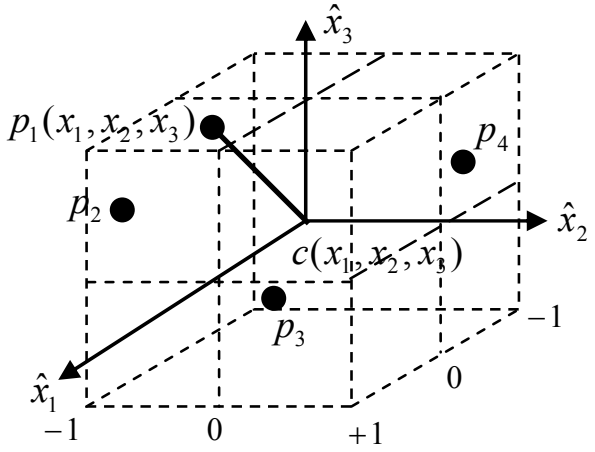


Fig. 3.4 A distribution of data points in three-dimensional space.

To generalise the solution, the pairs of the input and output data points are combined so as to become the design space *S* in this evaluation, which is given as:

$$S = \{X; Y\}. \tag{3.12}$$

Hence, the dimension of the distance measure for one targeted output has now become $(n+1)\times m$. The solution in (3.11) can be simplified further if we set a common reference centre at the zero coordinate by first normalising the design space *S* to the minimum of -1 and to a maximum of 1, given as:

$$S' = [-1, 1]^{(n+1)\times m}, \tag{3.13}$$

as shown in Fig. 3.4. And now solution in (3.11) becomes:

$$d(p^i) = \left[\sum_{j=1}^{n+1} \left(p^i(\hat{x}_j) \right)^2 \right]^{1/2}, \tag{3.14}$$

where $\hat{x}_j$ is the normalised value of the design space *S* which has been incorporated by the output variable.

A list of distance weight values which is given by:

$$D = \{ d_1, d_2, ...d_i \mid i = 1,2,..., m \}, \tag{3.15}$$

is then sorted and clustered by using an available clustering algorithm. In this work, a well-known k-means algorithm based on Voronoi iteration (MacKay, 2003) is used due to its fast computation especially for the one-dimension case. It uses a two-phase iterative algorithm to minimie the sum of point-to-centroid distances, summed over-all *k* clusters. There are several methods to choose the initial k-means points. In this evaluation, we replicate them randomly, which in typical will results in a solution that is a global minimum (Hamerly & Elkan, 2002). The maximum number of cluster *k* corresponds to the number data points that will be sampled. The determination of appropriate *k* value for this scheme is demonstrated throughout this work.

3.4.3. Test function

To evaluate the effectiveness of the proposed approach, a high dimensional and nonlinear mathematical test problem is employed, namely Problem 100 from Hock–Schittkowski (Meckesheimer et al., 2002). The Hock–Schittkowski Problem 100 is a test problem consisting of seven variables, one objective, and four constraints. In this analysis, we consider only the objective function without those constraints. However, we specify the design domain for this function where the function is given as follows:

$$\begin{aligned} f(x_i) = {} & (x_1 - 10)^2 + 5(x_2 - 12)^2 + {x_3}^4 + 3(x_4 - 11)^2 + 10{x_5}^6 \\ & + 7{x_6}^2 + {x_7}^4 - 4x_6x_7 - 10x_6 - 8x_7 \end{aligned}, \tag{3.16}$$

where $-10 \le x_i \le 10$. To prepare a full large dataset, a series of input-output data points are randomly generated (e.g. using *'randn'* code in Matlab) within the design space in which the maximum number of data points is set to 4000.

Different sample sizes and a fitting design method which involves weighted clustering design (WCD), n-level full factorial design (*n*-FFD) and Latin Hypercube design (LHD), will be evaluated. Some of the performance measures, the size of the RBFNN metamodel and the total execution time for the simulations will also be noted. The full program codes to run this simulation are listed in Appendix A (-1, -2 and -3).

3.4.4. Numerical analysis

By using three experimental design methods, the prepared datasets are sampled at different sample sizes, *N*. Each set of the sampled data is then mapped using the RBFNN metamodel by setting the spread parameter as 4 and the prescribed *MSE* goal as 0.001 for all the tests. A special RBFNN with a supervised learning process for the selection of basis centres is used in this evaluation (Wahid et al., 2012). A more detailed description of the algorithm will be discussed in Chapter 5.

Table 3.1 shows the results for three types of analysis which involve the performance indexes, the number of hidden neurons used to construct the neural network and also the total simulation time. For the proposed sampling scheme, the performance based on R^2 and d_2 is increased with the increment to the *N* value, which approaches unity for the possible best performance.

Table 3.1 Metamodel comparison results for the test function.

No.	*Design name*	*Configuration*	*Sample size, N*	*% of N*	*Performance measure*				*Network size*	*Simulation time (s)*
					RMSE	*MAE*	R^2	d_2		
1.	*WCD*		*400*	*10*	*1.79E06*	*1.23E06*	*0.443*	*0.861*	*298*	*42*
2.			*600*	*15*	*1.09E06*	*7.39E05*	*0.793*	*0.948*	*329*	*57*
3.			*1000*	*25*	*4.84E05*	*3.39E05*	*0.960*	*0.990*	*330*	*93*
4.			*1400*	*35*	*4.18E05*	*2.98E05*	*0.971*	*0.993*	*333*	*126*
5.			*1800*	*45*	*3.69E05*	*2.68E05*	*0.976*	*0.994*	*333*	*173*
6.			*2200*	*55*	*3.46E05*	*2.58E05*	*0.980*	*0.995*	*337*	*250*
7.			*2800*	*70*	*3.14E05*	*2.32E05*	*0.983*	*0.996*	*341*	*389*
8.			*3400*	*85*	*3.26E05*	*2.40E05*	*0.982*	*0.995*	*342*	*495*
9.	*n-FFD*	*[2 3 2 3 2 3 2]*	*432*	*11*	*1.37E06*	*9.75E05*	*0.672*	*0.918*	*330*	*43*
10.		*[2 3 2 3 2 3 3]*	*648*	*16*	*1.38E06*	*9.75E05*	*0.672*	*0.918*	*330*	*43*
11.		*[2 3 3 2 3 3 3]*	*972*	*24*	*4.66E05*	*3.27E05*	*0.962*	*0.991*	*351*	*99*
12.		*[2 3 3 3 3 3 3]*	*1458*	*36*	*3.70E05*	*2.66E05*	*0.976*	*0.994*	*349*	*143*
13.		*[2 3 3 3 3 3 4]*	*1944*	*49*	*3.31E05*	*2.43E05*	*0.981*	*0.995*	*347*	*233*
14.		*[3 3 3 3 3 3 3]*	*2187*	*55*	*3.23E05*	*2.37E05*	*0.982*	*0.996*	*352*	*279*
15.		*[3 3 3 3 3 3 4]*	*2916*	*73*	*2.99E05*	*2.21E05*	*0.985*	*0.996*	*344*	*467*
16.	*LHD*	*with 'maximin' criterion*	*400*	*10*	*2.00E06*	*1.34E06*	*0.304*	*0.826*	*305*	*42*
17.			*600*	*15*	*1.18E06*	*7.67E05*	*0.757*	*0.939*	*330*	*55*
18.			*1000*	*25*	*5.74E05*	*3.93E05*	*0.943*	*0.986*	*334*	*89*
19.			*1400*	*35*	*5.02E05*	*3.38E05*	*0.956*	*0.989*	*334*	*127*
20.			*1800*	*45*	*3.89E05*	*2.74E05*	*0.974*	*0.994*	*335*	*191*
21.			*2200*	*55*	*3.75E05*	*2.70E05*	*0.976*	*0.994*	*337*	*281*
22.			*2800*	*70*	*3.01E05*	*2.20E05*	*0.984*	*0.996*	*332*	*390*
23.			*3400*	*85*	*2.96E05*	*2.16E05*	*0.985*	*0.996*	*336*	*501*

Note: N_{full}=4000, σ=4, *MSE*=0.001

However, to compromise between the performance and the complexity of the approximate model, for a large dataset, it is suggested that the N number may be selected at between 25% to 30% of the full dataset as there is no significant improvement in the performance when N is more than this range of values (i.e. reaches the saturation region), see Fig. 3.5 to Fig. 3.8. As can be seen from the figures, the saturation point given by R^2 and d_2 is located at about 25% of the full dataset, whereas the point given by *RMSE* and *MAE* is at 30% of the full dataset. The resultant *RMSE* and the *MAE* values appear high, though they are relatively small (about ±4% errors) as compared to the maximum output value for the test problem.

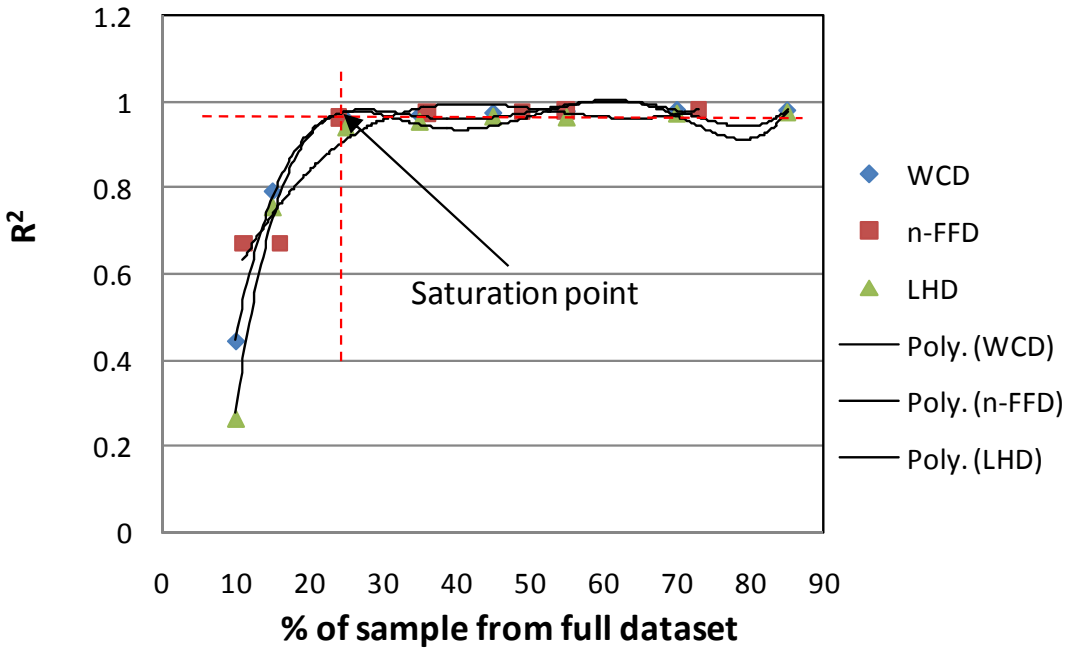


Fig. 3.5 The R^2 performance index against the sample number (in percentages).

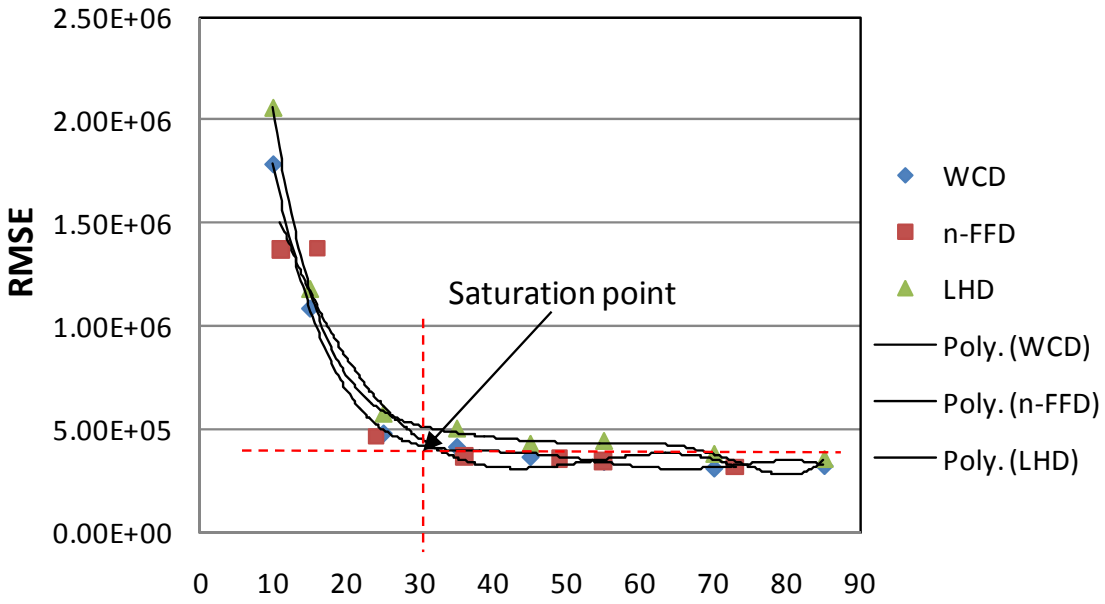


Fig. 3.6 The *RMSE* performance index against the sample number (in percentages).

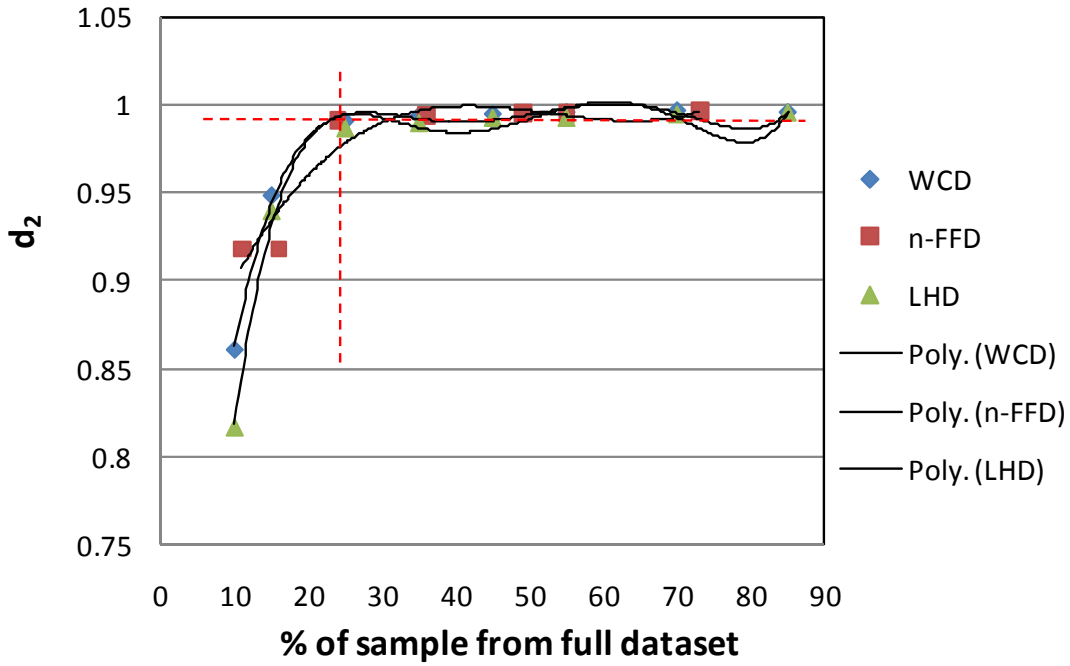


Fig. 3.7 The d_2 performance index against the sample number (in percentages).

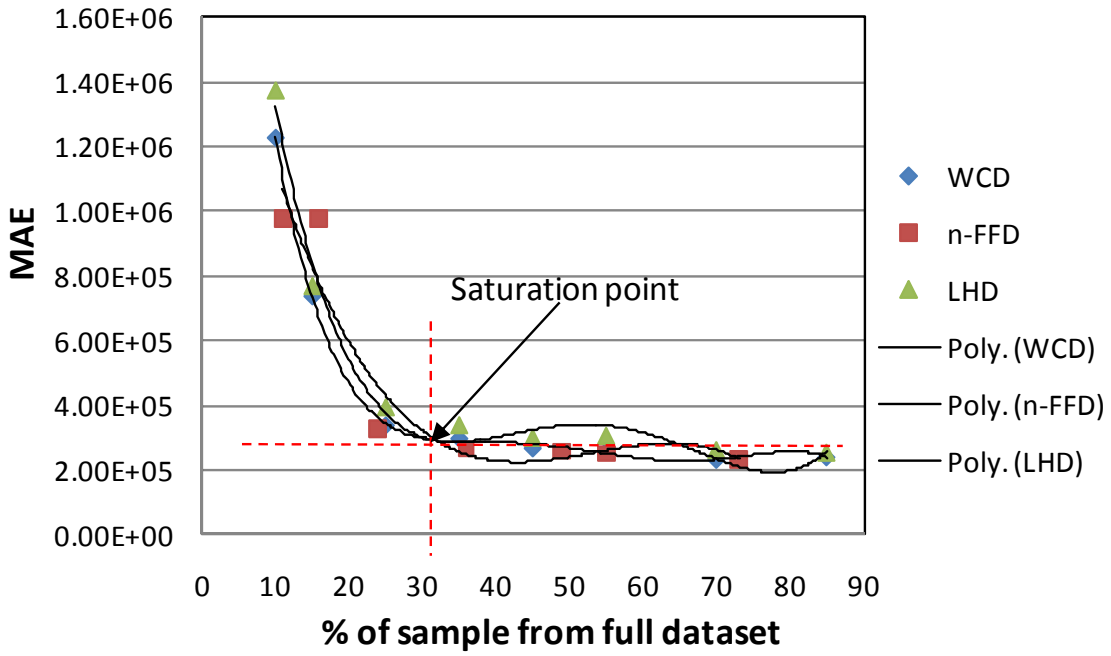


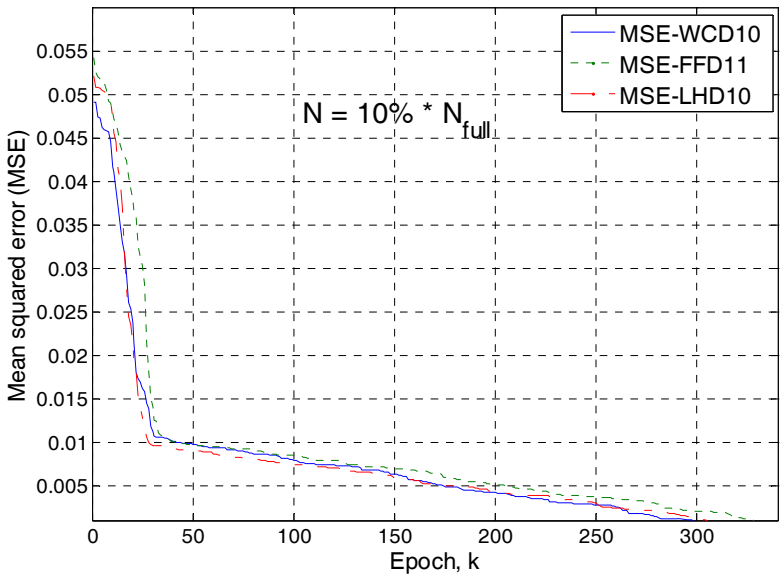
Fig. 3.8 The MAE performance index against the sample number (in percentages).

Next, the other two design methods are executed by using the same metamodel configurations. For the *n*-FFD method, each design variable is assigned a different number of levels so as to generate the different sampling sizes. For example, the *[2 3 3 2 3 3 3]* configuration of seven variables will give 972 sampling point locations, which are the product of the number of levels for each dimension. This design approach is affected in a uniform fashion by means of a rectangular grid of points. For the LHD technique, a 'maximin' metric which was introduced by Johnson et al. (1990) is considered in this study. This approach yields a randomised sampling plan, whose projections on to the axes are uniformly spread.

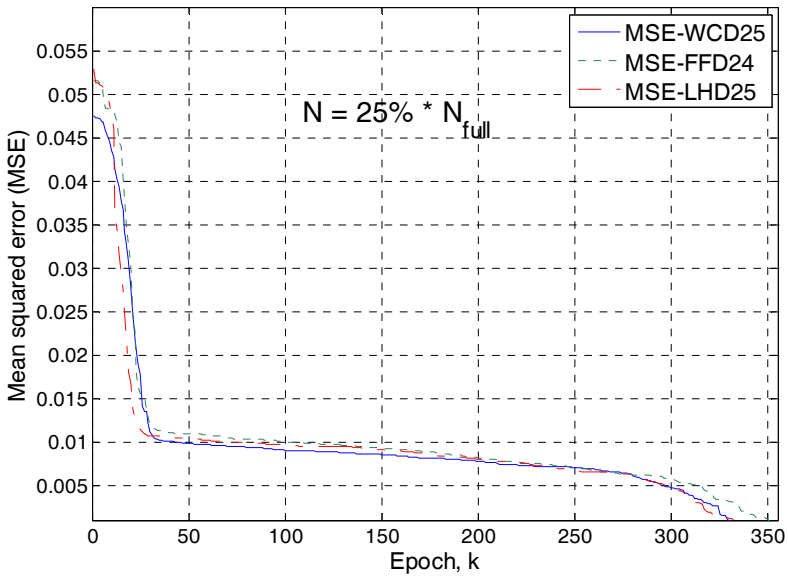
As compared to the *n*-FFD, at the same sample size, the proposed scheme (i.e. WCD) shows a great improvement in the size of constructed neural network, and produces nearly similar performance on the error indications. In the other comparison, the LHD requires a similar network size as the WCD, however poor in terms of the performance measure. Thus, in general, by compromising between the computational cost (i.e. execution time and the network size) and the performance of the model, the WCD method offers a better sampling solution.

The comparisons of the training evolution for different sampling numbers are illustrated in Fig. 3.9 (a–d). Therein, at the lower sampling numbers, the training performances in which the datasets are sampled by the WCD method reach the MSE goal of 0.001 faster than the benchmarked approaches. At the higher sampling numbers (more than 50% of the full dataset), the WCD and LHD have similar achievement in terms of the number of hidden neurons used; however, WCD obtains better performance indexes than LHD. An example of the estimated output for the case when the sample size is 30% of the full dataset is shown in Fig. 3.10. In the figure, the constructed metamodel is able to accurately approximate the true values at most of the points, except for the lower parts (i.e. less than zero level).

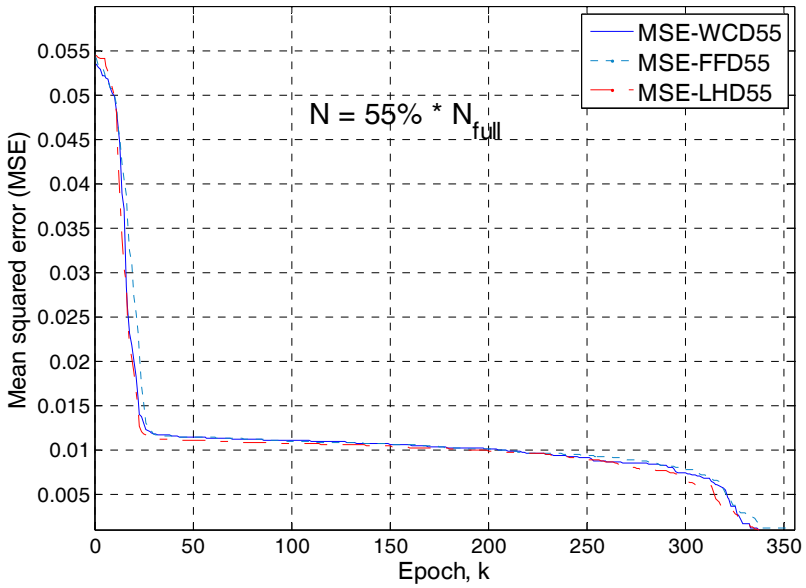
(a)



(b)



(c)



(d)

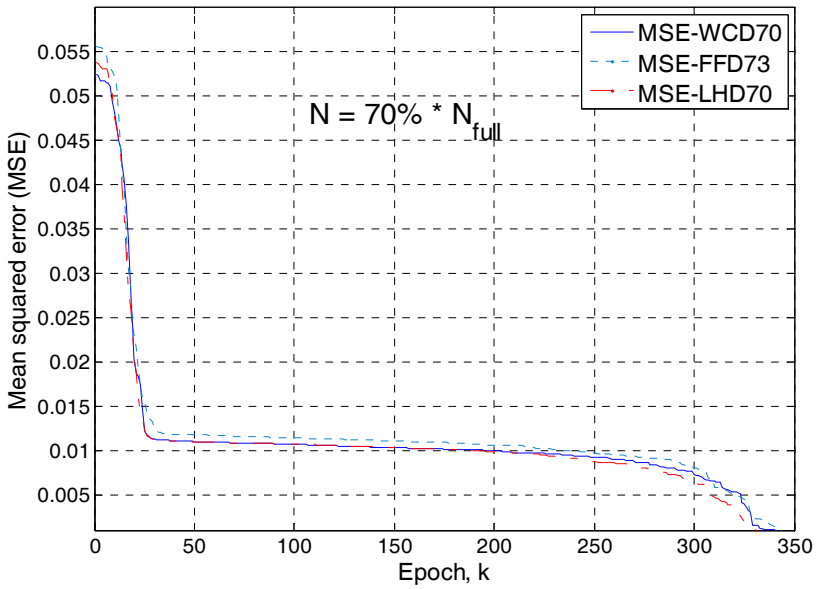


Fig. 3.9 The comparison of the training performance between three sampling methods, with several sample sizes.

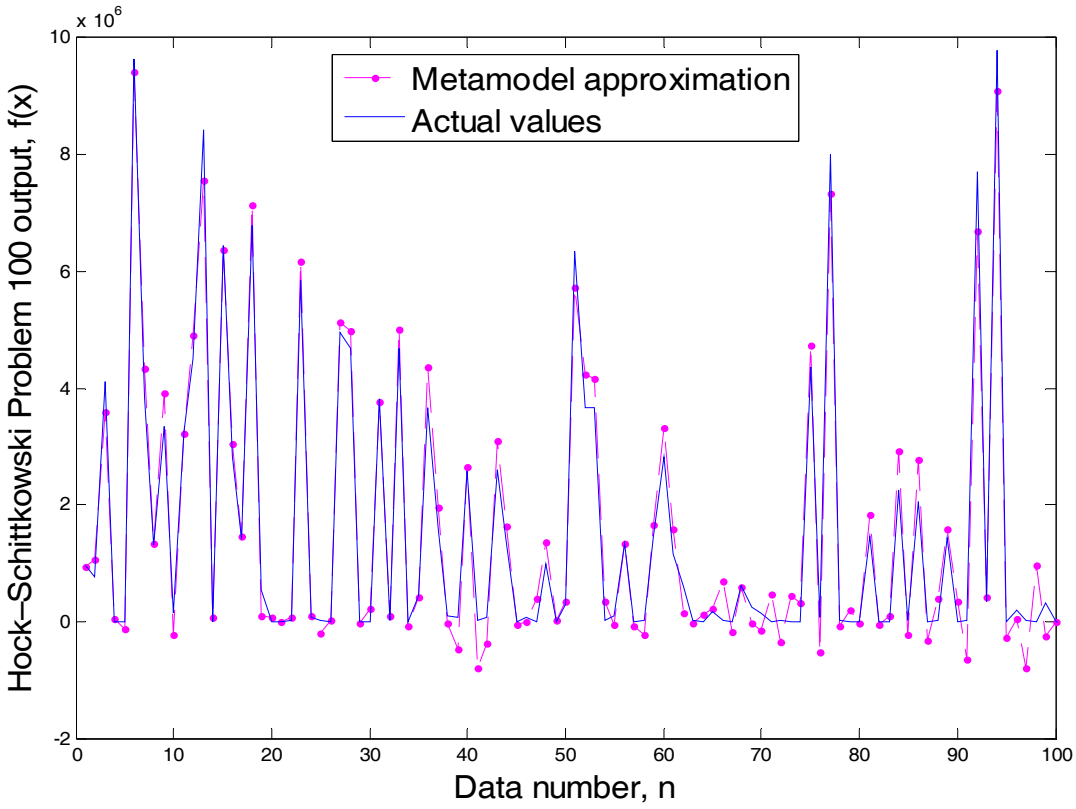


Fig. 3.10 The estimation output for 100 test data of problem 1 using N=1000 (i.e. case no. 3 in Table 3.1).

3.4.5. Discussion

A new method for the sampling design for a neural network metamodel has been presented. The validity and reliability of the proposed definition has been evaluated in several ways. By using the radial basis function neural network metamodel, the performance of the proposed approach was compared with two well-known sampling design strategies; the *n*-level full factorial design and the Latin hypercube design. A known non-linear test function, namely, The Hock–Schittkowski Problem 100 was used in the evaluation to validate the effectiveness of the proposed scheme. From this evaluation, for sampling a large dataset (typically more than 3000 design points), it has been determined that an appropriate sample number can be chosen between 25% to 30% of the full dataset as there are no significant improvements in the performance if more data is used. It is also noted that the proposed sampling method outperforms the other two evaluated methods in terms of several criteria, which are the performances indexes, the network size and the simulation time, in the tested function.

3.5. Chapter conclusion

This chapter covered the process involved in the metamodel development, which has focused on a neural network-based metamodel. The process includes data preparation and sampling, training process, and also validation and testing (evaluation) to assess the accuracy of a metamodel. Besides, the importance of some management strategies in executing the metamodel was also described. A reliable strategy has been presented, utilising the maximum potential of the sampling dataset to fit the metamodel. In the last section, a new method for the sampling scheme was introduced in which numerical analysis using a non-linear complex function was presented to evaluate the effectiveness of the proposed method. A significant improvement in the metamodel performance was found when using the weighted clustering design (WCD) for sampling the training dataset, as compared to benchmarked methods; the Latin hypercube design (LHD) and the *n*-level full factorial design (*n*-FFD).

Chapter 4
ORTHOGONAL LEAST SQUARES ALGORITHM

4.1. Introduction

In Chapter 2, a comprehensive review of the learning strategies of radial basis function neural network (RBFNN) has been discussed. Although RBFNNs have been proven to be able to model highly nonlinear data, their performance very much depends on the distribution of the centres. The strong point of RBFNN is that the input space can be clustered, and the centres are chosen in such a way that the radial basis function will have an effect only on certain regions of the input space (Jang et al., 1997). Consequently, RBFNN with clustering strategy reduces the "curse of dimensionality" problem that may reduce the over-fitting problem. However, when using a clustering method, a major concern is how to select the suitable set of RBF centres with the aim of reducing the network complexity as well as maintaining an adequate level of accuracy.

In this chapter, we will discuss a systematic method for the RBFNN learning algorithm in which the benchmark approach will be based on the *Orthogonal Least Squares* (OLS) method that was introduced by Chen et al. (1991). This is a well-known learning method and it has become a standard algorithm in the *Matlab* software toolbox. As well, some efforts will be demonstrated to improve the performance of the learning algorithm. First, an algorithm to optimally tune the spread parameter will be presented. Then, a method to prune the networks during the learning process will be introduced as well.

4.2. Orthogonal Least Squares (OLS) learning algorithm

The Orthogonal Least Squares (OLS) learning algorithm is a forward selection technique that computes the RBF centres or the significant terms from the input data (Lee & Billings, 2002), and the corresponding weights can be estimated in a very efficient manner. The minimisation of the cost function in the selection of a centre from input data is based on the computation of an error reduction ratio (*ERR*). The centres are chosen to maximise the *ERR*.

Recalling the RBFNN function from equations (2.10), (2.11) and (2.29) in Chapter 2, the Gaussian based RBFNN for i inputs and m outputs can be defined as:

$$\hat{y}_j = f(x) = \sum_{k=1}^{N} w_{jk} \exp\left(-\frac{\left\| x - c_k \right\|_2^2}{2\sigma^2} \right) + b_j, \qquad j = 1, 2, ...m, \tag{4.1}$$

where $x \in \Re^n$ is the input vector, j is the output index, $\| . \|_2$ denotes the Euclidean norm, w_{jk} are weights in the output layer, N is the number of hidden neurons (and centres), $c_k \in \Re^n$ are the RBF centres in the input vector space, σ is the spread parameter, and b_j is the bias of the network for each output.

The selection of centres in equation (4.1) can be estimated using a linear regression analysis (i.e. least squares method) if we have:

$$d(t) = \sum_{k=1}^{M} p_k(x(t))\theta_k + e(t), \tag{4.2}$$

where $d(t)$ is the desired output, p_k represents as regressors which are functions of $x(t)$, θ_k are parameters to be estimated, and $e(t)$ is an error signal included in the modelling. Rewriting (4.2) in matrix form, for $t = 1$ to N, yields:

$$D = P\theta + \xi, \tag{4.3}$$

where

$$D = \left[d(1), \ \ldots \ , d(N) \right]^T, \tag{4.4}$$

$$\theta = \left[\theta_1, \ \ldots \ , \theta_M \right]^T, \tag{4.5}$$

$$\xi = \left[e(1), \ \ldots \ , e(N)\right]^T, \text{ and} \tag{4.6}$$

$$P = \begin{bmatrix} p_1(1) & p_2(1) & \cdots & p_M(1) \\ p_1(2) & p_2(2) & \cdots & p_M(2) \\ \vdots & \vdots & & \vdots \\ p_1(N) & p_2(N) & \cdots & p_M(N) \end{bmatrix}. \tag{4.7}$$

An orthogonal decomposition of P is of the form:

$$P = TQ, \tag{4.8}$$

where Q is an $M \times M$ upper unit triangular matrix given as:

$$Q = \begin{bmatrix} 1 & \beta_{12} & \beta_{13} & \cdots & \beta_{1M} \\ 0 & 1 & \beta_{23} & \cdots & \beta_{2M} \\ 0 & 0 & 1 & \cdots & \beta_{3M} \\ \vdots & & & \ddots & \vdots \\ 0 & \cdots & 0 & \cdots & 1 \end{bmatrix}, \tag{4.9}$$

and T is a $N \times M$ matrix in which the orthogonal columns satisfy:

$$T^T T = H = \begin{bmatrix} h_1 & 0 & \cdots & 0 \\ 0 & h_2 & & 0 \\ \vdots & & \ddots & \\ 0 & 0 & & h_M \end{bmatrix}, \tag{4.10}$$

with the element $h_i = t_i^T t_i$ for $1 \le i \le M$.

The classical Gram–Schmidt (CGS) procedure computes Q one column at a time and orthogonalises P as follows: at the *k*-th stage, make the *k*-th column orthogonal to each of the $k-1$ previously orthogonalised columns and repeat the operations for $k = 2, \ldots, M$. The computational procedure is represented as:

$$\left.\begin{aligned} t_1 &= p_1 \\ \beta_{1k} &= \frac{\langle t_i, p_k \rangle}{\langle t_i, t_i \rangle}, \quad \text{for } 1 \le i \le k \\ t_k &= p_k - \sum_{i=1}^{k-1} \beta_{ik} t_i \end{aligned}\right\} k = 2, \ldots, M \tag{4.11}$$

where $\langle ., . \rangle$ denotes the inner product that is $\langle t_i, p_k \rangle = t_i^T p_k$.

From equation (4.8) we have $T = PQ^{-1}$, and rearranging equation (4.3) yields:

$$D = (PQ^{-1})(Q\theta) + \xi = Tg + \xi, \tag{4.12}$$

and the linear least squares estimate of g is given by:

$$\hat{g} = (T^T T)^{-1} T^T D \qquad or \qquad \hat{g}_i = \frac{t_i^T D}{t_i^T t_i}. \tag{4.13}$$

Next, the estimated weight $\hat{\theta}$ can be determined through the back substitution of $Q\theta = g$.

The sum squared errors cost function is given by:

$$J = \xi^T \xi, \tag{4.14}$$

By using (4.12) and (4.13), the cost function can be re-written as:

$$J = D^T D - \sum_{i=1}^{M} g_i^2 t_i^T t_i. \tag{4.15}$$

The error reduction ratio (ERR) due to each additional t_i can be defined as:

$$ERR_i = \frac{g_i^2 t_i^T t_i}{D^T D}, \qquad \text{for } 1 \le i \le M. \tag{4.16}$$

Based on the ERR_i, a simple and effective forward-selection procedure can be derived for choosing the RBF centres. This can be considered as finding a subset of significant regressors. The regressor selection procedure by using the CGS method is summarised as follows:

- At the first step (i.e. $k = 1$), for $1 \le i \le M$, compute:

$$\left.\begin{aligned} t_1^{(i)} &= p_i \\ g_1^{(i)} &= \frac{\left\langle t_1^{(i)}, D \right\rangle}{\left\langle t_1^{(i)}, t_1^{(i)} \right\rangle} \\ ERR_1^{(i)} &= \frac{\left(g_1^{(i)}\right)^2 \left\langle t_1^{(i)}, t_1^{(i)} \right\rangle}{\langle D, D \rangle} \end{aligned}\right\}. \tag{4.17}$$

  Find

$$ERR_1^{(i1)} = \max\left\{ ERR_1^{(i)} \right\}, \text{ and select } t_1 = t_1^{(i1)} = p_{i1} \tag{4.18}$$

- At the *k*-th step where $k \geq 2$, compute:

$$\left.\begin{aligned} \beta_{jk}^{(i)} &= \frac{\langle t_j, p_i \rangle}{\langle t_j, t_j \rangle}, \quad \text{for } 1 \leq j \leq k \\ t_k^{(i)} &= p_i - \sum_{j=1}^{k-1} \beta_{jk}^{(i)} t_j \\ g_k^{(i)} &= \frac{\langle t_k^{(i)}, D \rangle}{\langle t_k^{(i)}, t_k^{(i)} \rangle} \\ ERR_1^{(ik)} &= \frac{\left(g_k^{(i)}\right)^2 \langle t_k^{(i)}, t_k^{(i)} \rangle}{\langle D, D \rangle} \end{aligned}\right\}. \tag{4.19}$$

  Find

$$ERR_k^{(ik)} = \max\left\{ ERR_k^{(i)} \right\}, \text{ and select } t_k = t_k^{(ik)} = p_{ik} - \sum_{j=1}^{k-1} \beta_{jk} t_j \tag{4.20}$$

- The procedure is continued until the M_s-th step when:

$$1 - \sum_{j=1}^{M_s} ERR_j < \rho, \tag{4.21}$$

  where $0 < \rho < 1$ is the desired tolerance.

For practical implementation in MATLAB code, the p_i in the first step is set as an orthogonal and diagonal matrix of P (Demuth et al., 2009), which is given by:

$$P = radbas\ (dist\ (x^T, x) * \sigma_c)), \tag{4.22}$$

where x is the input space vector, $radbas$ is the radial basis function, $dist(.)$ is the Euclidean distance function, and σ_c is the spread parameter. To improve the sensitivity of the radial basis function, the authors suggested that the spread value is given by $\sigma_c = \sigma_c'/0.8326$ in which σ_c' is the user-defined spread parameter. This gives radial basis functions that cross 0.5 at weighted inputs of $+/-$ spread. To terminate the learning process, Demuth et al. (2009) use the set mean-squared-error (*MSE*) goal, rather than use equation (4.21).

The OLS strategy is a systematic and effective way of selecting centres as compared to random selection, but it is possibly a suboptimal solution (Sherstinsky & Picard, 1996). The minimal network structure may not be found for a given accuracy since the centres or the significant terms are selected based on a local optimisation. Previously chosen terms could affect the selection of subsequent terms as the *ERR* varied with the order in which the significant terms were orthogonalised into the orthogonal equation. On the other hand, the accuracy of constructed network is still dependent on the spread parameter value that affects the variance of each hidden neuron in which it is typically selected as a constant sp. In the next section, a potential method will be introduced to adaptively tune the spread parameter to possibly find the optimal parameters of basis functions.

4.3. Some improvements of OLS algorithm

4.3.1. Adaptively-tuned spread parameter

4.3.1.1. Introduction

In MATLAB, the spread parameter σ, is quite often set manually by trial and error. There still remains a question as to whether the results obtained are at the optimum point for various spread constants. As mentioned in Demuth et al. (2009), it was suggested that σ should be large enough so that neurons respond strongly to the overlapping region of the input space. Also, it must be selected at greater than 0.1 of the interval between inputs, and less than 2 of the distance between the leftmost and rightmost inputs. But it is still unclear with which spread parameter one should start, especially for beginner users. Thus, it may be interesting to tune for some optimal value of σ (Poshal & Ganesan, 2008).

4.3.1.2. Methodology

There are two possible implementations to learn the optimal value of the spread parameter. The first method is by using the fully supervised method to train the optimal points of RBFNN's weights, centres and spreads, the update equations having been discussed in Chapter 2 (see equations (2.28)–(2.20)). In this approach,

the gradients of the update equations are computed by minimising the sum of squared errors (*SSE*) as the cost function. The simultaneous calculation of the three equations incurs a significant computation cost.

Another possible approach is by computing the gradient using the slope information by taking the change of a cost function against the change of a variable to be learned, from their current and previous values. Here, the root-mean-squared-error (*RMSE*) is used as the cost function to train the spread value for every hidden neuron, the OLS algorithm is used to trains its centres, and the output weights are learned by the inversed linear regression (Wahid et al., 2010a). This idea can be illustrated here by testing a simple *quasi-sinusoidal* function. The *n*-dimensional *quasi-sinusoidal* function is generally defined as (Giunta & Watson, 1998):

$$f(x) = \sum_{i=1}^{n} \left[0.3 + \sin(16x_i/15 - \varepsilon) + \sin^2(16x_i/15 - \varepsilon) \right], \qquad (4.22)$$

where $\varepsilon = 0.7$, $i = 1$ for a one-dimensional problem, and $x_i \in [-1,2]$. The initial sample set is constructed by using eight randomly distributed points. By using the OLS algorithm in MATLAB and varying the spread parameter values, it is observed that the produced cost function (i.e. *RMSE*) decays until reaching a certain minimum point. This minimum point is considered as the optimal point of spread parameter, as shown in Fig. 4.1.

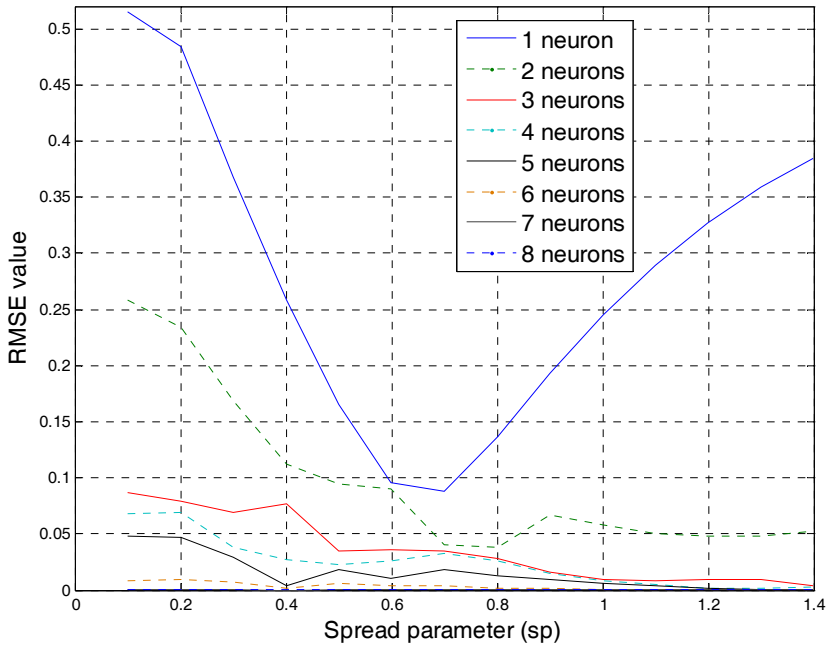


Fig. 4.1 *RMSE* value versus the spread parameter with various neuron numbers from a case study.

This illustration prompts the suggestion that an optimal value for the spread parameter is a function of *RMSE* at the minimum global point. Motivated by this idea, we propose to use the gradient criteria to adjust σ till the first derivative of *RMSE* approaches zero, i.e.

$$\nabla\sigma = \frac{\partial(RMSE)}{\partial\sigma} = 0. \tag{4.23}$$

From a given initial value σ_0, the optimal point can be achieved by an optimisation technique such as steepest descent, Newton's method, and the Marquardt method. It has been determined by many experiments that the σ_0 must be chosen between 0.1 and 10 in order to obtain the best convergence. In this development, we choose the steepest descent (Jacob, 1988) technique for simplicity, which method appears to be the best unconstrained minimisation technique (Rao, 2009). However, owing the fact that the steepest descent direction is a local property, it may fall into the local minimum rather than the global minimum. The procedure can be written as:

$$\sigma_{(new)} = \sigma_{(old)} - \beta\nabla\sigma\ , \tag{4.24}$$

where σ is the spread parameter, β is the step size (or learning rate) and $\nabla\sigma$ is the gradient as derived in (4.23). Commonly, the following criteria can be used to terminate the iterative process.

In order to terminate the iterative process, the following criteria can be used:

i. when the change in function value and two consecutive iterations are small, i.e.:

$$\left|\frac{RMSE_{(new)} - RMSE_{(old)}}{RMSE_{(old)}}\right| \le \varepsilon_1\,, \tag{4.25}$$

ii. or, when the component of the gradient of the function is small, i.e.:

$$\left|\frac{\partial RMSE}{\partial\sigma}\right| \le \varepsilon_2\,, \tag{4.26}$$

iii. or, when the change in design vector in two consecutive iterations is small:

$$\left|\sigma_{(new)} - \sigma_{(old)}\right| \leq \varepsilon_3 , \tag{4.27}$$

where ε_1, ε_2, and ε_3 are the suitable threshold points. Here, the first termination criterion is utilised because it shows better performance as compared to the rest.

The whole picture of the proposed algorithm is illustrated in the flowchart depicted in Fig. 4.2. Therein, from the training dataset (i.e. input dataset p, and target dataset *t*), the process starts by initialising the error goal (*eg*) and the initial value of the spread parameter (σ_0). By using the OLS algorithm, an RBF centre is selected, and the output weights and RMSE are computed. The optimal σ for first neuron is obtained by the steepest descent method until it reaches the termination point ε. If the first neuron is unable to meet the set error goal (*eg*), it will proceed to the next iteration where the new neuron is added. The new optimal σ will be calculated and the process continues until it meets the required accuracy, or will stop when the maximum number of neurons, which is equal to the number of inputs is reached.

The advantageous feature here is that the best performance is determined at every neuron, instead of using a fixed spread value in the standard RBFNN. The Matlab codes (*m-codes*) for the improved OLS algorithm are given in the Appendix B-1.

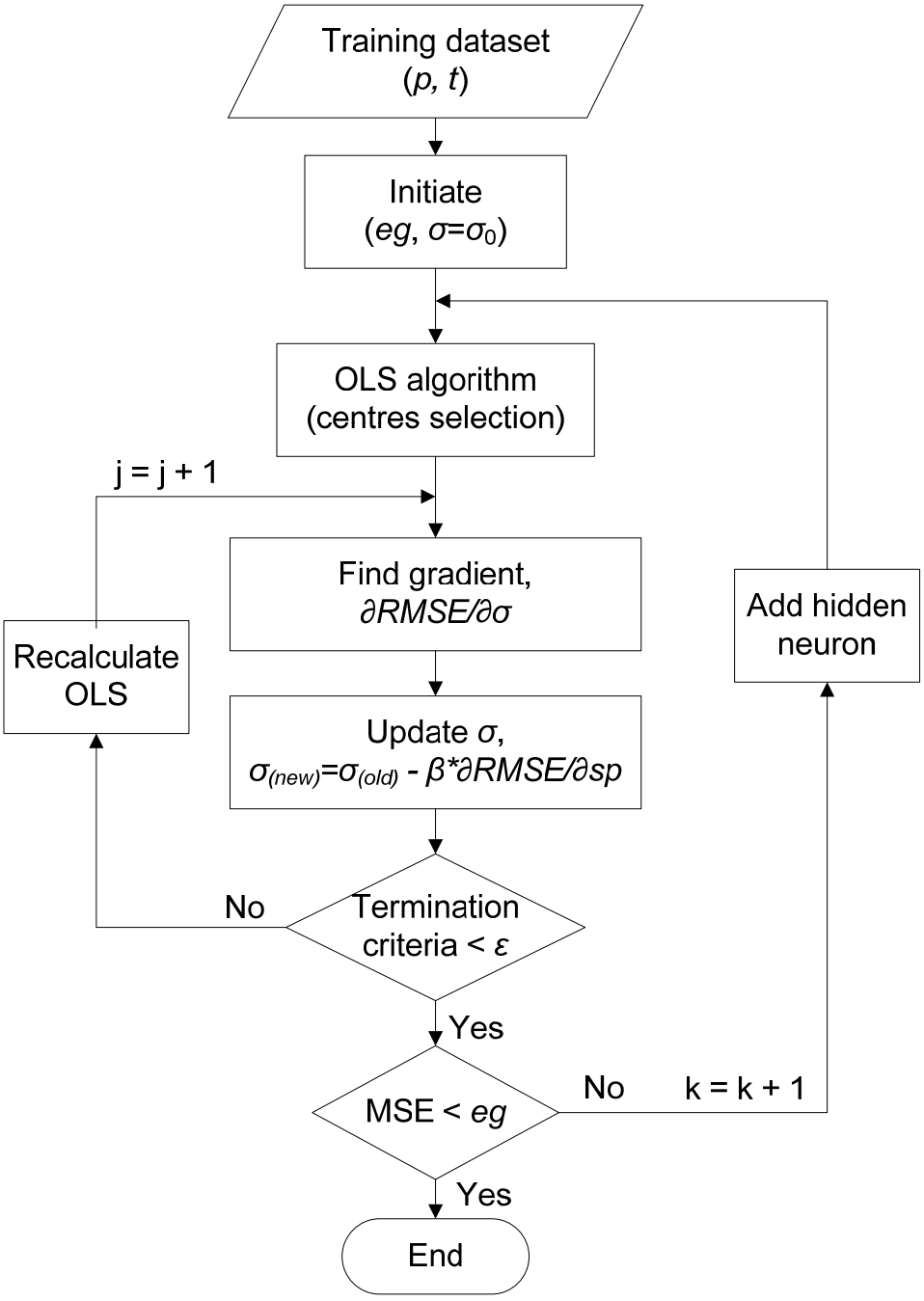


Fig. 4.2 RBFNN with adaptively tuned spread parameter algorithm flowchart.

4.3.2. A pruning algorithm for RBFNN

4.3.2.1. Introduction

In most of the developed RBFNN algorithms, the created hidden units will never be removed. This leads the network to produce some hidden units which are initially active but those end up contributing little to the network output. Thus, an appropriate method to prune the network is necessary in which inactive hidden units can be detected and removed during learning to produce a less wasteful network.

Liu et al. (1999) proposed a variable neural network based on the variable grid approach. The network selects the centres from the node set $\mathbb{N}$ of the variable grid. When the network needs some new basis functions, a new higher order subgrid is appended to the grid, and the new centres are chosen from the newly created subgrid. Similarly, if the network needs to be reduced, the highest order subgrid is deleted from the grid, and the associated centres are removed. In another paper, Mekki et al. (2006) applied a variable neural network of RBFNN for adaptive control of a nonlinear system, which combines a growth algorithm inspired from the adaptive diffuse element method with a pruning algorithm introduced by Fabri and Kadirkamanathan (1996). Basically, this approach also uses a grid based method in the selection of the hidden unit with a slightly different approach, and two criteria with their thresholds are used to assess the activeness of the current hidden units and remove them if necessary.

From the idea of the above papers, a new potential pruning method applied in the OLS learning algorithm will be introduced here. The method is simple to be implemented and requires only one criterion for its decision (Wahid et al., 2010a).

4.3.2.2. Methodology

For $m=1$ and $\mathrm{x}^{(i)} \in \Re^n$, where n is the number of training examples, the equation for RBFNN output in (4.1) can be rewritten in matrix form as:

$$F = W^T \Phi, \tag{4.28}$$

where

$$F = \left[f^{(1)}, \ \ldots \ , f^{(n)} \right], \tag{4.29}$$

$$W = \left[w_1, \ \ldots \ , w_k \right]^T, \text{ and} \tag{4.30}$$

$$\Phi = \begin{bmatrix} \phi_{11} & \phi_{12} & \cdots & \phi_{1n} \\ \phi_{21} & \phi_{22} & \cdots & \phi_{2n} \\ \vdots & \vdots & & \vdots \\ \phi_{k1} & \phi_{k2} & \cdots & \phi_{kn} \end{bmatrix}, \text{ or } \Phi = \begin{bmatrix} \phi_1^{(1)} & \phi_1^{(2)} & \cdots & \phi_1^{(n)} \\ \phi_2^{(1)} & \phi_2^{(2)} & \cdots & \phi_2^{(n)} \\ \vdots & \vdots & & \vdots \\ \phi_k^{(1)} & \phi_k^{(2)} & \cdots & \phi_k^{(n)} \end{bmatrix}. \tag{4.31}$$

Each radial basis of the hidden units from ϕ_1 to ϕ_k has its own strength of contribution to the build network. The energy could be assessed by computing the mean value of n numbers of radial basis output (i.e. RBF output of each training pattern) for each hidden neuron, as expressed in the following equation:

$$\bar{\phi}_i = \frac{1}{n} \sum_{j=1}^{n} \phi_i^{(j)}, \qquad i = 1,2,\cdots,k,, \tag{4.32}$$

where ϕ is the hidden layer output, i is the neuron number, and j is the pattern number. It is followed by normalising those mean outputs between 0 and 1, which resulted in a normalised $\bar{\phi}_i$ output, as given in the following equation:

$$s_i = \left\| \frac{\bar{\phi}_i}{\bar{\phi}_{(\max)}} \right\|, \qquad i = 1,2,3...k \, . \tag{4.33}$$

When the s_i value is less than a certain threshold, δ_s during the training process, the i^{th} hidden node could be removed. To incorporate the method into the OLS with the adaptive spread algorithm (in Fig. 4.2), additional steps are added between the adaptive spread termination criteria and the MSE goal, which is shown in Fig. 4.3. Please refer to the pruning algorithm codes in Appendix B-2.

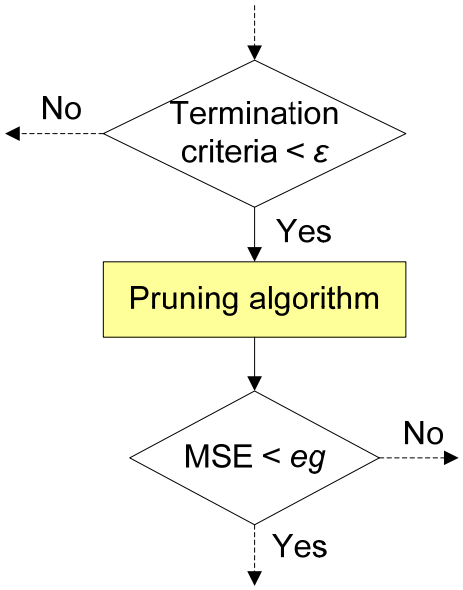


Fig. 4.3 RBFNN with an additional pruning algorithm.

4.3.3. Numerical analysis

To evaluate the performance of both proposed approaches, a nonlinear test function which was used in Chapter 3 will be utilised again here, namely the Problem 100 of Hock–Schittkowski. The WCD sampling method is used to prepare an appropriate sampling dataset for the purpose of this evaluation. Several types of simulations will be considered which involve the standard OLS algorithm, a combination of OLS with adaptively tuned spread parameter, and a combination of those three algorithms. Some of the performance measures, the size of RBFNN metamodel and the total execution time for the simulations will also be noted.

4.3.3.1. The OLS performance

First, the performance of the training process using the OLS algorithm is evaluated for the purpose of comparison. One thousand data points that have been sampled from 4000 points of the full dataset are involved in the analysis. The best achievement of this algorithm is basically dependent on the predefined values of the error goal and the spread parameter. Using the test problem, the performance of the algorithm is tested by varying the *MSE* goal values at a constant spread parameter of 1.0, which are shown in Fig. 4.4 (a–d).

It is general understood that the performance of the algorithm increased by setting a higher accuracy target, which also causes an increment in size of the network (i.e. given by the number of hidden neurons used). The lowest typical value used is 0.001 in which over-fitting problems always occur if the value is below that. However, for many cases, the error goal of 0.001 does not always produce an optimal solution as it may build unnecessary additional hidden units that could degrade its performance, thus a higher value may be chosen. Another possible solution is by developing an algorithm that can make an early stop decision, for example, by adding a cross validation step in the loop.

In another test, we look at the effect of changing the spread parameter (σ) on performance. The σ used here is called as an 'isotropic σ' which means the same value of σ is used for two purposes as follows:

i. To prepare a finite set of matrix *P* in equation (4.22) for the selection of radial basis centres using OLS algorithm. To avoid confusion, we note it as σ_c in this assessment.
ii. To produce the radial basis function for each hidden neuron during the training stage. The spread parameter is noted here as σ.

The results of testing several spread parameters with a constant *MSE* goal of 0.001 for the OLS algorithm is listed in Table 4.1. From the table, the optimum solution is found when the σ is set to 6.0. The network apparently converges very quickly when σ equals 0.1, but it produces the worst value in the performance measure. At a higher σ value, the performance increases such that the optimum performance with a smaller network size occurs at σ equals 6.0. When the σ is set to more than about 10, the network in which huge numbers of hidden neurons are used would not converge well. It should be keep in mind that the situation is not the same for a different test problem, hence a suitable mechanism is necessary to determine the appropriate value of isotropic σ or the optimal value of σ for each hidden unit.

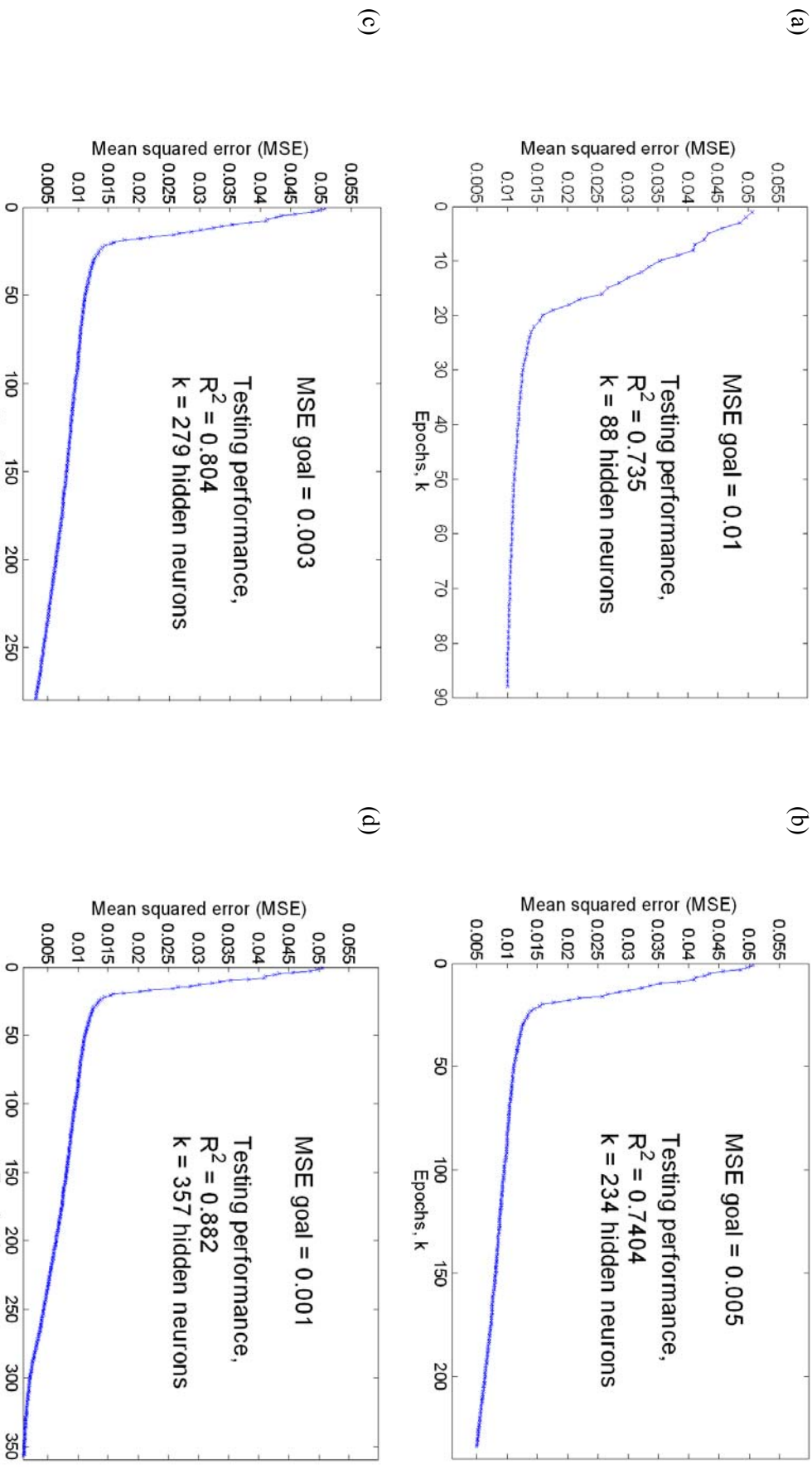


Fig. 4.4 The evolution of OLS learning process using different *MSE* goals.

Table 4.1 The performance of OLS using different spread parameters (i.e. $\sigma = \sigma_c$).

Spread parameter	R^2	Epochs
0.1	0.013	246
0.5	0.761	462
1.0	0.882	357
2.0	0.939	299
4.0	0.939	284
6.0	**0.943**	**279**
8.0	0.947	283
10.0	0.950	280
11.0	0.941	718

4.3.3.2. The OLS with adaptive tuned spread parameter

There are two parameters that have to be determined in the proposed tuning algorithm for the spread parameters (σ), which are the termination threshold ε and the learning rate β for the gradient descent. These parameters must be generic in offering a reliable result for training of any test function. For a demonstration, a similar test function as that used in the previous section is used to test the effect of the threshold, ε, values (in equation 4.25) to the accuracy of σ determination where the results are depicted in Fig. 4.5 (a–d). In this evaluation, for simplicity, a fixed value for the learning rate β is used in the iteration process. More effective ways may be implemented by varying values of β in every loop of iteration to speed up the computation.

In Fig. 4.5, the spread parameters are varied during the training process, and it is found that a lesser number of hidden units are produced when a smaller threshold value is used. The threshold values as in Fig. 4.5 (c) and (d) produce similar neuron numbers as those produced by the standard OLS algorithm. However, the results in (d) involve extremely expensive computation time, thus the threshold value of $\varepsilon = 0.00001$ (i.e. in (c)) is chosen so as to compromise between the performance and the computational cost. It is also observed that the mean values of the σ are increased with the reduction of the threshold value, but it does not much affect network performance. More comprehensive results of the evaluation for different σ_c values (spread parameter for centres selection) are shown in Table 4.2.

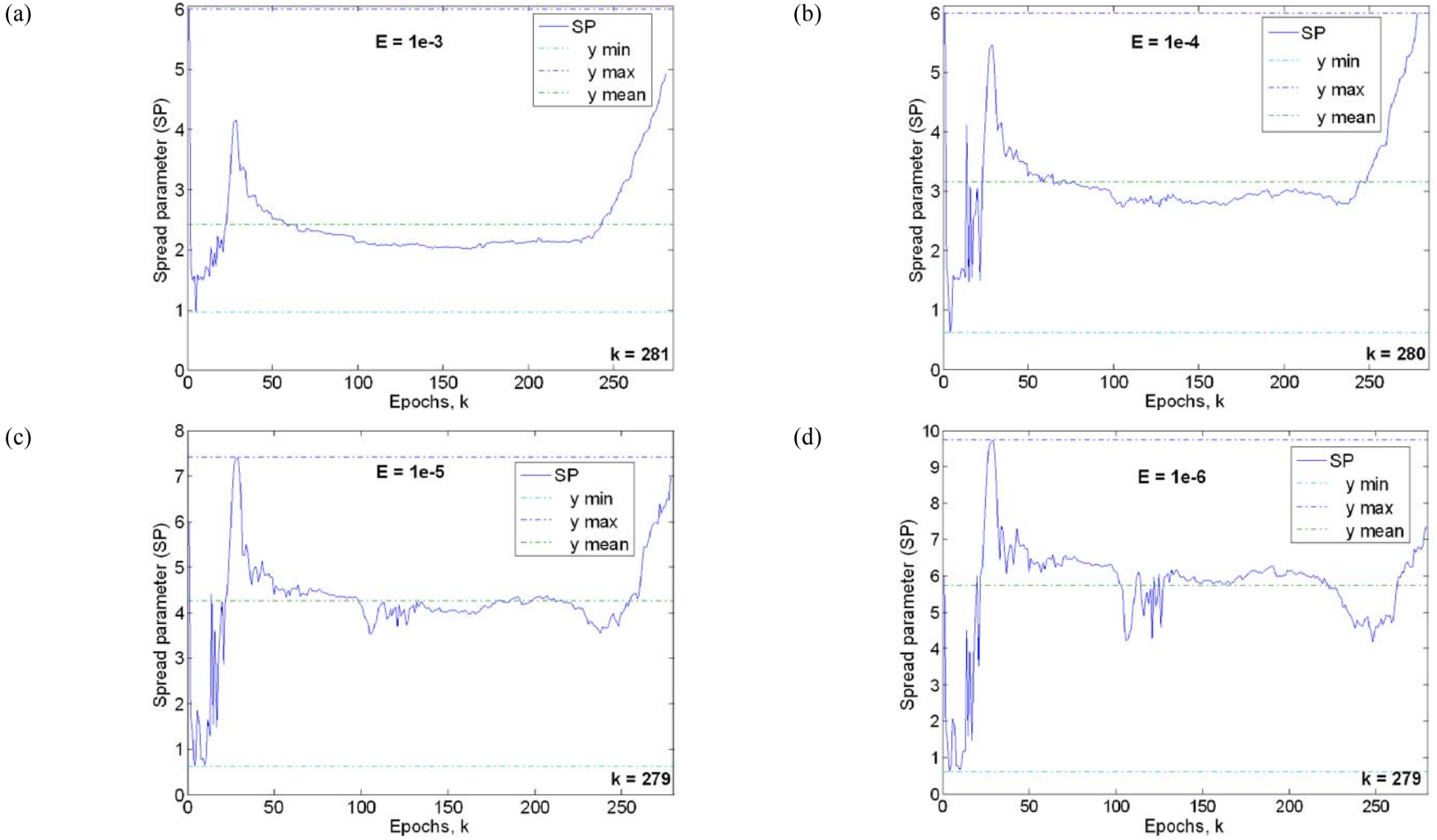


Fig. 4.5 The variations of spread parameters of the hidden units using different threshold values (ε = 0.001, 0.0001, 0.00001, 0.000001), and σ_c = 6.0 for the centres selection.

Table 4.2 The performance of improved OLS using different σ_c parameters (*MSE* goal=0.001).

σ_c	Threshold, ε	σ_{min}	σ_{max}	σ_{mean}	Epochs, k	R^2	Time (s)
1.0	1e-3	0.556	2.379	1.665	296	0.926	67
	1e-4	0.714	3.175	2.056	296	0.928	199
	1e-5	0.714	4.460	2.592	296	0.928	763
	1e-6	0.494	6.384	3.360	296	0.926	3576
2.0	1e-3	1.018	3.463	2.139	287	0.937	132
	1e-4	0.983	3.888	2.716	287	0.939	455
	1e-5	0.986	5.315	3.522	287	0.940	1254
	1e-6	0.488	7.417	4.609	287	0.940	6627
4.0	1e-3	0.960	4.670	2.193	283	0.941	123
	1e-4	0.960	4.670	2.193	283	0.941	107
	1e-5	0.507	7.624	3.882	281	0.938	2184
	1e-6	0.507	10.240	5.196	281	0.938	9168
6.0	1e-3	0.960	6.000	2.423	281	0.948	144
	1e-4	0.626	6.000	3.155	280	0.943	542
	1e-5	**0.611**	**7.408**	**4.201**	**279**	**0.943**	**2085**
	1e-6	0.605	9.742	5.731	279	0.943	> 2 hr
8.0	1e-3	-4.968	8.000	2.355	284	0.946	136
	1e-4	-6.447	8.000	3.030	283	0.946	585
	1e-5	-8.541	8.055	4.109	283	0.946	2157
	1e-6	Simulation more than two hours					
11.0	1e-3	1.500	11.000	2.351	283	0.946	135
	1e-4	1.131	11.000	3.051	282	0.944	569
	1e-5	1.560	11.000	4.115	282	0.944	2009
	1e-6	Simulation more than two hours					

From Table 4.2, it is clear that the training performance is increased with an increment in the σ_c value. The optimal solution appears when $\sigma_c = 6.0$ and the threshold $\varepsilon = 0.00001$. In the previous discussion of the OLS analysis, the network seems over-fitted when $\sigma_c > 11$ is used, however, with the improved algorithm, this limitation is compensated, however, the network size enlarged and the network performance deteriorated when the σ_c value is increased.

Some comparisons of the training performance between standard OLS and the improved OLS algorithm are illustrated in Fig. 4.6 (a–d). From this comparison, it is shown that the improved algorithm outperforms the regular OLS in terms of the network size, especially when a lower σ_c value is utilised. An identical performance appears when $\sigma_c = 6.0$. On the other hand, the proposed improvement also has the advantage of yielding a reasonable performance even when the optimum σ_c value is unknown. It is also pointed out that instead of tuning the σ values, an optimal σ_c value also needs to be determined to achieve a lesser number of hidden neurons.

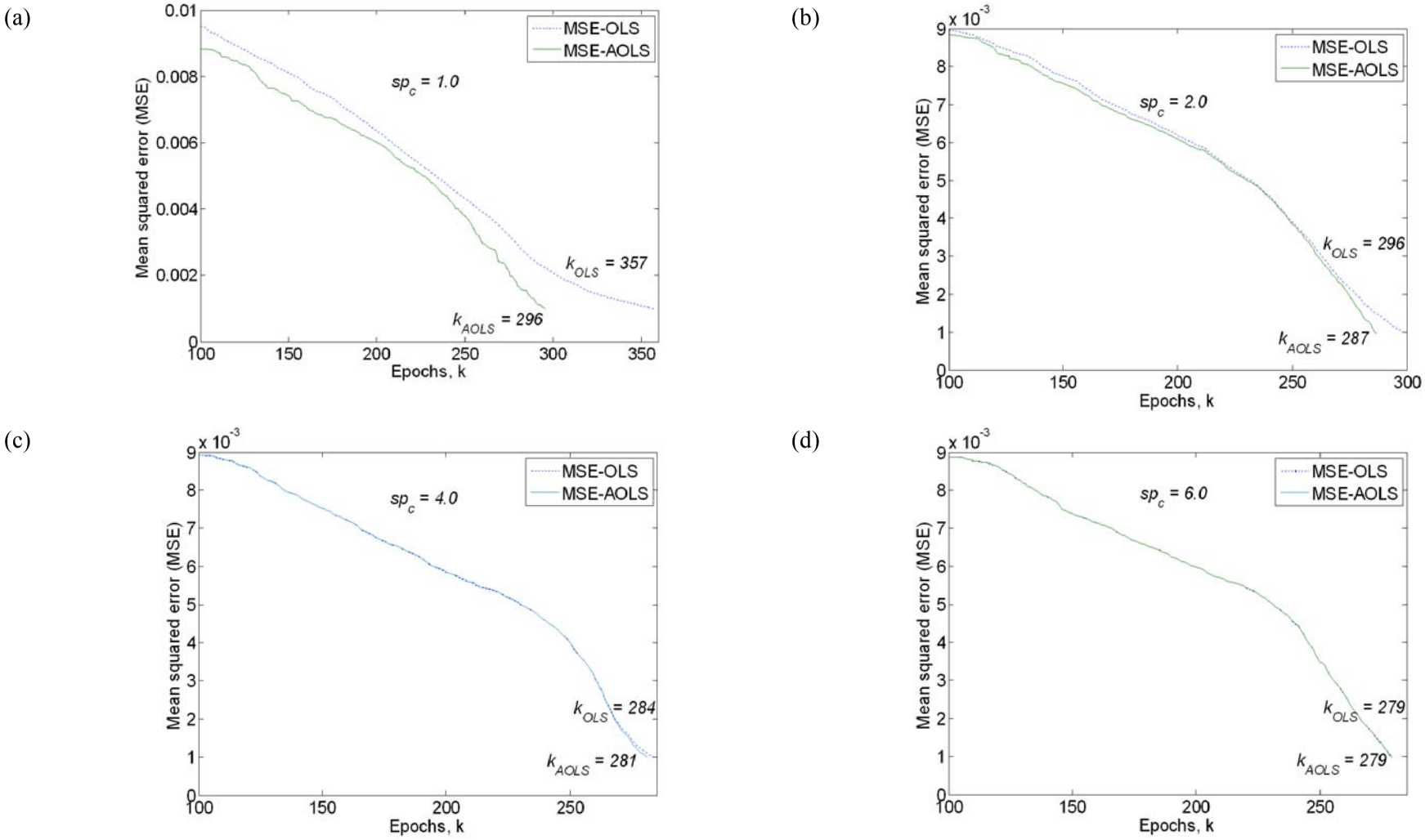


Fig. 4.6 The training performance of OLS and adaptive-OLS using different σ_c (or sp_c) values, at a threshold value, ε of 0.00001 and *MSE* goal of 0.001.

4.3.3.3. The OLS with adaptive spread and pruning algorithm

In this evaluation, a combined algorithm of OLS, adaptive spread and pruning method is considered. In the algorithm, an appropriate threshold of δ_s needs to be determined by running several experiments and considering several test functions. For the execution, in every epoch, the strength of the node (i.e. radial basis output of the node) is evaluated by using equations (4.32) and (4.33). If the s_i value is more than the threshold δ_s, the normal iteration is continued by adding a new neuron. If the s_i value falls below the threshold, the particular neuron is excluded and the number of excluded neuron n, is increased (i.e. $n=n+1$) in which the initial value of n was set to zero.

Table 4.3 The comparison of training performance using three methods, at different σ_c values and *MSE* goals.

σ_c	MSE goal	OLS		OLS + ASP*		OLS + ASP + PR**	
		Neurons	R^2	Neurons	R^2	Neurons	R^2
1.0	0.010	88	0.735	41	0.789	41	0.790
	0.005	234	0.740	227	0.761	230	0.763
	0.001	357	0.882	296	0.929	297	0.931
2.0	0.010	50	0.776	41	0.789	38	0.789
	0.005	232	0.753	227	0.761	231	0.765
	0.001	299	0.939	287	0.940	287	0.940
4.0	0.010	40	0.789	40	0.789	38	0.789
	0.005	231	0.754	231	0.754	243	0.768
	0.001	284	0.939	281	0.938	319	0.967
6.0	0.010	41	0.792	41	0.792	43	0.947
	0.005	231	0.771	231	0.771	255	0.791
	0.001	279	0.943	279	0.943	321	0.952

* ASP = adaptive spread parameter, ** PR = pruning algorithm

By using the same test problem, a comparison of the training performance using three approaches is shown in Table 4.3. Several σ_c and MSE goal values are used in the testing. It can be learnt that the combination of OLS and the adaptive spread algorithm will reduce the size of the network while maintaining a similar performance to OLS (or better in some cases). In general, the addition of the pruning algorithm will tend to increase the number of neurons, however, it will yield improved results in terms of the performance index. As referred to in Table 4.3, the R^2 values of the third approach (i.e. OLS + ASP + PR) are better than the rest of

approaches, for all the cases. In some cases of a higher *MSE* goal (e.g. 0.01), this approach also appeared to yield smaller network sizes.

4.4. Chapter conclusion

This chapter introduced two improvements in the RBFNN algorithm based on the orthogonal least squares (OLS) approach. Initially, the steps of OLS implementation have been highlighted for the purpose of deriving the proposed algorithms. Next, an algorithm to adaptively tune the spread parameter (σ) was explained. Instead of using a constant σ for the entire training process, the σ of each hidden neuron is updated by using the gradient descent method. The root mean squared error (*RMSE*) is used as the cost function and the slope of the change of a cost function against the change of σ is used in the adaptation rather than taking the derivation of the cost function, so as to quicken the computation. From a numerical evaluation, the proposed algorithm will tend to offer smaller network sizes whilst maintaining the performance of the developed metamodel.

The second improvement involves the introduction of a pruning algorithm to optimise the number of hidden neurons in the network. The idea is to exclude the hidden neurons as they make little contribution in the development of the network. From an analytical standpoint, the algorithm improves the network's performance based on the performance index, and it is able to reduce the number of hidden unit especially when a lesser accuracy of the error goals is required.

Chapter 5
PARAMETER DETERMINATION FOR RBFNN LEARNING ALGORITHM

5.1. Introduction

In the radial basis function neural network (RBFNN), three difficulties are involved in the training algorithm: (i) the selection of the radial basis centres; (ii) the selection of the basis function radius (spread); and (iii) the learning of network weights. The choice of network centres is crucial, as it will affect the size of the network and also part of its overall performance. Several methods have appeared in the literature, which can be grouped as random, unsupervised and supervised selection. The former method was first introduced by Broomhead and Lowe (1988), employing the subset of the input training data as the centres which are selected at random. This strategy could be executed quickly, however, it may require excessive centres that cause over-fitting. In another method, the centres are obtained from unsupervised learning via the clustering process such as the *k*-means algorithm, (see e.g. Ukyan & Gzelis, 1997; Lin et al., 2009), or by using the genetic algorithm and fuzzy logic (see e.g. Zhao & Huang, 2002; Wang et al., 2002). These methods are widely used and researched, however they only considered the input training features without evaluating the output error of the network.

A more systematic approach is by employing supervised selection, also known as the forward selection. The most popular method was introduced by Chen et al. (1991) who utilised the orthogonal least square (OLS) algorithm, which was explained in Chapter 4. Another strategy of forward selection was presented by Orr (1993), which uses the information of the hidden neuron output from a previous

iteration. It uses the subset forward selection by implementing ridge regression analysis.

In this chapter, we will introduce an approach in the selection of basis centres in which the idea is partially adopted from the forward selection method by Orr (1993) in conjunction with the generalised least squares (GLS) theory, which affords an advantage in dealing with noisy data or when the variances of the observation are unequal. A special case of GLS called weighted least squares (WLS) will be implemented here. Furthermore, the advantage of the Gram matrix, *P* (Demuth et al., 2009) as appeared in equation (4.21) will be utilised here during the learning process. An improved method to train the network output weights will then be described. Next, appropriate ways to estimate several parameters in RBFNN including the least squares weighting factor *H*, the regularisation parameter λ, and the spread parameter σ, will also be presented. Finally, the implementation steps for the proposed algorithms will be summarised.

5.2. A forward selection method for centres determination

5.2.1. Using regularised least squares to form a cost function

The regularised least squares method is basically a form of multi-objective least squares that incorporate the weighting factors for each objective function. Its general formulation (Mead & Renaut, 2009), which responds to the input matrix A, can take the form as follows:

$$\hat{z} = J(z) = \min_{z} \left\{ \|Az - b\|^2 W_b + \|F(z - z_0)\|^2 W_z \right\}, \tag{5.1}$$

where the first objective term corresponds to the original cost function due to the residual error between the estimated output, Az, and the desired output, b, the second objective term is a regularisation solution that represents the approximated noise out of the system, z is a least squares solution, z_0 is a reference solution, and A and F are the inputs of each objective function. The weighting factor W_b is

related to the variance of the observation output, whereas the weighting factor W_z is correlated to the regularisation parameter. If considering the generalised *Tikhonov Regularisation* (Tikhonov, 1973), matrix W_z is generally replaced by λI_N, where λ is an unknown regularisation parameter and $F = I_N$ in which it is necessarily not of full rank (Mead & Renaut, 2009). When z_0 is assumed to be zero, the cost function J can be rewritten as:

$$J(z) = \min_z \left\{ \|Az - b\|^2 W_b + \lambda I_N \|z\|^2 \right\}. \tag{5.2}$$

5.2.2. Solution for the regularised and weighted least squares

In the ordinary least squares (OLS) estimation procedure, it is always assumed that all observations of input-output $\left\{ \mathrm{a}^{(i)}, \mathrm{b}^{(i)} \right\}_{i=1}^{n}$ are equally important in estimating the model parameters. However, this is not the case in many real problems in which it may be that some observation are known to be less reliable than others, or the converse. Thus, all observations have unequal variance where the form of the equality is known. A better estimate than OLS can be obtained using weighted least squares (WLS), also called generalised least squares (GLS). The idea is to assign to each observation a weight that reflects the uncertainty of the measurement.

The incorporation of the W_b matrix coefficient in equation (5.2) represents the weighted least squares in the form of a cost function. In the equation, if we set $W_b = 1$ and $\lambda = 0$, the solution is called the OLS; if we set only $\lambda = 0$, the solution is called the WLS; and if we set only $W_b = 1$, the solution is called the ridge regression (i.e. regularised LS). Ridge regression is a solution invented by Tikhonov (1973) to regularise mathematical problems from ill-posed conditions. Typically, there is not enough information available in the trained problems, hence necessary extra information through the regularisation method could be supplied. For the regularised and weighted least squares solution in equation (5.2), the estimated $\hat{z}$ can be obtained by taking the first derivative of J equal to zero (i.e. $\partial J(z)/\partial z = 0$), in which the derivation is shown in the following equations:

$$\begin{aligned} \partial J(z)/\partial z &= 2A^T W_b\left(Az-b\right)+2\lambda z = 0 \\ 2\left(A^T W_b A+\lambda I_N\right)z - 2A^T W_b b &= 0 \\ \therefore \hat{z}_{rwls} &= \left(A^T W_b A+\lambda I_N\right)^{-1} A^T W_b b, \end{aligned} \tag{5.3}$$

where W_b is a diagonal matrix of the weighting coefficient with $W_{ii} = w_i$, and it is symmetrical. In general, the weight coefficient w_i assigned to the i^{th} observation, will be a function of the variance of the observation, denoted σ^2 (Bates & Watts, 1988).

5.2.3. Formulation of the error function

Generally, to apply supervised learning, the analysis may be started with a simple linear model with a scalar output, given by:

$$f(\mathrm{x}) = \sum_{j=1}^{m} w_j p_j(\mathrm{x}), \tag{5.4}$$

where p is the linear combination of a function vector, and w is the coefficient of the linear combination in which we refer to p and w here as functions of the hidden units and weights, respectively, in the context of neural networks. If the p function (e.g. radial basis) can change during the learning process, then the model becomes nonlinear.

If the training set is $\left\{\mathrm{x}^{(i)}, y^{(i)}\right\}_{i=1}^{n}$ where $\mathrm{x}^{(i)} \in \Re^n$ is the input vector, by using the regularised and weighted least squares, the sum of squared errors ε, is given by:

$$\begin{aligned} \varepsilon &= \sum_{i=1}^{n} h_i\left(y^{(i)} - f(\mathrm{x}^{(i)})\right)^2 + \sum_{j=1}^{m} \lambda_j {w_j}^2 \\ &= \sum_{i=1}^{n} h_i\left(y^{(i)} - w_j p_j(\mathrm{x}^{(i)})\right)^2 + \sum_{j=1}^{m} \lambda_j {w_j}^2, \end{aligned} \tag{5.5}$$

where $\left\{\lambda_j\right\}_{i=1}^{m}$ is the weight penalty or regularisation parameter, and $h_i > 0$ defines the relative importance of observation $\left\{\mathrm{x}^{(i)}, y^{(i)}\right\}_{i=1}^{n}$ which is affected by its individual variance. In matrix form, equation (5.4) can be re-written as:

$$F = \left[p_1^T w \quad p_2^T w \quad \ldots \quad p_n^T w \right]^T = PW. \tag{5.6}$$

By using the pseudo-inverse method, we have the solution of the least squares approach as derived in equation (5.3), which is now given by $W = \left(P^T HP + \lambda I_m\right)^{-1} P^T H\, y$, hence equation (5.6) can be further represented as:

$$F = P\left(P^T HP + \Lambda\right)^{-1} P^T H\, y = PA^{-1} P^T H\, y, \tag{5.7}$$

where y is the matrix of the desired output, $A = P^T HP + \Lambda$ is the variance matrix, and Λ is a diagonal matrix of the regularisation parameters. The cost function in (5.5) can be re-written in the matrix form as:

$$\begin{aligned}
\varepsilon &= tr\left\{ (y - F)^T (y - F) + W^T \Lambda W \right\} \\
&= tr\left\{ \left(y - PA^{-1} P^T H\, y\right)^T \left(y - PA^{-1} P^T H\, y\right) + \left(A^{-1} P^T H\, y\right)^T \Lambda A^{-1} P^T H\, y \right\} \\
&= tr\left\{ \left[\left(I_n - PA^{-1} P^T H\right) y\right]^T \left[\left(I_n - PA^{-1} P^T H\right) y\right] + y^T HPA^{-1} \Lambda A^{-1} P^T H\, y \right\} \\
&= tr\left\{ y^T Q^T H\, Q\, y + L \right\} \\
&= tr\left\{ y^T H\, Q^2 y + L \right\} \text{ (because Q is symmetric, i.e. } Q = Q^T\text{)},
\end{aligned} \tag{5.8}$$

where $Q = I_n - PA^{-1} P^T H$ is the projection matrix, and tr is the trace function which computes the sum of the elements in the main diagonal. The matrix L can be derived further as:

$$\begin{aligned}
L &= y^T HPA^{-1} \left(A - P^T HP\right) A^{-1} P^T Hy \\
&= y^T H\left(PA^{-1} P^T H - PA^{-1} P^T HPA^{-1} P^T H\right) y \\
&= y^T H\left((I_n - Q) - (I_n - Q)^2\right) y \\
&= y^T H\left(Q - Q^2\right) y.
\end{aligned} \tag{5.9}$$

Substituting equation (5.9) into (5.8), the error function can be simplified as,

$$\begin{aligned}
\varepsilon &= tr\left\{ y^T H\left(Q^2 + Q - Q^2\right) y \right\} \\
&= tr\left\{ y^T HQy \right\}.
\end{aligned} \tag{5.10}$$

5.2.4. Forward selection with regularised and weighted least squares

Now, we will look into the derivation of a forward selection algorithm, which is incorporated by the regularised and weighted least squares that were discussed earlier, in the context of the radial basis function neural network (RBFNN). The implementation of only the weighted least squares theory in the selection of radial

basis function centres has been demonstrated in Wahid et al. (2011), and the results will not be presented here.

Recalling the RBFNN function from equations (2.10) and (2.11) in Chapter 2, the general form of RBFNN for l inputs and m outputs can be defined as:

$$\hat{y}_j = f(\mathrm{x}^{(\mathrm{i})}) = \sum_{k=1}^{N} w_{jk}\phi\left(\left\|\mathrm{x}^{(\mathrm{i})} - c_k\right\|_2\right), \qquad j = 1,2,...m, \tag{5.11}$$

where $\mathrm{x}^{(\mathrm{i})} \in \Re^n$ is the input vector, j is the output index, $\phi(\cdot)$ is a basis function, $\|.\|_2$ denotes the Euclidean norm, w_{jk} are weights in the output layer, N is the number of hidden neurons (and centres) in which generally $N << n$, and $c_k \in \Re^n$ are the RBF centres in the input vector space. In matrix notation, equation (5.11) can also be written as:

$$F = \Phi^T W, \tag{5.12}$$

where F is the matrix of the network output with $n \times m$ dimension, Φ is the matrix of hidden nodes with $N \times n$ dimension, $W = \left[w_{jk}\right]^T$ is a network weight matrix with $N \times m$ dimension, and n is the number of dataset samples.

By using equation (5.3) with the notations and dimensions used for the RBFNN problem in (5.12), the RBF network weight can be computed by the following equation:

$$\hat{W}_{rbfn} = \left(\Phi H \Phi^T + \lambda I_N\right)^{-1} \Phi H D, \tag{5.13}$$

where D is the matrix of the desired output with $n \times m$ dimension and H is a diagonal matrix of the least square weighting factors which is given as follows:

$$\mathrm{H} = \begin{bmatrix} h_{11} & 0 & \cdots & 0 \\ 0 & h_{22} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & h_{ii} \end{bmatrix}, \quad \text{with } 1 \le i \le n. \tag{5.14}$$

From (5.13), the network weight at the k-th iteration can be trained using the following equation:

$$\begin{aligned} W_k &= \left(\Phi_k H_k \Phi_k{}^T + \lambda I_k\right)^{-1} \Phi_k H_k D \\ &= A_k{}^{-1} \Phi_k H_k D \ , \end{aligned} \tag{5.15}$$

where A_k is the variance matrix, and the network weight at the $(k-1)$-th iteration can be formed as:

$$\begin{aligned} W_{k\text{-}1} &= \left(\Phi_{k\text{-}1} H_{k\text{-}1} \Phi_{k\text{-}1}{}^T + \lambda I_{k-1}\right)^{-1} \Phi_{k\text{-}1} H_{k\text{-}1} D \\ &= \left(A_{k\text{-}1}\right)^{-1} \Phi_{k\text{-}1} H_{k\text{-}1} D \ . \end{aligned} \tag{5.16}$$

If the network is grown by adding a new hidden neuron ϕ_k, the matrix of the hidden neuron thus acquires an extra row given by $\Phi_k = \left[\Phi_{k-1}, \phi_k{}^T\right]^T$. Hence the variance matrix $A_k = \Phi_k H_k \Phi_k{}^T + \lambda I_k$ in (5.15) can now be formed as:

$$\begin{aligned} A_k &= \begin{bmatrix} \Phi_{k-1} \\ \phi_k{}^T \end{bmatrix} \left[H_k\right] \left[\Phi_{k-1}{}^T \quad \phi_k\right] + \lambda I_k \\ &= \begin{bmatrix} \Phi_{k-1} H_k \Phi_{k-1}{}^T + \lambda I_{k-1} & \Phi_{k-1} H_k \phi_k \\ \phi_k{}^T H_k \Phi_{k-1}{}^T & \lambda + \phi_k{}^T H_k \phi_k \end{bmatrix} \\ &= \begin{bmatrix} A_{k-1} & \Phi_{k-1} H_k \phi_k \\ \phi_k{}^T H_k \Phi_{k-1}{}^T & \lambda + \phi_k{}^T H_k \phi_k \end{bmatrix}, \end{aligned} \tag{5.17}$$

and the inverse matrix of (5.17) is given by:

$$\begin{aligned} A_k{}^{-1} &= \frac{1}{\det\left(A_k\right)} \begin{bmatrix} \lambda + \phi_k{}^T H_k \phi_k & -\Phi_{k-1} H_k \phi_k \\ -\phi_k{}^T H_k \Phi_{k-1}{}^T & A_{k-1} \end{bmatrix} \\ &= \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}, \end{aligned} \tag{5.18}$$

Here, we are concerned with a direct relationship between A_k and A_{k-1} matrices, thus only the first matrix entry in (5.18) is taken into account in which $A_{11} = \left(A_{k-1} - \left(\lambda + \phi_k{}^T H_k \phi_k\right)^{-1} \Phi_{k-1} H_k \phi_k \phi_k{}^T H_k \Phi_{k-1}{}^T\right)^{-1}$. Thus, we now have:

$$A_{k(A_{11})} = A_{k-1} - \left(\lambda + \phi_k{}^T H_k \phi_k\right)^{-1} \Phi_{k-1} H_k \phi_k \phi_k{}^T H_k \Phi_{k-1}{}^T , \tag{5.19}$$

or it can be written as:

$$A_{k(A_{11})} = A_{k-1} - \Phi_{k-1} H_k \phi_k \left(\lambda + \phi_k{}^T H_k \phi_k\right)^{-1} \phi_k{}^T H_k \Phi_{k-1}{}^T . \tag{5.20}$$

By using the small rank adjustment (Horn & Johnson, 1985), if we have the following matrix equation:

$\mathrm{A}_1 = \mathrm{A}_0 - \mathrm{XRY}$, it will result in $\mathrm{A}_1^{-1} = \mathrm{A}_0^{-1} - \mathrm{A}_0^{-1}\mathrm{X}(\mathrm{YA}_0^{-1}\mathrm{X} + \mathrm{R}^{-1})^{-1}\mathrm{YA}_0^{-1}$.

Thus, from equation (5.20) we obtain:

$$A_k^{-1} = A_{k-1}^{-1} - A_{k-1}^{-1} q_k \left(\lambda + \phi_k^T H_k \phi_k - q_k^T A_{k-1}^{-1} q_k\right)^{-1} q_k^T A_{k-1}^{-1}, \qquad (5.21)$$

where $q_k = \Phi_{k-1} H_k \phi_k$, and $q_k^T = \phi_k^T H_k \Phi_{k-1}^T$.

Substituting (5.15) into (5.12), at the k-th iteration, the RBF network output over the training set is given by:

$$F_k = \Phi_k^T W_k = \Phi_k^T A_k^{-1} \Phi_k H_k D. \qquad (5.22)$$

Now, we can estimate the sum of squared error ε_k at the k-th iteration by utilising equation (5.10), as given by:

$$\begin{aligned} \varepsilon_k &= tr\left\{(D-F)^T (D-F)\right\} \\ &= tr\left\{ D^T H Q_k D \right\}, \end{aligned} \qquad (5.23)$$

where

$$Q_k = I_n - \Phi_k^T A_k^{-1} \Phi_k H_k. \qquad (5.24)$$

is a projection matrix, I_n is the identity matrix with the dimension of $n \times n$ and tr is the trace function which computes the sum of the elements in the main diagonal. Using equation (5.21), matrix Q_k can be re-written as follows:

$$Q_k = I_n - \Phi_k^T A_{k-1}^{-1} \Phi_k H_k - \frac{\Phi_k^T A_{k-1}^{-1} q_k q_k^T A_{k-1}^{-1} \Phi_k H_k}{\lambda + \phi_k^T H_k \phi_k - q_k^T A_{k-1}^{-1} q_k}. \qquad (5.25)$$

Substituting matrix Φ_k, Φ_k^T and $H_k = H_{k-1}$, where A_{11} is used for A_k^{-1}, yields:

$$Q_k = Q_{k-1} - \frac{Q_{k-1}\phi_k \phi_k^T Q_{k-1} H_k}{\lambda + \phi_k^T H_k Q_{k-1} H_k \phi_k}, \qquad (5.26)$$

where $Q_{k-1} = I_n - \Phi_{k-1}{}^T A_{k-1}{}^{-1} \Phi_{k-1} H_{k-1}$, in which the denominator part (i.e. $\lambda + \phi_k{}^T H_k Q_{k-1} H_k \phi_k$) always returns a scalar number. Thus, by implementing equation (5.23) the error can be calculated as follows:

$$\varepsilon_k = \varepsilon_{k-1} - \frac{tr\left\{D^T H_k Q_{k-1} \phi_k \phi_k{}^T Q_{k-1} H_k D\right\}}{\lambda + \phi_k{}^T H_k Q_{k-1} \phi_k}. \tag{5.27}$$

This also means that we can minimise the error by maximising M_k given by:

$$M_k = \frac{\left\| D^T H_k Q_{k-1} \phi_k \right\|^2}{\lambda + \phi_k{}^T H_k Q_{k-1} \phi_k}, \tag{5.28}$$

Accordingly, the selection of the network centre can be made by taking the vector number from a finite set (i.e. iterated evaluation of different vectors ϕ_k) of possible centres corresponding to the maximum value of M_k. However, this procedure may again cause the ill-computation which hinders the advantage of RBFNN training. To avoid the iteration process as well as the over-fitting problem, one solution is to choose a smaller number of centres than the dimension of the input space (Broomhead & Lowe, 1988). Thus, we suggest that the set of possible centres can be assessed by the Gram matrix *P*, as appeared in Demuth et al. (2009) in which it is a symmetrical and orthogonal matrix of all the possible radial basis outputs of given training data for the case of exact interpolation, as given in equation (4.21). Thus equation (5.28) can be rewritten as:

$$M_k = \frac{\left\| D^T H_k Q_{k-1} P \right\|^2}{\lambda K + sum(P^T H_k Q_{k-1} P)}, \tag{5.29}$$

where $sum(.)$ returns the sum of the values for each matrix column, and K is an array of all ones (i.e. 1) with $1 \times k$ dimension. In equation (5.31), the M_k value is dependent of the projection matrix Q_{k-1}, and the chosen regularisation parameter λ, whereas the *P* matrix is fixed for each loop of the additional hidden neuron. However, the multiplications with the orthogonal matrix *P* in the denominator will require a huge amount of memory for computation. Thus, equation (5.29) can be re-written as follows to give the same meaning, but faster calculation:

$$M_k = \frac{\left\| D^T H_k Q_{k-1} P \right\|^2}{\lambda K + sum(P^T) H_k Q_{k-1} P}. \tag{5.30}$$

5.3. Training the network output weights

Once a centre has been selected, the hidden layer output (i.e. the RBF output) for each neuron can now be computed, e.g. for Gaussian based RBF, $\phi_k = \exp(-\|x - c_k\|^2 / 2\sigma_k)$. Basically, the output weights at the k-th hidden neuron can be trained directly using equation (5.15). It should be remembered that we have introduced the least square weighting factors H_k, which is included in the pseudo inverse of matrix A_k in (5.15), hence the network output needs to be re-scaled back to certain gains. To do this systematically and to follow a standard configuration of the neural networks layout, a generalised method (Haykin, 1994) is implemented here.

Basically, generalisation is a simplified version of the regularisation strategy for RBFNN by giving the penalty values via the bias weights. The framework of the generalisation network for a regularised radial-basis function neural network has been discussed in Chapter 2 as shown in Fig. 2.9. Therein, one of the linear weights in the output layer of the network is set equal to a bias and the associated node ϕ_0, is treated as a constant equal to +1. The bias is added to compensate for the difference between the average value over the data set of basis functions and the corresponding values of the targets. By applying this bias, the network weights formula in (5.15) becomes:

$$\begin{aligned} W_k' &= \left(\Phi_k' H_k \Phi_k'^T + \lambda I_k'\right)^{-1} \Phi_k' H_k D \\ &= \{W_{net}, W_{bias}\}, \end{aligned} \tag{5.31}$$

where

$$\Phi_k' = \left[\Phi_k^T, \Psi\right]^T \tag{5.32}$$

in which Ψ is an array of values (i.e. 1) with $n \times 1$ dimension, and I_k is an identity matrix with $(N+1)\times(N+1)$ dimension, which produces a series of network weights that correspond to the hidden neurons and bias weights that correspond to the

independent neuron (i.e. a bias node). The network output in (5.11) is now mathematically represented as:

$$\hat{y}_j = f(\mathrm{x}^{(\mathrm{i})}) = \sum_{k=1}^{N} w_{jk}\phi_k\left(\mathrm{x}, c_k, \sigma_k, \lambda_k, H_k\right) + w_{j0}\phi_0, \quad \text{with } j = 1,2,...m\,. \quad (5.33)$$

5.4. Selection of H, λ and σ parameters

5.4.1. Estimation of the H parameter

In general, the least squares weighting coefficient h_i assigned to the i-th observation, will be a function of the variance of this observation, denoted σ^2. If information on the noise structure of the measurements d in equation (5.23) is available, a straightforward weighting scheme is to take the inverse of C_d, the error covariance for d, but other schemes such as iterative method could also be used. Hence, for a coloured noise, the i-th diagonal component weight factor of d is defined as:

$$h_i = C_d^{-1} = \left(\sigma_{di}^2\right)^{-1}. \quad (5.34)$$

In this work, for simplicity, we only consider the covariance of the white noise in which it has a common variance in the components of d, thus the least squares weight matrix is defined as:

$$H = \left(\sigma_d^2\right)^{-1} I_n\,. \quad (5.35)$$

5.4.2. Optimal selection of λ using cross-validation

In the Gaussian based radial basis function, the spread σ_k of the k-th basis function is typically identical for all input dimensions. The performance of the regularised radial basis function neural network is strongly dependent on the suitable choice of the spread parameter σ_k, and the proper selection of the regularisation parameter λ, for a given spread parameter.

Several approaches have been presented in the literature to compute the optimal value of the regularisation parameter λ^*, including generalised cross-validation, GCV (Golub et al., 1979), leave-one-out cross-validation, LOOCV (Golub et al., 1996; Shahsavand, 2009), unbiased predictive risk estimator, UPRE (Vogel, 2002), and χ^2 Newton based algorithm (Mead & Renaut, 2009). Here, a modified generalised cross-validation to incorporate the least squares weighting factors will be used.

The equation in (5.22) can be rewritten as:

$$F_k = S_{\lambda,H} D\,, \tag{5.36}$$

where $S_{\lambda,H} = \Phi_k{}^T A_k{}^{-1} \Phi_k H_k$. The GCV formula in the context of the RBFNN problem in this work, which is incorporated by the LS weighting factors, is given as:

$$GCV(\lambda, h) = \frac{\frac{1}{n}\sum_{i=1}^{n} h_{ii}\left(d_i - f_{\lambda,h}(x_i)\right)^2}{\left(1 - \frac{tr(S_{\lambda,h})}{n}\right)^2}, \tag{5.37}$$

where h_{ii} is the least squares weighting factor at the i-th observation. Equation (5.37) can be represented in matrix form as:

$$\begin{aligned} GCV(\lambda, H) &= \frac{\frac{1}{n}(D - F_k)^T H (D - F_k)}{\left(1 - \frac{tr(\Phi_k{}^T A_k{}^{-1} \Phi_k H_k)}{n}\right)^2} \\ &= \frac{\frac{1}{n}(D^T Q_k{}^T H Q_k D)}{\left(1 - \frac{tr(I_n - Q_k)}{n}\right)^2} \end{aligned} \tag{5.38}$$

where the matrix Q_k is similar to that defined in equation (5.24). With a few steps of derivation as follows, this equation can finally be simplified as in equation (5.39):

$$GCV(\lambda,H)=\frac{\frac{1}{n}(D^T{Q_k}^T HQ_kD)}{1-\frac{2tr(I_n-Q_k)}{n}+\left(\frac{tr(I_n-Q_k)}{n}\right)^2}$$

$$=\frac{\frac{1}{n}(D^T{Q_k}^T HQ_kD)}{1-\frac{2}{n}\left(n-tr(Q_k)\right)+\frac{1}{n^2}\left(n-tr(Q_k)\right)^2}$$

$$=\frac{\frac{1}{n}(D^T{Q_k}^T HQ_kD)}{1-2+\frac{2}{n}tr(Q_k)+\frac{1}{n^2}\left(n^2-2n\,tr(Q_k)+\left(tr(Q_k)\right)^2\right)}$$

$$GCV(\lambda,H)=\frac{nD^T{Q_k}^T HQ_kD}{\left(tr(Q_k)\right)^2}. \tag{5.39}$$

By using a set of possible λ values, i.e. $\lambda=\{\lambda_{\min}\ldots\lambda_{\max}\}\geq 0$, the optimal λ^* is the minimum point of $GCV(\lambda,H)$.

The process can be executed at each *k*-th step, however, the computation will becomes very tedious if higher decimal values of λ are considered. Thus, the Gram matrix P is used to replace the matrix Φ_k in equation (5.24) in which the average λ is approximated from the exact interpolation case.

5.4.3. Estimation of the spread parameter, σ

The spread parameter σ, in the RBFNN is often set manually by trial and error. Thus, it may be interesting to tune for some optimal value of σ. It can either be chosen to be the same for all nodes (i.e. isotropic spread) or it can be different from each other. It has been suggested that in the determination of σ, for example, the common width can be a set of some multiple of average distance between the basis centres (Orr, 1996; Howlett & Jain, 2001). The variance may also be adjusted optimally using iteration methods (Ghosh et al., 1992; Wahid et al., 2010b), but it often involves an expensive computation. Moreover, few problems like multiple minima, minimum local point and no convergence point may arise, that may lead to wrong decisions on the selected σ values. For many situations, the first technique

(i.e. common variance) leads to little loss in the quality of the final solution as compared to the optimal solution, and if chosen properly it may provide better performance.

In this work, we prefer to use the first technique, i.e. a common (or an isotropic) spread parameter for the network. It is suggested that the range of the possible spread parameter lies between the negative and the positive values of the standard deviation of its average distance between the basis centres, given by:

$$\left\{\sigma : \bar{d} \pm \sigma_{dist}\right\} > 0 . \tag{5.40}$$

where $\bar{d}$ is the mean of the distance between input points and σ_{dist} is the one-standard deviation from the mean value. If the input vector $\mathrm{x}^{(i)} \in \Re^{l \times n}$, where l is the input number and n is the number of dataset sample, the distance function between each data point can take form as follows:

$$d(\mathrm{x}, \mathrm{x}^T) = dist(\mathrm{x}^T, \mathrm{x}) . \tag{5.41}$$

where dist is an Euclidean distance function (available in *Matlab*) which produces an orthogonal matrix of distance values at upper and lower triangular, and diagonal values of zero. Next, the average and the standard deviation values of (5.41) are computed, and the spread parameter range can be obtained from (5.40). Now a trial and error may be used, however, we have narrow down the complexity of the problem from infinity values to a possible range of σ values. Note that the selected value of σ must not only goodly fit the trial data, but the constructed model from using this value must also capable to avoid the over-fitting when using the validation or testing data, especially when a higher accuracy (i.e. lower error goal value) is set. Thus, this selection may reach an optimal point if a validation test can be made available in the learning loop via the cross-validation strategy.

5.5. Implementation of the proposed algorithm

The overall proposed network scheme is depicted in Fig. 5.1, wherein the network centre, c_k at the k-th loop is a function of M_k. The implementation of the suggested algorithms are summarised in following steps:

For the first node (k=1)

1. Set $\lambda_0 = 0$, Q_{k-1} as the identity matrix with $n \times n$ dimension, and find the suitable value of spread parameter σ., using equations (5.40) and (5.41).
2. Compute the H matrix using equation (5.35).
3. Compute M_k from equation (5.30), and find the maximum value and its vector number.
4. Select the centre from the training dataset $\mathrm{x}^{(i)} \in \Re^n$ for the chosen vector number in step (3).
5. Compute the radial basis output ϕ_k, using the selected centre.
6. Set the possible values of λ, and minimise the GCV formula in (5.39) to find the optimal value of the generalisation parameter λ^*.
7. Calculate the new value of Q_k using equation (5.24), which will be used in the next iteration.
8. Determine the network output weight W_k, using regularised least squares incorporated by the generalisation network (using '\' operation for faster linear inversion process in Matlab software).
9. Compute the mean square error (*MSE*) and terminate the process if the MSE is less than the prescribed goal.

For the following nodes (k>1)

1. Set the matrix Q_{k-1} as Q_k value which was computed in the previous node.
2. Repeat steps (2) - (9) when k = 1, excluding the step in number (6).

The Matlab codes implementation is listed in the Appendix C-1.

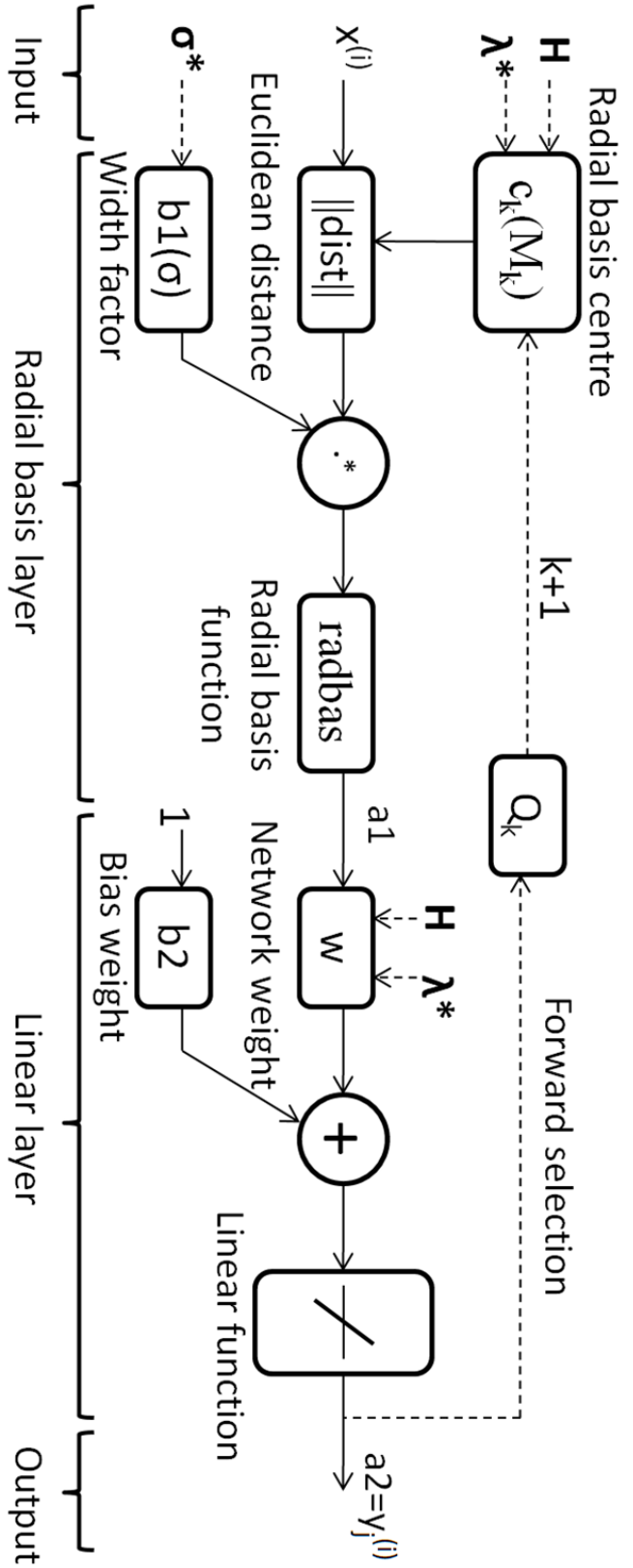


Fig. 5.1 Generalised radial basis function network scheme with regularised forward selection and weighted least square (GRFSWLS).

5.6. Chapter conclusion

A new method for training the radial basis function neural network (RBFNN) training has been successfully developed, which features the selected RBFNN centres using the regularised forward selection with weighted least squares, a generalised and regularised method for training the network weights, a reliable approximation method to define the least squares weighting factor, and an optimal method to tune the regularisation parameter using a generalised cross-validation approach. The effectiveness of the proposed method will be evaluated by application in air pollutant estimation, which will be covered in Chapter 8. Accordingly, the performance of several criteria including the performance indexes, the number of hidden neurons and the simulation time will be compared with other available training methods for RBFNN, such as the orthogonal least squares (OLS) and the forward selection (FS).

Chapter 6
COLLECTION METHODS FOR SYDNEY BASIN AIR QUALITY DATA

6.1. Introduction

To further assess the effectiveness of the proposed methodologies described in the previous chapters, they will be applied in several practical problems mainly related to air pollutant estimations. The methods are expected to provide alternative approaches to the current practices used by the environment regulatory agencies, and furthermore the incorporation of the proposed techniques with the technique typically used may offer improved outcomes.

In this work, numerous data sets will be used from three different sources: chronological data sets which include the air pollutant concentrations and meteorological data, which are collected at the monitoring stations at various sites scattered in the applied domain; the pollutant emissions inventory data which is typically managed by the local regulatory authorities (e.g. Department of Environment in New South Wales); and input-output data that could be extracted from the simulations of deterministic air quality model(s).

Basically, these three different stage approaches are the main keys used by the authorities in conducting air quality assessment. Each has its own usefulness to the policy maker in understanding the nature of air pollution attributable to various sources in the urban setting, in terms of both temporal and spatial aspects. The brief description of each type of data source will be discussed in this chapter to give a better understanding of how the data is collected and used in the several applications, which will be covered in the next following chapters.

6.2. The application domain: Sydney basin

6.2.1. New South Wales Greater Metropolitan Region

The Office of Environment and Heritage (OEH), New South Wales operates a comprehensive air quality monitoring network throughout the state, focused on the three main population centres: greater Sydney, the Lower Hunter (north of Sydney) and the Illawarra (south of Sydney), known collectively as the Greater Metropolitan Region (GMR), as depicted in Fig. 6.1. As a result of the Metropolitan Air Quality Study (MAQS), which was initiated in 1992, the number of monitoring stations was significantly increased as well as the number of air pollutants and meteorological parameters to be measured (DECC, 2007a).

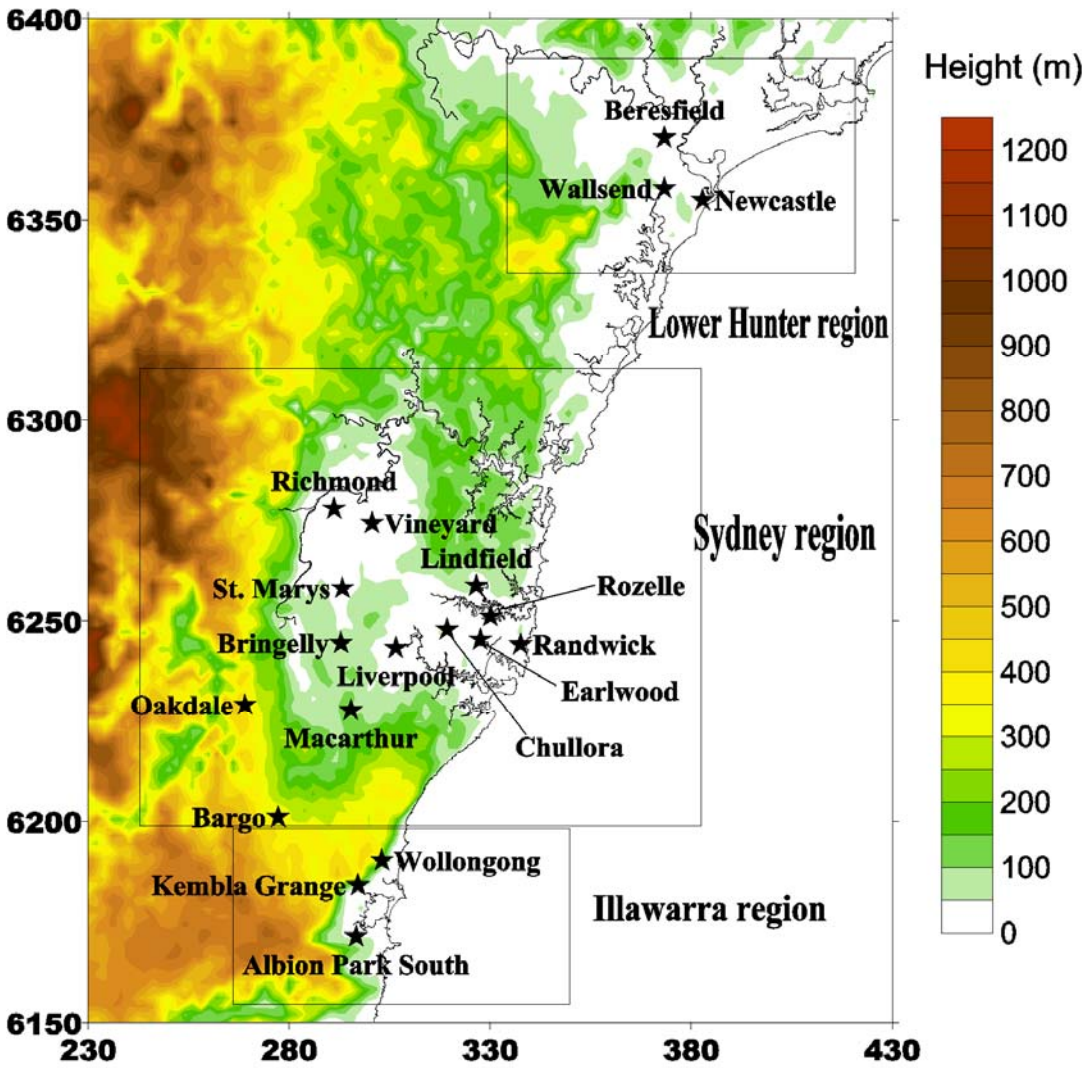


Fig. 6.1 Greater Metropolitan Region in New South Wales (NSW), Australia and its monitoring sites. (Map source: DECC, 2007a)

Sydney basin

This work will focus on the Sydney basin as the applied domain. The Sydney basin is the largest population centre in New South Wales (NSW), and its basin is bounded by elevated terrain to the north, west and south. Generally, the Sydney basin area can be divided into four main regions; East, North West, West and South West based on the geographical population settlement pattern. The basin currently has 14 monitoring stations scattered throughout the region, as depicted in Fig. 6.1.

The meteorology in the basin follows a general pattern. In the morning after sunrise, an onshore sea breeze flows from the east and north-east across Sydney toward the south-west causing an elevated level of ozone in the south-west and west of Sydney in the afternoon (Hart et al. 2006). In the evening and night-time, a drainage flow of cold air from the mountains in the west is directed to the coastal east and from the south-west to the north. It has been known that the chemical transport of nitrogen oxides (NO_x) and ozone also occur between the three regions, from the lower Hunter to the Sydney basin and from the Sydney basin to the Illawarra.

Lower Hunter and Illawarra basin

The lower Hunter basin has been determined as the second most populated region in NSW. The lower Hunter region is defined as that part of the Hunter River valley where it opens out to the coastal plain. It is bounded to the east by the coast and inland by higher terrain. It is separated from the upper Hunter River valley by a rise in the valley floor north-west of Maitland.

The Illawarra is the next largest population centre in NSW. The region is located on a thin coastal strip with a steep escarpment to the west. The width of the coastal strip increases from north to south until the strip terminates in a ridge of hills running from the escarpment to the sea. As the significant topographic feature, the escarpment has a major influence on meteorology and hence on air quality in the region. The Illawarra region lies only 80 km to the south of the Sydney region, and pollutants can be transported between the two, particularly from Sydney to the Illawarra. Most ozone events in the Illawarra occur as a result of local emissions combined with pollution transported from other regions.

The measurement stations in the Lower Hunter and the Illawarra regions are shown in Fig. 6.1. The details about the exact location of each monitoring station in New South Wales for the three main regions, are listed in Appendix D-1.

6.3. Data sets from the measurements

6.3.1. Air pollutants measurement

All the air pollutants as well as the meteorological data measurements are effected at the monitoring stations scattered in the region (see an example of a station in Fig. 2.1). Each monitoring station is designed according to a standard set-up similar to that shown in Fig. 6.2. Air quality monitoring devices are housed in or on a portable shed generally of dimensions 4.8m × 2.4m. All of the monitoring stations are made of aluminium cladding over a steel frame, and have one or two doors, an air conditioner and no windows. They are insulated throughout with 50mm rigid foam.

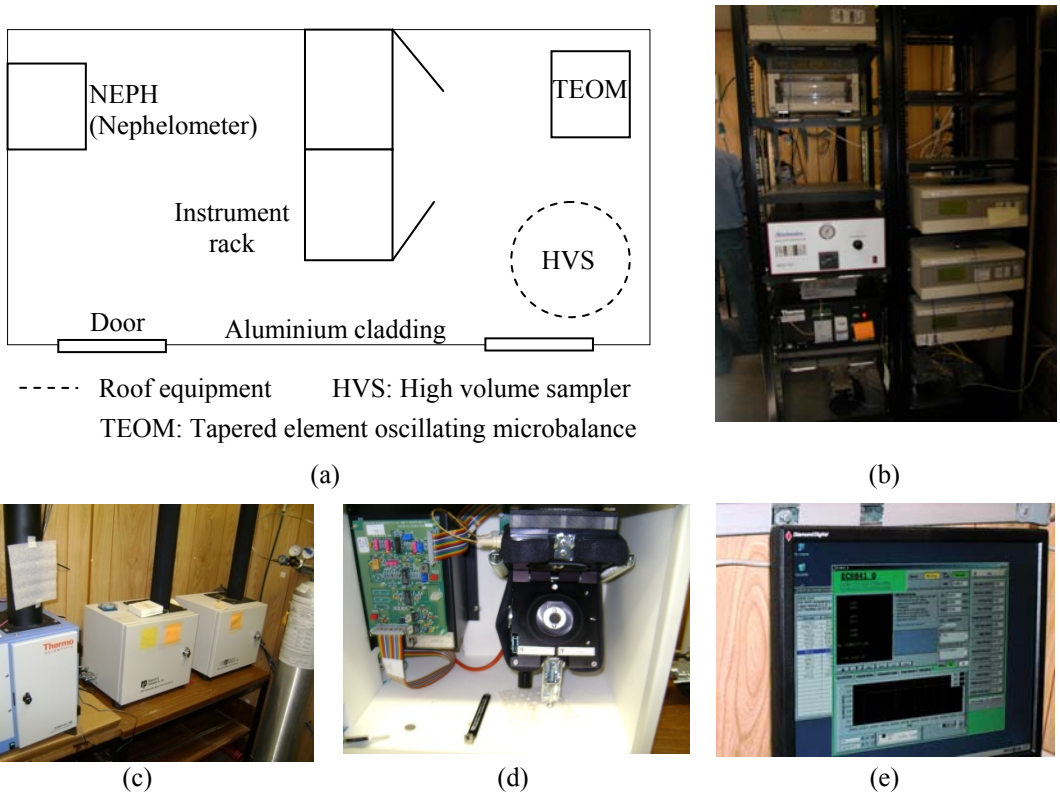


Fig. 6.2 (a) Typical monitoring station layout; (b) some air pollutants analysers in the instrument rack; (c) TEOM, devices for measuring fine particles; (d) internal view of TEOM device; and (e) data acquisition software for collecting the measurements data. (Photos courtesy of OEH, NSW)

The five pollutants to be measured under the 1998 national standards in Australia (i.e. National Environment Protection Measure – NEPM) are ozone, carbon monoxide, sulphur dioxide, nitrogen dioxide and air particles. These pollutants are generally measured in part per billion (ppb) units on an hourly basis. The following section briefly outlines the sampling methods used by the Environment Protection Agencies (EPA) and industry in their monitoring programs, being based on the Australian Standards for ambient air monitoring. However, for the purpose of this work, only two types of pollutant measurements will be described: ozone and oxides of nitrogen. The instrumentation used for other air pollutants can be found in Appendix D-2.

6.3.2. Instrumentation and its operation

Ozone (O_3) measurement

The measurement technique for ozone is based on the ultraviolet spectroscopy principle. In principle, ozone (O_3) molecules absorb UV light at a wavelength of 254 nm, and the degree to which the UV light is absorbed is directly related to the ozone concentration as described by the Beer-Lambert Law given as follows:

$$\frac{I}{I_o} = e^{-KLC}, \tag{6.1}$$

where:

K = molecular absorption coefficient (in cm^{-1});

L = length of cell (in cm);

C = ozone concentration (in parts per million, ppm);

I = UV light intensity of sample with ozone (sample gas);

I_o = UV light intensity of sample without ozone (reference gas).

For general operation, first, sample air is drawn into a cell where a beam of ultraviolet light is passed through it to an ultraviolet detector. Some of the light is absorbed by ozone in the sample, the amount being proportional to the number of molecules present. The decrease in intensity between the transmitted light and that of the source is used to determine the ozone concentration in the sample (AS 3580.6.1, 2011). There are several approved ozone analysers operating with the

same principle used by the Australian EPA, such as Monitor Labs Model 8810, Thermo Environmental Instruments Model TE49C, and Ecotech Ozone Monitor Model 9810 which were recently used at some sites, and shown in Fig. 6.3.

(a) (b) (c)

Fig. 6.3 Examples of ozone analyser models used by EPA: (a) ML8810, (b) TE49C, and (c) EC9810.

An example of the model flow schematic for TE49C is illustrated in Fig. 6.4 (TEC, 2003). As referred to in the diagram, the sample is split into two gas streams. One gas stream flows through an ozone scrubber to become the reference gas (I_o). The reference gas then flows to the reference solenoid valve. The sample gas (I) flows directly to the sample solenoid valve. The solenoid valves alternate the reference and sample gas streams between cells A and B every 10 seconds. When cell A contains reference gas, cell B contains sample gas and vice versa.

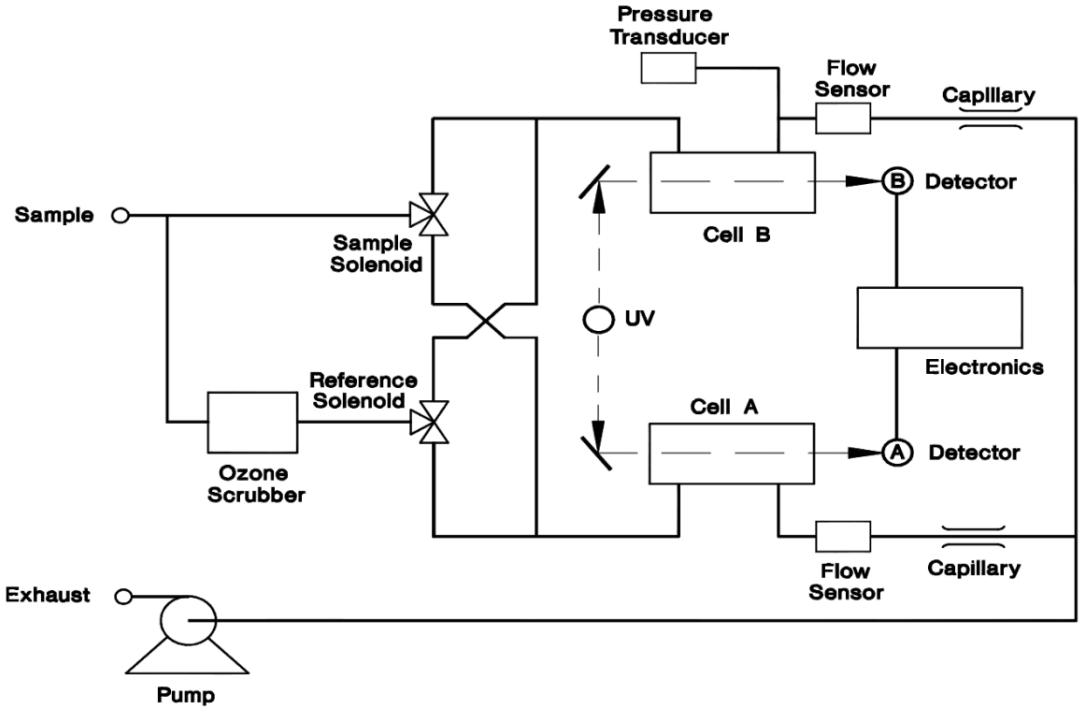


Fig. 6.4 Ozone analyser flow schematic for model TE49C.

The UV light intensities of each cell are measured by detectors A and B. When the solenoid valves switch the reference and sample gas streams to opposite cells, the light intensities are ignored for several seconds to allow the cells to be flushed. The device calculates the averaged ozone concentrations as measured by both detectors.

Oxides of nitrogen (NO, NO_2 and NO_x) measurement

The measurement technique of oxides of nitrogen is based on the chemiluminescence principle. According to this principle, the nitric oxide (NO) and ozone (O_3) react to produce a characteristic luminescence with the intensity linearly proportional to the NO concentration. Infrared light emission results when electronically excited nitrogen dioxide (NO_2) molecules decay to lower energy states, specifically for the following reaction:

$$NO + O_3 \rightarrow NO_2 + O_2 + h\upsilon . \quad (6.2)$$

Thermo Environmental Instruments Model TE42C and Ecotech Model 9841 are examples of approved oxides of nitrogen analysers used by the Australian EPA. For operation, sample air is drawn into a reaction chamber where NO in the sample reacts with a stream of O_3 produced by an ultraviolet lamp in dried air. The reaction produces light in the wavelength range 600 nm to 3000 nm. The light is detected by a photomultiplier tube (PMT), where the intensity is proportional to the concentration of NO. The concentration of total nitrogen oxides (NO_x) is measured in a separate sample stream. They are first reduced to NO oxide using a selective converter and its concentration determined as above. The concentration of nitrogen dioxide NO_2 reported is assumed to be the difference between NO_x and NO (AS 3580.5.1, 2011). A model flow schematic for TE42C is illustrated in Fig. 6.5 (TEC, 2004).

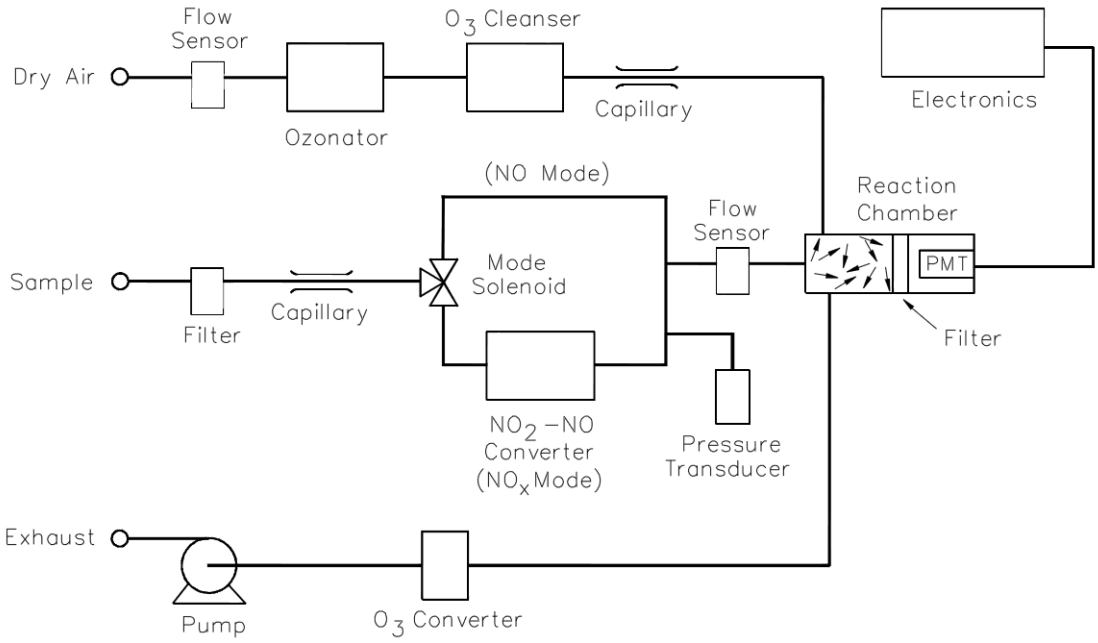


Fig. 6.5 Oxides of nitrogen analyser flow schematic for model TE42C.

6.3.3. Calibration of instrumentation

All measurement data is carefully recorded at the monitoring stations and post-processed to be trusted as reliable data sets. Hence, all instruments for capturing particulate air samples and gas analyser equipment are subject to frequent calibration and consistency checks. Generally, two types of calibration are carried out, as follows:

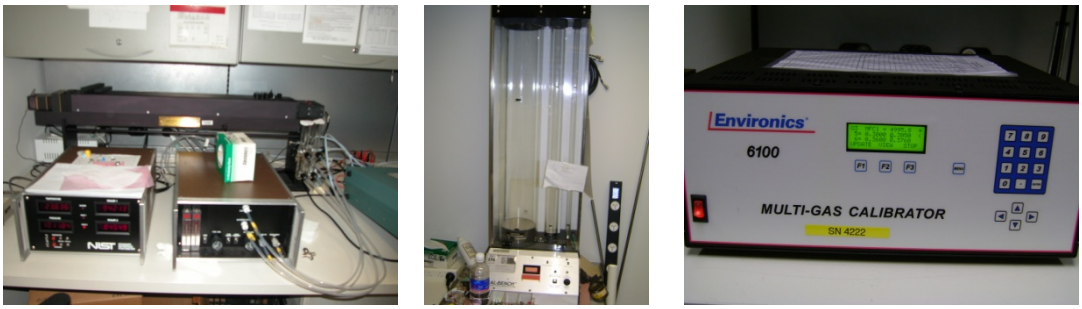
1. ***Online calibration:*** Recently, most of the gas analyser instruments were equipped with the capability of undertaking the auto-calibration process, mainly to correct the measured pollutant data, for example using 'background corrections'. Typically, zero calibration and full scale calibration are automatically performed every day in the early morning following midnight.

   The background correction is determined during zero calibration. The pollutant background is the amount of signal read by the analyser while sampling zero air. Before the analyser sets the pollutant reading to zero, it stores the value as the air pollutant background correction.

2. ***Offline calibration:*** This is a more accurate approach to calibration where the analyser itself is calibrated by comparing it with a standard, typically using a

special calibration instrument (see e.g. Fig. 6.6), and normally done offline in the designated calibration room.

Some gas analyser equipment can be used as a self-calibrator with few modifications from its standard operation. For example, the Model TE49C can be modified to operate as a calibration photometer for ozone by removing the ozone scrubber (as in Fig. 6.4) and plumbing zero air into the common port of the ozone-free solenoid valve (Model 49C Instruction Manual, 2004).



(a) (b) (c)

Fig. 6.6 (a) devices to produce pollutants gas standards; (b) a device to keep the ozone gas standard; (c) a multi-gas calibrator used in the instrument calibration. (Photos courtesy of OEH, NSW)

6.4. Data sets from the emissions inventory database

6.4.1. Air emissions inventory in New South Wales

The Office of Environment and Heritage (OEH) (formerly known as Department of Environment and Climate Change NSW – DECC) has conducted the air emissions inventory project, in which the study area covers 57,330 km^2 of the Greater Metropolitan Region (GMR) in New South Wales (NSW). It has been determined that approximately 76% of the NSW population resides in the GMR. This project commenced in 2004 and took nearly three years to complete.

The air emissions inventory includes emissions from biogenic (i.e. natural) and anthropogenic (i.e. human derived) sources, the details of which were described in Chapter 2 (section 2.2.1.2). During the inventory project, a number of surveys were conducted to obtain activity data from industry groups, government departments and other service providers. Air emissions have been estimated by combining *activity*

data with *emission factors* which are dependent on industrial and commercial sources (DECC, 2007d).

The emissions have been assigned to map coordinates for each 1 km by 1 km grid cell for biogenic, domestic-commercial, off-road mobile and on-road mobile area sources. Emissions are then calculated for months, weekdays/weekend days and hours using factors derived from the activity data. The base year of the inventory represents activities that took place in the 2003 calendar year and emission projection factors have been developed for every year from 2004 to 2031 using the methodologies published by US-EPA, which is given in the following equation (DECC, 2008):

$$E_{i,j,n} = E_{i,j,2003} \times PF_{j,n}\,, \tag{6.3}$$

where:

$E_{i,j,n}$ = Emission of substance i from source type j for year n (tonnes/year),

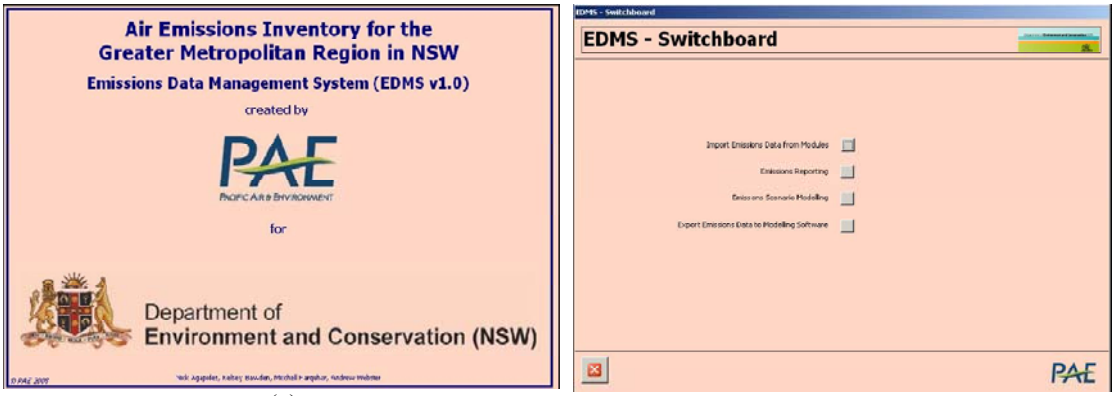
$E_{i,j,2003}$ = Emission of substance i from source type j for the base year 2003 (tonnes/year), and

$PF_{j,n}$ = Emission projection factor for source type j for year n.

6.4.2. Air emissions inventory database system

The Emissions Data Management System (EDMS v1.0) is the air emissions inventory database that links to individual source-specific databases comprising all the data necessary to service policy and technical related queries (DECC, 2008). The EDMS uses the Microsoft® SQL Server 2005™ relational database management system which is a comprehensive, integrated data management and analysis software package. The splash screen of the database system is shown in Fig. 6.7 (a).

Generally, the EDMS has the same function as other inventory systems in other countries providing a database for air quality models, to provide emissions modelling to test policy scenarios, and to chart and report emissions by air pollutant, by source or region. The main form of the EDMS, where users can choose which type of functions they would like to perform, is shown in Fig. 6.7 (b).



(a) (b)

Fig. 6.7 (a) EDMS splash screen; (b) the main form where a user can choose the function to be performed.

In this work, the EDMS is mainly used to export the emissions data to the air quality modelling software. It is quite tedious to directly use the emissions data from EDMS because this system will normally generate the separate data, based on the type of emission sources which are the point source, area source and motor vehicle source, rather than the total summation of the grid data. Therefore, a further pre-processing step is necessary to produce the hourly emissions data by grid coordinates, where this task could be performed by the air quality models. An example of the EDMS output for the area source (i.e. with the output filename as 'aems.in') is depicted in Fig. 6.8.

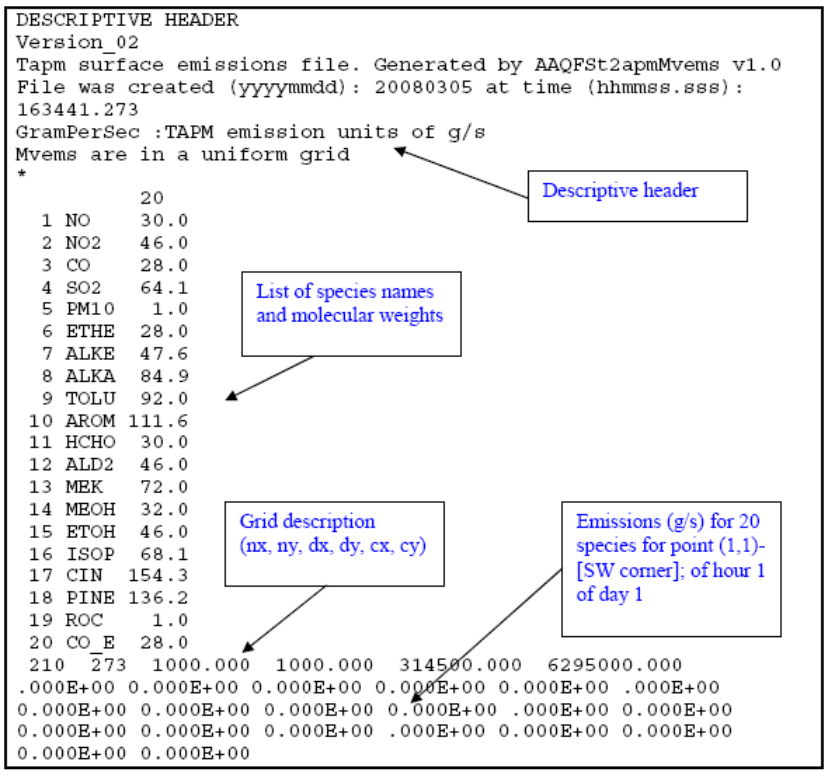


Fig. 6.8 EDMS output file structure for a TAPM-CTM area source emission file.

The EDMS is capable of producing a variety of emission input files for a selection of air quality models, including the California Puff Transport Model (CALPUFF), The Air Pollution Model (TAPM), CIT Airshed Photochemical Model, and Chemical Transport Model (CTM). For this work, the CTM feature is used to generate emissions input files for the TAPM-CTM model, a photochemical air quality model that is developed in Australia and used widely by the policy makers. A layout for this feature is illustrated in Fig. 6.9 where some information needs to be provided before generating the files such as grid dimensions that are to be modelled, a day to model, emission files that are to be generated, and the photochemical scheme to be used for the outputs. Two photochemical schemes can be chosen in the system: the Lurmann, Carter and Coyner (LCC) mechanism, and the Carbon Bond IV (CBIV) mechanism (Bawden et al., 2004).

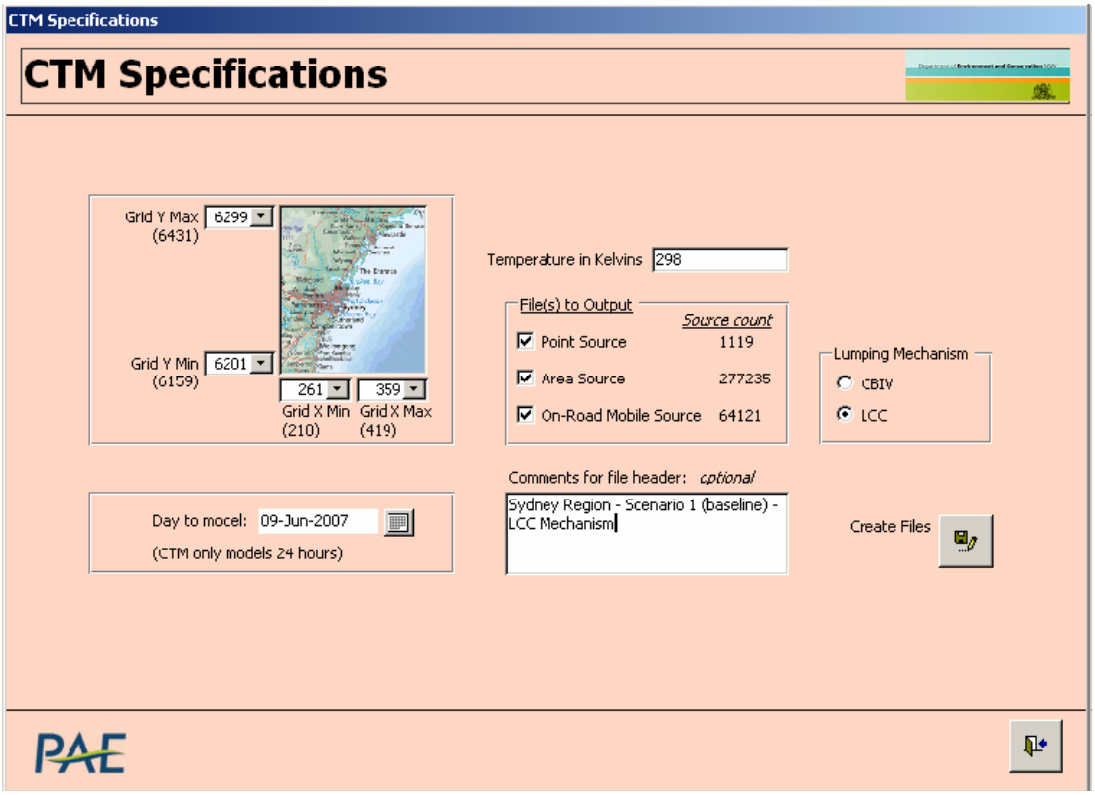


Fig. 6.9 Creating emissions files for TAPM-CTM model.

6.5. Data sets from the air quality model: TAPM–CTM

6.5.1. Overview of TAPM-CTM model

The Air Pollution Model with Chemical Transport Model (TAPM–CTM) is a three-dimensional prognostic meteorological and air pollution model, which has been developed since 1997 by the Commonwealth Scientific and Industrial Research Organization (CSIRO), in Australia, for use in air quality studies on a local, regional or inter-regional scale (Hurley, 2008). It was originally developed as TAPM, using a Generic Reaction Set (GRS) photochemical mechanism (Azzi et al., 1992). Recently, a modified version of TAPM called TAPM–CTM was developed to include the LCC and carbon bond IV photochemical mechanism as well as the GRS photochemical component, and was released in 2008. The Chemical Transport Model (CTM) features are mostly adopted from the CIT model (McRae et al., 1992).

Due to the limitation of the measurement data in which there are only certain points available in the domain, some input-output data for training the metamodel needs to be extracted from the TAPM-CTM simulation outputs. By doing this, a more generalised solution in the modelling process, using a metamodel approach, is expected to be achieved. More details on what parameters are necessary to be used from the TAPM–CTM will be discussed in Chapter 8 (metamodel application).

Three important processes are involved in executing the complete simulation using the TAPM–CTM model as shown in Fig. 6.10, and briefly described as follows:

1. **The preparation of emissions data using the EDMS system** (as discussed in the previous section).
2. **Air quality modelling using TAPM–CTM:** Three main input files are used, which are emission files from the EDMS, topographical information of the terrain heights, and synoptic data which is multi-layer meteorological data in grid form and typically consists of temperature, wind direction and wind speed data. In Australia, the topographical information can be obtained from the Australia National Mapping Agency (AUSLIG), while the synoptic data can be made available from the Bureau of Meteorology in NSW. Fig. 6.11 shows the main graphical user interface (GUI) for TAPM–CTM software.

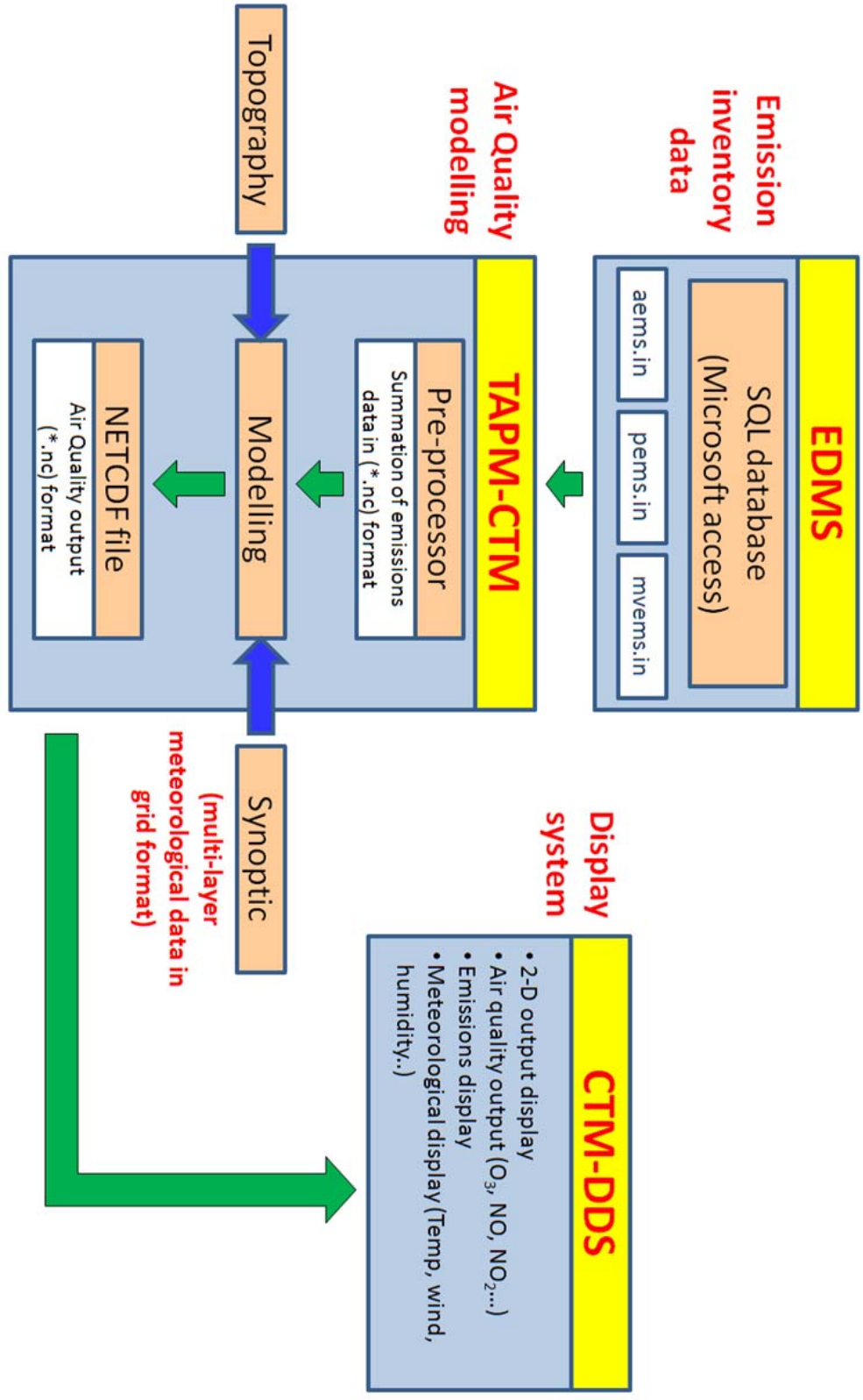


Fig. 6.10 The three processes involved in the simulation of TAPM-CTM model.

3. **Visualisation of the simulation output:** Air quality, emissions and meteorological data from TAPM–CTM can be displayed using a GUI-driven display system such as CTM Data Display System (CTM–DDS).

The TAPM–CTM also involves the pre-processing stage in which the three different emission sources are first merged to become one emission output in the form of grids and hourly basis data. This type of emission data could serve as one type of data set for this work.

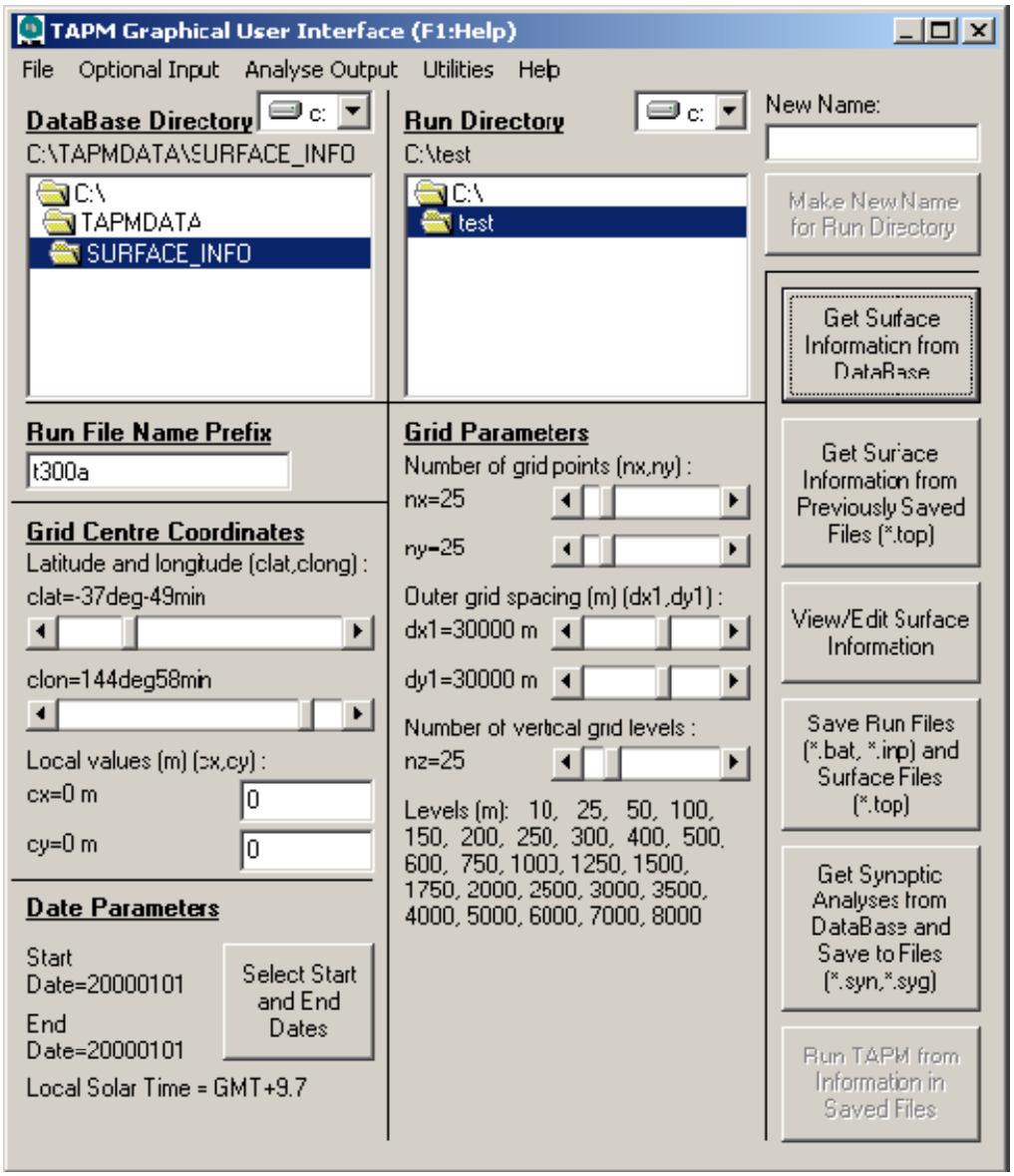


Fig. 6.11 Main GUI of the TAPM software.

6.5.2. Data display system for the air quality model

The CTM Data Display System (CTM–DDS) is a GUI-driven display system that is specially designed to view the TAPM–CTM output file (i.e. *NetCDF* data packet in *.nc format) as shown in Fig. 6.12, as well as to provide some other types of output. The display system may be accessed from the '*Analyse Output'* menu of the TAPM–CTM GUI, or alternatively by double-clicking on the relevant *NetCDF* file.

CTM–DDS generates two-dimensional animations of pollutant, emissions and meteorological fields. The CTM–DDS can also be used to generate time series plots of observed and modelled meteorological and air pollutant parameters, such data being used to verify the performance of TAPM–CTM. In addition, the system is also able to generate pollutants, emissions, terrain heights and meteorological outputs in (*.csv) format, thus it could be used as part of the input-output data sets for this work, with several refinements that will be discussed later in Chapter 8. A more detailed explanation on how to run the TAPM–CTM model and the CTM–DDS software is described in Cope and Lee (2009).

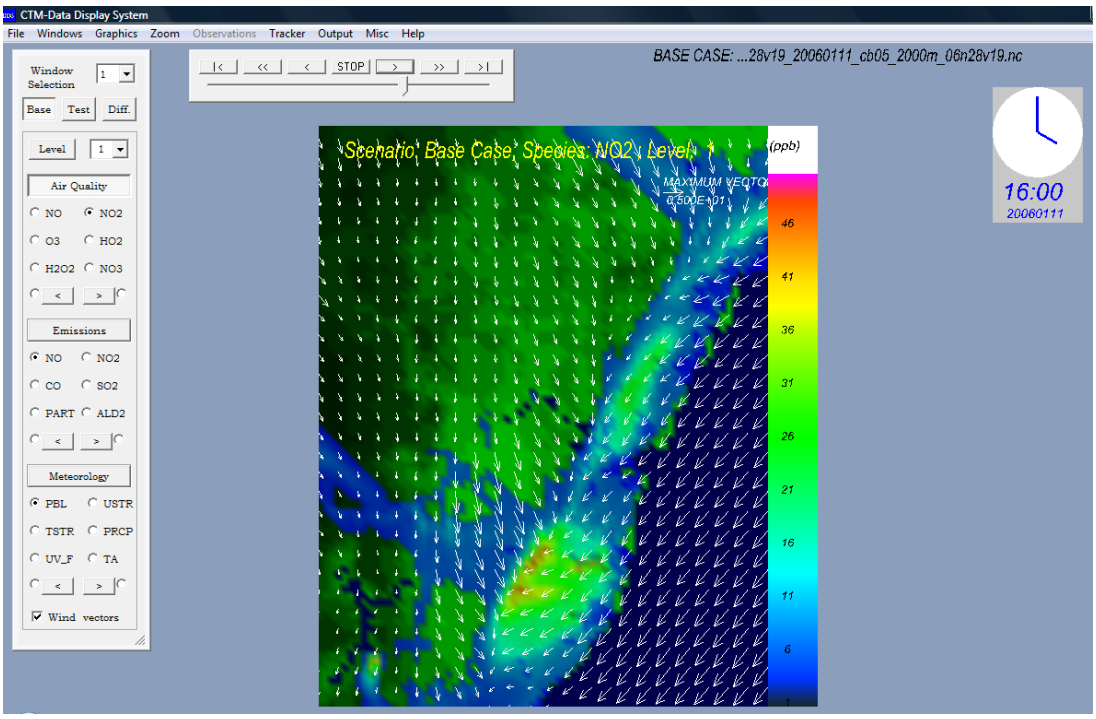


Fig. 6.12 Air pollutant level of NO_2 at 4:00pm as displayed by CTM–DDS system.

6.6. Chapter conclusion

This chapter has provided some explanations of the tools and methods for collecting the data sets for this work. An overview of the applied domain, the Sydney basin in New South Wales, Australia, was first described.

There are three different methods used for preparing the data sets: 1) by measuring the air pollutants at the monitoring sites located at various locations across the region, using some special instruments – this would be the most accurate and reliable method that could be used as the reference data set, although this data is only valid adjacent to the monitoring sites; 2) by using the emissions data for the sources of pollutants from the Emission Data Management System (EDMS) – this type of data is useful as inputs in the modelling process; and 3) by using special air quality modelling software (i.e. TAPM–CTM in this work) to generate some important information about the input-output from its simulation of outputs – these data sets have the capacity to model the spatial distribution problem and could generate some of the information that cannot be covered by the measurement method, although this type of data is less accurate and necessary refinements or calibrations of the data need to be considered.

From some of the explanations about the data collection methods in this chapter, it is expected that the following discussions in Chapters 7 and 8 may be more readily understood.

Chapter 7
BACKGROUND OZONE LEVEL DETERMINATION IN THE SYDNEY BASIN

7.1. Introduction

As discussed in the literature review in Chapter 2, background ozone level (BOL), in the context of the photochemical smog process, is defined as the ozone level that is formed by purely natural processes. The concept is easily understood but the problem remains how to determine the BOL and distinguish between natural and anthropogenic effects.

BOL is important as it sets the reference level against which anthropogenic impacts can be ascertained by the measured ozone level in a particular area. It provides the basis for human health risk assessment estimates and also determines whether policy expectations are realistic about the levels to which hourly average ozone concentrations can be lowered as a result of emission reduction requirements. Health risks are known to be associated with the ozone concentration in excess of the allowable background concentration. Furthermore, long-term exposure to background ozone levels can also affect plant growth (Diaz-de-Quijano et al. 2009).

Clean pristine sites in rural or bushland areas have been cited as suitable locations to measure the level of background ozone. Several works of BOL determination at remote sites have been reviewed, however, this approach is rather unlikely to be implemented at all locations especially in the metropolitan area, e.g. in the Sydney basin, Australia. On the other hand, the determination of BOL by an air quality model (AQ) could cover a wider region, but is difficult to prepare. The reliability of

its estimation is much dependent on the correctness of the biogenic emissions data as the inputs of the model.

In this work, we present several approaches for determination of the BOL, which are based on the ambient measurement data. The proposed methods are generic, which may be used to determine the background level at any part of the globe and in any season without relying on data obtained at remote sites. From the definition of BOL, this work will assess its concentration in the Sydney basin from ambient air quality data measured at several monitoring stations. As it has been recognised that the background ozone level in urban areas is changing over the years, our objective is to derive a temporal profile of BOL in the region.

7.2. The night-time BOL determination

7.2.1. Theoretical background

The US EPA has defined Policy-Relevant Background (PRB) ozone concentrations used for the purposes of informing decisions about National Ambient Air Quality Standards, as ozone concentrations that would occur in the absence of anthropogenic emissions, including contributions from natural sources everywhere in the world and from anthropogenic sources outside continental North America (US, Canada, and Mexico). According to US EPA (2006), 'contributions to PRB ozone include photochemical actions involving natural emissions of VOCs, NO_X, and CO as well as the long-range transport of ozone and its precursors from outside North America and the stratospheric-tropospheric exchange of ozone, whereby natural sources of ozone precursors are mainly biogenic emissions, wildfires and lightning.

Abiding strictly by the definition of PRB, it is difficult to determine the PRB ozone level by using measurements obtained at various background "pristine" sites, and only a chemical transport model would be suitable to estimate the range of PRB values. Hence, a measurement approach combined with analytical modelling methods may provide a more tractable way to determine the background ozone level (Gadner & Dorling, 2000; Vingarzan, 2004).

7.2.2. Methodology

Policy relevant background ozone (PRB), as defined by US-EPA, which excludes ozone formation contributed from outside the continent is rather difficult, or impossible, to measure and can only be determined by modelling. In this work, we are concerned with assessing BOL in a practical way considering the night-time non-photochemical condition. Here, ambient air quality measurements are used to estimate the background ozone concentration, taking into account local site conditions.

The background ozone level in a local (e.g. in the Sydney basin) or regional area is defined as the ozone level which would be measured if there were no ozone precursor anthropogenic source emissions within that area. This definition helps in our understanding of background ozone, its local, regional and global evaluation, and in the determination of its concentration and temporal profile, which can be useful to the policy maker. Notably, the proposed background ozone definition can also allow for BOL estimation by using a modelling method only or in combination with observations at remote clean sites. There are a number of estimated background ozone levels of interest, which can be calculated (one-hour daily maximum, eight-hour daily maximum and the daily mean) using the hourly ambient air quality data at the stations, summer being the period of most interest during the year.

Night-time non-photochemical background ozone is defined as the average of ambient measurements of hourly ozone values from night time to early morning (i.e. from 7.00 pm to 8.00 am the next morning), when there is no nitric oxide present for at least two hours consecutively (Duc et al., 2012). This prevents the reaction of ozone with nitric oxide (i.e. NO scavenging of ozone). This definition of night-time background ozone allows for excluding the photochemical process that would occur during daytime, in which both natural and anthropogenic sources are present. Thus, it includes the case of no ozone loss due to scavenging (NO=0) as if only a natural precursor but no anthropogenic sources were present in the local area. Here nitrogen oxide is assumed to be a surrogate for the presence of anthropogenic sources and ozone deposition loss is not accounted for. This night-time background ozone can generally include ozone formed from precursor emission from natural and

anthropogenic sources inside and outside the area. From our understanding of Sydney's meteorology, residual ozone formed during the day is mostly carried off-shore by westerly winds and drainage flow from the mountains during night-time to the early morning. Hence it is expected that the night-time background ozone does not contain much ozone formed previously in the area.

7.2.3. Case study: Analysis of night-time BOL in Sydney

7.2.3.1. Night-time BOL statistical results

From the proposed night-time background ozone definition and data collected at a number of different monitoring sites in the Sydney region, a statistical summary of the background ozone concentration is given in Table 7.1, where the 1st quartile, median (2nd quartile), 3rd quartile and statistical mean are shown for each station. These stations are mostly urban sites (excepting Vineyard and Richmond, both located in semi-rural or suburban areas), where ozone concentration data were being collected over the period of 1998 to 2005. It is noted that the statistical properties of BOL distributions were different between regions and within regions, and that BOL in the East and North West of Sydney in general was higher than in the West and South West of Sydney. A possible reason for this is that during night-time and early morning, North West Sydney is downwind from the South West due to the southerly flow and Eastern Sydney is downwind from the easterly drainage flow from the mountains in the west of the Sydney basin.

Table 7.1 Statistics of non-photochemical night-time BOL at monitoring sites in the Sydney basin.

Region	Site	Period	1st Qtr. (ppb)	Median (ppb)	3rd Qtr. (ppb)	Mean (ppb)
Sydney East (urban)	Woolooware	01/01/1998 to 30/08/2004	16	21	25	21
	Rozelle	01/07/1998 to 17/11/2005	13	18	22	18
	Randwick	01/01/1998 to 18/11/2005	17	21	26	21.4
	Earlwood	01/02/1998 to 17/11/2005	15	19	23	19.2
	Lindfield	01/01/1998 to 11/02/2005	12	17	22	17.2
North West (semi-rural/ suburban)	Vineyard	01/01/1998 to 18/11/2005	12	17	23	18
	Richmond	01/01/1998 to 17/11/2005	12	17	23	18
South West (suburban)	Bringelly	01/01/1998 to 18/11/2005	10	17	25	17.7
	Liverpool	01/01/1998 to 17/11/2005	12	19	25	18.8
Sydney West (urban)	Westmead	01/01/1998 to 06/08/2004	11	15	20	16
	St Marys	01/01/1998 to 18/11/2005	11	17	22	17.2
	Lidcombe	01/01/1998 to 01/05/2002	11	16	22	16.9
	Blacktown	01/07/1998 to 03/06/2004	12	16	21	17.2

7.2.3.2. Night-time BOL temporal trend

In theory, excluding all anthropogenic sources in the region and Australia wide, the average background ozone level in Sydney should be stable with respect to time. It is however possible that not only the emission of precursors outside the Sydney region, but also global emissions outside Australia could also influence the level of background ozone over time. It is therefore beneficial to study the BOL change in the Sydney region, as derived from ozone measurements at various monitoring sites, in the temporal domain.

The trend of the night-time background ozone can be found, using night-time hourly ozone monitoring data collected from the period 1998 to 2005 at various monitoring stations in the Sydney basin. To analyse all hourly ozone data, the linear regression method was used with ozone as an affine function of nitrogen oxide and the BOL is derived when the NO concentration is zero. Fig. 7.1 shows an example of the results for the St Marys site, a suburban site in the west of Sydney. It reveals an increasing trend with an increasing rate in the ozone concentration of 3.2±0.3 ppb (standard error is 0.3 ppb) over the 1998 to 2005 period (or 0.43±0.04 ppb per year) with a zero probability (p-value) obtained for the null hypothesis of the slope and an intercept of 15.6±0.2 ppb in the linear regression trend line, as shown in Fig. 7.1.

Considering the level of oxidant, which consists of ozone (O_3) and nitrogen dioxide (NO_2), a similar upward trend can be observed in Fig. 7.2 with an increase of 4.6±0.3 ppb over the same period and intercept of 20.1±0.2 ppb in the linear regression trend line. Again the p-value is zero for the slope, and the oxidant concentration exhibits an average increase of 0.66±0.04 ppb per year.

The same trend is found in our analysis for all other sites in the Sydney basin, except at Lidcombe where the monitoring site was discontinued in mid-2002 and was replaced by a nearby station, a few streets away in Chullora. With a possible extension of data points up to 2005 using data collected at Chullora, a similar temporal profile could be obtained at Lidcombe in comparison to other stations.

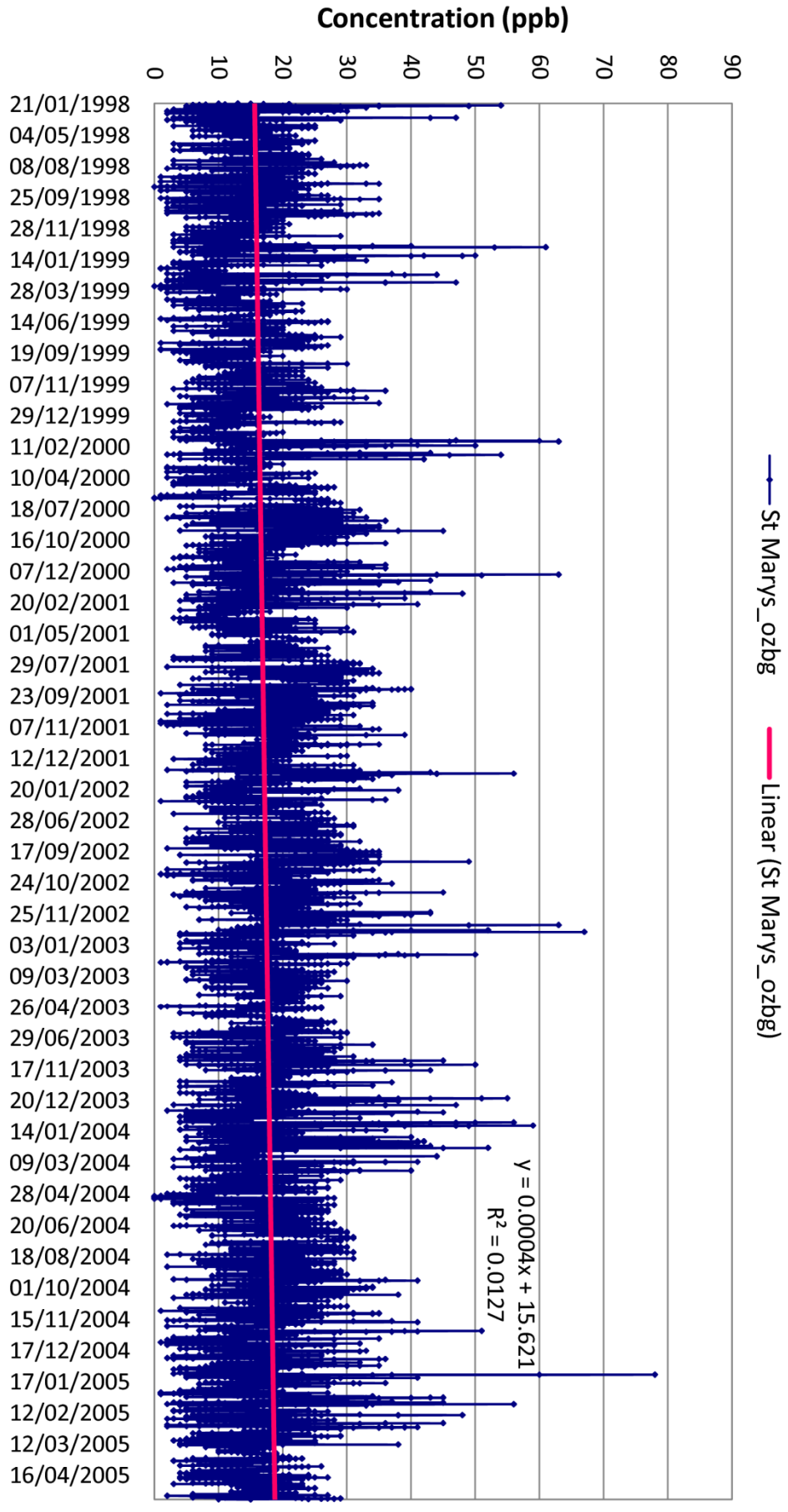


Fig. 7.1 Background trends from night-time to early morning at St Marys' station when nitrogen oxide (NO) is zero, based on ozone concentration

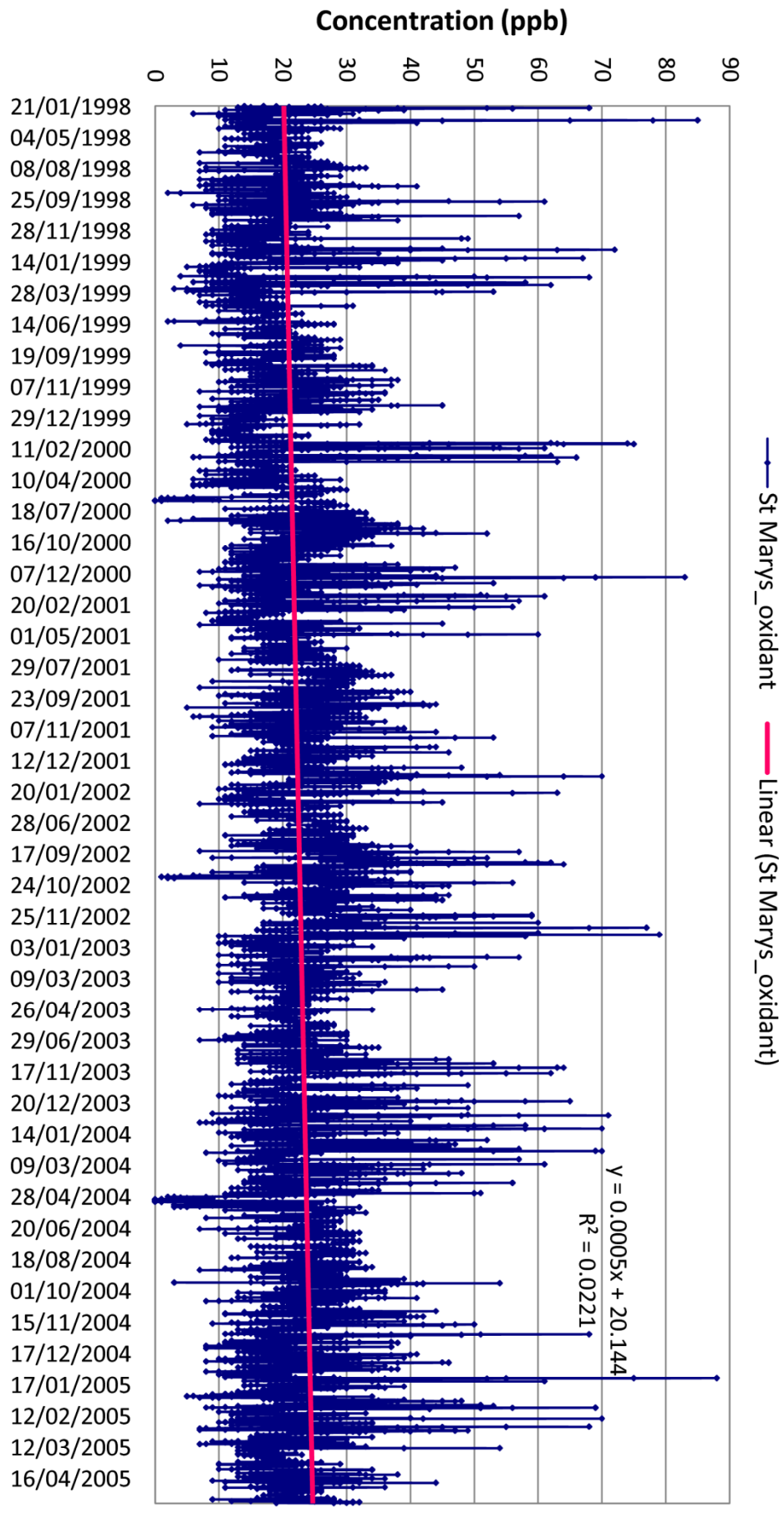


Fig. 7.2 Background trends from night-time to early morning at St Marys' station when nitrogen oxide (NO) is zero, based on oxidant ($Ozone+NO_2$) concentration.

7.2.3.3. Discussion

As summer is the time when photochemistry is most active, only ozone data during that time of the year, rather than for the whole year, are used to determine BOL (Fiore et al., 2002). Indeed, our results show that there is not much difference in the trend lines from using summer data only, and using all year data.

The Sydney background ozone trend is similar to the trend in US and Europe. Indeed, Jaffe and Ray (2007) have reported similarly of the ozone trend at 11 remote rural sites in north and western US (including Alaska), in which seven sites showed a statistically significant increase in ozone with an annual increase of 0.26 ppb on average. These sites are considered 'pristine' background ozone sites. Temperature changes can account for only part of the trend, and the authors explained these trends as possibly arising from increasing regional emissions, changes in the distribution of emissions, increasing biomass burning or increasing global background ozone, and especially due to rapid growth in emissions within North East Asia. In the same context, Simmonds et al. (2004) showed an increasing trend in background ozone observations at Mace Head on the west coast of Ireland from 1987 to 2003 with an average of 0.49±0.19 ppb per year. They concluded that there has been at least one major perturbation of the ozone trend during the 1998–1999 timeframe that was associated with global biomass burning coupled to an intense El Nino event in 1997.

7.3. The daytime BOL determination

7.3.1. Methodology

It is recognised that a better way for determination of background ozone level should include also the daytime photochemical process when only natural sources are present locally. For this, the regression and extrapolation method proposed in Clapp and Jenkin (2001), can be used to estimate the daytime background ozone.

The method involves the use of regression analysis to find the linear relationship between oxidant (including ozone and nitrogen dioxide, O_3+NO_2) and nitrogen

oxides (NO_x=NO+NO_2), using measurement data at various sites in the analysed domain. As suggested by Clapp and Jenkin (2001), this oxidant level at a given location is made up of NO_x-independent and NO_x-dependent contributions. The NO_x-independent contribution is the intercept, which equals the daytime background oxidant level, while the slope of the regression line represents the NO_x-dependent contribution or the level of primary pollution from the local sources.

This approach of using the regression line to find the BOL is also similar to that of Altshuller and Lefohn (1996) but instead of using the relationship between ozone and peroxyacetyl nitrate (PAN) as well as ozone versus the total reactive nitrogen species (NO_y) or ozone versus (NO_y -NO_x), we use the relationship between oxidant (ozone+nitrogen dioxide) versus NO_x as outlined by Clapp and Jenkin (2001). The reason is that most ambient monitoring stations do not measure PAN or NO_y and this situation is the case in the Sydney basin.

7.3.2. Case study: Analysis of daytime BOL in Sydney

7.3.2.1. Analysis of local background oxidant level

The analysis has covered the eight-year period from 1998 to 2005, collected in the Sydney region during the photochemistry-active summer in the southern hemisphere. The results obtained from the measurements, observed locally or regionally, can be extrapolated to find the derived background ozone at the local sites.

Using data for the summer 1998 period at monitoring sites in the Sydney basin, the plots of daylight average (O_3+NO_2) versus NO_x are shown in Fig. 7.3 for Blacktown, Bringelly, St Marys and Richmond, respectively. Most of these sites are considered urban except for Richmond, which is located in a semi-rural area. A linear regression line can be fitted to the corresponding data as shown but some variance can be expected. This is explained by a major variation in the regional contributions resulting from frequent elevated levels of ozone during the summer period. Two separate regression lines can be fitted to the ‘ozone non-episode’ and ‘ozone episode’ days (Clapp and Jenkin, 2001). Here, the episode day is defined as a day

when one or more stations in the Sydney basin have an ozone level greater than 80 parts per billion (ppb). The daytime background level for 'non-episode' and 'episode' days can be obtained from the intercepts of the regression lines.

As shown in Fig. 7.3, various slopes of the regression lines for the episode and non-episode daytime oxidant level appear at most of the sites. If the slopes are similar, e.g. at Bringelly site, it is suggested that the local contribution of nitrogen oxides to the oxidant level (NO_x-dependent) is the same at that site during episode and non-episode days. Thus, only the daytime background level (NO_x-independent) is different with a higher concentration value being obtained for episode days. The derived background oxidant levels at the four sites in the Sydney West, as presented in Fig. 7.3, range from 27 to 46 ppb for episode days and from 15 to 27 ppb for non-episode days. Notably, the results are coincident with those derived by Clapp and Jenkin (2001), using ambient air quality data measured at rural and urban sites in the UK. Their background values for oxidant level for non-episode days are about 35 ppb and about 55 ppb for episode days (Clapp and Jenkin, 2001). These values can be comparable with BOL of about 35 ppb of air entering the west coast of the US as reported by Oltmans et al. (2008).

Contrary to the findings using ambient data measured at Sydney West sites, the overall analysis for East Sydney sites, such as Randwick and Rozelle, does not show a distinct relation between oxidant (O_3 and NO_2) and nitrogen oxides (NO_X) and the scatter around the regression lines appears to be relatively large. Under the photo-stationary assumption for photochemical reactions, a linear relationship between oxidant and nitrogen oxide levels is generally expected. This means the ideal photo-stationary state of smog reaction rarely occurs in the East Sydney area.

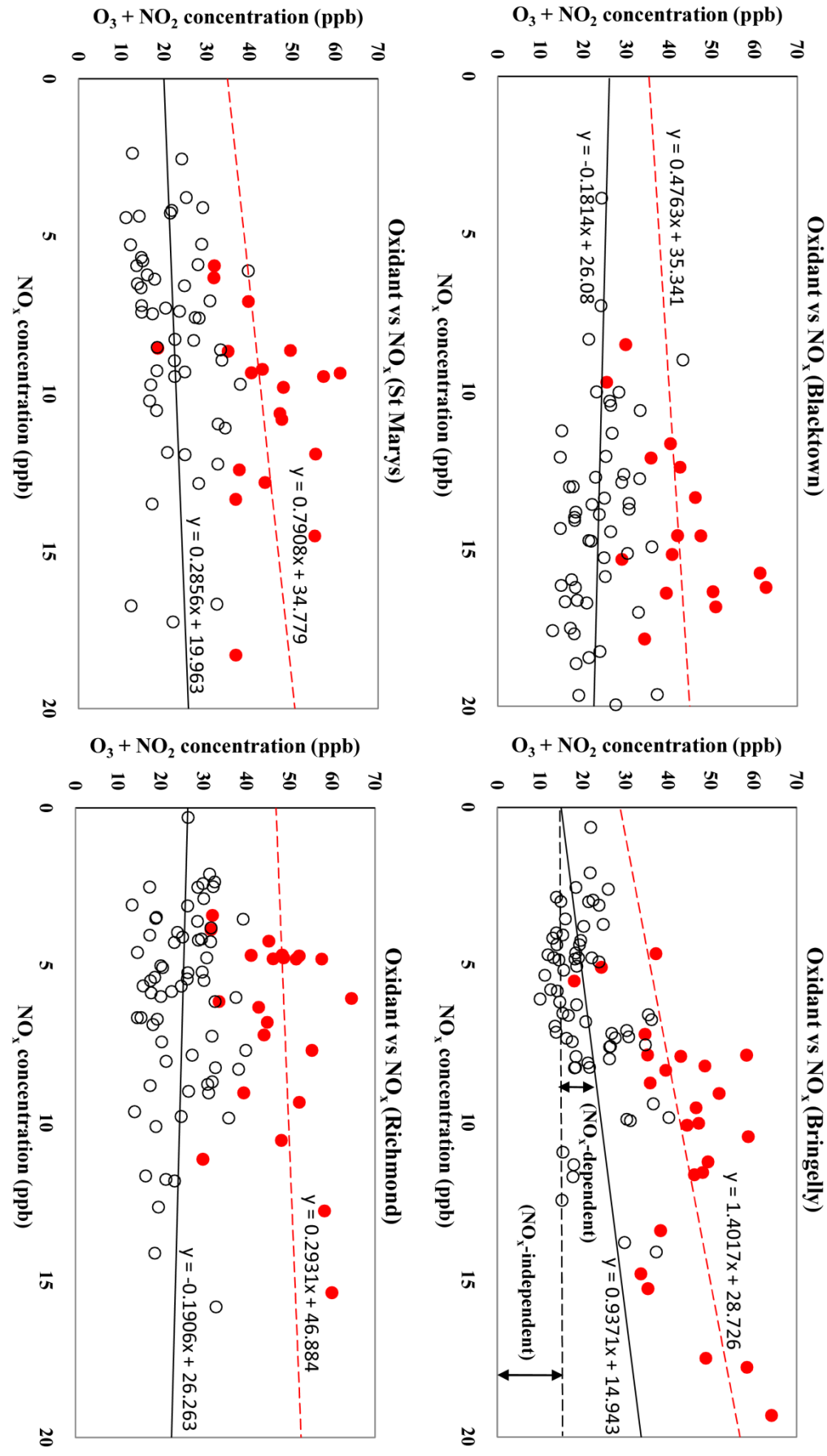


Fig. 7.3 Daytime oxidant level versus NOX in Sydney West in 1998: broken lines for episode, and solid lines for non-episode days.

Similar analysis was conducted with the 1999 summer data, covering three different regions in the Sydney basin, i.e. western, eastern and central Sydney. The daytime background levels of oxidant for non-episode and episode days obtained from the intercepts of the regression lines are summarised in Table 7.2, where the local oxidant contributions are illustrated by the rate of change in background oxidant level with respect to the NO_X concentration (slope). On average, the derived background oxidant for the Sydney area is about 16 ppb for non-episode days and about 4 ppb higher for the episode days. Generally, the daytime background ozone level was more consistent during the non-episode days because it has a small change between 10 ppb to 20 ppb, while for episode days the level varies from 11 ppb to 37 ppb. However, only a few sites such as St Marys, Bargo, Randwick and Liverpool display a clear relationship for the local contribution of the oxidant level (NO_x-dependent) in this year, as the slopes for regression lines between non-episode and episodes days are similar. This implies that the local oxidant level depended on the local activities and pollution emissions, thus its trends varied from site to site and also between episode and non-episode days. It is also noted that the NO_x-dependent contribution apparently existed at the semi-rural inland sites such as Vineyard, Richmond and Bringelly (with higher value slopes). The reason was probably due to the long range transportation of the ozone produced due to emission from the industrial activities and vehicle transports in the coastal area to these inland sites.

Table 7.2 Comparison between episode and non-episode background oxidant levels at Sydney basin sites in 1999.

Region	Site	Non-episode		Episode	
		Background level (ppb)	Slope	Background level (ppb)	Slope
West	Blacktown	15	0.65	16	1.95
	Bringelly	16	0.95	14	2.25
	St Marys	21	0.40	37	0.70
	Richmond	18	0.75	11	3.70
	Bargo	13	0.45	32	0.60
	Vineyard	17	1.15	17	3.10
Eastern	Randwick	16	0.40	21	0.65
	Rozelle	10	0.40	13	0.85
	Earlwood	13	0.35	17	0.80
	Woolooware	16	0.50	17	1.15
Central	Lidcombe	16	0.40	23	1.25
	Westmead	13	0.40	13	1.40
	Liverpool	16	0.50	28	0.70
	Lindfield	18	0.30	18	2.10
	Average	**15.5**	**0.54**	**19.8**	**1.51**

Table 7.3 shows the derived results for background oxidant levels in the year 2000. On average, the non-episode level was 3 ppb higher than the previous year. However, there was not much change in the level for episode days. Furthermore, the local contributions also followed similar trends as in year 1999 but with slightly higher values. A clear oxidant-NO_X relationship between episode and non-episode days can be observed at St Marys and Bargo (in West Sydney), Randwick, Earlwood and Woolooware (in East Sydney), and Liverpool (in central Sydney). Nevertheless, the results for these three years 1998–2000 show that the ideal photo-stationary state cannot be achieved at every site due to some variation in the local contributions, especially during episode days.

Table 7.3 Comparison between episode and non-episode background oxidant levels at Sydney basin sites in 2000.

Region	Site	Non-episode		Episode	
		Background level (ppb)	Slope	Background level (ppb)	Slope
West	Blacktown	18	0.65	17	1.85
	Bringelly	16	1.45	15	2.25
	St Marys	22	0.40	37	0.75
	Richmond	21	0.85	10	3.80
	Bargo	21	0.85	32	0.65
	Vineyard	22	0.45	18	3.10
Eastern	Randwick	20	0.50	21	0.60
	Rozelle	18	0.40	13	0.85
	Earlwood	18	0.50	17	0.85
	Woolooware	15	1.15	17	1.10
Central	Lidcombe	19	0.45	22	1.30
	Westmead	16	0.60	13	1.40
	Liverpool	21	0.35	28	0.65
	Lindfield	19	0.40	18	2.10
	Average	**19.0**	**0.64**	**19.9**	**1.52**

In the subsequent five-year analysis from 2001 to 2005, we focus on the NO_X-independent contributions. From Table 7.4, the upward tendency of the daytime background oxidant level generally occurred for the non-episode days, except for Westmead and Lindfield (in the central part of Sydney). For the episode days, there were some sites that showed an upward trend such as Bringelly, Bargo and Vineyard, all located in the west of Sydney, and which are consistent with the findings stated above. However, the background oxidant level at some other sites was observed to vary irregularly for each year.

Table 7.4 Episode and non-episode background oxidant levels at Sydney basin sites from 2001 to 2005.

Region	Site	Non-episode background level (ppb)					Episode background level (ppb)				
		2001	2002	2003	2004	2005	2001	2002	2003	2004	2005
West	Blacktown	22	23	24	*	*	30	20	24	*	*
	Bringelly	21	21	19	32	37	15	32	32	50	56
	St Marys	22	24	23	39	38	22	19	33	53	57
	Richmond	21	26	25	38	38	35	24	26	45	55
	Bargo	20	23	21	27	31	19	29	32	44	45
	Vineyard	27	25	29	38	38	17	18	28	43	42
Eastern	Randwick	21	21	22	20	24	25	28	23	22	22
	Rozelle	21	20	20	20	20	25	20	17	38	27
	Earlwood	20	19	20	20	24	21	21	21	53	50
	Woolooware	20	19	21	*	*	21	20	27	*	*
Central	Lidcombe	*	*	*	30 (Chullora)	33 (Chullora)	*	*	*	49 (Chullora)	52 (Chullora)
	Westmead	24	18	22	*	*	29	13	23	*	*
	Liverpool	24	*	17	20	35	27	-	15	58	58
	Lindfield	25	18	22	30	*	33	25	37	60	*
	Average	**22.2**	**21.4**	**21.9**	**28.5**	**31.8**	**24.5**	**22.4**	**26.0**	**46.8**	**46.4**

Asterisk (*) represents the non-available data for the correspondence year.

7.3.2.2. Background oxidant level temporal trend

The overall trend of the background oxidant level for non-episode days at a number of monitoring sites in the Sydney basin for the period from 1998 to 2005 is shown in Fig. 7.4. Therein, an upward trend can be seen almost all sites excepting Richmond, where a higher value of the average concentration is observed in the beginning year of the period under investigation. The average trend shown in Fig. 7.4 is computed by considering the values from several monitoring sites in Sydney during the corresponding year of the period.

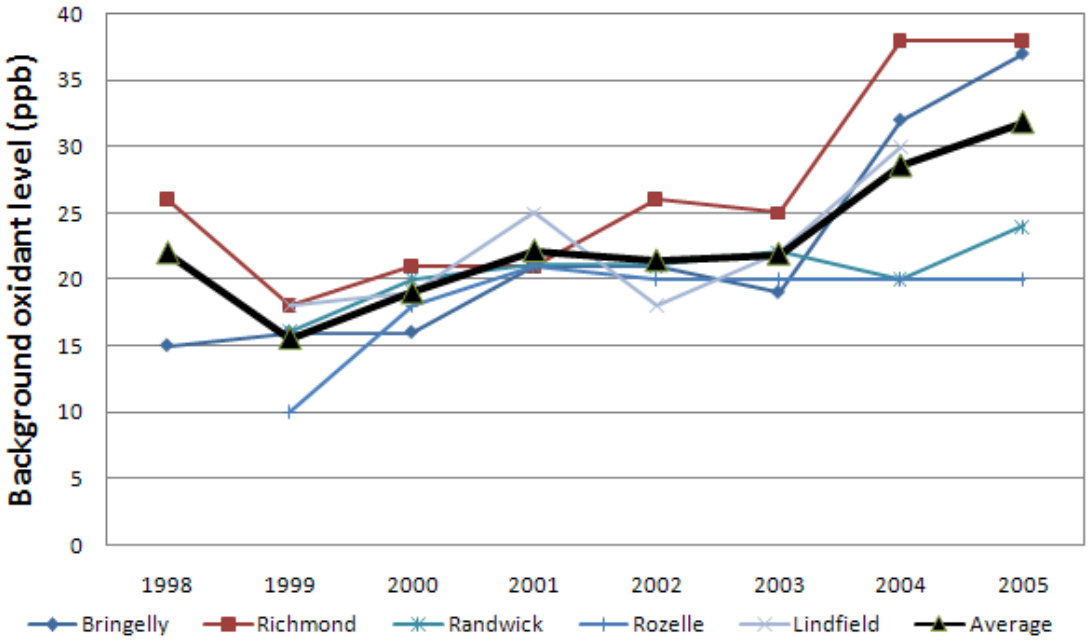


Fig. 7.4 Daytime background oxidant trend for non-episode days at several Sydney sites from 1998 to 2005.

On average, the same upward trend appears for the episode days, as can be seen in Fig. 7.5. However, some sites show no clear trend for the daytime background levels of oxidant (e.g. Randwick and Rozelle in Eastern Sydney), in which the concentration values were fluctuating year by year. It implies that the estimation of BOL by using the Clapp-Jenkin (C-J) method is more suitable for analysis during non-episode days rather than episode days. In addition, the regression analysis is more comprehensive by using data of the oxidants and nitrogen oxides for the entire basin rather than considering the individual regression analysis for each site.

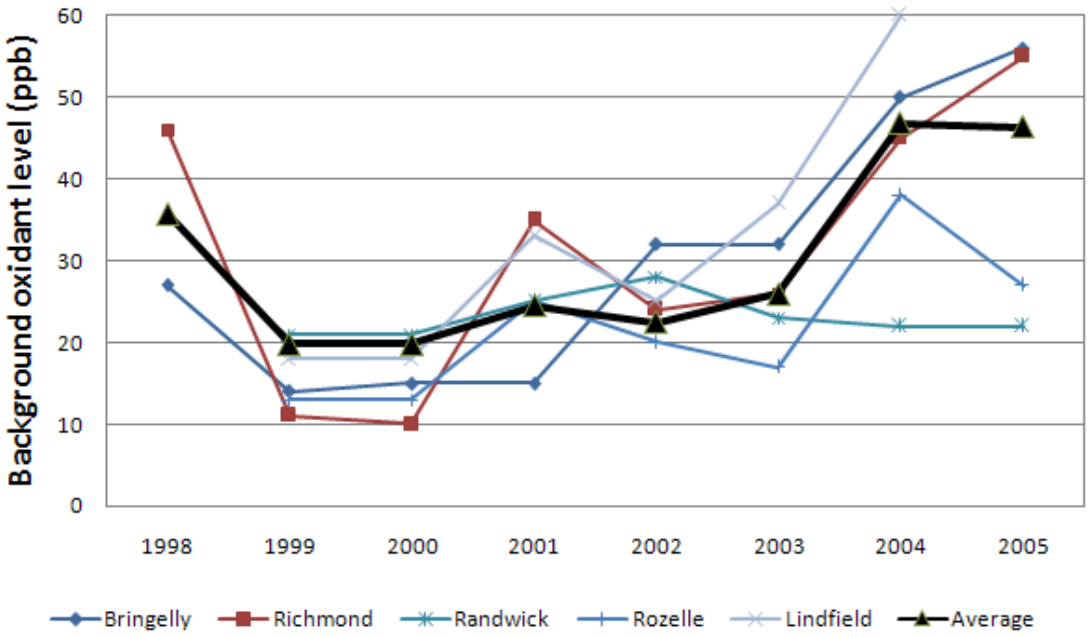


Fig. 7.5 Daytime background oxidant trend for episode days at several Sydney sites from 1998 to 2005.

7.3.2.3. Discussion

The overall comparison for the defined background ozone level for 1998–2005 data (e.g. the St Marys site) using several methods is now summarised in Table 7.5. It is clear that methods 2 and 3 show similar levels of the night-time and daytime background oxidant, however, higher values occurred in 2004 and 2005 using the C-J method. This partially confirms the validity of our proposed approach for the determination of background ozone level. Furthermore, the annual BOL increase (in ppb/yr) by using the C-J linear trend line is slightly higher whereas a consistent trend is observed by the proposed methods 1 and 2. As shown, an offset of about 5 ppb exists consistently between the night-time background ozone and the night-time oxidant (O_3+NO_2) to account for the nitrogen dioxide concentration.

On a larger scale, the upward trend may be explained as due to the increasing global emission of ozone precursors, especially in North Asia in recent years. The link between a local BOL and a continental source of emission has been suggested in Derwent et al. (2008), who used a global chemical transport model, STOCHEM, to show that NO_x emission pulses emitted in one continent (e.g., North America or Asia) can generate surface ozone in another (e.g. Europe) via the transport of precursors and ozone into the troposphere and then the mixing down of the air above

to the surface level. On a continental scale, Oltmans et al. (2006) recorded a slight increase in surface ozone per year at some (but not all) locations of the world and they also emphasised the importance of the relative contribution of the stratosphere to tropospheric ozone. Notably, the upward trend reported in this paper for the local BOL in Sydney is also in line with the results obtained by Jaffe and Ray (2007) in the US and by Simmonds et al. (2004) in Europe.

Table 7.5 Background ozone level determination (in ppb) at St Marys using several methods.

	Methods	1998	1999	2000	2001	2002	2003	2004	2005	Trend (ppb/yr)
1	Night-time background ozone (daily average night-time when NO=0)	15.6	16.0	16.5	16.9	17.3	17.8	18.2	18.6	+0.43
2	Night-time background oxidant (daily average night-time when NO=0)	20.1	20.8	21.4	22.1	22.7	23.4	24.1	24.7	+0.66
3	Daytime background oxidant for non-episode days (by C-J method)	20.0	21.0	22.0	22.0	24.0	23.0	39.0	38.0	+2.63
4	Daytime background oxidant for episode days (by C-J method)	27.0	37.0	37.0	22.0	19.0	33.0	53.0	57.0	+3.27

7.4. Quantisation methods to refine the night-time BOL definition

7.4.1. A method to deal with unavailable pollutant data

As described previously, the proposed method for the night-time BOL determination was based on the relationship between ozone, and nitric oxide measured data. The condition of 'no nitric oxide present for at least two hours consecutively' is rather difficult to be made available especially in the daytime, as the photochemical effects were dominant during day time. Therefore, we only considered the evening and the early morning ozone data (i.e. from 7.00 pm to 8.00 am the next morning) for the background ozone level, in which the anthropogenic effect can be minimised.

Unfortunately, there still exists some hourly night-time data where the background ozone cannot be determined in which NO=0 is not present. Thus, we replaced that missing data by the linear regression of the previous and subsequent measured values at the station. This can be done by taking the local correlation of the ozone

concentration and the NO data, O_3:NO (Wahid et al., 2010a). The intercept of the linear regression line from this correlation indicates the background ozone level at the particular missing point, which could be approximated as the ozone concentration when NO is equal to zero. Fig. 7.6 shows an example of the O_3:NO correlation at an absence point, in which we take the data two hours before the absence point and two hours after, for that regression. This procedure was repeated for every missing point for the entire data. The intercept value is commonly given by the following equation,

$$b = \frac{\left(\sum y\right)\left(\sum x^2\right) - \left(\sum x\right)\left(\sum xy\right)}{n\sum x^2 - \left(\sum x\right)^2}, \tag{7.1}$$

where in this case, b is the estimated background ozone level (in ppb) at the missing point, y is the ozone concentration (in ppb), x is the NO concentration (in ppb), and n is the number of data used for the linear regression.

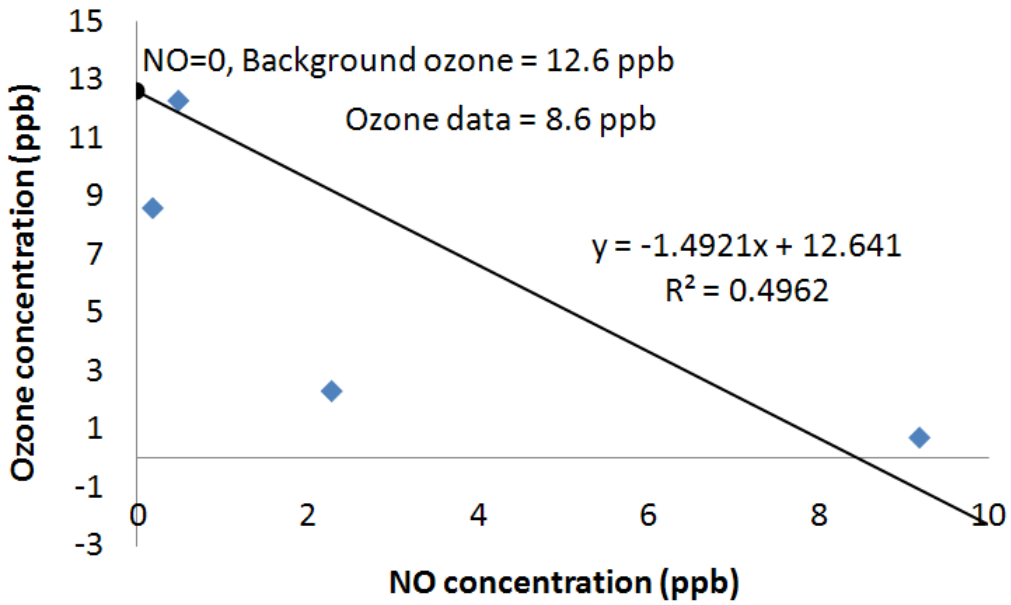


Fig. 7.6 Correlation of O_3 and NO to estimate the background ozone level at an absence point.

To verify the reliability of the approach, 25 data points were sampled for pollutant levels at two monitoring stations in Sydney and compared, as shown in Table 7.6. This data is consecutive hourly data taken at a specified time (i.e. from 7.00 pm to 8.00 am the next morning). As can be seen from the table, we are unable to obtain zero NO data at every hour to fulfil the proposed BOL definition. By using the described method as summarised in equation (7.1), the estimated BOL when zero

NO is not available can be calculated. Generally, the calculated BOLs show higher values than the measured ozone data, but interestingly, there are not many changes in the level when zero NO is present (shown in bold values in Table 7.6). This partially confirms the reliability of the proposed solution.

Table 7.6 Comparison of pollutant concentrations and the estimated BOL at two sites in Sydney.

Data no.	Randwick site				Vineyard site			
	NO	**NO_x**	**O_3**	**BOL**	**NO**	**NO_x**	**O_3**	**BOL**
1	**0.0**	0.0	**27.1**	**24.9**	**0.0**	4.3	**13.8**	**13.0**
2	**0.0**	0.0	**27.5**	**27.0**	**0.0**	3.4	**13.3**	**11.8**
3	0.2	0.2	27.2	27.2	**0.0**	5.9	**10.9**	**11.7**
4	0.1	1.1	26.2	26.9	1.1	9.0	7.2	10.7
5	0.5	2.5	25.7	26.1	5.8	15.0	8.5	14.9
6	2.0	4.5	25.4	25.9	5.9	14.9	12.5	18.7
7	**0.0**	2.6	**25.0**	**25.6**	**0.0**	2.4	**24.5**	**22.2**
8	**0.0**	1.8	**26.5**	**25.6**	**0.0**	4.0	**20.9**	**20.0**
9	**0.0**	2.6	**25.3**	**25.6**	**0.0**	3.7	**21.1**	**20.7**
10	**0.0**	2.3	**25.4**	**24.4**	0.5	10.4	12.3	17.2
11	1.0	9.4	17.7	20.5	0.2	12.2	8.6	12.6
12	0.6	13.5	6.7	18.6	2.3	19.0	2.3	9.0
13	5.5	22.6	1.2	11.3	9.2	25.2	0.7	6.4
14	5.3	21.8	1.0	3.3	8.6	23.6	0.3	2.4
15	10.6	23.6	0.8	0.9	11.2	24.7	0.4	0.1
16	27.9	41.9	1.3	-0.7	13.6	25.8	0.3	-0.5
17	65.2	79.8	1.9	8.8	25.0	36.9	1.0	0.2
18	127.9	145.5	11.6	17.2	41.8	57.3	2.5	15.5
19	4.0	7.5	25.0	22.3	25.4	41.3	7.8	20.5
20	0.2	3.4	23.9	24.1	**0.0**	3.6	**22.7**	**20.1**
21	**0.0**	3.5	**23.4**	**24.0**	**0.0**	3.8	**20.8**	**18.8**
22	**0.0**	2.5	**23.9**	**24.3**	**0.0**	5.7	**17.2**	**17.4**
23	**0.0**	1.7	**24.9**	**24.3**	0.3	7.5	14.4	14.9
24	**0.0**	1.3	**25.1**	**24.7**	**0.0**	**9.8**	**9.0**	**11.8**
25	0.3	2.0	24.2	24.6	**0.0**	**7.3**	**12.5**	**10.9**

* Note: All concentrations in part per billion (ppb) units.

7.4.2. Night-time BOL based on O_3-NO_x relationship

In Duc et al. (2012), from the ambient measurement data, the stipulation of no nitrogen dioxide present, that is for at least two hours consecutively to ensure no ozone reaction with nitrogen oxides, is rather unlikely in an urban area. As the zero concentrations of nitric oxide and nitrogen dioxide (i.e. $[NO]=0$, $[NO_2]=0$) are

difficult to obtain via measurements, it would be hard to determine accurately the non-photochemical night-time BOL. Motivated by this idea, we will refine the definition by Duc et al. (2012) to be more generic, such that it could be used globally without the presence of zero NO or NO_2 data, by incorporating the idea of Clapp and Jenkin (2001).

7.4.2.1. Methodology

Fig. 7.7 shows the linear relationships of the mixing ratio of $NO+NO_2+O_3$ with NO_x during night-time by using the regression analysis. From the idea proposed by Clapp and Jenkin (2001), the NO_x–independent and NO_x–dependent contribution appears in the plot. The linear equation is given as follows:

$$[O_3 + NO_x] = m[NO_x] + c, \qquad (7.2)$$

where m is the slope value which depends on the NO_x local contribution and c is the intercept value. At the intercept point, $[NO_x]=0$, we can consider this intercept as the average daily non-photochemical night-time BOL because we only use the night-time data to develop the regression profile which predominantly excludes the photochemical effect. The variation of hourly BOL is much dependent on the NO_x concentration from local anthropogenic contributions that still exist from the daytime.

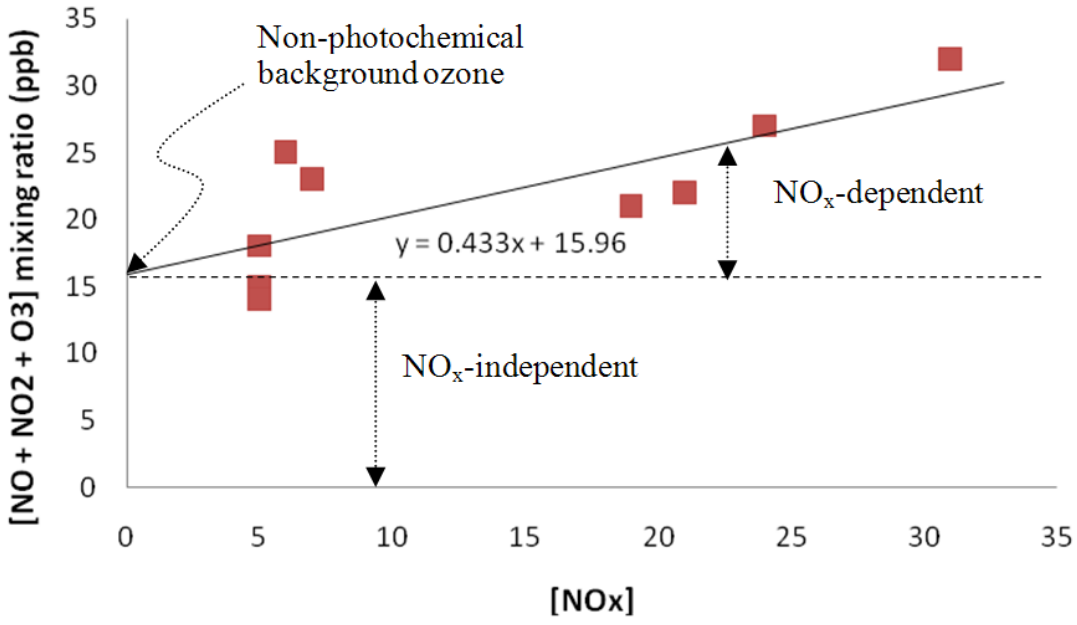


Fig. 7.7 Daily average of non-photochemical night-time BOL shown by intercept of the linear regression line.

To simplify the procedure, we can also get the daily BOL value by plotting the oxidant profile against NO_x data and finding the intercept value from the regression analysis, as depicted in Fig. 7.8. Next, the hourly BOL can be determined by extrapolating the main curve from the daily average trend to the respective hourly data and getting their intercept values, as illustrated therein. The results for the application of the proposed methodology are as appeared in Wahid et al. (2011).

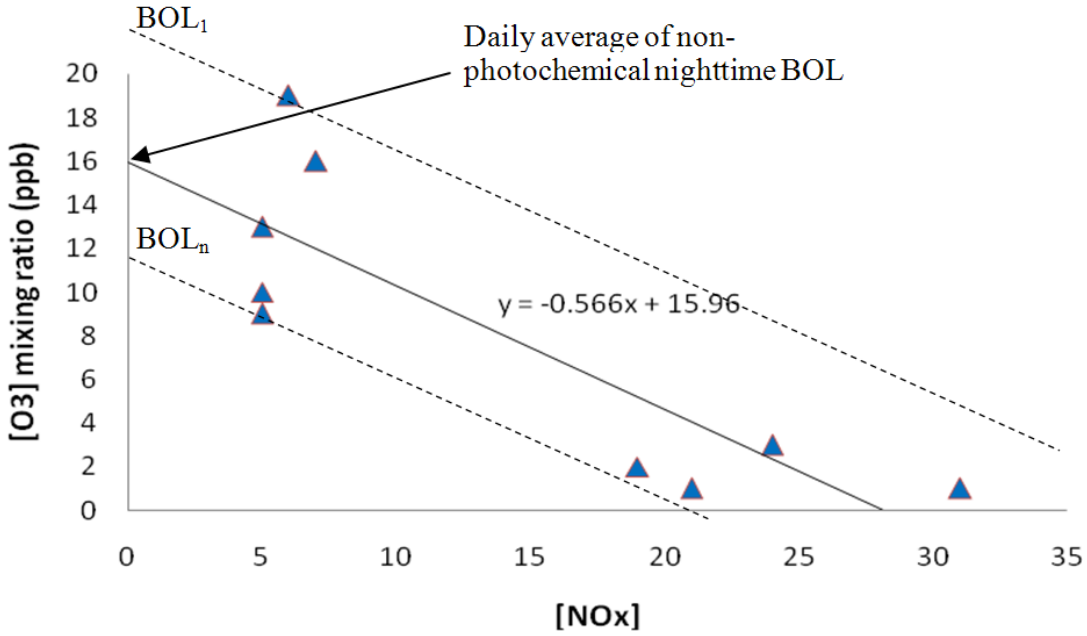


Fig. 7.8 Hourly average of non-photochemical night-time BOL shown by the intercept of the dashed lines.

7.4.2.2. Validity of the proposed method

To verify the proposed method for BOL determination, we test the consistency with several techniques using monitoring site data. First, we try to determine the BOL based on daily mean ozone concentration data collected during summer 2003 at a semi-pristine site in Bargo located in the south-west of Sydney which may be considered as a clean site. Fig. 7.9 shows the difference of the BOL obtained from the proposed method and the mean 24-hour ozone concentration (i.e. including the daytime ozone) measured for about one month. As can be seen, the levels are similar on most of the evaluated days. Some differences exist on the event days, which are expected due to the site being not a completely clean site. We assume that some high ozone concentration during the daytime is also affected by the photochemical ozone which is transported to this area instead of its natural processes. The average

background ozone concentration given by the proposed method is 18.4 ppb compared to 20.7 ppb as obtained from measurement data at the site.

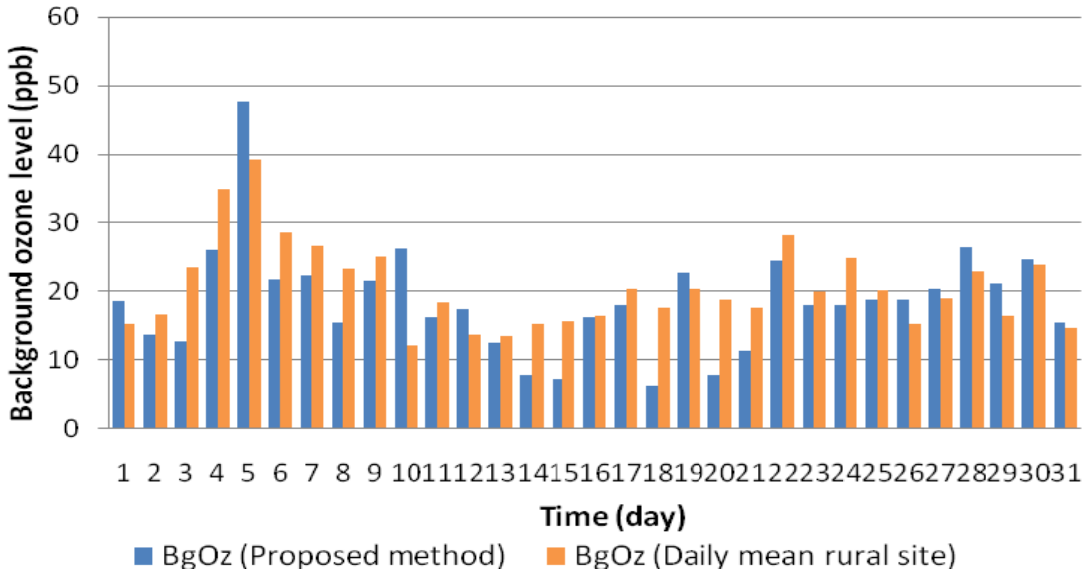


Fig. 7.9 Comparison of background ozone level between the proposed method and mean concentration at a semi-pristine site.

Furthermore, we compare the hourly BOL corresponding to other air pollutant concentrations. Fig. 7.10 shows an example of hourly data for BOL, ozone concentration and NO_x concentration. We found that the concentration level is similar when the NO_x value is equal to zero, which prompts the absence of photochemical reactions. Thus, the ozone concentration at those points can be considered as BOL, which is in line with the proposed definition.

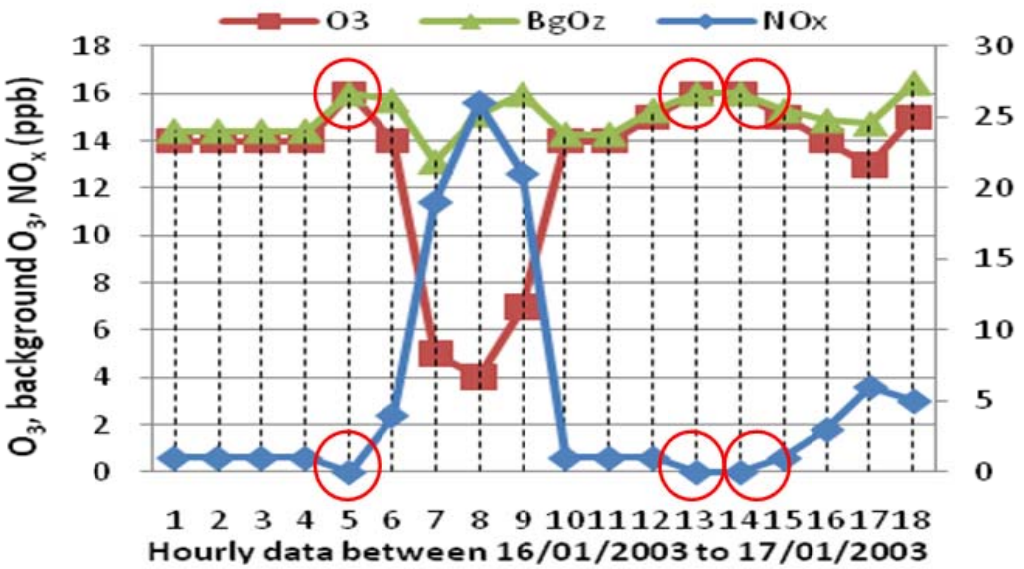


Fig. 7.10 Hourly data at Randwick station showing BOL equal to ozone concentration when NO_x=0. (The right axis represents NOx scale in ppb).

7.4.3. A generic method to determine the duration time for night-time BOL

Duc et al. (2012) used ambient measurements to estimate the non-photochemical BOL by taking the mean value of night-time and early morning hourly ozone concentrations (i.e. from 7.00 pm to 8.00 am the next morning), when nitric oxide (NO) is empirically considered as being absent, assuming the photochemical processes occur only during daytime. However, the duration time from the definition is too specific and depending on different locations as well as seasons. The aim of this work is to estimate the start- and end-time of a day period in which the ozone concentration can be used for the determination of BOL. The suggested method can be used to determine the background level at any part of the globe and in any season without relying on data obtained at remote sites (Wahid et al., 2011).

7.4.3.1. Methodology

For the case study, the investigation is based on the ambient measurement data collected at various monitoring stations across the Sydney basin in Australia. By analysing the diurnal distribution of the ozone concentration (O_3, in ppb), nitrogen oxide concentration (NO_x=NO+NO_2, in ppb) and hourly temperature (in °C), we found a pattern which can be illustrated in Fig. 7.11. The figure shows the mixing ratio of O_3 and NO_x concentrations against the hourly time at the Bringelly station, located in the West of Sydney by using two months' data in summer 2003. Two profiles representing each pollutant are plotted by using a higher-order polynomial line via the least square values for every hour. From the figure, we can see that the ozone starts to form in the morning and will increase with temperature. During that time, the NO_x concentration decreases, as contributing to the ozone formulation, and will rise again in the evening. It is suggested the two intersections of O_3 and NO_x concentration profiles in the morning and evening are the limits in the day period to be used for the determination of the non-photochemical night-time BOL. This is meant to exclude the photochemical part contributing to the ozone concentration, present during the daytime due to photochemical reactions, which is mainly caused by anthropogenic emissions.

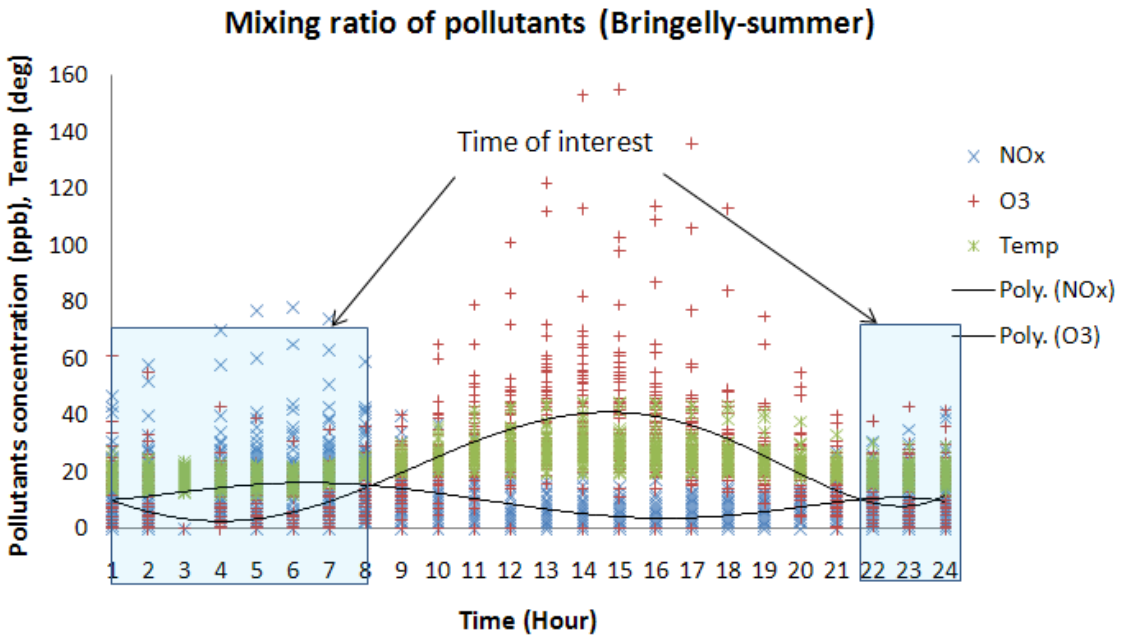


Fig. 7.11 Diurnal distributions of O_3 and NO_x at the Bringelly site during summer 2003.

The patterns and the corresponding intersection points seem to be different for every site and season. Fig. 7.12 shows the pollutants' mixing ratio during the winter season at the same site, wherein the period for the ozone production appears to be shorter during daytime. Thus the time interval to be considered for BOL in winter is longer than in other seasons.

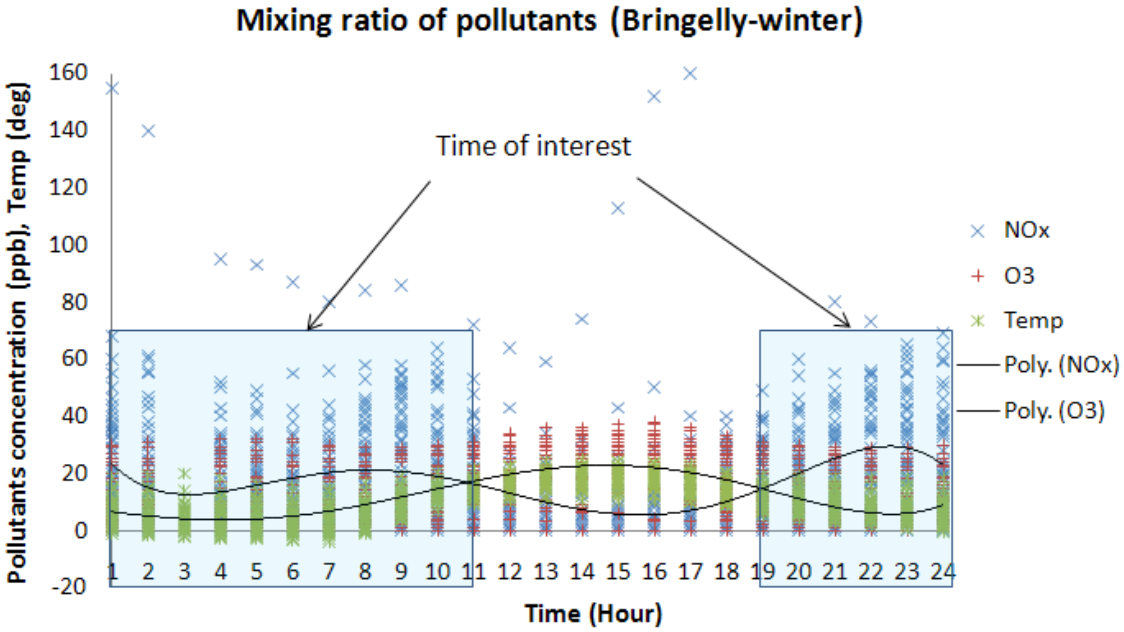


Fig. 7.12 Diurnal distributions of O_3 and NO_x at the Bringelly site during winter 2003.

For the purposes of generalisation, we could repeat the same method by considering data from every site in the region to obtain the average points. The start- and end-times by using the above method at various sites in Sydney are listed in Table 7.7. From here, the BOL definition for the Sydney basin can be determined by considering ozone concentration data from 9.00 pm to 8.00 am the next morning (i.e. as summer is the period of interest). It shows that the start-time begins later than that specified by the empirical definition (Duc et al., 2012), i.e. 7.00 pm.

Table 7.7 The suggested start (evening time) and end (next morning time) to determine the non-photochemical night-time BOL.

Site	Start time	End time	Site	Start time	End time
Bargo	9.00 pm	8.00 am	Randwick	7.00 pm	7.00 am
Bringelly	10.00 pm	8.00 am	Richmond	10.00 pm	7.00 am
Earlwood	8.00 pm	10.00 am	Rozelle	9.00 pm	9.00 am
Liverpool	8.00 pm	11.00 am	St Marys	8.00 pm	8.00 am
Oakdale	10.00 pm	7.00 am	Vineyard	10.00 pm	8.00 am
Average for Sydney basin	**9.00 pm**	**8.00 am**			

Hence, we can form a generic function for the period of interest for BOL definition, which probably can be used globally at various geographical locations and in different seasons, as given in the following equation:

$$t' = interval\left(f_1(\mathrm{NO_x}'), f_2(\mathrm{O_3}')\right), \tag{7.3}$$

where *interval* is the function to find the two intersection points of the higher order polynomial lines of f_1 and f_2, NO_x' is the mixing ratio of the NO_x component, and O_3' is the mixing ratio of the O_3 component for the season under consideration.

7.5. Chapter conclusion

This chapter has presented some refinements of the BOL definition and presented measurement-based methods to determine the night-time and daytime background ozone levels. The night-time BOL is defined as the average of ambient measurements of hourly ozone values from night-time to early morning when nitric oxide (NO) is not present for at least two hours consecutively. Ambient air quality data collected at monitoring sites in the Sydney basin is used to derive the local BOL and to investigate the BOL trend over the years. The night-time BOL results have

been compared with a method to estimate the background oxidant concentration introduced by Clapp and Jenkin (2001).

The night-time BOL as defined and derived in this work is shown to be suitable for the Sydney basin. From several analyses, there is a clear upward trend in background ozone concentration at nearly all the Sydney monitoring sites. For other regions, such as the Lower Hunter in the north and Illawarra to the south of the Sydney basin, the BOL and their trends can be determined and the results may be different from those of Sydney as the BOL (and hence its trend) is dependent on a number of conditions such as the meteorological flow, pollution sources and terrain conditions in those regions.

Moreover, several refinements of the BOL quantisation methods were also presented involving a method to deal with unavailable zero nitric oxide (NO) data in the determination of the proposed BOL, an alternative method to define night-time BOL using the O_3-NO_x relationship, and a method to determine the duration time for night-time BOL. Notably, the approaches are generic in determining BOL according to the period of interest for non-photochemical activities, hence it may be applied at any location and in any season.

Chapter 8
METAMODEL APPLICATION IN AIR QUALITY MODELLING

8.1. Introduction

The proposed methodological approaches have been described comprehensively in Chapter 3 to Chapter 6, while a significant problem to be addressed has been explained and elaborated in detail in Chapter 7. This chapter will discuss several possible practical problems to be addressed in the atmospheric studies using the neural network based metamodel approach, for which the theories and algorithms were considered in previous chapters.

Ozone is known as a secondary pollutant gas; its formation is extremely complicated and non-linear as compared to other air pollutants. A special measurement could be used to assess its current level, however, more difficult tasks are to make a future prediction of its levels temporally and to estimate the distribution of its concentration spatially. Deterministic air quality (AQ) models are always used by the policy maker to deal with this type of problem, however, a simulation by an AQ model is quite tedious because of the nonlinear nature of some particular chemical reactions involved in the model formulation, which is also subjected to some uncertainties. In this chapter, the metamodel method will be presented to assist the AQ models or other air quality assessment methods in order to improve the reliability of its estimation and to build a computationally effective model, particularly for the case of ozone and its background level. As the conceptual framework of the approach is generic, the proposed implementation can be extended for the estimation of other air pollutants in the temporal and spatial domains.

8.2. Short-term temporal prediction model of BOL

The idea in the metamodelling approach is that we can design a model network for short-term prediction of the background ozone level (BOL) for each measuring station in the analysed domain. In a similar attempt, the one hour ahead prediction model of BOL at selected sites, namely Blacktown, Lidcombe and St. Marys, has been presented in Wahid et al. (2010a). However, the methodology is much dependent on a limited number of input parameters, namely time, NO, NO_2 and O_3 concentrations. To obtain a more comprehensive model, other parameters especially the meteorological data should be taken into consideration, as it has a direct effect on the level of background ozone.

An extended analysis will be presented here that includes the incorporation of some meteorological data, the hourly prediction for 24 hours ahead, and the performance comparison between the OLS algorithm and the improved OLS algorithm in the applied problem. The addition of the two parameters is expected to improve the performance of the prediction as compared to the previous attempt.

8.2.1. Model development

A proper selection of the input and output characteristic is essential in order to make the RBFNN learn with a fast convergence. Typically, more input data is better so as to make the model more comprehensive and the interpretation more convincing. In this work, six input parameters are used (i.e. including the addition of two meteorological parameters from the previous work), namely the hourly time information, the nitric oxide (NO) concentration, the nitrogen dioxide (NO_2) concentration, the ozone (O_3) concentration, the wind speed (WSP) and the ambient temperature (TEMP) as measured at the monitoring stations. The wind direction (WDR) is not considered here as it will not allow significant performance improvement unless it is used for constructing the spatial model.

As described in Chapter 7, the absence of NO indicates that there is no scavenging of ozone by NO to produce NO_2 and hence no photochemical reaction is occurring. However, the availability of zero NO cannot be obtained at every hour, especially at

the urban sites. Therefore, we use the local time regression analysis (as appeared in section 7.4.1) to replace the missing data in order to define the BOL at that particular hour. Since we intend to predict the BOL, we use the defined BOL from the measured data at some interval time as the target output for training the model. The size of the interval depends on the prediction horizon. For short-term prediction, one hour to twenty four hours are adopted as the interval times for this evaluation. The data for seven hours of specific interest are selected as the outputs to analyse the performance of the methodology. The input-output of the model is illustrated in Fig. 8.1.

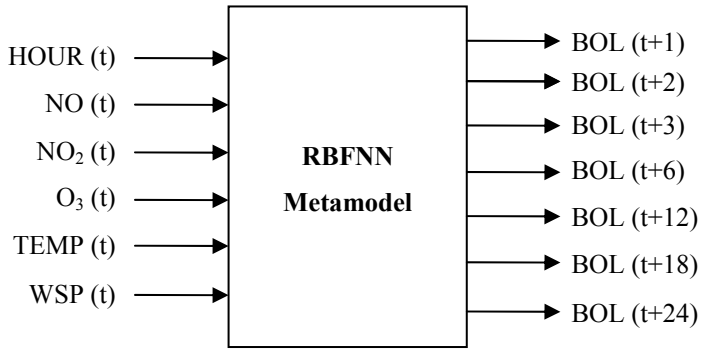


Fig. 8.1 Inputs and outputs of RBFNN model for short-term prediction of BOL.

The entire inputs and targets have to be normalised (e.g. for a minimum of zero to a maximum of unity, by using *mapminmax* function in MATLAB), in order to make them contribute with the same influence to the RBFNN. This also allows the Gaussian activation function to squash all incoming data and to make the execution faster. Furthermore, we should only consider 'interesting data', which means that the data patterns contain as many possible different significant input and output situations. Thus, if there is too much similar data we can remove a part of it and accept only the more interesting data. Once everything is ready for the training set, we can execute to train the network by using the RFBNN metamodel approach, i.e. a method as proposed in Chapter 4 will be used in this evaluation.

The test set should follow the same manner as with the training set, i.e. the time interval for the test is the same as the time interval used for the training set. Once the learning process is finished and the accuracy by some test sets is satisfactory, the network can be used for the prediction of other available data, at the same site.

The codes for running the metamodel construction are listed in Appendix E-1.

8.2.2. Analysis and discussion

In this analysis, we used the data recorded in the year of 2001, at two monitoring stations in the Sydney basin, namely at Blacktown and Randwick. By using an improved RBFNN method that featured an adaptively tuned spread parameter (as discussed in Chapter 4), we initialise the training process by setting the initial spread parameter and prescribed error goal. After several epochs, the network is constructed once having met the set goal. The model setting for both analysed sites are summarised in Table 8.1. The network will then be tested with the sample testing data for the monitoring stations.

Table 8.1 The RBFNN model setting for short-term prediction of BOL.

	Model setting		**Constructed model**		**Dataset**	
Site	Initial σ	*MSE* goal	Hidden neurons no.	Activation function	Training data points	Testing data points
Blacktown	0.1	0.014	20	Gaussian	248	100
Vineyard	0.1	0.012	17	Gaussian	202	100

8.2.2.1. Prediction results

Blacktown is considered a suburban site that is located in the west of Greater Sydney. By using the constructed model, the results of prediction on the testing set are shown in Fig. 8.2 (a–f). As can be seen, most of the values have shown reasonable results for lesser prediction horizon, where it follows the pattern of the actual values derived from ambient measurements. However, the performance of the model based on the R^2 index deteriorates towards the twenty-four-hour horizon, as expected and which is depicted in the figure. The performance for the twenty-four-hour prediction horizon could be improved further by using a cascaded model structure, for example as presented by Coman et al. (2008) for the prediction of ozone concentration, but the methodology is quite tedious as it requires more sub-models to predict for each hour.

Notably, it is quite difficult to obtain the non-photochemical condition at every hour, thus information of the background ozone level during that hour cannot be obtained. By using this model, the background ozone still can be predicted for events as in past hours under the non-photochemical conditions. Therefore, it is envisaged that the model could be used on-line for continuous prediction of the BOLs.

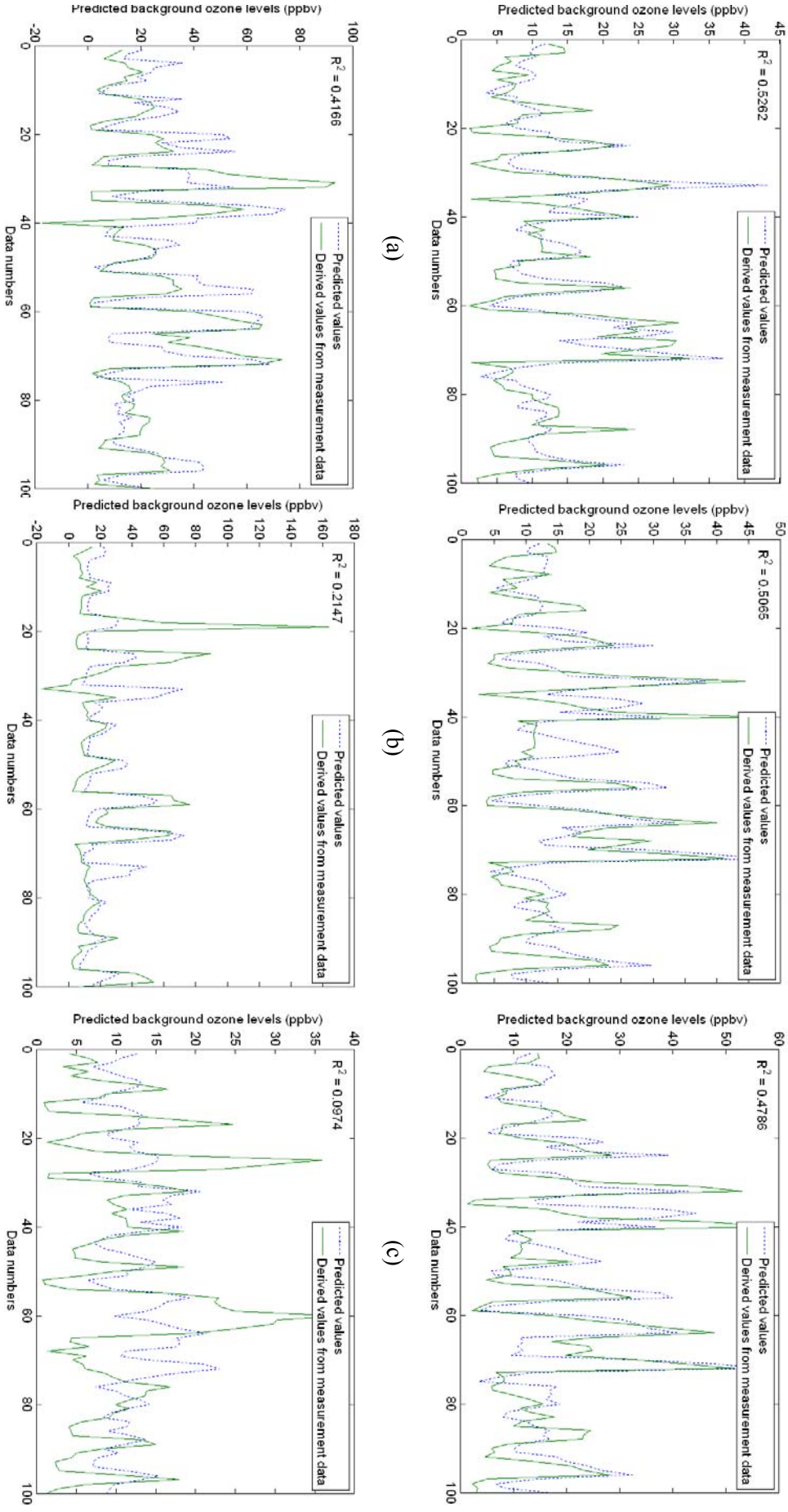


Fig. 8.2 Short-term prediction results of BOL at Blacktown site for: a) 1-hour; b) 2-hour; c) 3-hour; d) 6-hour; e) 18-hour; and f) 24-hour, respectively.

The same routine was applied for Vineyard, a site located in the north-west region of Sydney. The example of prediction results for one-hour and six-hours ahead are depicted in Fig. 8.3, showing better results than the Blacktown model where only a few points were different from the values derived from measurements, probably due to more non-photochemical data being available during the training stages. Overall, the Vineyard model gives better performance for lesser prediction horizons but the performance decreases drastically towards the higher prediction hours (e.g. for eighteen-hours and twenty-four-hours).

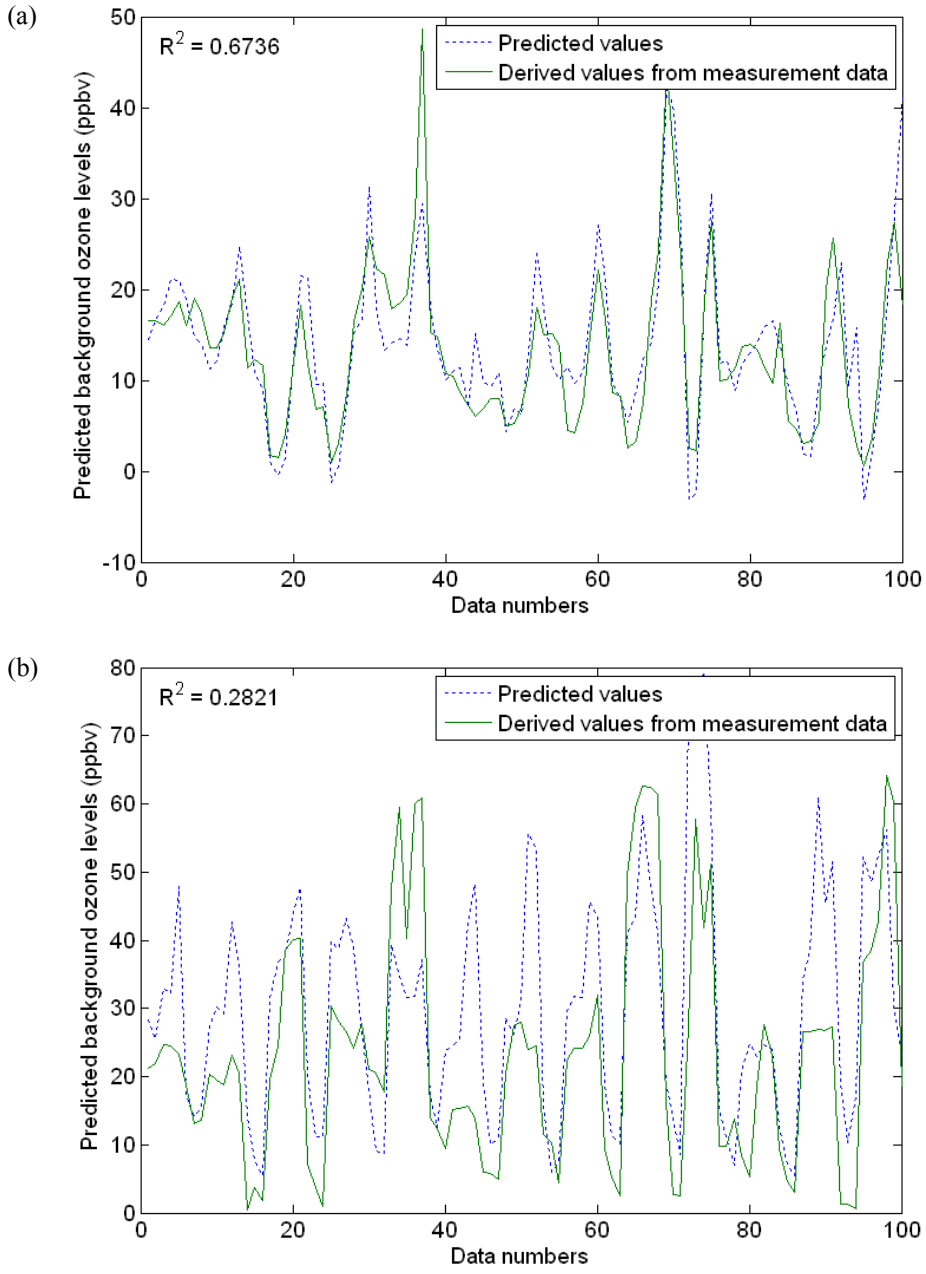


Fig. 8.3 Short-term prediction results of BOL at Vineyard for: a) 1-hour; b) 6-hour, respectively.

8.2.2.2. Performance analysis and discussion

This section aims to evaluate the performance of the models for different prediction horizons in a twenty-four hour period, and to compare the performance of the model which is constructed by an improved RBFNN featuring the orthogonal least squares algorithm with adaptively tuned spread parameters (i.e. OLS-ASP). Fig. 8.4 and Fig. 8.5 show the performance of the models to predict the BOL for different horizons at the Vineyard and Blacktown sites, respectively. As can be seen from the figures, some good results can be shown for the less than six-hour prediction horizons (i.e. give higher values of R^2 and lower values of mean absolute error, *MAE*, index) but the performance of the model degrades towards the twenty-four hour prediction horizon.

In a typical scenario, an exponential trend line of performance is expected for this type of evaluation. However, in this case the worst performances appeared in the six to eighteen-hour prediction horizon). This occurs because most of the prediction horizons in the middle fell in the daylight hours in which the BOL cannot be defined correctly from the measurement data, thus affecting the model performance during the training process.

We have presented a new metamodel using an adaptive radial basis function neural network for predicting the hourly background ozone level with reasonable accuracy. The results obtained indicate the promising application of the proposed method in the short-term analysis of background ozone levels and emission impact assessments for air quality modelling.

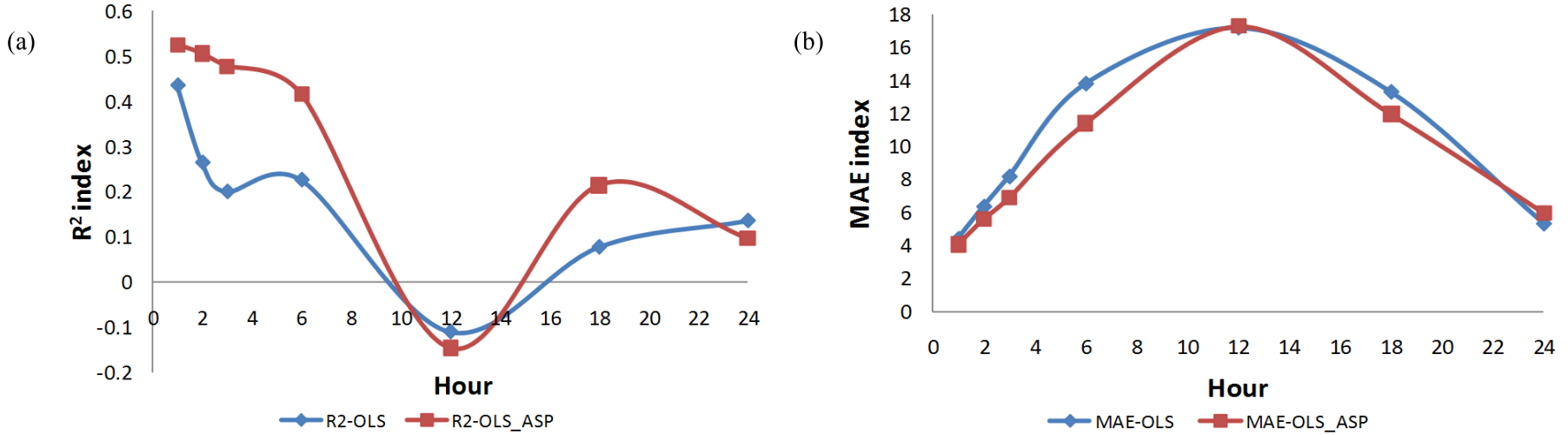


Fig. 8.5 The models performance for hourly BOL prediction for 24 hours at Blacktown site: a) performance based on R^2 index; b) performance based on *MAE* index.

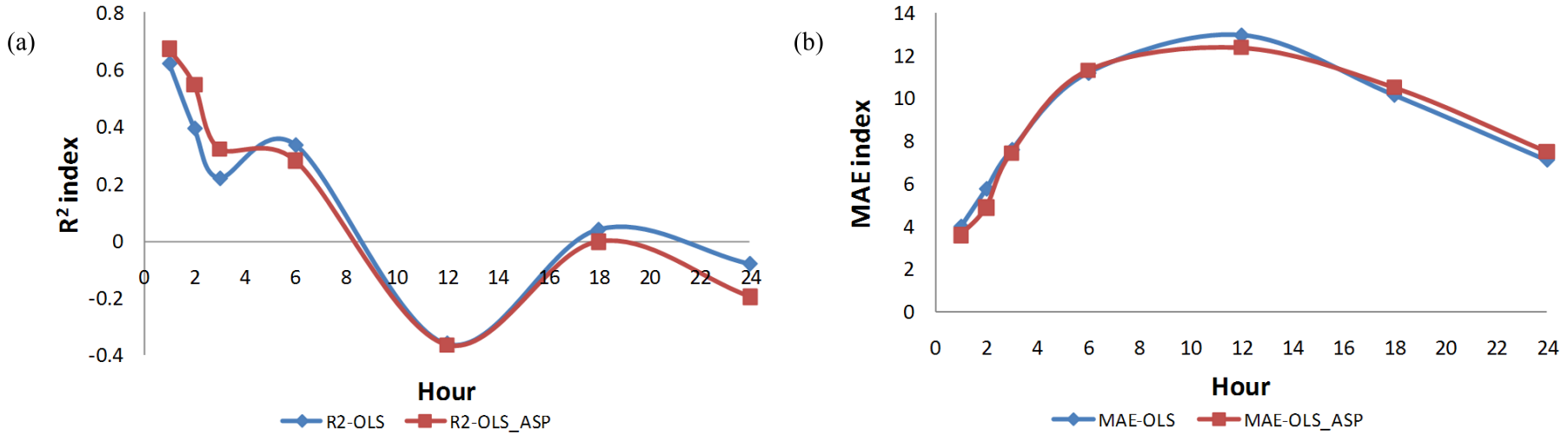


Fig. 8.4 The models performance for hourly BOL prediction for 24 hours at Vineyard site: a) performance based on R^2 index; b) performance based on *MAE* index.

8.3. Long-term estimated model of BOL

The idea in our metamodelling approach here is slightly different when compared to the short-term prediction, described in the previous section. For the short-term prediction, we were trying to predict the pollutant levels at certain horizons in which we used a 24 hour time-frame. In this work, we are analysing the effect of some factors (i.e. time, pollutant data and meteorological data) on the level of background ozone (Wahid et al., 2010b). It is expected that a generic model for BOL could be constructed in which it can be used for estimating the long-term trend of the BOL. In this study, our scope is to model a network to predict the BOL at several urban sites in Sydney, from which the results may be extended to the whole region by constructing a generic model that considers the entire monitoring stations' data in the domain.

8.3.1. Model development

In this work, we analysed several input variables including pollutants data and the related meteorological data, to look into their variation effect on the accuracy of the constructed model. The pollutants data include the concentrations of ozone (O_3), nitrogen oxide (NO), nitrogen dioxide (NO_2), carbon monoxide (CO), sulphur dioxide (SO_2), volatile organic compounds (VOC), particulate matter of size <10μm (PM_{10}), and size <25μm (PM_{25}). Other meteorological data included the wind speed (WSP), wind direction (WDR), and the air temperature (TEMP). In addition, the time information was also considered as an input variable.

The background ozone level has been set as the target output of the network. From the several definitions of BOL in Chapter 7, the non-photochemical night-time BOL is considered. Fiore et al. (2002) suggested that the estimation could be more accurate if the correlations with reactive nitrogen oxides (NO_Y) are taken into consideration. It implies that a component of the background ozone is produced from natural precursor sources, which include NO_X emissions from soil and lightning, and hydrocarbon emissions from vegetation. However, according to the specified definition of BOL, it is typically difficult to obtain data that contains zero NO in which the BOL cannot be determined. Thus we replaced that missing data by

the linear regression of the previous and subsequent measured values at the station, which method was discussed in section 7.4.1.

8.3.2. Analysis and discussion

We used the hourly data collected by the Department of Environment, Climate Change and Water (DECCW), New South Wales. The data used in this study covered a period of five months from November 2000 to March 2001, at three sites in the Sydney region namely Randwick, Blacktown and Vineyard (Wahid et al., 2010b). Two of the sites' results will be presented here, and an extended work to construct a more generic model for all sites will be covered in the next following section.

Case A: Randwick station

The Randwick site is located in the east region of Sydney (within a 5 km radius from the Sydney CBD). As reported in (Duc & Azzi, 2009), this urban site was recorded as having the highest average level of non-photochemical background ozone in the east region for the years 1998 to 2005.

Five simulations of different combinations of inputs as shown in Table 8.2 have been performed using an RBFNN metamodel. The first simulation was performed using the time information, the photochemical concentration and the ozone measurements. For the next three sets, we added to the first set the meteorological information that includes the air temperature, the wind speed and the wind direction with three different combinations. For the entire simulations, we initialised the training process by setting the initial spread parameter to be 0.1 and the error goal to be 0.005. After several epochs that depended on the inputs data used, the network was constructed once the set goal had been met.

A suitable compromise between the network size and the selected variable inputs in the background ozone estimation showed that the best results in terms of the coefficient and index of agreement are obtained by using the inputs in the set number 3 as depicted in Table 8.2. The network size is 17, which means that 17

radial basis functions exist in the hidden layer of the network. The network size is increasing gradually with the number of input variables, according to the growing complexity of the training process. Furthermore, it can be learnt that the wind speed and wind direction much affect the accuracy of the constructed model. We expected that the temperature would influence the performance, but the results did not support that, as shown in the set number 4. This is probably due to a small variation of the recorded temperature level, hence affecting the convergence in only a minor way.

Table 8.2 Performance indexes on the test set for the simulation performed by different combination of inputs (at Randwick site).

Set	Inputs	RMSE	MAE	R^2	d_2	Network size
1	Time, NO, NO_2, 0_3	4.878	3.560	0.464	0.866	11
2	Time, NO, NO_2, 0_3, Temp	5.244	3.949	0.381	0.845	15
3	Time, NO, NO_2, 0_3, WSP, WDR	4.850	3.635	0.471	0.868	17
4	Time, NO, NO_2, 0_3, Temp, WSP, WDR	6.069	4.859	0.171	0.793	21
5	Time, NO, NO_2, 0_3, Temp, WSP, WDR, PM_{10}, SO_2	5.677	4.444	0.275	0.819	27

An example of approximation on the testing set from the best results is shown in Fig. 8.6. As we can observe from the figure, most of the values have revealed acceptable results, in which the predicted values using the metamodel follow the pattern of the expected BOL values derived from ambient measurements. As observed from the graph, the predicted values vary from 6.5 ppb to 47 ppb, demonstrating an increasing trend in the three months at the rate of about 0.002 ppb/hour using the linear fit line.

Case B: Blacktown station

Blacktown, which is an urban site located at the Sydney West region, was investigated. This station was also showing the highest average background ozone level in this region from the years 1998 to 2004 (Duc & Azzi, 2009). The same procedure explained previously for Randwick site was also applied to the Blacktown station. Five combination sets of input variables were used, with the addition of carbon monoxide as the pollutant agent. During the training process, we set the initial spread to be 0.1 and the error goal to be 0.006.

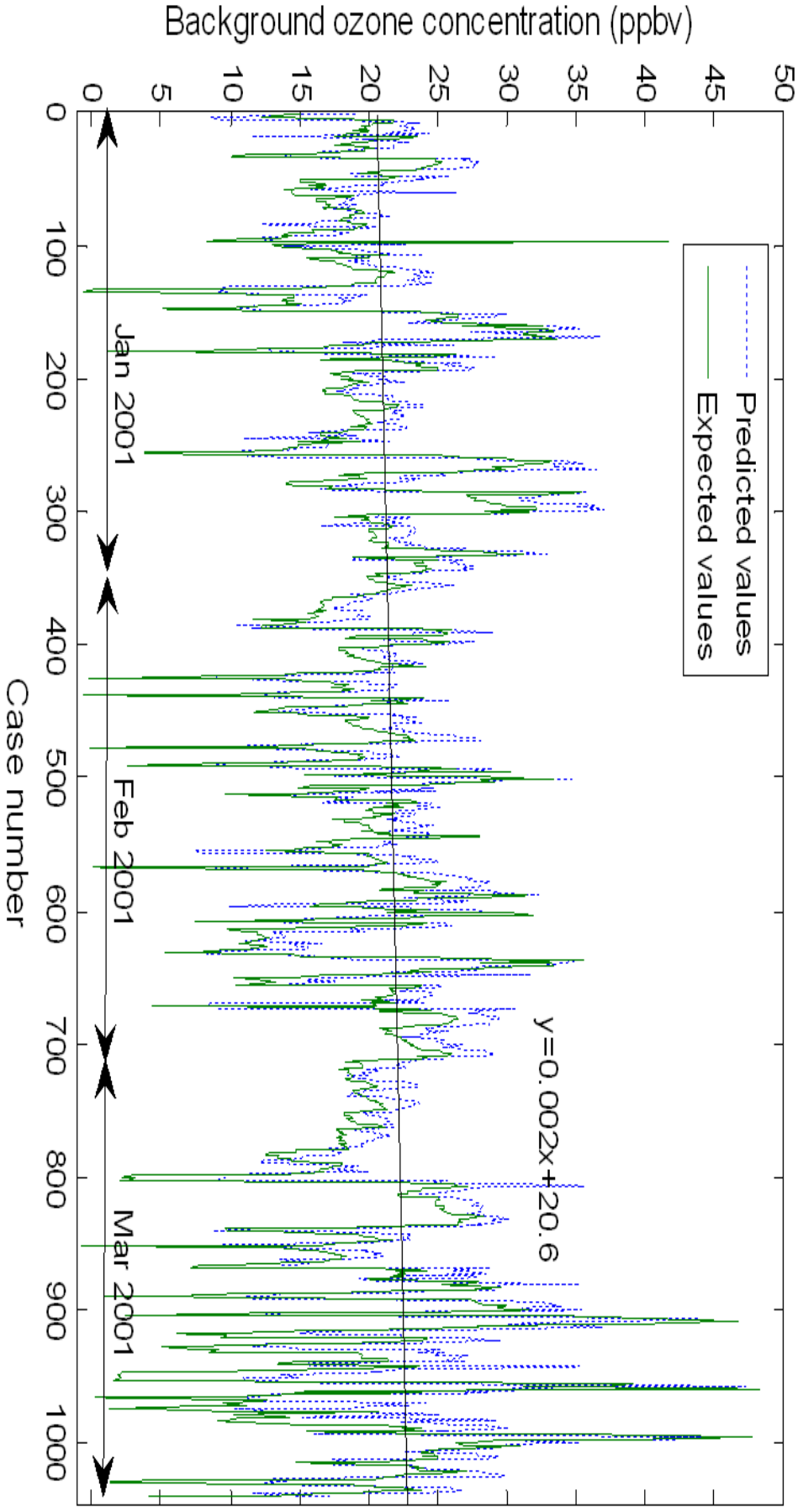


Fig. 8.6 Predicted background ozone level and the observed data at Randwick station over night-time and early morning hourly data.

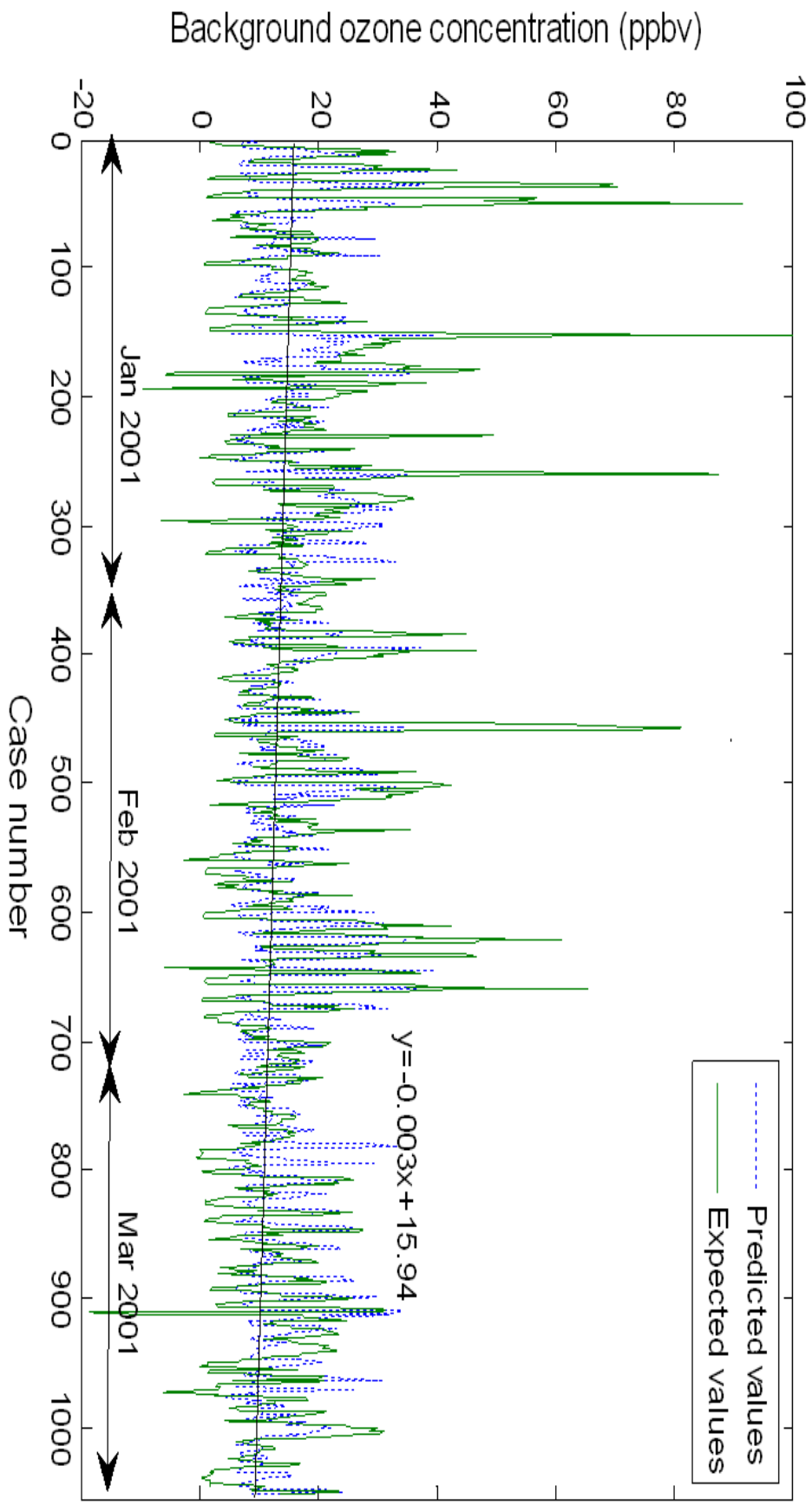


Fig. 8.7 Predicted background ozone level and the observed data at Blacktown station over night-time and early morning hourly data.

By referring to Table 8.3, the best combination with the highest accuracy is given by the inputs in the set number 4. Again, meteorological parameters are determined to be dominant in influencing the model performance. Unfortunately, the produced errors as derived by the *RMSE* and *MAE* are slightly higher than was generated by the Randwick model. Furthermore, the correlation given as a determination coefficient R^2 was also low. The deviation in performance may be due to only a small number of non-photochemical conditions that were obtained from the dataset while most of the background ozone level data has been determined by using the linear regression method. An example of prediction results is depicted in Fig. 8.7. The predicted background ozone level data was found to be between 4 ppb to a maximum of 40 ppb, with a decreasing trend of 0.003 ppb/hour in a three month period.

Table 8.3 Performance indices on the test set for the simulation performed by different combination of inputs (at Blacktown site).

Set	Inputs	RMSE	MAE	R^2	d_2	Network size
1	Time, NO, NO_2, 0_3	11.241	7.483	0.223	0.806	7
2	Time, NO, NO_2, 0_3, Temp	11.163	7.365	0.234	0.808	11
3	Time, NO, NO_2, 0_3, WSP, WDR	11.590	7.917	0.174	0.793	13
4	Time, NO, NO_2, 0_3, Temp, WSP, WDR	11.025	7.479	0.252	0.813	22
5	Time, NO, NO_2, 0_3, Temp, WSP, WDR, PM_{10}, SO_2, CO	11.727	7.876	0.154	0.789	32

8.3.3. Extended model for long-term estimation of BOL

8.3.3.1. Model input-output

For the determination of BOL, several combinations of the input variables have been investigated in Wahid et al. (2010b) as discussed in the previous section, which showed that time information, pollutant precursor data, ozone data and meteorological data have a major influence on the model performance. However, each site was separately treated with a different model for the network training and validation. In this work, we will construct a generic model that covers several sites in the region, which may be used to interpolate and extrapolate the values for other sites not appearing in the training data set. To achieve such a model, we have to develop an appropriate training technique that can be mixed between the sites.

For the training dataset, we use the following variables as the inputs: the x and y coordinate locations as the site's identifier, the NO concentration, the NO_2 concentration, the O_3 concentration, the ambient temperature (TEMP), the wind direction (WDR), and the wind speed (WSP). To generalise the solution, the hour information will be excluded from the inputs because a further analysis has shown that the model performance could be slightly improved without it in the new model structure. The suggestion of the new model structure and its analysis appeared in Wahid et al. (2011).

The considered input datasets are the hourly data at the specified time period of interest for the BOL determination which has been described in section 7.4.3. Furthermore, the BOL defined in section 7.4.1 is used as the target output of the network. As usual, all input-output variables are normalised using their minimum and maximum values so that they are in the same range of the radial basis function used. After the learning process by RBFNN (i.e., an OLS method explained in Chapter 4) is finished and the accuracy obtained with some test sets is satisfactory, the network can be used for predictive purposes with other available data for the following years. The proposed model input-output is depicted in Fig. 8.8, and the codes for building the model appear in Appendix E-2.

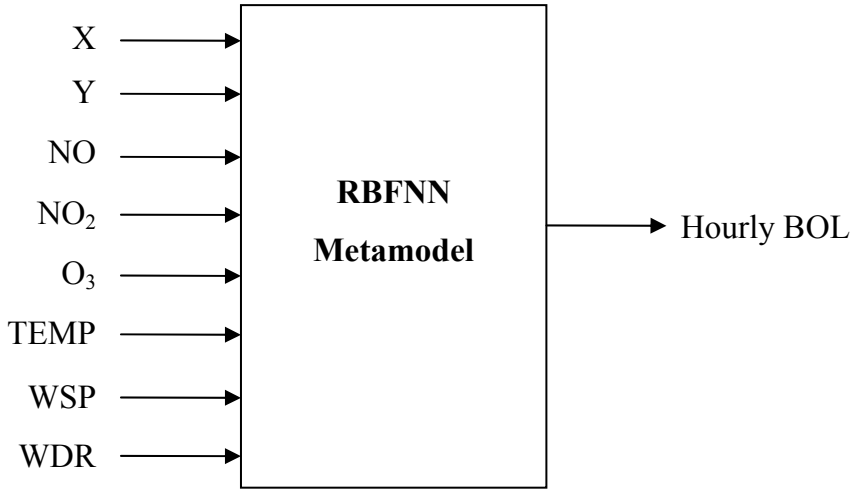


Fig. 8.8 A generic metamodel structure to estimate the BOL in the evaluated domain.

8.3.3.2. Results and analysis

For the BOL model development, the hourly data of the pollutant concentrations and meteorological data which have been carefully recorded at 10 monitoring stations in the Sydney region are utilised. The data covers six years from 2005 to 2010 in which summer is chosen as the season of interest. For the Sydney region, the hourly data from 8.00 pm to 8.00 am the next morning is used for the determination of background ozone level.

Two groups of data were prepared, which were for training and testing purposes. For the training data set, the input and target data for the summer of 2005 was adopted from every site, whereas the remaining summer data from 2006 to 2010 were used as the testing data sets. The training dataset consisted of 6327 data points and by using a proposed experimental design method presented in Chapter 3 (i.e. Wahid et al., 2012), the dataset was sampled to 35% of the full data set. By using RBFNN with an initial setting of 0.6 for the spread parameter and 0.002 for the MSE goal, the model is constructed with 55 hidden neurons.

Fig. 8.9 and Fig. 8.10 show two examples of the reliability of the model to estimate the BOL, with pleasing results. Therein, the solid lines show the expected BOL derived by the measured data, and the broken lines represent the estimated values using the RBFNN metamodel. Fig. 8.9 illustrates the estimation results for St Marys, a site located in the west of the Sydney region. Therein is revealed a decreasing trend with a decreasing rate in the BOL of 3.1±0.2 ppb (standard error is 0.23 ppb) over the 2006 to 2010 period (or 0.62±0.04 ppb per year) and an intercept of 11.0±0.2 ppb in the linear regression trend line. The enlarged figures for the two periods show the capability of the metamodel to estimate the true values of the BOL at most of the points, with the R^2 value of 0.577 which presents an improved model performance from the previous attempt. By using the same constructed model, another site namely Bringelly was also evaluated, as depicted in Fig. 8.10. A similar decreasing trend of the regression line is also shown for this site.

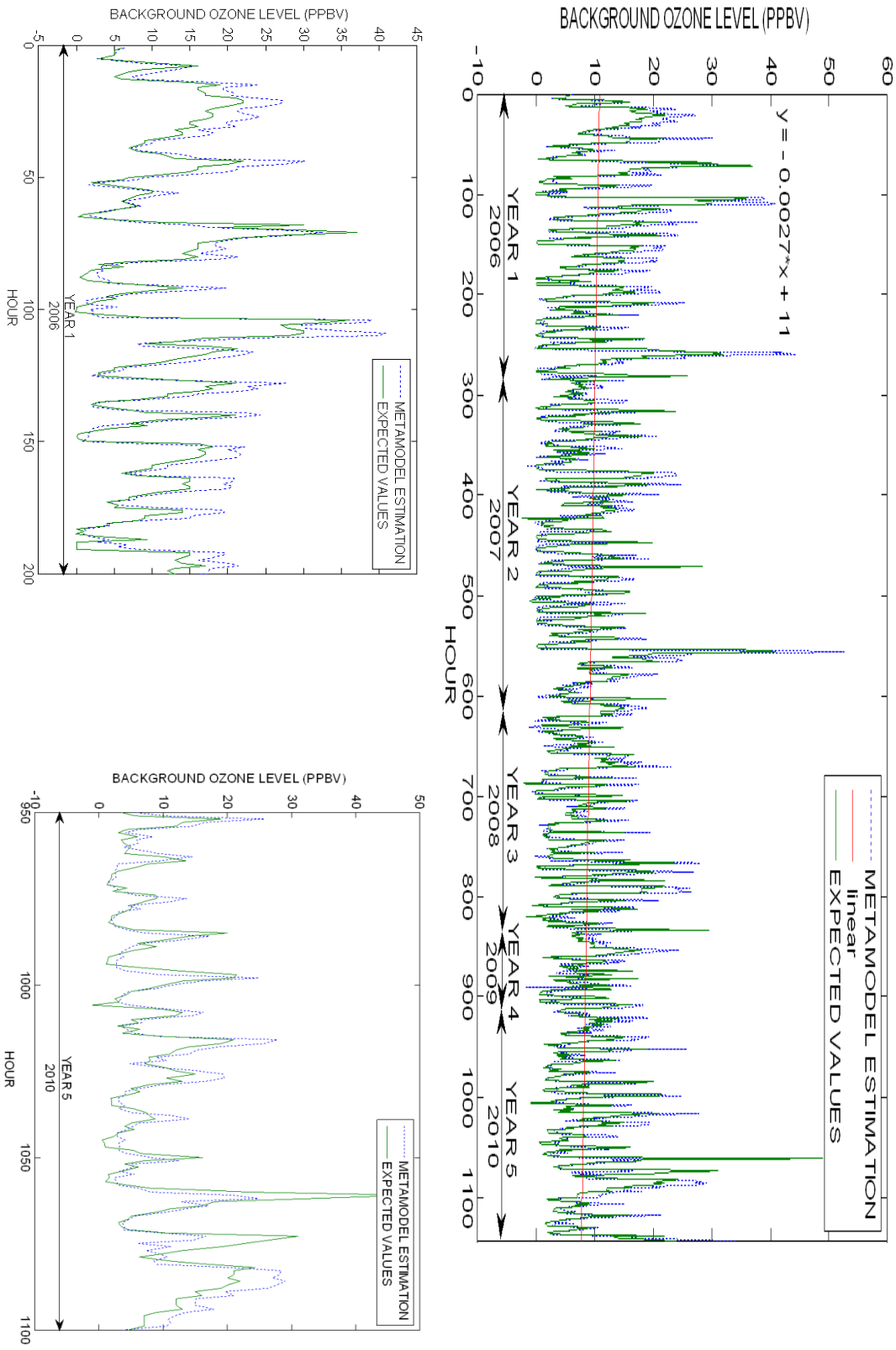


Fig. 8.9 Comparison between night-time BOL estimated by metamodel, and expected values derived by the measured data at St Marys site.

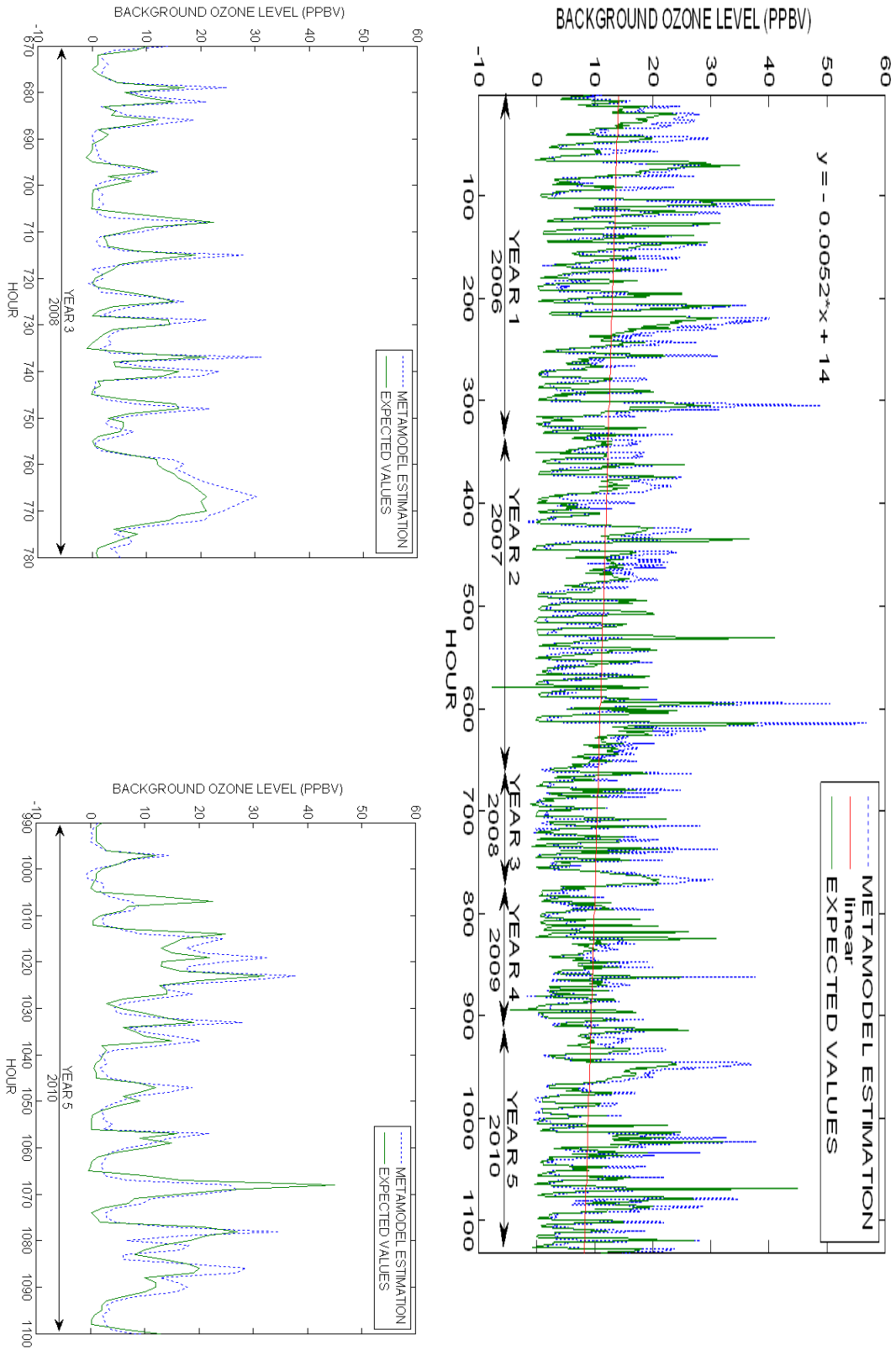


Fig. 8.10 Comparison between night-time BOL estimated by metamodel, and expected values derived by the measured data at Bringelly site.

The yearly BOL mean values for the period of six years from 2005 to 2010 at all monitoring sites in Sydney are summarised in Table 8.4. Most of these sites are considered as urban except for Richmond, which is located in a semi-rural area. The table shows decreasing trends at most of the sites from 2005 to 2008, but their levels rise again from 2008 to 2010. A clearer illustration of the trends is shown in Fig. 8.11. It was expected that the increasing trend would continue for the following years (after 2010) but with a lower increasing rate, to be confirmed with the availability of the measurement data.

Table 8.4 Yearly night-time BOL mean values at various sites around Sydney from 2005 to 2010.

	2005	**2006**	**2007**	**2008**	**2009**	**2010**
Bringelly	12.1	11.5	9.6	6.3	7.2	9.6
Chullora	13.1	11.2	8.4	7.5	8.3	7.9
Earlwood	14.1	13.5	8.8	8.8	10.0	7.7
Liverpool	12.2	11.5	8.2	8.1	8.0	6.8
Oakdale	21.0	19.0	15.3	13.5	16.0	19.9
Randwick	18.3	17.9	14.8	14.8	12.4	17.7
Richmond	14.1	13.2	9.5	8.5	12.7	7.0
Rozelle	15.2	13.2	10.1	8.9	9.2	14.3
St Marys	12.7	10.4	8.0	6.4	8.0	8.9
Vineyard	15.2	10.4	8.5	9.4	9.6	8.3
Mean	**14.8**	**13.18**	**10.12**	**9.22**	**10.14**	**10.81**

Using a regression analysis, Duc et al. (2012) reported that there was an increasing trend of the background ozone level in Sydney from 1998 to 2005. In another evaluation by Wahid et al. (2011) using metamodel estimation, an upward tendency was found at each site in Sydney from 2003 to 2005 which was also in line with the results reported in (Duc et al., 2012). In the report, it was also shown that the patterns start to decrease after 2005 until 2008 which is similar to the analysis shown here. The only difference is that the BOL defined in Wahid et al. (2011) used zero nitrogen oxides (i.e. NO_x=0), while this work uses zero nitric oxides (i.e. NO=0) in its determination, which will result in about 3 ppb difference in the background ozone levels (i.e. the later method gives slightly lower values).

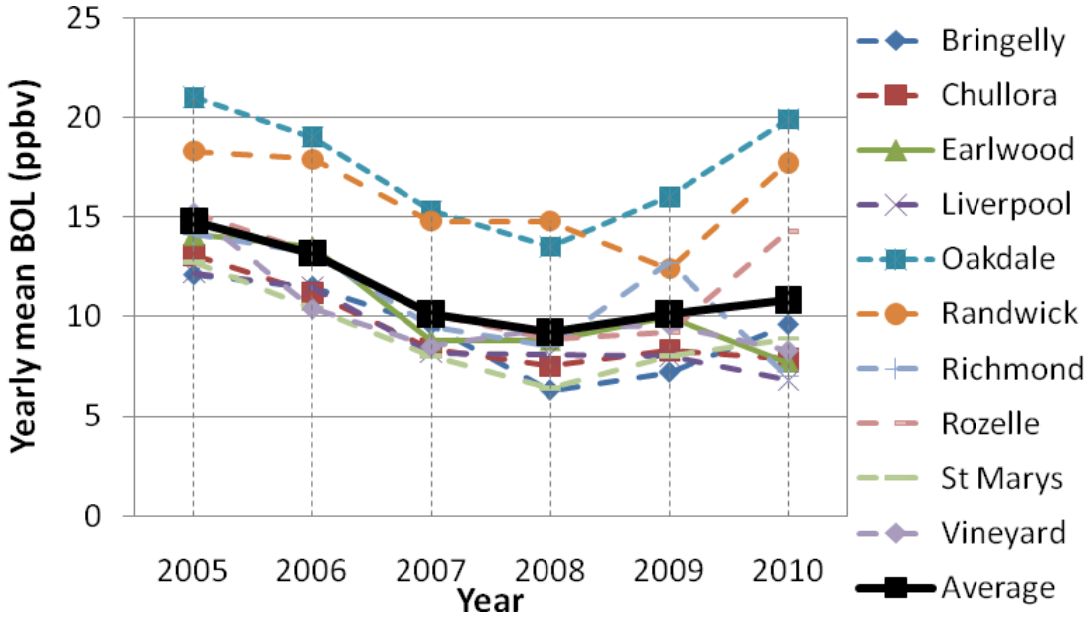


Fig. 8.11 BOL profiles in the Sydney basin over a six year period.

To further validate the constructed metamodel, two monitoring sites' data that were not used in the training data set were used to estimate the BOL in which the results are depicted in Fig. 8.12. Interestingly, the model is still capable of roughly following the pattern of the expected true values defined by the measured data. Hence, the developed model may be used for quick scanning for the spatial estimation of BOL of the whole domain, if the inputs' data can be made available, for example, from the simulation outputs of the deterministic air quality model.

8.3.4. Discussion

A metamodel using a radial basis function neural network has been presented for estimating the background ozone level in several monitoring stations in the Sydney area. In the first attempt, several combinations of the input variables have been used to evaluate their influence on the accuracy of the constructed model. It has been observed that the photochemical data with the existing meteorological data, especially the wind speed and wind direction, dominantly affect the model's performance. It is noted that the model performance does not depend much on the existence of other pollutant agents such as SO_2, CO and PM_{10}. Their influence is probably not crucial for the background ozone prediction, hence their existence in the training data may cause some reduction in the model accuracy.

The second approach proposes a more generic regional model for estimating the BOL using an RBFNN metamodel, which can be used for each monitoring site in the domain under consideration. Coordinate location of the sites, nitrogen oxides and meteorological data are used as the input, while night-time background ozone was used as the target output. The results obtained indicate the promising application of the proposed method in predicting the long-term BOL with fair accuracy.

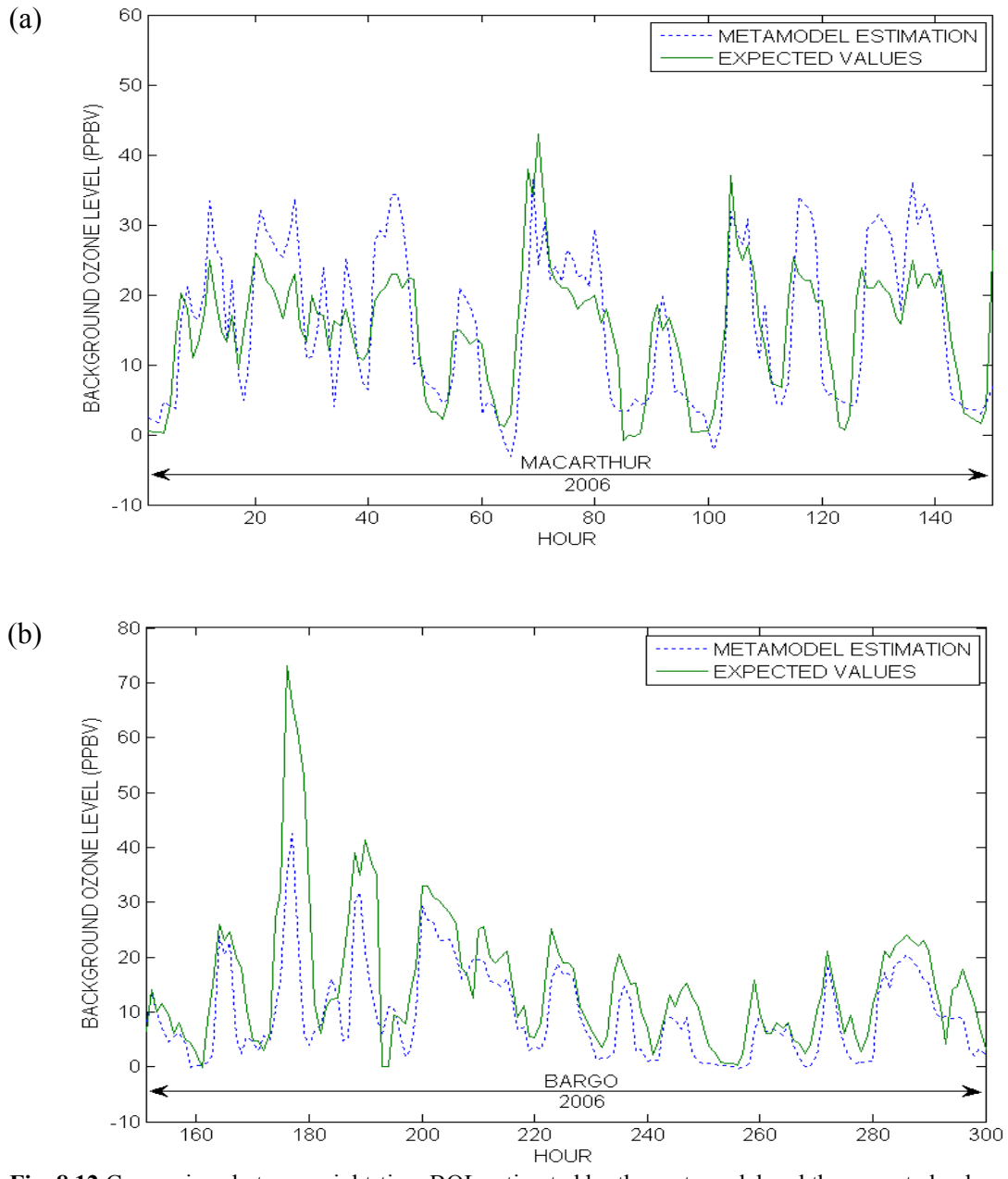


Fig. 8.12 Comparison between night-time BOL estimated by the metamodel and the expected values derived by the measured data at two sites that were not used in the training data set, namely Macarthur and Bargo.

8.4. A metamodel approach for air pollutant spatial estimation

Continuous measurements of the air pollutant concentrations at monitoring stations serve as a reliable basis for formulating air quality regulations. Their availability are however limited only to the location of interest. In most situations, the spatial distribution beyond these locations still remains uncertain as it is highly influenced by other factors such as emission sources, meteorological effects, dispersion conditions and topography. To overcome this issue, a larger number of monitoring stations could be installed, but it would involve a high investment cost. An alternative solution is via the use of a deterministic air quality model, which is mostly adopted by regulatory authorities for prediction in the temporal and spatial domain as well as for policy scenario development. Nevertheless, the results obtained from a model are subject to some uncertainties and they generally require significant computation time.

In this work, a meta-modelling approach based on neural network evaluation is proposed to improve the estimate of the spatial distribution of the pollutant concentrations. From a dispersion model, it is suggested that the spatially-distributed pollutant levels (i.e. ozone, in this study) across a region under consideration is a function of the grid coordinates, topographical information, solar radiation and the pollutant's precursor emission. Initially, for training the model, the input-output relationship is extracted from a photochemical dispersion model called The Air Pollution Model and Chemical Transport Model (TAPM–CTM). The proposed metamodel is then applied to estimate the ozone concentrations in the Sydney basin, Australia. Once executed, apart from the advantage of inexpensive computation, it provides more reliable results of the estimation and offers better predictions of ozone concentrations than those by using the TAPM–CTM model alone when compared to the measurement data collected at monitoring stations.

8.4.1. Model development for ozone distribution

8.4.1.1. Input-output parameters of the model

The neural network model could be considered a black box for mapping the best relationship between the inputs and the outputs of the dataset without knowing the

underlying physics of the system. In this work, an improved RBFNN is proposed for the modelling where suitable input parameters should be selected to get the best possible network configuration. To this end, we utilise specific ambient measurement data and also input-output data from the deterministic air quality model (DAQM), to train the RBFNN. In this work, we adopt a specialised DAQM model known as the TAPM–CTM.

Since ozone is the pollutant to be considered in this work, the most related input parameters for training the model are the ozone precursors, the *x*-*y* coordinates, the topography information and the solar radiation levels. Basically, there are two important classes of precursors involved in the formation of ozone, namely volatile organic compounds (VOCs) and NO_x. However, VOCs are apparently very difficult to measure, hence the VOC data is fully based on the emission rate data extracted from the emission inventory system, whereby the NO_x data could be enhanced by incorporating its measurement data collected at the monitoring stations.

The *x*-*y* coordinates represent the cell locations (in km) in *x* and *y* directions, which normally form a group of 2 km×2 km domain cells. By using statistical modelling, the coordinate information is adequate for quick interpolation of measurements between the monitoring stations, but it is not quite accurate, especially for a large distance between sites. To improve the estimation, topography information is added, consisting of the height information above sea level (in metres) at each domain cell.

Here, ambient temperature data is used to represent, at each cell, the solar radiation level, which basically is a good variable indicator proxy for the formation of ozone and has a strong correlation to the ozone concentration. Generally, a temperature dataset could be made available from a local meteorological institution such as the Bureau of Meteorology for the Sydney region. The lowest layer data (about 20 m above the sea level) will be considered. These datasets need to be post-processed as daily maximum temperatures, taken from the daylight hourly temperature, as to represent the activeness of the daily ozone production.

The network output consists of daily averaged eight-hour maximums of the ozone concentration (in parts per billion, *ppb*), which are extracted from the DAQM

simulation output. The eight-hour average is selected here in this work as a demonstration of the approach. The four-hour or one-hour data can be analysed similarly. As for the ozone predictions, the simulation is only run for the months of the summer season (e.g. December, January and February, in Australia), during which the formation of ozone is most intense. To correlate with the actual measurement data, this dataset will be calibrated using regression by analysing the correlation ratio between DAQM output and actual concentration data at all available monitoring sites, for each recorded day. This correlation ratio will then be multiplied for the entire cell parameters in the simulated domain. For illustration, the topology of the model network is shown in Fig. 8.13.

8.4.1.2. NO_x emission distribution

Generally, the amount of the daily NO_x emission (in kg/day) taken from the emission inventory does not change much for each day, except there is a small difference between the weekdays and the weekend days. Thus, the daily emission can be assumed to be identical over time at one location, however, they are apparently different between each domain cell. To make the significant variations of daily emissions for the purpose of neural network training, the actual measured NO_x concentration at monitoring stations (typically in pphm) will be converted to an emission rate, distributed to the entire domain and added to the original emission data. This can be done by assuming the emission source is at ground level and thus, the produced concentration is contaminated at the ground level and using the basic Gaussian dispersion model developed by Pasquill (1976) as appeared in equation (2.1) in Chapter 2, which can be simplified as follows:

$$C(x,y,z) = \frac{Q}{2\pi\sigma_y\sigma_z u}, \tag{8.1}$$

where C is the pollutant's concentration (in μg/m^3) at distance x downwind from the source (in meters), Q is the emission rate (in g/sec), u is the average wind speed (in m/sec), σ_y and σ_z are the dispersion coefficient respectively in y- and z-direction. Here, the plume is always assumed to follow the wind direction in x-direction, thus the distance y in crosswind direction and the distance z in vertical direction can be approximated as zero.

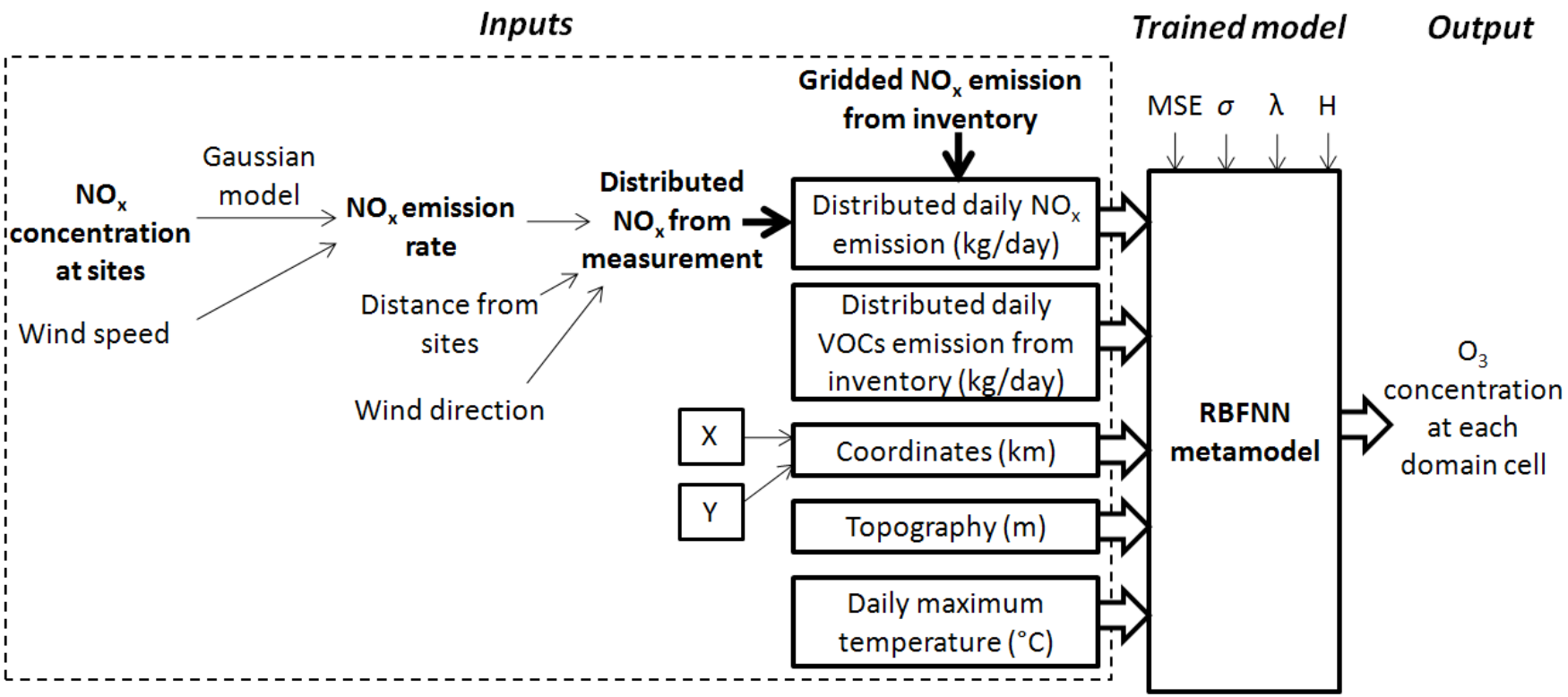


Fig. 8.13 Inputs and output for training the RBFNN model for spatial estimation of ozone concentration.

As described in the literature review in section 2.2 (Chapter 2), the values of σ_y and σ_z can be determined roughly from the dispersion coefficient graphs, or more accurately determined by the following equations (Cooper & Alley, 2011):

$$\sigma_y = ax^{0.894}, \text{ and} \tag{8.2}$$

$$\sigma_z = cx^d - f, \tag{8.3}$$

where values of a, c, d and f can be obtained by curve-fitting, depending on the atmosphere's stability condition in which the values are summarised in Table 8.5. Note that the measurement unit, in pphm, for pollutant concentration is consistently converted to μg/m^3 using the molecular mass of NO and NO_2 at 25°C and 1 atm.

Table 8.5 The coefficient values for calculating the σ_y and σ_y.

Stability	**Sigma-*y***	**Sigma-*z***					
		if *x*<1 km			**if *x*>1 km**		
	a	***c***	***d***	***f***	***c***	***d***	***f***
A	213	440.8	1.041	9.27	459.7	2.094	-9.6
B	156	106.6	1.149	3.3	108.2	1.098	2.0
C	104	61.0	0.911	0	61.0	0.911	0
D	68	33.2	0.725	-1.7	44.5	0.516	-13.0
E	50.5	22.8	0.678	-1.3	55.4	0.305	-34.0
F	34	14.35	0.740	-0.35	62.6	0.180	-48.6

The calculated emission rate at the stations will then be assumed to be coarsely distributed to other cells. The emission rates at these cells are estimated by considering the nearest distance to the station, adjusted by the wind direction factor. Finally, the calculated distributed NO_x emission will be added to the gridded emission rate from the inventory database.

8.4.1.3. Training and verification of the model

To start the modelling process, first we need to define the frame area for the simulation. The border of the domain is selected about 30 km distant from the outer-most monitoring stations for a reasonable correlation. For the network training purpose, the entire domain will be divided to groups of 6 km×6 km grid cells for the input dataset from these groups to be able to represent the behaviour of the whole frame. This choice reduces the number of datasets to be trained. The dataset will be

trained by using an RBFNN with the appropriate selection of spread parameter (σ), least squares weighting matrix (H), regularisation parameter (λ) and prescribed error goal, i.e. mean squared error (MSE).

In the validation stage, the denser input-output dataset (i.e. smaller cell size, for e.g. 2 km×2 km) from the same simulation is used to confirm the correctness of the trained model. The developed model will be tested with other datasets which have not been used in the training stage to predict the spatial distribution of ozone concentration, and the results are compared with the measured ozone level collected at the continuous monitoring sites.

8.4.2. Study case: results and analysis

8.4.2.1. The application domain

The methodology has been applied to the Sydney basin in New South Wales, Australia. The basin currently has 14 monitoring stations scattered throughout the Sydney metropolitan region, from the coastal area in the East to the edge of the Blue Mountain in the North West and in the West. Most of the measuring sites are located in the urban area except for some locations, which can be considered as suburban in the greater West, and a semi-rural area in the North West.

The whole Sydney basin covers an area of about 24,242 km^2. For the station location, in order to obtain reasonable prediction results using the proposed methodology, the selected domain begins from 246 km to 384 km easting and from 6207 km to 6305 km northing, by using the Australian map grid (AMG) coordinates.

8.4.2.2. Neural network model implementation

The model development is based on the ambient measurement of pollutant data, meteorological data and primary or precursor pollutant emission sources data for the year 2004, considered as the base year for this study. For preparing the output dataset, a few simulations for the summer days in 2004 are performed by using the TAPM–CTM model. As the regulatory agency, the NSW Office of Environment and Heritage (formerly known as Department of Environment and Climate Change) is

mostly interested in the prediction of peak ozone scenarios, for which only episode days are chosen for the simulation in this study. The spatial distributions of eight-hour maximum average of the ozone level are extracted from those simulations for the smallest grid cell (i.e. 2 km×2 km).

It is noted that there are some differences in the ozone level as predicted by the TAPM–CTM model, compared to the actual measurement data at the monitoring stations. Most of the TAPM–CTM predicted outputs are under-predicted, especially during the episode days. Moreover, their correlation is usually nonlinear and different for each day. To correct the under-prediction and improve the correlation between the model output and the measurement data, the modelled ozone datasets need to be calibrated, e.g. by using the regression analysis via comparison of the actual and the simulated data at all the monitoring stations to determine the correlation ratio between them. For example, Fig. 8.14 shows a correlation of daily eight-hour maximum average of ozone for a day in summer. A regression line is drawn by setting the intercept point at zero. Therein, the correlation ratio is determined as 1.326, i.e. all the daily ozone distribution data from TAPM–CTM output are multiplied with this ratio. This is assuming that the spatial distributions of the pollutant are in general predicted correctly enough by the deterministic model, but need to be further compensated due to the under-predict or over-predict situations. The aim here is to form a dataset that is close to the actual data for the whole domain, based on the available correlation ratio at all monitoring stations, i.e. by a regression technique.

For the NO_x input dataset, the measured concentration data for the same days as the TAPM–CTM simulations will be utilised to compute the variation of the NO_x emission rate. The hourly NO and NO_2 concentration for each day will be converted to the emission rate according to their molecular mass values and average wind speeds. The downwind distance is estimated accordingly to cover the 2 km×2 km grid cells, and the other coefficients are set based on the environment stability conditions using the Pasquill Table (i.e. Table 2.4, in Chapter 2). The calculated hourly emission rate will be summed to obtain the daily emission rate of NO_x at every monitoring station. The emission values for other cells in the domain will be approximated in accordance with the nearest distance to the station at which the

wind direction and the cell-station direction make the smallest angle. Within a certain radius from the stations, pollutant concentrations are assumed to be similar and hence the same emission rate level is expected. On the other hand, the gridded inventory emission rate data for NO_x is extracted from the TAPM–CTM pre-processing outputs. Finally, both types of emission (i.e. inventory and calculated) for each cell are added to form distributed daily NO_x emissions (in kg/day).

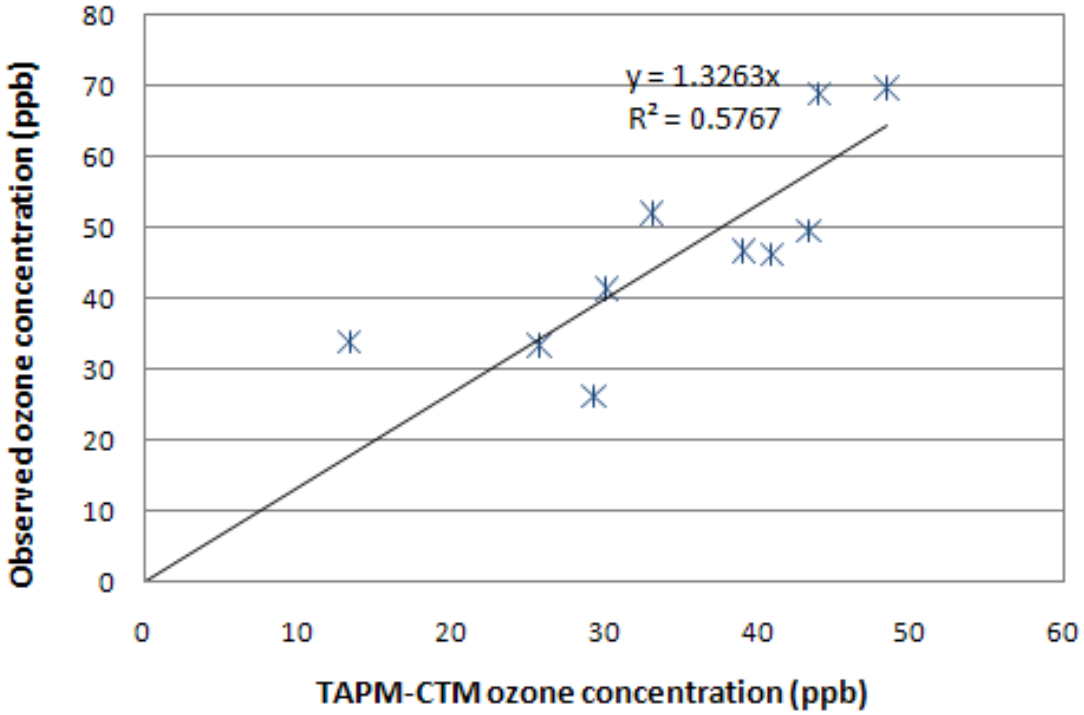


Fig. 8.14 Regression analysis for determination of the correlation ratio between simulated and observed ozone level at a monitoring site.

Fig. 8.15 shows a comparison of the daily distribution before and after the summation for a summer day in 2004, where the daily emission is concentrated mostly in the Sydney metropolitan area. Obviously, this area has a high population concentration and also dense road networks, as well as a large number of industrial activities. The high emissions also appeared along the roadways from North to South, and to the West. Fig. 8.15 (b) shows that the emissions are more scattered in the domain, while it is not distributed well in the East area because there is no measurement data in that area (i.e. the Tasman Sea).

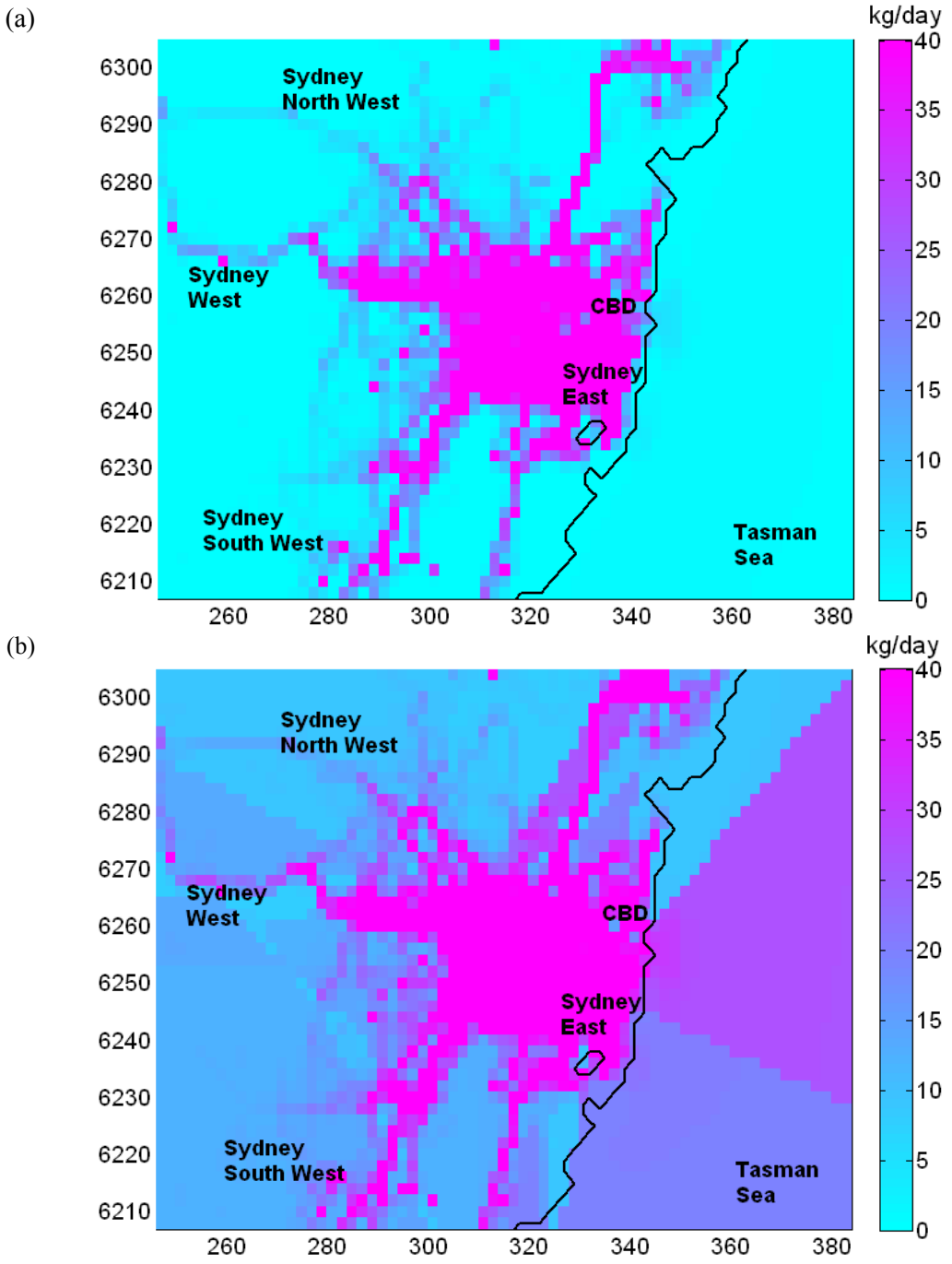


Fig. 8.15 Daily NOx distribution for a day in summer: (a) post-process by TAPM-CTM from the emission inventory, (b) added with the calculated emission.

The rest of the input dataset (i.e. coordinate, height from sea level and temperature) could also be extracted from the TAPM–CTM model which uses synoptic data collected by the Australian Bureau of Meteorology. Using the proposed algorithms for training the RBFNN metamodel as was discussed in Chapter 5, the optimal parameters for the model have been determined as: the spread parameter $\sigma = 0.1$, the regularisation parameter $\lambda = 1.0$, and the least squares weighting constant

$h_{ii} = 34.03$. By setting the MSE goal to 0.005, the model network was created by having a total of 168 hidden neurons from 2448 (i.e. n) training samples, in just about six minutes of the simulation time. Notably, with the TAPM–CTM simulation, it will require about one whole day to complete a similar task.

8.4.2.3. Model performance

To validate the trained model, denser datasets (from the same simulation days in the training stage), which involve 21000 data patterns consisting of data collected from January to February 2004, are used. The performance of the validation phase is shown in the scatter plot of Fig. 8.16. It consists of 3500 data points, corresponding to 3500 cells, each 2 km×2 km over the whole domain (i.e. 70 cells to the East × 50 cells to the North). The plot represents a correlation between the prediction results by using the constructed RBFNN model against the target outputs in the dataset. As depicted, most of the scatter points are located close to the bisecting line for every data point with the determination coefficient R^2 of 0.96, which can be considered an acceptable performance.

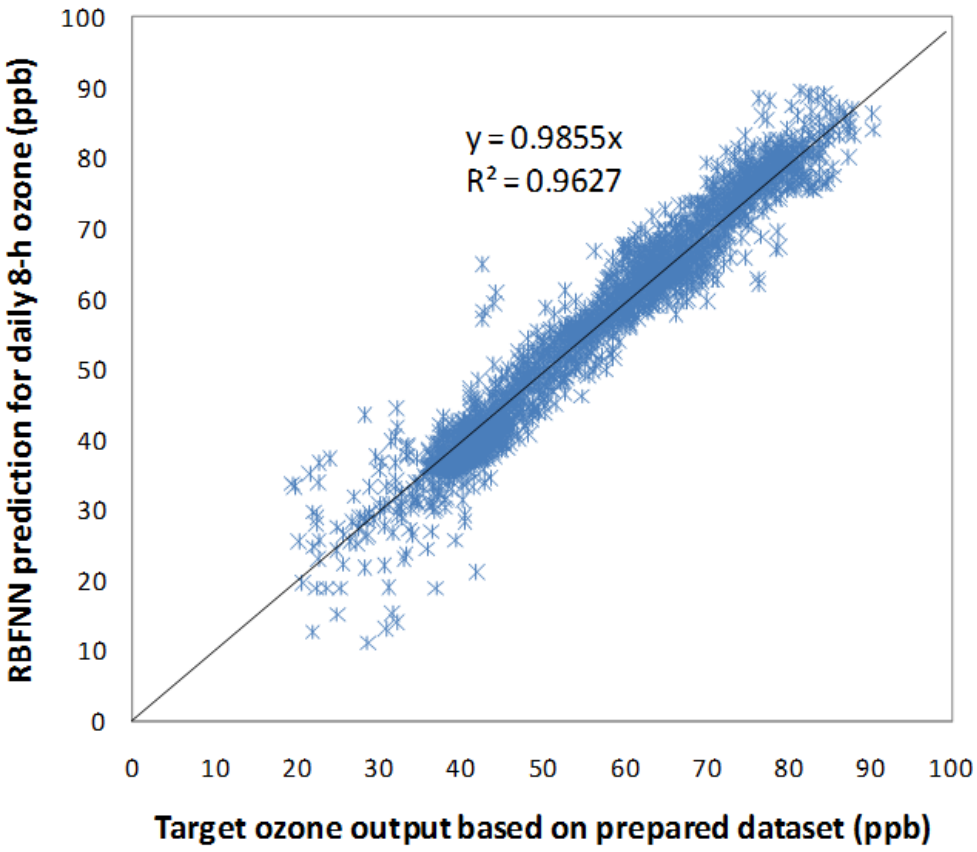


Fig. 8.16 Scatter plot to illustrate the performance of validation phase.

The spatial distribution, obtained by using the RBFNN and TAPM–CTM models, is shown in Fig. 8.17. The results of two episode days are presented, wherein both models generate similar patterns of the spatial lines but with different ranges of concentrations. For the first day, the higher levels are concentrated in the West area from North to South with a range from 19 to 90 ppb for the RBFNN, and from 8 to 62 ppb for the TAPM model. On the second day, the high concentration is scattered about the whole domain in which the peak levels appear mostly in West towards the South West area. However, the RBFNN output gives a maximum level of 113 ppb while the maximum level by TAPM is only 72 ppb, which exhibits an under-prediction. This uncertainty is confirmed by comparing those levels with actual data collected at the monitoring points.

From these spatial distribution results, it can be observed that most of the high ozone levels always appeared, especially during the episode days, in the West of Sydney, which consists of suburban and semi-rural areas. This is the general pattern of ozone occurrence in the Sydney basin which is consistent with the meteorological condition of the West and South West being downwind of the sea breeze during the day. In the morning after sunlight, an off-shore sea breeze flows from the East and North East across Sydney towards the South West causing an elevated level of ozone in the South West and West of Sydney during the afternoon.

However, the most important issue is the number of excessive observations recorded (i.e. more than 80 ppb for an eight-hour maximum average standard), which could have an adverse impact on human health as well as on the vegetation. This situation rises due to the increase of the ozone level caused by the accumulation of ozone formed previously in the East of Sydney, which is the transported to the West and South West areas.

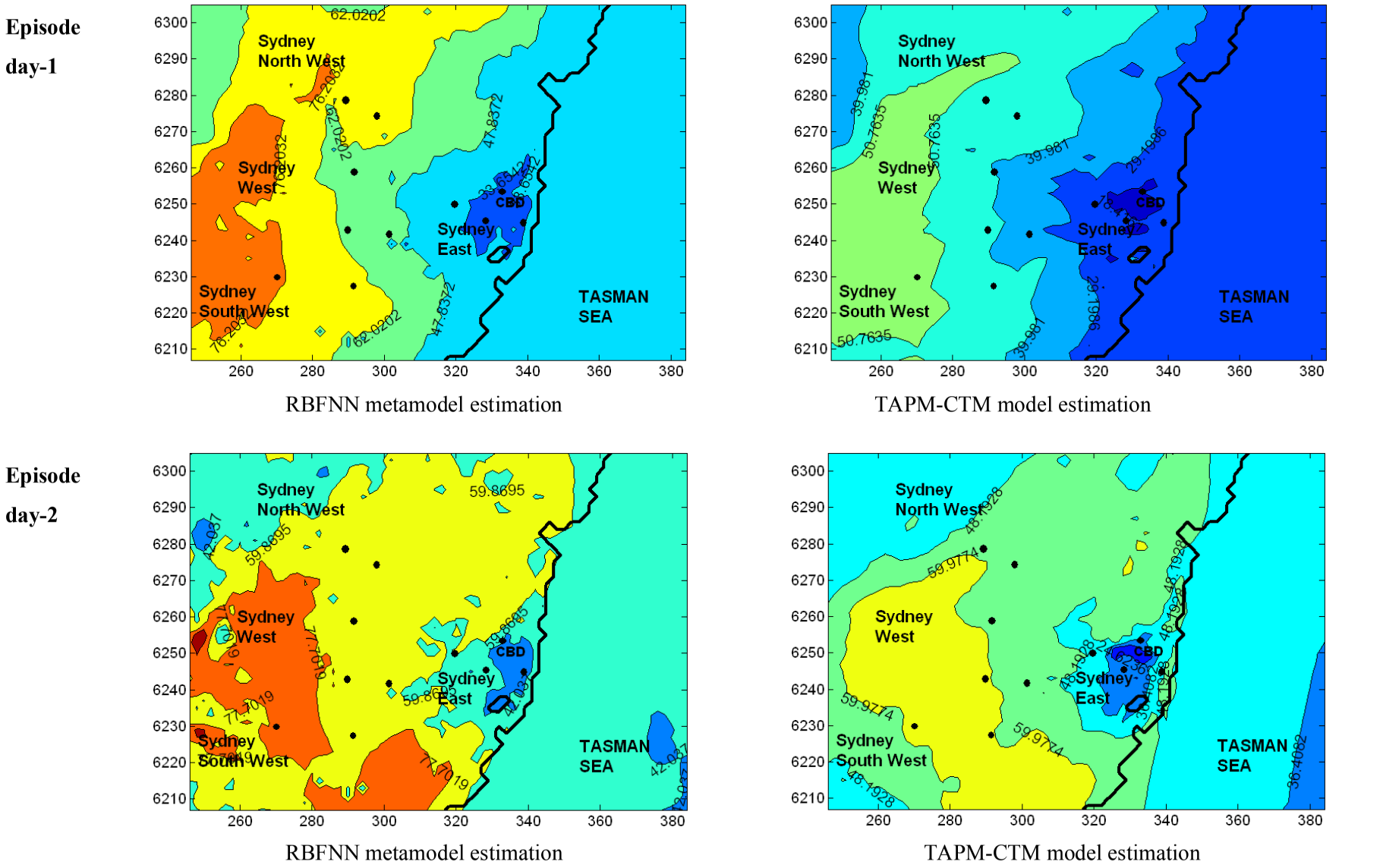


Fig. 8.17 Spatial distribution for 8-hour maximum average of ozone by using RBFNN and TAPM-CTM model (Note: the bullet dots show the location of the monitoring stations).

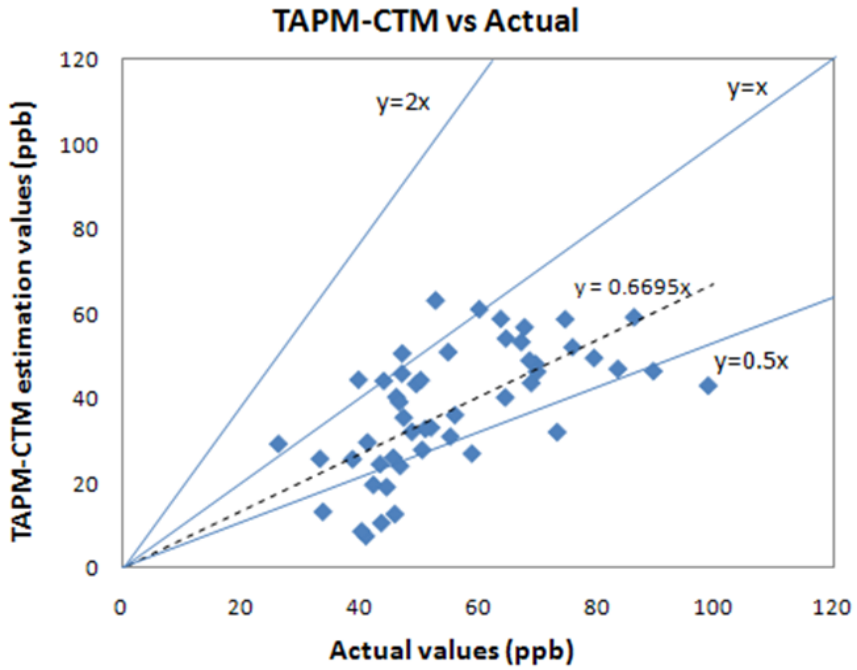
8.4.2.4. Performance comparison

To assess the reliability of the models, five days of simulation results of the spatial distribution are compared with the actual measurement data at 10 monitoring stations for each day. Fig. 8.18 shows the scatter plots of the models versus the actual data, whereby each plot consists of 50 data points (i.e. 5 days × 10 monitoring stations). Five episode ozone days in a summer season are selected in the analysis. As can be seen from the first figure, most data points are located close to the bisecting lines, and all lie in between the upper-half section line and lower-half section line. This is an improvement as compared to the TAPM estimations in which most of the TAPM values show an under-prediction of results, as presented in Fig. 8.18(a).

In terms of R^2 values, the RBFNN yields 0.7658 while the TAPM yields 0.3521, which can be claimed as another advantage of the proposed approach. However, this indicative value shows that further improvement in the approach needs to be carried out, as there are some estimation points that do not achieve the actual measurement value. This is probably due to the preparation of the output dataset (for training the model) which is much dependent on the regression analysis to correlate with the actual measurement data, and on other uncertainty arising from the TAPM–CTM simulation outputs.

In another analysis, the performance of the proposed RBFNN's training scheme (i.e. a generalisation network with regularised forward selection and weighted least squares, GRFSWLS), as proposed in Chapter 5, is compared with two available RBFNN algorithms, i.e. the orthogonal least square (OLS) by Chen et al. (1991), and the forward selection (FS) by Orr (1996). By using a common σ value of 0.1 which has been determined as the best isotropic spread parameter, the comparison of the training evolution for different mean squared error (*MSE*) goals is illustrated in Fig. 8.19. It is shown that the proposed algorithm outperforms the other two approaches at almost every error goal, in terms of the number of hidden neurons used and the total simulation times.

(a)



(b)

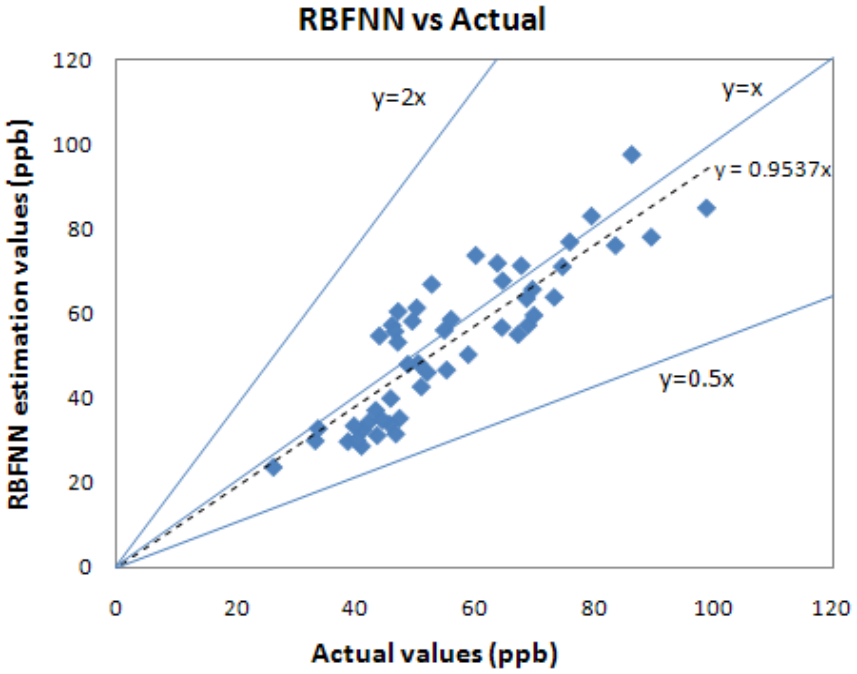


Fig. 8.18 Performance comparison between RBFNN and TAPM–CTM predictions for eight-hour maximum average of ozone at 10 sites in the Sydney basin.

Table 8.6 shows the comparison of the models' performances in terms of the determination coefficient (R^2) index, which is used to determine the accuracy of each method for the addressed problem. From the table, it is found that for each error threshold, smaller network sizes are used and higher R^2 values are always obtained by GRFSWLS as compared to the rest of the methods. It has been determined that the lowest possible error goal is 0.005, as the network tends to be over-fitted when

lower values are selected. It is also learnt that the optimal value of the regularisation parameter λ, can be selected more easily by incorporating the least squares weighting matrix H, as the FS method is required to fine tune the λ (to several decimal places, e.g. 0.0001) in order to obtain its optimal value, hence will result in a slower computation (e.g. 1000 times slower). This situation can be illustrated in Fig. 8.20, where the λ value is found as 0.1 when using one decimal place, whereas the optimal value can only be obtained using four decimal places which is found as 0.0299. It implies that the proposed method has a higher sensitivity in the solution to the regularisation problem.

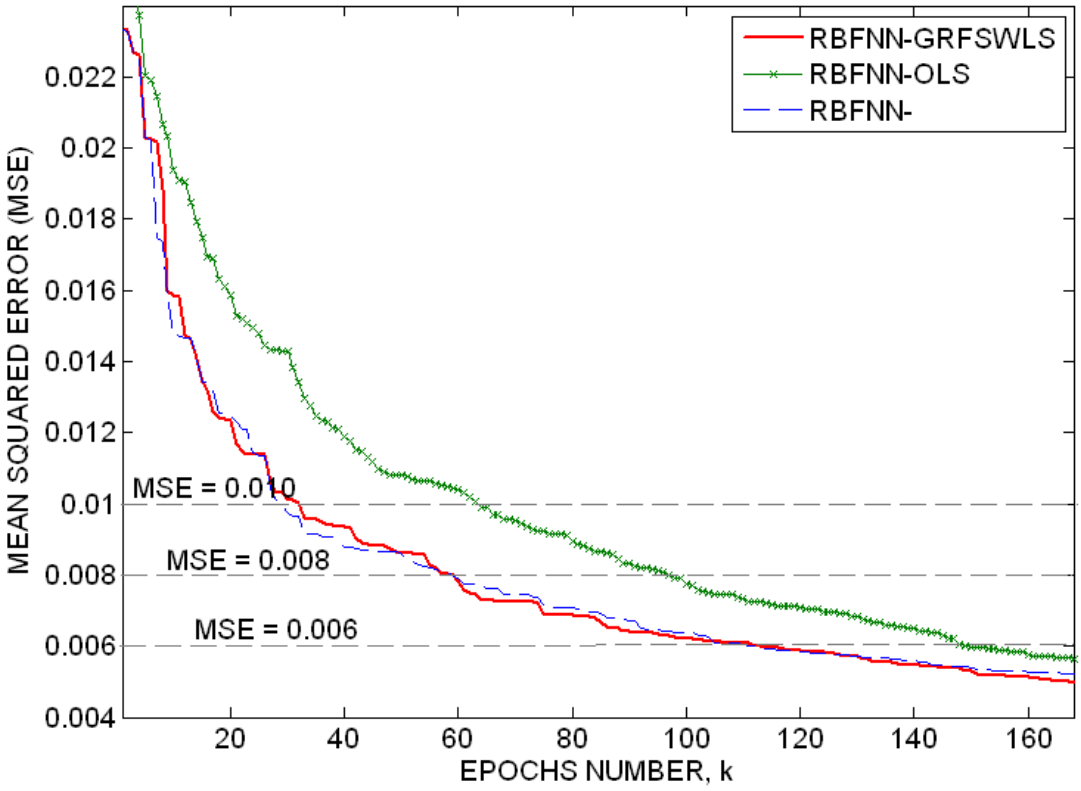


Fig. 8.19 The comparison of the training performance between GRFSWLS, OLS and FS methods.

Table 8.6 Comparison of the training performance between three methods for different *MSE* goals.

MSE	GRFSWLS		OLS		FS	
	Neurons no.	R^2	Neurons no.	R^2	Neurons no.	R^2
0.0100	31	0.673	64	0.678	29	0.667
0.0090	42	0.700	80	0.703	40	0.670
0.0080	58	0.733	97	0.727	59	0.729
0.0070	75	0.767	124	0.757	82	0.766
0.0060	111	0.793	150	0.781	113	0.792
0.0050	168	0.818	211	0.805	189	0.819

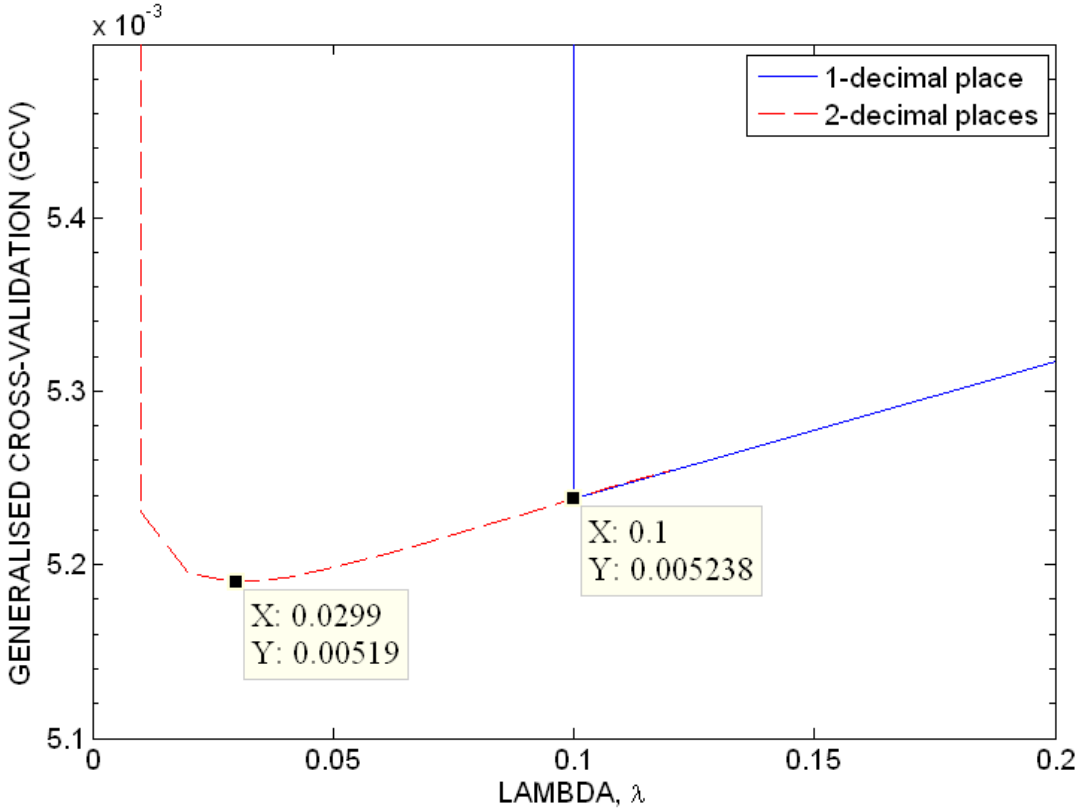


Fig. 8.20 The results of using higher decimal places for finding the regularisation parameter value in the FS method.

8.4.3. Discussion

This work has presented a neural network metamodel approach to effectively estimate the spatial distribution of daily ozone concentrations with adequately fast computation time. The model approximates the nonlinear relationship between the NO_x emission, ambient temperature, location coordinates and topography, considered as the inputs, and the eight-hour maximum average of ozone concentration as the output. For the distribution of the NO_x emission, the emission rate is derived from the measured concentration by using the Gaussian dispersion model, and then added to the emission rate obtained from the emission inventory data.

In the training stage, target output data for ozone distribution is extracted from a deterministic air quality model and calibrated to correlate with the actual data from the monitoring stations by using regression analysis. Here, data from the deterministic model and the actual measurements are combined to construct the

neural network model to enhance its training performance. Moreover, the proposed approach features the selection RBFNN centres using the regularised forward selection with weighted least squares, offering some performance improvement over the orthogonal least squares and the forward selection methods in terms of a smaller number of hidden neurons used and better estimation results. The methodology is then applied to air pollutant data collected from the monitoring stations in the Sydney basin. The results obtained indicate a promising application of the proposed method in the estimation of ozone concentration with a reasonable accuracy. Compared with the TAPM–CTM model, the proposed method yields higher performance, in which most of the estimated values are closer to the measurement data, while experiencing faster computation.

8.5. Chapter conclusion

In this chapter, three applications of the RBFNN metamodel in the atmospheric environment field have been presented. In the first attempt, the metamodel has been used to construct a model for predicting the hourly background ozone level up to twenty-four hours ahead. Several air pollutants and meteorological components have been chosen as the input parameters, being the NO concentration, the NO_2 concentration, the O_3 concentration, the wind speed and ambient temperature. The night-time BOL has been used as the output in which a local regression method was used to define its level. It has been observed that the developed model offered an acceptable accuracy when predicting lower hours until a six hour horizon, and the performance decreased when considering prediction horizons up to twenty-four hours.

In another application, the metamodel was employed to build a model for estimating the BOL in an attempt to analyse the long-term trend of its levels. Several combinations of the input variables were used to investigate their effect on the performance of the built model. It has been determined that the photochemical data incorporated by the meteorological data dominantly affected the model performance, while other pollutant agents such as SO_2, CO and PM_{10} did not influence performance to any significant degree. The next attempt introduced a more generic model for BOL estimation which could estimate its levels at each monitoring station

with acceptable results, as well as could roughly follow the pattern of BOL profiles for the case of other monitoring sites which are not included in the training stage. Hence the developed model has the potential to be extended for spatial estimation of the background ozone level in a region.

The next work presented a method to effectively estimate the spatial distribution of daily ozone concentrations with fair accuracy and adequately fast computation time. The NO_x emission data for each day was varied by adding the actual emission rate extracted from the emission database system (i.e. their values were typically similar for each day), to the emission rate which is derived from the measured concentration by using the Gaussian dispersion model. The target output data was extracted from the TAPM–CTM model and calibrated to correlate with the actual data from the monitoring stations by using regression analysis. Once the method was applied to the Sydney domain, the results showed promise in application to the spatial estimation of ozone concentration with acceptable accuracy. Hence, it has been shown that a combination of the deterministic approach (e.g. TAPM–CTM model) and a neural network will offer better estimates of the spatial distribution of an air pollutant (e.g. ozone) concentration rather than only using the dispersion model as currently applied by most regulatory agencies.

Chapter 9
CONCLUSION AND FUTURE WORK

9.1. Conclusion

This thesis has described in detail the advantageous use of the metamodel approach in several applications in atmospheric studies, particularly for measuring ozone pollution and its background level. In the early chapters following the literature review, the proposed methodology was elaborated, starting with the processes involved in the building of a metamodel based on the radial basis function neural network (RBFNN), which includes data preparation and sampling, the training process, and also validation and testing. Due to the importance of the sampling process, a new potential method for the design of experiment (DOE) was investigated. The new strategy was a weighted clustering design (WCD), which was based on distance measures and the clustering process. Several numerical analyses using non-linear functions were demonstrated, and the obtained results showed that the proposed sampling method outperformed the other two evaluated methods (i.e. LHD and *n*-FFD) with respect to several criteria, which included the performance indexes, the network size and the simulation time. It has also been learnt that the sampling number can be selected at about 30% of the full dataset as there is no significant improvement in the performance if more data is used.

Next, two improvements of the RBFNN algorithm based on orthogonal least squares (OLS) methodology was introduced. The first improvement featured the OLS algorithm with adaptively tuned spread parameter (σ) in which the σ of each hidden neuron was updated by using the steepest (gradient) descent method for the entire training process. From a numerical investigation viewpoint, the proposed algorithm offered a lesser number of hidden neurons whilst maintaining the

performance of the metamodel. The optimality of the networks could be improved further by using variable learning rates or other optimisation approaches such as the conjugate-gradient, Newton's and Marquardt method. In the second improvement, a pruning algorithm was introduced to exclude the hidden neurons making little contribution to the development of the network. Once executed, the algorithm was able to reduce the hidden number of units, especially when higher error goals were used as well as slightly improving the network's performance.

Furthermore, a new approach to training the RBFNN involving several algorithms was presented. The first attempt was to introduce a supervised training algorithm for the selection of the basis centres based on a forward selection strategy. A special case of generalised least squares (GLS) called the weighted least squares was implemented, which affords an advantage when the variances of the observations are unequal. A regularisation method was also considered to deal with the ill-condition problem. Other efforts included a method to train the network output weights and suggestions for the selection of the RBFNN model parameters. The combination of the proposed approaches, namely a generalisation network with regularised forward selection and weighted least squares (GRFSWLS), offer some performance improvement over the two benchmarked methods in terms of the total training time, the number of hidden neurons used and the estimation results in the applied problem as compared with the measurement data.

The concept of background ozone is easily understood, however the problem is how to distinguish between natural and anthropogenic effects, which requires measurement in a "clean" environment. The available best solution for this is by measuring the ozone concentration at pristine sites but it is unfortunately nearly impossible to be implemented in highly urbanised areas. This work introduced a generic method to determine background ozone levels using ambient measurement of night-time data for ozone and nitrogen oxides incorporated by some other related factors, which were specially named as the "night-time background ozone level". It was defined as the average of ambient measurements of hourly ozone values from night-time to early morning (e.g. from 8.00 pm to 8.00 am the next morning) when nitric oxide (NO) is not present for at least 2 hours consecutively. In addition, extension work also involved the exploration of some quantisation techniques to

deal with unavailable ambient measurement data using regression analysis, and to introduce a method to determine the duration time for night-time BOL. The proposed approaches here are shown to be suitable for Sydney basin, and as the concept is generic in determining BOL according to the period of interest for non-photochemical activities, it could be performed in other regions in the world and in any season of interest.

In this work, the neural network based metamodel was then been used as a statistical approximation technique to construct several models particularly for the prediction of ozone and its background ozone level, temporally and spatially, and on either a short-term or long-term scale. The idea was to design a network model for each measuring station and from this information, to construct a more generic model for application over the region of interest. Several air pollutants and meteorological components have been identified as strongly influencing the model's performance, namely the NO concentration, the NO_2 concentration, the O_3 concentration, the wind speed, the wind direction, the ambient temperature, the time information and coordinates of the locations. Suitable combinations of those inputs have been determined for each of the applied problems. A more demanding problem is to estimate the spatial distribution of air pollutants (e.g. ozone) levels across the region. The deterministic air quality model is often used for this task, but it has some shortcomings in terms of its computation time and reliability. This work has shown that the proposed metamodel approach in combination with the deterministic approach has provided better estimates of the spatial distribution of ozone concentrations with fast computation, rather than just using only the deterministic model when compared to the sites' measurement data. Complementary to the often expensive direct measurement or numerical modelling approach in air quality predictions, simpler statistical techniques including the neural network based metamodel in this work, have shown their advantages on the accuracy (i.e. better estimate results), complexity handling (i.e. significant reduction of requirements for computational resources and prior knowledge) and robustness (i.e. its capability of estimating the varying levels of ozone and background ozone from year to year), as have been demonstrated and validated thoroughly in chapters 7 and 8. Moreover, the proposed approach may increase the trustworthiness of the air quality predictions

and its future trends, thus assisting regulatory authorities in making suitable policy decisions related to air quality.

From the analytical investigation undertaken in the Sydney region, it reveals that the trend of ozone as well as the background ozone level varies greatly from year to year according to the analysis for the period from 1998 to 2010. It has been found that a great upward trend in background ozone concentration was evident from 1998 to 2005, with a slightly down trend occurring from 2005 to 2008, and with the trend slowly increasing again at a lower rate of increase from 2008 onward. The variation of the trends was related to varying weather conditions and the level of the ozone precursor. The implication of this background ozone trend is believed to be important for the regulatory authorities in many countries in setting the ozone goal and target for emission reduction, as it may be not possible to act independently in the local and regional context without a coordinated action on a global scale to also reduce the emissions of precursors. If the ozone standard is lowered it would mean that the efforts and policy measures for reduction of excessive readings in the Sydney region are less effective. Indeed, it would be more difficult to keep the ozone level within the maximum threshold in terms of human health risk and plant growth concerns.

9.2. Direction for future work

To enhance the findings of this research, several suggestions are suggested for future work to be carried out, and are listed as follows:

i. The performance of the metamodel for the spatial estimation of ozone may be improved further by considering other factors such as the population distribution and the roads network. The metamodel performance was also much dependent on the target outputs of the training process, which are extracted from the deterministic air quality model. In this work, the TAPM–CTM model has been used, however many underestimated points exist especially during the episode days which significantly affected the performance of the constructed metamodel. Therefore, the performance could be compared to alternative models such as the AUSPLUME and CMAQ models.

ii. As the concept of the proposed approach in this thesis is generic in the spatial estimation of air pollutants (i.e. ozone in this work), it has a high potential to be extended to other air pollutants such as sulphur dioxide (SO_2), carbon monoxide (CO), nitrogen dioxide (NO_2) and fine particles. For example, for the NO_2 pollutant, these input parameters could be considered: x-y location coordinates, terrain information, and emission rates for nitrogen oxides (NO_x); while for fine particles less than 10 micrometer (PM_{10}) pollutant, these input variables may be used: x-y location coordinates, terrain information, and emission rates for nitrogen oxides (NO_x), ammonia (NH_3), sulphur oxides (SO_x), VOCs, and primary PM_{10}.

iii. The construction of the metamodel requires the Matlab neural network toolbox and involves several complicated steps, which may give difficulty to the regulatory agencies in implementing the proposed methodology. To avoid this trouble, a suitable user interface (UI) or standalone toolbox should be developed so that its operation is not reliant on the Matlab software.

iv. Other regions in Australia as well as in other countries could be considered in the future evaluation, for example in South-East Asia region, to study the air pollutant transportation profiles between inland areas. As the presented methods developed in this work are generic in the temporal and spatial predictions of air pollutant (i.e. ozone in this work), they can be applicable for air quality modelling for different regions of Australia and expected to achieve similar modelling performance. The detail steps on how to run the modelling processes have been explained clearly, hence, there are easy to be followed, especially for those who have some fundamental knowledge in neural networks.

BIBLIOGRAPHY

Abdellatif, A.S.; El Rouby, A.B.; Abdelhalim, M.B., and Khalil, A.H. (2010). "Hybrid Latin Hypercube Designs," *Proc. 7th IEEE International Conference on Informatics and Systems (INFOS),* pp.1-5, 28-30.

Abdul-Wahab, A., Bouhamra, W., Ettouney, H., Soweby, B., Crittenden, B.D. (1996). "Predicting ozone levels: a statistical model for predicting ozone levels in the Shuaiba Industrial area, Kuwait," *Environmental Science and Pollution Research*, vol. 3(4), pp. 195-204.

Altshuller, A.P. and Lefohn, A.S. (1996). "Background ozone in the planetary boundary layer over the United States," *Journal Air Waste Management Association,* vol. 46(2), pp. 134-141.

AS 3580.5.1 (2011). "Methods for sampling and analysis of ambient air – Determination of oxides of nitrogen – Direct-reading instrumental method," Australian Standard 3580.5.1, pp. 1-17.

AS 3580.6.1 (2011). "Methods for sampling and analysis of ambient air Method 6.1: Determination of ozone – Direct reading instrumental method," Australian Standard 3580.6.1, pp. 1-17.

Azzi M., Johnson G.M., and Cope M. (1992). "An introduction to the generic reaction set photochemical smog mechanism," *Proceedings of the 11th International Clean Air and Environment Conference*, Brisbane, 1992.

Barton, R.R. (1992). "Metamodels for Simulation Input-Output Relations," *Proceedings of the 1992 Winter Simulation Conference*, December 13-16, 1992, Arlington, VA, pp. 289-299.

Barton, R.R. (1998). "Simulation metamodel," *Proceedings of the 1998 Winter Simulation Conference*, Washington DC, USA, pp. 167-174.

Bates, D.M. and Watts, D.G. (1988). "Nonlinear regression analysis and its applications," New York: Wiley.

Bawden, K., Farquhar, M., and Farquhar, M. (2004), "Air Emissions Inventory For The Greater Metropolitan Region in NSW – Emissions Data Management System: Design Documentation," *Consultancy report for New South Wales Department of Environment & Conservation, Sydney, NSW*.

Blanchard, C.L. (1999). "Methods for Attributing Ambient Air Pollutants to Emission Sources", *Annual Review of Energy and the Environment*, Vol. 24, pp. 329-365.

Boznar, M., Lesjak, M., Mlakar, P. (1993). "A neural network-based method for short-term predictions of ambient SO_2 concentrations in highly polluted industrial areas of complex terrain," *Atmospheric Environment*, Vol. 27(2), pp. 221-230.

Broomhead, D.S. and Lowe, D. (1988). "Multivariable functional interpolation and adaptive networks," *Complex Systems*, Vol. 2, pp. 321-355.

Buragohain, M., Mahanta, C. (2008). "A novel approach for ANFIS modelling based on full factorial design," *Applied Soft Computing*, Vol. 8(1), pp. 609-625.

Byun, D., Schere, K.L. (2006). "Review of the Governing Equations, Computational Algorithms, and Other Components of the Models-3 Community Multiscale Air Quality (CMAQ) Modeling System," *Applied Mechanics Reviews*, Vol. 59, pp. 51-77.

Carnevale, C., Finzi, G., Pisoni, E.,Volta, M. (2009). "Neoro-fuzzy and neural network systems for air quality control," *Atmospheric Environment*, Vol. 43, pp. 4811-4821.

Chan, E., Vet, R.J. (2010). "Baseline levels and trends of ground level ozone in Canada and the United States," *Atmospheric Chemistry and Physics*, Vol. 10, pp. 8629-8647.

Chang, L-C., Chang, F-J., and Wang, Y-P. (2009). "Auto-configuring radial basis function networks for chaotic time series and flood forecasting," *Hydrological Processes*, Vol. 23, pp. 2450–2459.

Chang, T., Rudy, S.J. (1989). "Urban air quality impact of methanol-fuelled vehicles compared to gasoline-fueled vehicles", *Johns Hopkins University Conference on Methanol as a Fuel Choice: An Assessment*, Washington, D.C., December, pp. 4-5.

Chen, S., Cowman, C.F. and Grant, P. (1991). "Orthogonal least squares learning algorithm for radial basis function networks," *IEEE Transaction on Neural Networks*, Vol. 2, pp. 302-309.

Chen, W., Allen, J.K., Mavris, D. and Mistree, F. (1996). "A Concept Exploration Method for Determining Robust Top-Level Specifications," *Engineering Optimization,* Vol. 26, pp. 137-158.

Chng, E.S., Yang, H.H., Bos, S. (1996). "Orthogonal least-squares learning algorithm with local adaptation process for the radial basis function networks," *Signal Processing Letters, IEEE* , Vol. 3(8), pp.253-255.

CHNPWA (1993). "Protecting Visibility in National Parks and Wilderness Areas," First edition, The National Acedemic Press, Committee on Haze in National Parks and Wilderness Areas, National Research Council, Washington, D.C.

Clapp, L., Jenkin, M. (2001). "Analysis of the relationship between ambient levels of O_3, NO_2 and NO as a function of NO_x in the UK," *Atmospheric Environment*, vol. 35, pp. 6391-6405.

Clarke, S.M., Griebsch, J.H., and Simpson, T.W. (2005). "Analysis of support vector regressionin approximation of complex engineering analysis," *Journal of Mechanical Design,* Vol. 127, pp. 1077-1087.

Coman, A., Ionescu, A., Candau, Y. (2008). "Hourly ozone prediction for a 24-h horizon using neural networks," *Environmental Modelling & Software*, Vol. 23(12), pp. 1407-1421.

Cooper, D., Alley, F.C. (2011). "Air Pollution Control: A Design Approach," Waveland Press, Inc., 4th edition, 2011.

Cope, M., and Lee, S. (2009). "Chemical Transport Model – User Manual," *The Centre for Australian Weather and Climate Research*, *NSW, Australia*, pp. 1-41.

Currin, C., Mitchell, T., Morris, M., Ylvisaker, D. (1988). "A Bayesian Approach to the Design and Analysis of Computer Experiments," *Technical Report ORNL-6498*, Oak Ridge National Laboratory, October 1988.

deBoor, C. (1978). A Practical Guide to Splines. New York: Springer-Verlag.

DECC (2007a). "Current and projected air quality in NSW," *A technical paper supporting the Clean Air Forum 2007*, Department of Environment and Climate Change NSW, pp. 1-30.

DECC (2007b). "Air Emissions Inventory for the Greater Metropolitan Region in New South Wales," *A technical paper supporting the Clean Air Forum 2007*, Department of Environment and Climate Change NSW, pp. 1-49.

DECC (2007c). "Criteria Pollutant Emissions for all Sectors: Results," Technical Report No. 1, Department of Environment and Climate Change NSW, pp. 1-546. (Available online)

DECC (2007d). "Air pollution – where does it come from?," Department of Environment and Climate Change NSW, pp. 1-6.

DECC (2008). "Air Emissions Inventory for the Greater Metropolitan Region in New South Wales – EDMS v1.0," *Technical Report No. 9*, Department of Environment and Climate Change NSW, pp. 1-73.

Demuth, H., Beale, M., and Hagan, M. (2009). "Radial Basis Networks", *Neural Network Toolbox™ 6, MATLAB User's Guide*, pp. 8-1 – 8-12.

Derwent, R.G., Stevenson, D.S., Doherty, R.M., Collins, W.J., Sanderson, M.G. (2008). "How is surface ozone in Europe linked to Asian and North American NOx emission," *Atmospheric Environment*, vol. 42, pp. 7412-7422.

Diaz-de-Quijano, M., Penuelas, J. and Ribas, A. (2009). "Increasing interannual and altitudinal ozone mixing ratios in the Catalan Pyrenees," *Atmospheric Environment*, vol. 43, pp. 6049-6057.

DoEH (2004). "State of the Air: Community Summary 1991-2001," Department of the Environment and Heritage, Australian Government. Available online: http://www.environment.gov.au/atmosphere/airquality/publications/status/pubs/community-summary.pdf

Donev, E., Zeller, K., Avramov, A., (2002), "Preliminary background ozone concentrations in the mountain and coastal areas of Bulgaria," *Environmental Pollution*, Vol. 117(2), pp. 281-286.

Duc, H., Azzi, M. (2009). "Analysis of Background Ozone in the Sydney Basin," *Proc. 18th IMACS & MODSIM Congress*, Cairns Australia, 2009, pp. 2307-2313.

Duc, H., Azzi, M., Wahid, H., Ha, Q.P. (2012) "Background ozone level in the Sydney basin: Assessment and trend analysis", *Journal of Climatology*, In Press.

Duc, H., Shannon, I., Azzi, M. (2000). "Spatial distribution characteristics of some air pollutants in Sydney," *Mathematics and Computers in Simulation*, Vol. 54, pp. 1–21.

EEA (2009). "EMEP/CORINAIR Emission Inventory Guidebook – 2006," Technical report No 9/2009, European Environment Agency. (Available online)

Elkamel, A., Abdul-Wahab, S., Bouhamra, W., Alper, E. (2001). "Measurement and prediction of ozone levels around a heavily industrialized area: a neural network approach," *Advances in Environmental Research*, Vol. 5(1), pp. 47-59.

EU (1999). "Council Directive 1999/30/EC of 22 April 1999 relating to Limit Values for Sulphur Dioxide, Nitrogen Dioxide and Oxides of Nitrogen, Particulate Matter and Lead in Ambient Air," *Official Journal of European Communities*, L163/41, pp. 41-60.

EU (2000). "Directive 2000/69/EC of the European Parliament and of the Council of 16 November 2000 relating to Limit Values for Benzene and Carbon Monoxide in Ambient Air," *Official Journal of European Communities*, L313, pp. 12-21.

EU (2002). "Council Directive 2002/3/EC of the European Parliament and of the Council of 12 February 2002 relating to Ozone in Ambient Air," *Official Journal of European Communities*, L067, pp. 14-30.

Eubank, R.L. (1988). Spline Smoothing and Nonparametric Regression. New York: Marcel Dekker.

Fabri, S., Kadirkamanathan, V. (1996). "Dynamic structure neural networks for stable adaptive control of nonlinear systems," *IEEE Trans. Neural Networks*, Vol. 7(5), pp. 1151–1167.

Fahey, D.W. (2006). "Twenty questions and answers about the ozone layer: 2006 update," *Scientific assessment of the ozone depletion: 2006*. Available online:

Fang, H., Rais-Rohani, M., Liu, Z., Horstemeyer, M.F. (2005). "A Comparative Study of Metamodeling Methods for Multiobjective Crashworthiness Optimization," *Computers and Structures*, Vol 83, pp. 2121-2136.

Fazel Zarandi, M.H., Faraji, M.R., Karbasian, M. (2012). "Interval type-2 fuzzy expert system for prediction of carbon monoxide concentration in mega-cities," *Applied Soft Computing*, Vol. 12(1), pp. 291-301.

Fiore, A., Jacob, D.J., Liu, H., Yantosca, R.M., Fairlie, T.D., Li, Q. (2003). "Variability in surface ozone background over the United States: implifications for air quality policy," *Journal of Geophysical Research*, Vol. 108, pp. 19-1 – 19-12.

Fiore, A.M., Jacob, D.J., Bey, I., Yantosca, R.M., Field, B.D., Fusco, A.C. (2002). "Background ozone over the United States in summer: Origin, trend, and contribution to pollution episodes," *J. of Geophysical Research*, vol. 107, pp. 11.1-11.25.

Forrester, A.I.J., Sobester, A., and Keane, A.J. (2008). "Engineering Design via Surrogate Modeling: A Practical Guide," Wiley, University of Southampton, UK, First ed.

Franke, R., 1982. “Scattered data interpolation: tests of some methods,” *Mathematics of Computation*, Vol. 38(157), pp. 181–200.

Friedman, J.H. (1991). “Multivariate Adaptive Regression Splines,” *Analysis of Statistic*, vol. 19(1), pp. 1-67.

Friedman, J.H. (1994). “An overview of predictive lerning and function approximation,” *From Statistics to Neural Networks, Proc. NATO/ASI Workshop*, pp. 1-61, Springer Verlag.

Fusco, A.C. and Logan, J.A. (2003). “Analysis of 1970-1995 Trends in Tropospheric Ozone at Northern Hemisphere Midlatitudes with the GEOS-Chem Model,” *J. Geophysical Research*, Vol. 108 (D15), doi:10.1029/2002JD002742, pp. 1-45.

Gadner, M.W., and Dorling, S.R. (2000). “Statistical surface ozone models: an improved methodology to account for non-linear behaviour,” *Atmospheric Environment*, Vol. 34, pp. 21-34.

Gano, S.E., Renaud, J.E., Martin, J.D., Simpson, T.W. (2006). “Update strategies for kriging models used in variable _delity optimization,” *Structural and Multidisci-plinary Optimization*, Vol. 32, pp. 287-298.

Garson, G.D. (1991). “Intreprating neural networks connection weights,” *AI Expert*, Vol. 6(4), pp. 46-51.

Ghosh, J., Deuser, L., and Beck, S. (1992). “A neural network based hybrid system for detection, characterisation and classification of short-duratrion oceanic signal,” *IEEE Journal of Ocean Engineering*, Vol. 17(4), pp. 351-363.

Giunta, A.A., and Watson, L.T. (1998). “A comparison of approximation modeling techniques: polynomial versus interpolating models,” American Institute of Aeronautics and Astronautics, AIAA-98-4758, pp. 1-13.

Godish, T. (2004). “Air Quality,” Fourth ed., Lewis Publishers, Boca Raton London New York Washington, D.C.

Golub, G.H. and Van Loan, C.G. (1996). Matrix Computations, John Hopkins University Press, Baltimore, 3rd. ed.

Golub, G.H., Heath, M., and Wahba, G. (1979). “Generalised cross-validation as method for choosing a good ridge parameter,” *Technometrics*, vol. 21, no. 2, pp. 215-223.

Gomez, P., Nebot, A., Mugica, F., Wotawa, F., (2001). "Fuzzy inductive reasoning for the prediction of maximum ozone concentration," *In Proceeding of the 13th European Simulation Symposium (EES'2001)*, pp. 535-542.

Hamerly, G. and Elkan, C. (2002). "Alternatives to the K-means Algorithm that Find Better Clusterings," *Proc. of the 11th Int. Conf. on Information and Knowledge Management (CIKM)*, November 4-9, , McLean, Virginia, USA, pp. 600-607.

Harpham, C. and Dawson, C.W. (2006). "The effect of different basis functions on a radial basis function network for time series prediction: A comparative study," *Neurocomputing*, Vol. 69(16-18), pp. 2161-2170.

Hart, M., de Dear, R., Hyde R. (2006). "A symnoptic climatology of tropospheric ozone episodes in Sydney, Australia," Int. Journal Climatology, Vol. 26, pp. 1635–1649.

Haykin, S. (1994). "Neural Networks a Comprehensive Foundation," New Jersey, Prentice Hall.

Heo, J.-S., Kim, D.S., (2004). "A new method of ozone forcasting using fuzzy expert and neural network systems," *Science Total Environment*, Vol. 325, pp. 221-237.

Horn, R. and Johnson, C. (1985). *Matrix analysis*, Cambridge University Press, Cambridge, UK, pp. 18-19.

Howlett, R.J., and Jain, L.C. (2001). "Radial basis function networks 2: New advances in design," Phisica-Verlag, Heidelberg, New York.

Hurley, P. (2008). "TAPM V4 Part 1: Technical description," *CSIRO Atmospheric Research Paper No. 25*. (Available at online)

Jacob, R.A. (1988). "Increased rates of convergence through learning rate adaptation," *Neural Networks,* Vol. 1, pp. 295-307.

Jaffe, D., Ray, J. (2007). "Increase in surface ozone at rural sites in the western US," *Atmospheric Environment*, vol. 41, pp. 5452-5463.

Jang, J.–S. R., Sun, C.-T., and Mizutani, E. (1997). Neuro-fuzzy and Soft Computing: A Computational Approach to Learning and Machine Intelligence. Prentice Hall of India Pvt. Ltd.

Jin, R., Chen, W., Simpson, T. (2001). "Comparative studies of Metamodeling techniques under multiple modeling criteria," *Structural & Multidisciplinary Optimization*, Vol. 23, pp. 1-13.

Johnson, M.E., Moore, L.M., Ylvisaker, D. (1990). "Minimax and maximin distance designs," *J. of Satatistical Planning and Infernce*, Vol. 26, pp. 131-148.

Kleijnen, J.P.C. (2005). "An Overview of the Design and Analysis of Simulation Experiments for Sensitivity Analysis," *European Journal of Operational Research*, Vol. 164, pp. 287-300.

Lee, K.L., and Billings, S.A. (2002). "Time series prediction using support vector machines, the orthogonal and the regularized orthogonal least-squares algorithms," *International Journal of Systems Science*, Vol. 33, pp. 811-821.

Lin, C.L., Wang, J.F., Chen, C.Y., Chen, C.W., Yen, C.W. (2009). "Improving the generalization performance of RBF neural networks using a linear regression technique," *Expert Systems with Applications*, Vol. 36, pp. 12049-12053.

Liu, G.P., Kadirkamanathan, V., Billings, S.A. (1999). "Variable Neural Networks for Adaptive Control of Nonlinear Systems," *IEEE Transactions on Systems, Man, And Cybernetics - Part C: Applications And Reviews*, Vol. 29(1), pp. 34-43.

Liu, X., Chance, K., Sioris, C.E., Kurosu, T.P., Spurr, J.D., Martin, R.V., Fu, T.M., Logan, J.A., Jacob, D.J., Palmer, P.I., Newchurch, M.J., Megretskaia, I.A., Chatfield, R. (2006). "First directly-retrieved global distribution of tropospheric column ozone from GOME: comparison with the GEOS-Chem model," *J. Geophysical Research*, Vol. 111, pp. 1-17.

Lu, W.Z., Wang, W.J., Wang, X.K., Yan, S.H., and Lam, J.C. (2004). "Potential assessment of a neural network model with PCA/RBF approach for forecasting pollutant trends in Mong Kok urban air, Hong Kong," *Environmental Research*, Vol. 96(1), pp. 79-87.

Luhar, A.K., Hurley, P.J. (2003). "Evaluation of TAPM, aprognostic meteorological and air polution model, using urban and rural point-source data," *Atmospheric Environment*, Vol. 37(20), pp. 2795-2810.

Ma, L., Xin, K., and Liu, S. (2008). "Using radial basis function neural networks to calibrate water quality model," *World Academy of Science, Engineering and Technology*, Vol. 38, pp. 385-393.

MacKay, D. (2003). "Chapter 20. An Example Inference Task: Clustering," *Information Theory, Inference and Learning Algorithms*, Cambridge University Press., pp. 284–292.

Martin, J.D. and Simpson, T.W. (2003). "A Study on the Use of Kriging Models to Approximate Deterministic Computer Models," *ASME Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, ASME, Chicago, Illinois, USA, 2-3 Sept 2003.

Matveev, V., Luria, M., Alper-Siman Tov, D., Peleg, M. (2002). "Long range transportation of air pollutants from Europe towards Israel," *Israeli Journal of Earth Sciences*, Vol. 51, pp. 17-28.

McKay, M.D., Beckman, R.J., and Conover, W.J. (1979). "A Comparison of Three Methods for Selecting Values of Input Variables in the Analysis of Output From a Computer Code," *Technometrics*, Vol. 42(1), pp. 55-61.

McRae, G.J., Russell, A.G. and Harley, R.A. (1992). "CIT Photochemical Airshed Model: Systems Manual," Carnegie Mellon University and California Institute of Technology.

ME (2001). "Good practice guides for preparing emission inventories," The Crown, Ministry for the Environment, New Zealand, pp. 1-78. (Available online)

Mead, J.L., and Renaut, R.A. (2009). "A Newton root-finding algorithm for estimating the regularization parameter for solving ill-conditioned least squares problems," *Journal of Inverse Problems*, vol. 25, no. 025002, pp. 1-19.

Meckesheimer, M., Barton, R.R., Limayem, F., Yannou, B. (2000). "Metamodeling of Combined Discrete/ Continuous Responses," Design Theory and Methodology – DTM'00 (Allen, J.K., Ed.), ASME, Baltimore, MD, Paper No. DETC2000/DTM-14573.

Meckesheimer, M., Booker, A.J., Barton, R.R., and Simpson, T.W. (2002). "Computationally inexpensive metamodel assessment strategies," *AIAA Journal*, Vol. 40(10), pp. 2053-2060.

Mekki, H., Chtourou, M., Derbel, N. (2006). "Variable structure neural networks for adaptive control of nonlinear systems using the stochastic approximation," *Simulation Modelling Practice and Theory*, Vol. 14(7), pp. 1000-1009.

Mintz, R., Young, B.R., Svreek, W.Y., (2005). "Fuzzy logic modelling of surface ozone concentration," *Computers & Chemical Engineering*, Vol. 29(10), pp. 2049-2059.

Monks, P.S. (2003). "TROTREP-Tropospheric ozone and precursors, trends, budgets and policy, synthesis and integration report," *Report to the EU FPV Energy*, Environment and Sustainable Development Program, European Union.

Monteiro, A., Miranda, A.I., Borrego, C., Vautard, R. (2007). "Air quality assessment for Portugal," *Science of The Total Environment*, Vol. 373(1), pp. 22-31.

Moody, J., and Darken, C.J. (1989). "Fast leaning in networks of locally tuned processing units," *Neural Computation*, Vol. 1, pp. 281–294.

Morris, R.E., Yarwood, G., Emery, C.A., Wilson, G.M. (2000). "Recent advances in CAMx Air Quality Modelling", *Paper No. 934*, ENIRON International Corporation, Carlifornia, pp. 1-16. Available online: http://www.camx.com/publ/pdfs/camx934_AWMA_2001.pdf.

Mullur, A.A. and Messac, A. (2006). "Metamodeling using extended radial basis functions: a comparative approach," *Engineering with Computers*, Vol. 21, pp. 203-217.

Nassar, R., Logan, J.A., Megretskaia, I.A., Murray, L.T., Zhang, L., Jones, D.B.A. (2009). "Analysis of tropical tropospheric ozone, carbon monoxide and water vapor during the 2006 El Niño using TES observations and the GEOS-Chem model," *J. Geophysical Research*, Vol. 114, D17304, doi:10.1029/2009JD011760, pp. 1-23.

NEPC (2003). "National Environment Protection (Ambient Air Quality) Measure," National Environment Protection Council, Canberra, pp. 1-20. (Available online)

Olden, J.D., Joy, M.K., and Death, R.G. (2004). "An accurate comparison of methods for quantifying variable importance in artificial neural networks using simulated data," *Ecological Modelling*, Vol. 178(3–4), pp. 389-397.

Olea, R.A. (1999). "Geostatistics for Engineers and Earth Scientists," Kluwer Academic Publishers.

Oltmans, S., Lefohn, A., Harris, J., Galbally, I., Scheel, H. E., Bodeker, G., Brunke, E., Claude, H., Tarasick, D., Johnson, B.J., Simmonds, P., Shadwick, D., Anlauf, K., Hayden, K., Schmidlin, F., Fujimoto, T., Akagi, K., Meyer, C., Nichol, S., Davies, J., Redondas, A., Cuevas, E. (2006). "Long-term changes in troposphere ozone," *Atmospheric Environment*, Vol. 40, pp. 3156-3173.

Oltmans, S.J., Lefohn, A., Harris, J., Shadwick, D. (2008). "Background ozone levels of air entering the west coast of the US and assessment of long-term changes," *Atmospheric Environment*, Vol. 42, pp. 6020-6038.

Orr, M.J.L. (1993). "Regularised centre recruitment in radial basis function networks," Neural Computation, vol. 59, pp. 1-11.

Orr, M.J.L. (1996). "Introduction to Radial Basis Function Networks. Technical report, Centre for Cognitive Science," University of Edinburgh. Available online: http://www.anc.ed.ac.uk/rbf/papers/intro.ps

Parish, D.D., Millet, D.B., Goldstein, A.H. (2009). "Increasing ozone in marine boundary layer inflow at the west coasts of North America and Europe," *Atmospheric Chemistry and Physics*, Vol. 9, pp. 1303-1323.

Park, J., and Sandberg, I.W. (1991). "Universal approximation using radial-basis function networks," *Neural Computation*, Vol. 3, pp. 246–257.

Park, J.S. (1991). "Optimal Latin-hypercube Designs for Computer Experiments," *Journal of Statistical Planning and Inference*, Vol. 39, pp. 95-111.

Pasquill, F. (1976). "Atmospheric dispersion parameters in Gaussian plume modeling," U.S. Environmental Protection Agency Rep. EPA-600/4-76-030B.

Pfeiffer, H., Baumbach, G., Sarachaga-Ruiz, L., Kleanthous, S., Poulida, O., Beyaz, E. (2009). "Neural modelling of the spatial distribution of air pollutants," *Atmospheric Environment*, Vol. 43 (20), pp. 3289-3297.

Phillips, S.B., Finkelstein, P.L. (2006). "Comparison of spatial patterns of pollutant distribution with CMAQ predictions," *Atmospheric Environment*, Vol. 40(26), pp. 4999-5009.

Poggio, T. and Girosi, F. (1988). "Networks for approximation and learning," Proceedings of the IEEE, Vol. 78, pp.1481-1497.

Poshal, G., and Ganesan, P. (2008). "An analysis of formability of aluminium performs using neural network," *J. of Materials Processing Technology*, Vol. 205, pp. 272-282.

Powell, M. (1987). "Radial basis functions for multivariable interpolation: A review," *Algorithms for Approximations*, pp. 143-167.

Protonotariou, A.P., Tombrou, M., Giannakopoulos, C., Kostopoulou, E., Le Sager, P. (2010). "Study of CO surface pollution in Europe based on observations and nested-grid applications of GEOS-Chem global chemical transport model," *Tellus B*, Vol. 62, doi: 10.1111/j.1600-0889.2010.00462.x, pp. 209-227.

Rao, S.S. (2009). "Engineering Optimization Theory and Practice", Fourth Edition, John Wiley & Sons Inc.

Roth, P.M., Blanchard, C.E., Reynolds, S.D. (1989). "The Role of Grid-Based Reactive Air Quality Modeling in Policy Analysis: Perspectives and Implications as Drawn from a Case Study", U.S. Environmental Protection Agency, Research Triangle Park, N.C., EPA/600/3-89-082.

Sacks, J., Welch, W.J., Mitchell, T.J., Wynn, H.P. (1989). "Design and Analysis of Computer Experiments," *Statistical Science*, Vol. 4, pp. 409-435.

Saitanis, C.J. (2003). "Background ozone monitoring and phytodetection in the greater rural area of Corinth-Greece," *Chemosphere*, Vol. 51(9), pp. 913-923.

Sarimveis, H., Alexandridis, A., Mazarakis, S., and Bafas, G. (2004). "A new algorithm for developing dynamic radial basis function neural network models based on genetic algorithms," *Computers and Chemical Engineering*, Vol. 28, pp. 209-217.

Sathyanarayanamurthy, H., Chinnam, R.B. (2009). "Metamodels for variable importance decomposition with applications to probabilistic engineering design," *Computers & Industrial Engineering*, Volume 57(3), pp. 996-1007.

Seigneur, S. (2001). "Current status of air quality models for particulate matter," *Journal of the Air and Waste Management Association*, Vol. 51, pp. 1508–1521.

Seinfeld, J.H. (1989). "Urban air pollution: state of science," *Science*, Vol. 243, pp. 745-752.

Shahsavand, A. (2009). "An optimal radial basis function (RBF) neural network for hyper-surface reconstruction," *Chemistry and Chemical Engineering Transactions*, vol. 16, no. 1, pp. 41-53.

Sherstinsky, A., and Picard, R.W. (1996). "On the Efficiency of the Orthogonal Least Squares Training Method for Radial Basis Function Networks," *IEEE Transactions on Neural Networks*, Vol. 7, pp. 195-200.

Simmonds, P., Derwent, R., Manning, A., Spain, G. (2004). "Significant growth in surface ozone at Mace Head, Ireland, 1987–2003," *Atmospheric Environment*, vol. 38, pp. 4769-4778.

Simpson, T. W., Peplinski, J., Koch, P. N. and Allen, J.K. (1997). "On the Use of Statistics in Design and the Implications for Deterministic Computer Experiments," *Design Theory and Methodology - DTM'97*, Sacramento, CA, ASME, Paper No. DETC97/DTM-3881.

Simpson, T.W., Booker, A.J., Ghosh, D., Giunta, A.A., Koch, P.N., and Yang, R.J. (2004). "Approximation Methods in Multidisciplinary Analysis and Optimization: a Panel Discussion," *Structural and Multidisciplinary Optimization*, Vol. 27, pp. 302-313.

Simpson, T.W., Mauery, T.M., Korte, J.J., Mistree, F. (1998). "Comparison of Response Surface and Kriging Models for Multidisciplinary Design Optimization," *7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis & Optimization,* St. Louis, MO, AIAA, Vol. 1, pp. 381-391. AIAA-98-4755.

Sousa, S.I.V., Martins, F.G., Pereira, M.C., Alvim-Ferraz, M.C.M. (2005). "Prediction of ozone concentrations in Oporto city with statistical approaches," *Chemosphere*, Vol. 64, pp. 1141-1149.

Srivastava, A., Rao, B.P.S. (2011). "Urban air pollution modeling," *Air Quality – Models and Application*, pp. 15-32.

Sung, A.H. (1998). "Ranking importance of input parameters of neural networks, *Expert Systems with Applications*," Vol. 15(3-4), pp. 405-411.

Swiler, L.P., Slepoy, R., and Giunta, A.A. (2006). "Evaluation of Sampling Methods in Constructing Response Surface Approximations," *47th AIAA/ASME/ ASCE/AH-S/ASC Structures, Structural Dynamics and Materials Conference*, No. AIAA-2006-1827, Newport, USA, pp. 1-24.

Tao, K.M. (1993). "A closer look at the radial basis function (RBF) networks," *Conference Record of The Twenty-Seventh Asilomar Conference on Signals, Systems and Computers*, 1993, Vol. 1, pp. 401-405.

TEC (2003). "Model 49C – UV photometric O_3 analyzer," *Instruction manual, Thermo Electron Corporation*, p/n: 9999, pp. 1-1 – 9-8.

TEC (2004). "Model 42C – Chemiluminescence NO-NO_2-NO_x analyzer," *Instruction manual, Thermo Electron Corporation*, p/n: 100174-00, pp. 1-1 – 9-10.

Tesche, T.W. (1983). "Photochemical dispersion modeling: Review of model concepts and applications studies", *Environment International*, Vol. 9, pp. 465-490.

Tikhonov, A.N. (1973). "On Regularization of Ill-posed Problems," *Doklady Akademii Nauk USSR*, Vol. 153, pp. 49-52.

Trabelsi, A., Lafont, F., Kamoun, M., Enea, G. (2007). "Fuzzy identification of a greenhouse," *Applied Soft Computing*, Vol. 7(3), pp.1092-1101.

Tunali, S., Batmaz, I. (2003). "A metamodeling methodology involving both qualitative and quantitative input factors," *European Journal of Operational Research*, Vol. 150, pp. 437-450.

Tunali, S., Batmaz, I. (2003). "A metamodeling methodology involving both qualitative and quantitative input factors," *European Journal of Operational Research*, vol. 150(16), pp. 437-450.

Turner, D.B. (1970). "Workbook of Atmospheric Dispersion Estimates," U.S. Department of Health, Education, and Welfare, pp. 1-69.

Ukyan, Z., Gzelis, C.G. (1997). "Input-output clustering for determining the centers of radial basis function network," *Proc. of ECCTD-98*, Budapest, Hungary, pp. 435-439.

Unal, R., Lepsch, R.A., Engelund, W., Stanley, D.O. (1996). "Approximation Model Building and Multidisciplinary Design Optimization Using Response Surface Methods," *6th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization,* Bellevue, WA, AIAA, Vol. 1, pp. 592-597.

US-EPA (1986). "Guideline on Air Quality Models", U.S. Environmental Protection Agency, Research Triangle Park, N.C., EPA/450/2-78-027R.

US-EPA (2001). "Community Multiscale Air Quality (CMAQ)", U.S. Environmental Protection Agency, Research Triangle Park, NC.

US-EPA (2006a), "Air quality criteria for ozone and related photochemical oxidants (final)," Research Triangle Park, NC: Office of Research and Development, U.S. Environmental Protection Agency, 2006 EPA/600/R-05/004af.

US-EPA (2006b). "Review of the Process for Setting National Ambient Air Quality Standards," NAAQS Process Review Workgroup, U.S. Environmental Protection Agency, Research Triangle Park, NC. (Available online)

US-EPA (2011a). "The Ambient Air Monitoring Program," U.S. Environmental Protection Agency, Research Triangle Park, NC. (Available online)

US-EPA (2011b). "National Ambient Air Quality Standards (NAAQS)," U.S. Environmental Protection Agency, Research Triangle Park, NC. (Available online)

US-EPA (2011c). "List of the Designated Reference and Equivalent Methods," U.S. Environmental Protection Agency, National Exposure Research Laboratory, NC. (Available online)

US-EPA (2011d). "Emission Inventory," U.S. Environmental Protection Agency, Research Triangle Park, NC. (Available online)

US-EPA (2011e). "National Ambient Air Quality Standards (NAAQS)," U.S. Environmental Protection Agency. (Available online)

US-EPA (2012). "Clearinghouse for Inventories and Emission Factors," U.S. Environmental Protection Agency, Technology Transfer Network, NC. (Available online)

Vardoulakis, S., Fisher, B.E.A., Pericleous, K., Gonzalez-Flesca, N. (2003). "Modelling air quality in street canyons: a review", *Atmospheric Environment*, Vol. 37, pp. 155-182.

Vingarzan, R. (2004). "A review of surface ozone background levels and trends," *Atmospheric Environment*, vol. 38, pp. 3431-3442.

Vogel, C.R. (2002). "Computational methods for inverse problems," *Frontiers in Applied Mathematics*, SIAM.

Wahid, H., Ha, Q.P. and Duc, H. (2010a). "Adaptive Neural Network Metamodel for Short-term Prediction of Natural Ozone Level in an Urban Area," *Proc. Int. Conf. on Computing and Communication Technologies* (2010 IEEE-RIVF), Hanoi, Vietnam, 1-4 Nov 2010, pp. 250-253.

Wahid, H., Ha, Q.P. and Duc, H.N. (2010b), "A Metamodel for Background Ozone Level Using Radial Basis Function Neural Networks," *Pro. 11th International Conference on Control, Automation, Robotics and Vision (ICARCV 2010)*, Singapore, 7-10 Dec 2010, pp. 958-963.

Wahid, H., Ha, Q.P., Duc, H.N. (2011). "Computational intelligence estimation of natural background ozone level and its distribution for air quality modelling and emission control," *Pro. 28th International Symposium on Automation and Robotics in Construction (ISARC 2011)*, Seoul, Korea, 29 Jun-2 Jul 2011, pp. 551-557.

Wahid, H., Ha, Q.P., Duc, H. (2012). "New sampling scheme for neural network-based metamodelling with application to air pollutant estimation," *Proc. ISG-ISARC 2012*, Eindhoven, Netherlands; appears in *Gerontechnology Journal*, vol. 11(2), 2012, pp. 336, doi:10.4017/gt.2012.11.02.325.00.

Wang, G. and Shan, S. (2007). “Review of Metamodeling Techniques in Support of Engineering Design Optimization,” Journal of Mechanical Design, Vol 129, pp. 370-380.

Wang, G.G. (2003). “Adaptive Response Surface Method Using Inherited Latin Hypercube Design Points," *Journal of Mechanical Design*, Vol. 125, No. 2, pp. 210-220.

Wang, H.R. Wang, H.B., Wei, L.X., Li, Y. (2002). “A new algorithm of selecting the radial basis function networks center,” *Proc. Int. Conf. on Machine Learning and Cybernetics*, Beijing, China, pp. 1801-1804.

Wang, L., Beeson, D., Wiggs, G., and Rayasam, M. (2006). “A Comparison of Meta-modeling Methods Using Practical Industry Requirements,” *47th AIAA/AS-ME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference*, No. AIAA 2006-1811, Newport, Rhode Island, 1-4 May 2006.

Wang, W., Lu, W., Wang, X., Leung, A. (2003). “Prediction of maximum daily ozone level using combined neural network and statistical characteristics,” *Environment International*, Vol. 29(5), pp. 555-562.

Whitten, G.Z., Yonkow, N., Myers, T.C. (1986). “Photochemical Modeling of Methanol-use Scenarios in Philadelphia”, U.S. Environmental Protection Agency, EPA/460/3-86-001.

WHO (2005). “WHO Air Quality Guidelines Global Update 2005,” *Report on a Working Group meeting*, Bonn, Germany, 18-20 October 2005, Copenhagen, World Health Organization, WHO Regional Office for Europe.

Xie, D., Sun, X., Bai, B. Yang, S. (2008). “Multiobjective Optimization Based on Response Surface Model and its Application to Engineering Shape Design,” *IEEE Transactions on Magnetics*, Vol. 44(6), pp. 1006-1009.

Yi, J., Prybutok, V.R. (1996). “A neural network model forecasting for prediction of daily maximum ozone concentration in an industrialized urban area,” *Environmental Pollution*, Vol. 92(3), pp. 349-357.

Zainuddin, Z., Pauline, O. (2011). “Modified wavelet neural network in function approximation and its application in prediction of time-series pollution data,” *Applied Soft Computing*, Vol. 11(8), pp. 4866-4874.

Zhao, W. and Huang, D.S. (2002). “The structure optimization of radial basis probabilistic neural networks based on genetic algorithm,” *Proc. World Congress on Computational Intelligence*, Honolulu, Hawaii, pp. 1086-1091.

APPENDICES

Appendix A: Matlab codes for three DOE methods.

A-1 Matlab codes for Weighted Clustering Design (WCD) sampling method.

```
function [] = Model_sampling_WCD_iteration()
% Data sampling using the Weighted Clustering Design (WCD) method
% -----------------------------------------------------------------
%
% The input-output data have to be provided (i.e. large data to be
   sampled)
% e.g.: 'Test_7d' file contains of TESTIN and TESTOUT, which
   considered as large dataset

clear all;
clc;

dataname='Test_7d';
% dataname='Test_5d_ozone';

    disp(' ');
    disp('Data sampling in process.  This may take a while, please
          wait...');
    disp(' ');
    load(dataname);

perc = 0.25;        % percentage number of data points to be sampled
n_iter = 1;         % the number of data points' group (e.g. 3 days
= 3 groups)
u = 1;              % the first number of a group data points
v = 4000;           % maximum number of a group data points
max_n = v;
INPUT = [];
OUTPUT = [];

for q = 1:n_iter
    q
    INPUTFULL = TESTIN(u:v,:);
    OUTPUTFULL = TESTOUT(u:v,:);
    DATA = [INPUTFULL,OUTPUTFULL];
    [pn,ps] = mapminmax(DATA',-1,1);
    pn = pn';
    [a b] = size(pn);

    c = zeros(b,1);
    d = dist(pn,c);
    [S SI] = sort(d,'ascend');
    n = round(perc*length(S));
    T = kmeans(S,n);
    used = [];
```

```
    for k=1:n
        R = find(T==k);
        RR = R(1,1);
        pick = SI(RR,1);
        used = [used pick'];
    end

    used = used';
    INPUTq = INPUTFULL(used,:)';
    OUTPUTq = OUTPUTFULL(used,:)';
    INPUT = [INPUT,INPUTq];
    OUTPUT = [OUTPUT,OUTPUTq];
    u = u + max_n;
    v = v + max_n;
end
INPUT = INPUT';
OUTPUT = OUTPUT';

% Scatter plot of the sampled data
    figure;
    scatter3(pn(:,1),pn(:,2),pn(:,3),100,T,'filled');
    figure;
    scatter3(pn(:,3),pn(:,4),pn(:,5),100,T,'filled');
    figure;
    scatter3(INPUT(:,3),INPUT(:,4),INPUT(:,5),100);
save result_DOE;
```

A-2 Matlab codes for Latin Hypercube Design (LHD) sampling method.

```
function [] = Model_sampling_LHD_iteration()
% Data sampling using the n-level Latin Hypercube Design (LHD
% ------------------------------------------------------------
%
% The input-output data have to be provided (i.e. large data to be
   sampled)
% e.g.: 'Test_7d' file contains of TESTIN and TESTOUT, which
   considered as large dataset

clear all;
clc;

dataname='Test_7d';
%dataname='Test_5d_ozone';

disp(' ');
disp('Data sampling in process.  This may take a while, please
wait...');
disp(' ');
load(dataname);

perc = 0.85;        % percentage number of data points to be sampled
n_iter = 1;         % the number of data points' group (e.g. 3 days
                       = 3 groups)
nvar_LHD = 7;       % the number of design variables
```

```
u = 1;              % the first number of a group data points
v = 4000;           % maximum number of a domain data points
max_n = v;
INPUT = [];
OUTPUT = [];

for q = 1:n_iter
    q
    INPUTFULL = TESTIN(u:v,:);
    OUTPUTFULL = TESTOUT(u:v,:);
    DATA = INPUTFULL;
    [pn,ps] = mapminmax(DATA,0,1);
    [a b] = size(pn);

    % FFD sampling points
    % -------------------
    n = round(perc*a);
    dLH = lhsdesign(n,nvar_LHD,'criterion','maximin');

    % Compute distances between all possible pairs of points
    % ------------------------------------------------------
    J = dist(pn,dLH');
    [Jmin I] = min(J(:,1));
    [m n] = size(J);

    Index = I;
    for i=2:n
        [Jmin I] = sort(J(:,i),'ascend');
        for k=1:m
            %[Jmin I] = sort(J(:,i),'ascend');    %
            pick_I = I(k,1);
            if pick_I~=Index, break
            end
        end
        Index = [Index pick_I];
    end

    used1 = Index';
    used = unique(used1,'first');
    INPUTq = INPUTFULL(used,:)';
    OUTPUTq = OUTPUTFULL(used,:)';
    %save result_DOE;
    INPUT = [INPUT,INPUTq];
    OUTPUT = [OUTPUT,OUTPUTq];
    u = u + max_n;
    v = v + max_n;
end
INPUT = INPUT';
OUTPUT = OUTPUT';

save result_DOE;
```

A-3 Matlab codes for n-level Full Factorial Design (n-FFD) sampling method.

```
function [] = Model_sampling_FFD_iteration()
% Data sampling using n-level full factorial design (FFD)
% -----------------------------------------------------
%
% The input-output data have to be provided (i.e. large data to be
   sampled)
% e.g.: 'Test_7d' file contains of TESTIN and TESTOUT, which
   considered as large dataset
% The sampling number defined by the combination of n-level in 'dFF'
% function

clear all;
clc;

dataname='Test_7d';
%dataname='Test_5d_ozone';

disp(' ');
disp('Data sampling in process.  This may take a while, please
wait...');
disp(' ');
load(dataname);

n_iter = 1;         % the number of data points' group (e.g. 3 days
= 3 groups)
u = 1;              % the first number of a group data points
v = 4000;           % maximum number of a domain data points
max_n = v;
INPUT = [];
OUTPUT = [];

for q = 1:n_iter
    q
    INPUTFULL = TESTIN(u:v,:);
    OUTPUTFULL = TESTOUT(u:v,:);
    DATA = INPUTFULL;
    [pn,ps] = mapminmax(DATA,0,1);

    % FFD sampling points
    % -------------------
    dFF = fullfactorial([3 3 3 3 3 3 4],1); %for Test1_7 variables
    % dFF = fullfactorial([4 4 4 4 5],1); %for Test2_5
                                          variables_ozone

    % Compute distances between all possible pairs of points
    % ------------------------------------------------------
    J = dist(pn,dFF');
    [Jmin I] = min(J(:,1));
    [m n] = size(J);

    Index = I;
    for i=2:n
        [Jmin I] = sort(J(:,i),'ascend');
        for k=1:m
            %[Jmin I] = sort(J(:,i),'ascend');    %
            pick_I = I(k,1);
```

```
            if pick_I~=Index, break
            end
        end
        Index = [Index pick_I];
    end

    used1 = Index';
    used = unique(used1,'first');
    INPUTq = INPUTFULL(used,:)';
    OUTPUTq = OUTPUTFULL(used,:)';
    %save result_DOE;
    INPUT = [INPUT,INPUTq];
    OUTPUT = [OUTPUT,OUTPUTq];
    u = u + max_n;
    v = v + max_n;
end
INPUT = INPUT';
OUTPUT = OUTPUT';

save result_DOE;

function X=fullfactorial(q, Edges)
% Generates a full factorial sampling plan in the unit cube
%
%Inputs:
% q – k – vector containing the number of points along each
% dimension
% Edges – if Edges=1 the points will be equally spaced from
% edge to edge (default), otherwise they will be in
% the centres of n=q(1)? q(2)? _ _ _ q(k) bins filling
% the unit cube.
%
% Output:
% X – full factorial sampling plan
if nargin < 2, Edges=1; end
if min(q) < 2
    error('You_must_have_at_least_two_points_per_dimension.');
end
% Total number of points in the sampling plan
n=prod(q);
% Number of dimensions
k=length(q);
%Pre–allocate memory for the sampling plan
X=zeros(n,k);
%Additional phantom element
q(k+1)=1;
for j=1:k
    if Edges==1
        one_d_slice =(0:1/(q(j)-1):1);
    else
        one_d_slice =(1/q(j)/2:1/q(j):1);
    end
    column=[];
    while length(column) <n
        for l=1:q(j)
            column=[column; ones(prod(q(j+1:k)),1)* one_d_slice(l)];
        end
    end
    X(:,j)=column;
end
```

A-4 Codes to generate large data points from a known function

```
% Codes to generate large data points from a known function, y
% ------------------------------------------------------------
%
% All variables have the same constraint (i.e. range)

clear all;
clc;

hi = 10;        % low range
lo = -10;       % high range
nr = 4000;      % the number of data points to generated
nvar = 7;       % the number of tyhe design variables
vari = 0.001;   % the variance of the data

inp = lo+(hi-lo)*rand(nvar,nr);
x1 = inp(1,:); x2 = inp(2,:); x3 = inp(3,:); x4 = inp(4,:); x5 =
inp(5,:); x6 = inp(6,:); x7 = inp(7,:);

s = length(x1);
n = ones(1,s);
y = (x1-10*n).^2 + 5*(x2-12*n).^2 + x3.^4 + 3*(x4-11*n).^2 +
10*x5.^6 + 7*x6.^2 + x7.^4 - 4*x6.*x7 - 10*x6 - 8*x7;
%y = (x1-10*n).^2 + 5*(x2-12*n).^2;
yd = y + sqrt(0.001).*randn(1,s);

figure;
plot(x7,y,x7,yd,'o','MarkerFaceColor',[0 0 0]);
% figure;
% plot(x,y,INPUT30,OUTPUT30,'o','MarkerFaceColor',[0 0 0]);
```

Appendix B: Matlab codes for some improvement of OLS algorithm

B-1 Codes for RBFNN with adaptively tuned spread parameter

```
function [net,tr]=newrb3b_sp3(p,t,goal,spread,mn,df,sp1)
%NEWRB Design a radial basis network (with adaptively tuned spread
(sp)
%-------------------------------------------------------------------
%
%    [net,tr] = newrb(P,T,GOAL,SPREAD,MN,DF,SP1)
%
%  Description
%
%    Radial basis networks can be used to approximate
%    functions.  NEWRB adds neurons to the hidden
%    layer of a radial basis network until it meets
%    the specified mean squared error goal.
%
%   NEWRB(P,T,GOAL,SPREAD,MN,DF) takes these arguments,
%     P      - RxQ matrix of Q input vectors.
%     T      - SxQ matrix of Q target class vectors.
%     GOAL   - Mean squared error goal, default = 0.0.
%     SPREAD - Spread to build the P matrix
%     MN     - Maximum number of neurons, default is Q.
%     DF     - Number of neurons to add between displays,default=25.
%     SP1    - Initial spread parameter for the gradient descent
%
% Copyright 1992-2007 The MathWorks, Inc.
% Modified by Herman Wahid, June, 2012

if nargin < 2, error('NNET:Arguments','Not enough input arguments'),
end

% Defaults
if nargin < 3, goal = 0; end
if nargin < 4, spread = 1; end
if nargin < 6, df = 10; end
if nargin < 7, sp1 = 1; end

% Format
if isa(p,'cell'), p = cell2mat(p); end
if isa(t,'cell'), t = cell2mat(t); end

% Error checks
if (~isa(p,'double')) | (~isreal(p)) | (length(p) == 0)
  error('NNET:Arguments','Inputs are not a non-empty real matrix.')
end
if (~isa(t,'double')) | (~isreal(t)) | (length(t) == 0)
  error('NNET:Arguments','Targets are not a non-empty real matrix.')
end
if (size(p,2) ~= size(t,2))
  error('NNET:Arguments','Inputs and Targets have different numbers
of columns.')
end
if (~isa(goal,'double')) | ~isreal(goal) | any(size(goal) ~= 1) |
(goal < 0)
  error('NNET:Arguments','Performance goal is not a positive or zero
real value.')
```

```
end
if (~isa(spread,'double')) | ~isreal(spread) | any(size(spread) ~=
1) | (spread < 0)
  error('NNET:Arguments','Spread is not a positive or zero real
value.')
end
if (~isa(df,'double')) | ~isreal(df) | any(size(df) ~= 1) | (df < 1)
| (round(df) ~= df)
  error('NNET:Arguments','Display frequency is not a positive
integer.')
end

% More defaults
Q = size(p,2);
if nargin < 5, mn = Q; end

% More error checking
if (~isa(mn,'double')) | ~isreal(mn) | any(size(mn) ~= 1) | (mn < 1)
| (round(mn) ~= mn)
  error('NNET:Arguments','Maximum neurons is not a positive
integer.')
end


% Dimensions
R = size(p,1);
S2 = size(t,1);

% Architecture
net = network(1,2,[1;1],[1; 0],[0 0;1 0],[0 1]);

% Simulation
net.inputs{1}.size = R;
net.layers{1}.size = 0;
net.inputWeights{1,1}.weightFcn = 'dist';
net.layers{1}.netInputFcn = 'netprod';
net.layers{1}.transferFcn = 'radbas';
net.layers{2}.size = S2;
net.outputs{2}.exampleOutput = t;

% Performance
net.performFcn = 'mse';

% Design Weights and Bias Values
warn1 = warning('off','MATLAB:rankDeficientMatrix');
warn2 = warning('off','MATLAB:nearlySingularMatrix');
[w1,b1,w2,b2,tr] = designrb(p,t,goal,spread,mn,df,sp1);
warning(warn1.state,warn1.identifier);
warning(warn2.state,warn2.identifier);

net.layers{1}.size = length(b1);
net.b{1} = b1;
net.iw{1,1} = w1;
net.b{2} = b2;
net.lw{2,1} = w2;

%=========================================================
function [w1,b1,w2,b2,tr] = designrb(p,t,eg,sp,mn,df,sp1)

[r,q] = size(p);
[s2,q] = size(t);
```

```
b = sqrt(-log(.5))/sp;
%sp1 = 0.1;
SP = [];
SP = [SP sp];
MSEALL = [];

% RADIAL BASIS LAYER OUTPUTS
P = radbas(dist(p',p)*b);
PP = sum(P.*P)';
d = t';
dd = sum(d.*d)';

% CALCULATE "ERRORS" ASSOCIATED WITH VECTORS
e = ((P' * d)' .^ 2) ./ (dd * PP');

% PICK VECTOR WITH MOST "ERROR"
pick = findLargeColumn(e);
used = [];
left = 1:q;
W = P(:,pick);
P(:,pick) = []; PP(pick,:) = [];
e(:,pick) = [];
used = [used left(pick)];
left(pick) = [];

% CALCULATE ACTUAL ERROR
b = sqrt(-log(.5))/sp1;
w1 = p(:,used)';
a1 = radbas(dist(w1,p)*b);
[w2,b2] = solvelin2(a1,t);
a2 = w2*a1 + b2*ones(1,q);
MSE = mse(t-a2);
MSEALL = [MSEALL MSE];
RMSE0 = sqrt(MSE);

% Start
tr = newtr(mn,'perf');
tr.perf(1) = mse(t-repmat(mean(t,2),1,q));
tr.perf(2) = MSE;
if isfinite(df)
    fprintf('NEWRB, neurons = 0, MSE = %g\n',tr.perf(1));
    fprintf('NEWRB, neurons = 1, MSE = %g\n',tr.perf(2));
    fprintf('NEWRB, neurons = 1, SP = %g\n',sp);
end
flag_stop = 0;

iterations = min(mn,q);
%sp1=sp;
for k = 2:iterations

  % CALCULATE "ERRORS" ASSOCIATED WITH VECTORS
  wj = W(:,k-1);
  a = wj' * P / (wj'*wj);
  P = P - wj * a;
  PP = sum(P.*P)';
  e = ((P' * d)' .^ 2) ./ (dd * PP');

  % PICK VECTOR WITH MOST "ERROR"
  pick = findLargeColumn(e);
  W1 = [W, P(:,pick)];
  P(:,pick) = []; PP(pick,:) = [];
```

```
  e(:,pick) = [];
  used = [used left(pick)];
  left(pick) = [];

  %b = sqrt(-log(.5))/sp;
  % CALCULATE ACTUAL ERROR
  w1 = p(:,used)';
%   a1 = radbas(dist(w1,p)*b);
%   [w2,b2] = solvelin2(a1,t);
%   a2 = w2*a1 + b2*ones(1,q);
%   MSE = mse(t-a2);

  % FIND OPTIMAL SPREAD
  % -------------------
  %RMSE0 = sqrt(MSE);
  spz = sp1 + 0.5*sp1;
  sp0 = sp1;
  kk = 1;
  while kk>0 && kk<1000
  b = sqrt(-log(.5))/spz;
  w1 = p(:,used)';
  a1 = radbas(dist(w1,p)*b);
  [w2,b2] = solvelin2(a1,t);
  a2 = w2*a1 + b2*ones(1,q);
  MSE = mse(t-a2);
  RMSE1=sqrt(MSE);

  step = 30;
  M = 1e-5;       % Threshold value to terminate gradient descent
  z = (spz-sp0);
  %if z==0, break, end
  gradrmse=(RMSE1-RMSE0)/z;
  %graddiff=(gradrmse-gradrmse0);
  ratio=abs((RMSE1-RMSE0)/RMSE0);
    if ratio>=M
        RMSE0=RMSE1;                %abs(z)
        sp0=spz;
        %gradrmse0=gradrmse;
        spz=spz-step*gradrmse;
        kk=kk+1;
        elseif ratio>=0 && ratio<M ||isnan(ratio)
            break;
    end
  kk;
  end
  W=W1;
  SP = [SP spz];
  MSEALL = [MSEALL MSE];
  % ---------------------

  % PROGRESS
  tr.perf(k+1) = MSE;

  % DISPLAY
  if isfinite(df) & (~rem(k,df))
    fprintf('NEWRB, neurons = %g, MSE = %g\n',k,MSE);
    fprintf('NEWRB, neurons = %g, SP = %g\n',k,spz);
    %flag_stop=plotperf(tr,eg,'NEWRB',k);
  end

  % STOP CONDITION
```

```
  if (MSE < eg), break, end
  if (flag_stop), break, end
end

[S1,R] = size(w1);
b1 = ones(S1,1)*b;

% Finish
tr = cliptr(tr,k);

save work;

%=========================================================

function i = findLargeColumn(m)

replace = find(isnan(m));
m(replace) = zeros(size(replace));

m = sum(m .^ 2,1);
i = find(m == max(m));
i = i(1);

%=========================================================

function [w,b] = solvelin2(p,t)

if nargout <= 1
  w= t/p;
else
  [pr,pc] = size(p);
  x = t/[p; ones(1,pc)];
  w = x(:,1:pr);
  b = x(:,pr+1);
end

%=========================================================
```

B-2 Pruning algorithm codes (additional codes of Appendix B-1)

```
  % CHECK THE STRENGTH OF THE NODE
  y = mean((a1)');
  yy = y(:,k-n);
  yyy = yy/max(y);
  Y = [Y yyy];
  if yyy >= 0.8;
     [w2,b2] = solvelin2(a1,t);
     a2 = w2*a1 + b2*ones(1,q);
     MSE = mse(t-a2);
  else
      used(:,k-n)=[];
      n = n + 1;
  end
```

Appendix C: The proposed algorithm codes for RBFNN

C-1 RBFNN with GRFSWLS strategy

```
function [net,tr]=newrbfs6dddd(p,t,goal,spread,mn,df,lamb)
% NEWRB Design a radial basis network
% ------------------------------------------------------------------
% Regularised and weighted least squares forward selection (RWLSFS)
% + regularised network weights + fine-tuned GCV
%
%  Synopsis
%
%    [net,tr] = newrb(P,T,GOAL,SPREAD,MN,DF,LAMB0)
%
%  Description
%
%    Radial basis networks can be used to approximate
%    functions.  NEWRB adds neurons to the hidden
%    layer of a radial basis network until it meets
%    the specified mean squared error goal.
%
%   NEWRB(P,T,GOAL,SPREAD,MN,DF) takes these arguments,
%     P      - RxQ matrix of Q input vectors.
%     T      - SxQ matrix of Q target class vectors.
%     GOAL   - Mean squared error goal, default = 0.0.
%     SPREAD - Spread of radial basis functions, default = 1.0.
%     MN     - Maximum number of neurons, default is Q.
%     DF     - Number of neurons to add between displays, default =
25.
%   and returns a new radial basis network.
%     LAMB0  - Initial regularisation parameter value
%
% Copyright 1992-2007 The MathWorks, Inc.
% Modified by Herman Wahid, August, 2012

if nargin < 2, error('NNET:Arguments','Not enough input arguments'),
end

% Defaults
if nargin < 3, goal = 0; end
if nargin < 4, spread = 1; end
if nargin < 6, df = 25; end
if nargin < 7, lamb = 0; end

% Format
if isa(p,'cell'), p = cell2mat(p); end
if isa(t,'cell'), t = cell2mat(t); end

% Error checks
if (~isa(p,'double')) | (~isreal(p)) | (length(p) == 0)
  error('NNET:Arguments','Inputs are not a non-empty real matrix.')
end
if (~isa(t,'double')) | (~isreal(t)) | (length(t) == 0)
  error('NNET:Arguments','Targets are not a non-empty real matrix.')
end
if (size(p,2) ~= size(t,2))
  error('NNET:Arguments','Inputs and Targets have different numbers
of columns.')
```

```
end
if (~isa(goal,'double')) | ~isreal(goal) | any(size(goal) ~= 1) |
(goal < 0)
  error('NNET:Arguments','Performance goal is not a positive or zero
real value.')
end
if (~isa(spread,'double')) | ~isreal(spread) | any(size(spread) ~=
1) | (spread < 0)
  error('NNET:Arguments','Spread is not a positive or zero real
value.')
end
if (~isa(df,'double')) | ~isreal(df) | any(size(df) ~= 1) | (df < 1)
| (round(df) ~= df)
  error('NNET:Arguments','Display frequency is not a positive
integer.')
end

% More defaults
Q = size(p,2);
if nargin < 5, mn = Q; end

% More error checking
if (~isa(mn,'double')) | ~isreal(mn) | any(size(mn) ~= 1) | (mn < 1)
| (round(mn) ~= mn)
  error('NNET:Arguments','Maximum neurons is not a positive
integer.')
end


% Dimensions
R = size(p,1);
S2 = size(t,1);

% Architecture
net = network(1,2,[1;1],[1; 0],[0 0;1 0],[0 1]);

% Simulation
net.inputs{1}.size = R;
net.layers{1}.size = 0;
net.inputWeights{1,1}.weightFcn = 'dist';
net.layers{1}.netInputFcn = 'netprod';
net.layers{1}.transferFcn = 'radbas';
net.layers{2}.size = S2;
net.outputs{2}.exampleOutput = t;

% Performance
net.performFcn = 'mse';

% Design Weights and Bias Values
warn1 = warning('off','MATLAB:rankDeficientMatrix');
warn2 = warning('off','MATLAB:nearlySingularMatrix');
[w1,b1,w2,b2,tr] = designrb(p,t,goal,spread,mn,df,lamb);
warning(warn1.state,warn1.identifier);
warning(warn2.state,warn2.identifier);

net.layers{1}.size = length(b1);
net.b{1} = b1;
net.iw{1,1} = w1;
net.b{2} = b2;
```

```
net.lw{2,1} = w2;

% =======================================================
% FORWARD SELECTION (k=1)
% =======================================================
function [w1,b1,w2,b2,tr] = designrb(p,t,eg,sp,mn,df,lamb)

% THE SPREAD PARAMETER
DIST = (dist(p',p)).^2;
MDIST = mean(DIST(:));
STDIST = std(DIST(:),1)*1.1;
spmin = MDIST - STDIST;
spmax = MDIST + STDIST;
num_dig = 1;
sp = round(spmin*(10^num_dig))/(10^num_dig);

% THE INITIAL SETTING
[r,q] = size(p);
[s2,q] = size(t);
MSEALL = [];
b = 1/(2*sp);
P = radbas(dist(p',p)*b);
used = [];
I = eye(q,q);

% THE LEAST SQUARES WEIGTHING FACTORS (H MATRIX)
d = t';
h1 = var(d(:));
h2=1/h1;
%H = h2*I;
H = h2;

% RBF CENTRE SELECTION
Q1 = I;
Pt = P;
z1 = ((t*H*Q1*P).^2);
z2 = lamb*ones(1,q);
z3 = sum(Pt)*H*Q1*P;
W = z1./(z2+z3);%
[W1 IX] = sort(W,'descend');
pick = IX(1,1);             % Pick vector with maximum W1
used = [used pick];         % Used vector number for RBF centre

% THE RBF OUTPUT
w1 = p(:,used)';
a1 = radbas(dist(w1,p)*b);
a1t = a1';

% GENERALISED CROSS-VALIDATION TO FIND OPTIMAL 'LAMBDA'
lamb_min = 0.;
res = 0.1;
lamb_max = 1;
fprintf('Lambda selection, Resolution = integer number...\n');
[lamb,lamb_min,lamb_max,res] =
solvegcv(P,Pt,H,t,lamb_min,lamb_max,res); %res=1
fprintf('Lambda selection, Resolution = one decimal place...\n');
[lamb,lamb_min,lamb_max,res] =
solvegcv(P,Pt,H,t,lamb_min,lamb_max,res); %res=0.1
% fprintf('Lambda selection, Resolution = two decimal places...\n');
```

```
% [lamb,lamb_min,lamb_max,res] =
solvegcv(P,Pt,H,t,lamb_min,lamb_max,res); %res=0.01
% fprintf('Lambda selection, Resolution = three decimal places');
% [lamb,lamb_min,lamb_max,res] =
solvegcv(P,Pt,H,t,lamb_min,lamb_max,res); %res=0.001
% fprintf('Lambda selection, Resolution = four decimal places');
% [lamb,lamb_min,lamb_max,res] =
solvegcv(P,Pt,H,t,lamb_min,lamb_max,res); %res=0.0001
% if lamb == lamb_min, lamb=0;end
% LAMB = [LAMB lamb];

% THE PROJECTION MATRIX
A1 = a1*H*a1t + lamb;
A1in = A1\eye(size(A1));
Q1 = I - a1t*A1in*a1*H;
% Q1 = I - a1t*(A1\a1)*H;
% Q1 = I - (a1t/A1)*a1*H;

% THE NETWORK OUTPUT
[w2,b2] = solvelin3(a1,t,H,lamb);
a2 = w2*a1 + b2*ones(1,q);
%a2 = w2*a1;

% NETWORK PERFORMANCE
MSE = mse(t-a2);
MSEALL = [MSEALL MSE];
SSE = sse(t-a2);
tr = newtr(mn,'perf');
tr.perf(1) = mse(t-repmat(mean(t,2),1,q));
tr.perf(2) = MSE;
if isfinite(df)
  fprintf('NEWRB, neurons = 0, MSE = %g\n',tr.perf(1));
  fprintf('NEWRB, neurons = %g, MSE = %g\n',1,MSE);
end
flag_stop = 0;

% ========================================================
% FORWARD SELECTION (k>1)
% ========================================================
iterations = min(mn,q);
for k = 2:iterations

  % RBF CENTRE SELECTION
  z1 = ((t*H*Q1*P).^2);
  z2 = lamb*ones(1,q);
  z3 = sum(Pt)*H*Q1*P;
  W = z1./(z2+z3);%
  [W1 IX] = sort(W,'descend');

  for kk = 1:q
      pick = IX(1,kk);          % Pick vector with maximum W1
      if pick~=used, break, end
  end
  used = [used pick];           % Used vector number for RBF centre

  % THE RBF OUTPUT
  w1 = p(:,used)';
  a1 = radbas(dist(w1,p)*b);
  a1t = a1';
```

```
  % THE PROJECTION MATRIX
  A1 = a1*H*a1t + lamb*eye(k);
  A1in = A1\eye(size(A1));
  Q1 = I - a1t*A1in*a1*H;

  % THE NETWORK OUTPUT
  [w2,b2] = solvelin3(a1,t,H,lamb);
  a2 = w2*a1 + b2*ones(1,q);

  % NETWORK PERFORMANCE
  MSE = mse(t-a2);
  SSE = sse(t-a2);
  MSEALL = [MSEALL MSE];

  tr.perf(k+1) = MSE;

  % DISPLAY
  if isfinite(df) & (~rem(k,df))
    fprintf('NEWRB, neurons = %g, MSE = %g\n',k,MSE);
    %flag_stop=plotperf(tr,eg,'NEWRB',k);
  end

  % CHECK ERROR
  if (MSE < eg), break, end
  if (flag_stop), break, end

end

[S1,R] = size(w1);
b1 = ones(S1,1)*b;

% Finish
tr = cliptr(tr,k);
save work;

% ============================================================
function [w,b] = solvelin3(p,t,H,lamb)

%H = 1;
%lamb = 0;
pt = p';
t1 = t';
[pr,pc] = size(pt);
p1 = [pt,ones(pr,1)];
p1t = p1';
x = [H*pt,H*ones(pr,1); lamb*eye(pc),zeros(pc,1)]\[H*t1;
zeros(pc,1)];
%x = [H*p1,H*ones(pr,1); lamb*eye(pc+1)]\[H*t1; ones(pc+1,1)];
%x = (p2t*H*p2 + lamb*eye(pc+1))\(p2t*H*t1);
%x = (p1t*H*p1 + diag(lamb,pc))\(p1t*H*t1);
w = x(1:pc,:); w = w';
b = x(pc+1,:); b = b';

% ============================================================
function [lamb,lamb_min,lamb_max,res] =
solvegcv(P,Pt,H,t,lamb_min,lamb_max,res)
```

```
GCV = [];
[r,q] = size(P);
I = eye(q);
lambd = lamb_min:res:lamb_max;
for m = 1:length(lambd)
    m;
    A1 = P*H*Pt + lambd(1,m)*eye(r);
    A1in = A1\eye(size(A1));
    Q1 = I - Pt*A1in*P*H;
    Q1t = Q1';
    v1 = q*t*Q1t*H*Q1*t';
    v2 = (trace(Q1)).^2;
    gcv = v1/v2;
    GCV = [GCV gcv];
end
[pickl p_i] = min(GCV);
lamb = lambd(1,p_i)
ratio = 0.5;
res = res*0.1;
if lamb<=1, lamb_min = 0;
else lamb_min = lamb - (ratio*res*lamb*10); end
if lamb_min<0,lamb_min=0;end
lamb_max = lamb + (ratio*res*lamb*10);
%save work2;

% ========================================================
```

Appendix D: Related documents for the data collection methods

D-1 Coordinate locations of the monitoring stations in NSW, Australia

Remarks:

1. Source of document: Air quality monitoring procedural guide (Issue No. 6), Department of Environment, NSW, Australia.
2. The active monitoring stations in the Sydney region are highlighted in yellow colour.

3 MONITORING STATIONS

3.1 SITING CRITERIA

Air Monitoring stations are generally located according to guidelines issued in **Australian Standard AS 2922-1987 Ambient Air– guide for the siting of sampling units.** Siting is generally determined on a regional basis. An exception is the City (Grace Bros.) station in Table 3.2.1 which measures peak level concentrations of CO and NO_x. Tables 3.2.1 to 3.2.6 detail monitoring stations in each of the nominated regions.

3.2 LOCATION OF MONITORING SITES (2001)

Table 3.2.1 -Central East Sydney

Site Name	Site ID	Location/ Address	AMG (km)* Easting	AMG (km)* Northing	Latitude **	Longitude **	Elevation **
Earlwood	206	*Canterbury Municipal Council Land* Beaman Park, Riverview Rd.	327.57	6245.38	33° 55' 04"	151° 08' 05"	7
Chullora	190		319.25	6247.88			
Lidcombe	141	*EPA Laboratories* Joseph St	318.90	6248.76	33° 53' 09"	151° 02' 30"	40
Lindfield	70	*CSIRO National Measurement Laboratory* Bradfield Road	328.71	6260.38	33° 46' 58"	151° 09' 00"	60
Randwick	33	*Commonwealth Property Army Barracks* Cnr Avoca and Bundock Sts, Kingsford	337.51	6243.83	33° 56' 00"	151° 14' 31"	28
Rozelle	39	*Rozelle Hospital* Balmain Rd in the hospital grounds	330.03	6251.19	33° 51' 57"	151° 09' 45"	22

Table 3.2.2- Newcastle

Site Name	Site ID	Location/ Address	AMG (km)* Easting	AMG (km)* Northing	Latitude **	Longitude **	Elevation **
Newcastle	300	*Newcastle Sportsground* Dumaresq St, Hamilton South	383.92	6355.50	32° 55' 57"	151° 45' 30"	5
Wallsend	287	*Newcastle City Council swimming pool* Frances/John St	375.53	6359.43	32° 53' 46"	151° 40' 09"	8
Beresfield	322	*Francis Greenway High School* Lawson Ave, Woodberry	374.53	6370.26	32° 47' 54"	151° 39' 36"	14

Table 3.2.3-North West Sydney

Site Name	Site ID	Location/ Address	AMG (km)* Easting	AMG (km)* Northing	Latitude **	Longitude **	Elevation **
Blacktown	148						
Richmond	573	*University of Western Sydney* Richmond	290.88	6277.87	33° 37' 06"	150° 44' 45"	21
St.Marys	760	*Mamre Plains Ltd* Mamre Rd.	293.17	6258.07	33° 47' 50"	150° 45' 57"	29
Vineyard	765	*Castle Hill Sewerage Treatment Plant* Bandon Rd	300.33	6273.69	33° 39' 28"	150° 50' 48"	35

Table 3.2.4-South West Sydney

Site Name	Site ID	Location/ Address	AMG (km)* Easting	Northing	Latitude **	Longitude **	Elevation **
Bargo (mobile 1)	574	*Novastar Pyt/Ltd* 105 Silica Road	277.30	6201.08	34° 18' 27"	150° 34' 48"	365
Bringelly	171	*Liverpool City Council land* Ramsay Rd.	293.03	6244.51	33° 55' 10"	150° 45' 40"	53
Macarthur							
Campbelltown***	60	*Pilkington (Australia) Limited* Kanbyugal Res. Cronulla Cres, Campbelltown	298.08	6229.45	34° 03' 22"	150° 48' 44"	
Liverpool	107	*Liverpool City Council* Rose St	306.44	6243.31	33° 55' 58"	150° 54' 21"	22
Oakdale (mobile 2)	1570	*Linda & Garry Heise (Farm)* 30 Ridge Rd.	268.99	6229.12	34° 03' 11"	150° 29' 50"	457

Table 3.2.5-Various

Site Name	Site ID	Location/ Address	AMG (km)* Easting	Northing	Latitude **	Longitude **	Elevation **
Albury***	640	*Jelbert Park* *Cnr of Kaylock Rd and Cambourne St*	497.67	6010.31	36° 03′ 06″	146° 58′ 27″	
Bathurst	795	*Waste Water Treatment Plant*					
Tamworth***	340	*Hyman Park, Tamworth*	301.08	6556.26	31° 06′ 38″	150° 54′ 51″	
Wagga Wagga***	650	*Corner of Morgan and Murray Streets*	534.28	6113.88	35° 07′ 02″	147° 22′ 35′	

Table 3.2.6-Wollongong

Site Name	Site ID	Location/ Address	AMG (km)* Easting	Northing	Latitude **	Longitude **	Elevation **
Albion Park South	1921						
Kembla Grange	526	*Kembla Grange Race Course* Princes Hwy	299.56	6182.85	34° 28' 35"	150° 49' 03"	5
Wollongong	500	*Australian Army Depot* Gipps St	305.76	6189.39	34° 25' 07"	150° 53' 11"	15

Note: * The AMG (Australian Map Grid) co-ordinates are converted from the longitude and latitude using Redfearn Formulae
** The elevation, longitude and latitude co-ordinates were obtained using a GPS (Global Positioning System).
*** Not totally operated by Atmospheric Science Section (Part Industry operated or Council assisted sites)

D-2 Typical instrument used in the monitoring stations in NSW

Source of document: Air quality monitoring procedural guide (Issue No. 6), Department of Environment, NSW, Australia.

3.4 INSTRUMENTATION

The suite of instrumentation and equipment used in the AQMN are listed in Table 3.4.1. Each site contains a standard instrument rack generally configured as per Figure 3.4.2.

Table 3.4.1 Instrumentation Listing

Instrument	Details
Datalogger	PC based Developed in-house
Calibrator and Zero Air Supply	Manufacturer: SABIO Model: 4010 &1001
Recorder	Manufacturer: Yokogawa Model: HR1300
NOx Analyser	Manufacturer: Thermo Environmental Instruments Model: TE42C
CO Analyser	Manufacturer: Thermo Environmental Instruments Model: TE48C
SO_2 Analyser	Manufacturer: Thermo Environmental Instruments Model: TE43C
O_3 Analyser	Manufacturer: Thermo Environmental Instruments Model: TE49C
TEOM	Manufacturer: Rupprecht & Patashnick Model: Series 1400A & 1400AB
Anemometer (WSP/WDR)	Manufacturer: MetOne Model: Sonic 50.5
Modem	Manufacturer: Netcom Model: SmartModem56
Nephelometer	Manufacturer: Ecotech Model: M9003

Appendix E: Codes for RBFNN metamodel application

E-1 Metamodel codes for hourly prediction of BOL

```
% Hourly prediction of background ozone level (BOL) for 24 hour
  ahead at Blacktown
% ------------------------------------------------------------
%
% DATA: BLACKTOWN_NEW6ee = HR,NO,NO2,O3,TEMP, WSP; 7 outputs
% RBFNN setting: mse=0.014,sp=0.6(OLS) -->
                 mse=0.014,sp_initial=0.1(OLS-ASP)

clear all;
clc;
load BLACKTOWN_NEW6ee;

[pn,ps] = mapminmax(INPUT',0,1);
[tn,ts] = mapminmax(OUTPUT',0,1);
[testn,tests] = mapminmax(TESTIN',0,1);

net=newrb3b_sp3(pn,tn,0.014,0.1,100,1);     % RBFNN with adaptively
                                              tune SP
y=sim(net,testn);
yy = mapminmax('reverse',y,ts);

% 1 HOUR
obs=TESTOUT(:,1)'; est=yy(1,:); mobs = mean(obs); [nz,ntv] =
size(obs);
RMSE_1=sqrt(mse(est-obs));
MAE_1=mae(est-obs)
R2_1 = 1 - ( sum( (obs-est).^2 ) / sum( (obs-mobs).^2 ) )
d2_1 = 1-(sum((obs-est).^2)/sum((abs(obs-mobs)+abs(obs-mobs)).^2));

i=1:1:length(est);
figure;
plot(i,est,i,obs);

% 2 HOURS
obs=TESTOUT(:,2)'; est=yy(2,:); mobs = mean(obs); [nz,ntv] =
size(obs);
RMSE_2=sqrt(mse(est-obs));
MAE_2=mae(est-obs)
R2_2 = 1 - ( sum( (obs-est).^2 ) / sum( (obs-mobs).^2 ) )
d2_2 = 1-(sum((obs-est).^2)/sum((abs(obs-mobs)+abs(obs-mobs)).^2));

i=1:1:length(est);
figure;
plot(i,est,i,obs);

% 3 HOURS
obs=TESTOUT(:,3)'; est=yy(3,:); mobs = mean(obs); [nz,ntv] =
size(obs);
RMSE_3=sqrt(mse(est-obs));
MAE_3=mae(est-obs)
R2_3 = 1 - ( sum( (obs-est).^2 ) / sum( (obs-mobs).^2 ) )
d2_3 = 1-(sum((obs-est).^2)/sum((abs(obs-mobs)+abs(obs-mobs)).^2));

i=1:1:length(est);
figure;
```

```
plot(i,est,i,obs);

% 6 HOURS
obs=TESTOUT(:,4)'; est=yy(4,:); mobs = mean(obs); [nz,ntv] =
size(obs);
RMSE_6=sqrt(mse(est-obs));
MAE_6=mae(est-obs)
R2_6 = 1 - ( sum( (obs-est).^2 ) / sum( (obs-mobs).^2 ) )
d2_6 = 1-(sum((obs-est).^2)/sum((abs(obs-mobs)+abs(obs-mobs)).^2));

i=1:1:length(est);
figure;
plot(i,est,i,obs);

% 12 HOURS
obs=TESTOUT(:,5)'; est=yy(5,:); mobs = mean(obs); [nz,ntv] =
size(obs);
RMSE_12=sqrt(mse(est-obs));
MAE_12=mae(est-obs)
R2_12 = 1 - ( sum( (obs-est).^2 ) / sum( (obs-mobs).^2 ) )
d2_12 = 1-(sum((obs-est).^2)/sum((abs(obs-mobs)+abs(obs-mobs)).^2));

i=1:1:length(est);
figure;
plot(i,est,i,obs);

% 18 HOURS
obs=TESTOUT(:,6)'; est=yy(6,:); mobs = mean(obs); [nz,ntv] =
size(obs);
RMSE_18=sqrt(mse(est-obs));
MAE_18=mae(est-obs)
R2_18 = 1 - ( sum( (obs-est).^2 ) / sum( (obs-mobs).^2 ) )
d2_18 = 1-(sum((obs-est).^2)/sum((abs(obs-mobs)+abs(obs-mobs)).^2));

i=1:1:length(est);
figure;
plot(i,est,i,obs);

% 24 HOURS
obs=TESTOUT(:,7)'; est=yy(7,:); mobs = mean(obs); [nz,ntv] =
size(obs);
RMSE_24=sqrt(mse(est-obs));
MAE_24=mae(est-obs)
R2_24 = 1 - ( sum( (obs-est).^2 ) / sum( (obs-mobs).^2 ) )
d2_24 = 1-(sum((obs-est).^2)/sum((abs(obs-mobs)+abs(obs-mobs)).^2));

i=1:1:length(est);
figure;
plot(i,est,i,obs);
```

E-2 A generic modelling codes for the estimation of BOL

```
% RBFNN metamodel estimation of background ozone level (BOL):
  Generic model
% --------------------------------------------------------------------
%
% RBFNN setting: mse=0.002,sp=0.6(OLS)
% INPUTS: X,Y,NO,NO2,O3,TEMP,WSP,WDR
% OUTPUT: Background ozone level (BOL)

clear all;
clc;
load NEW_ALL2;

[pn,ps] = mapminmax(INPUT',0,1);
[tn,ts] = mapminmax(OUTPUT',0,1);
[testn,tests] = mapminmax(TESTIN',0,1);

l=min(min(pn));
net=newrb1(pn,tn,0.002,0.6,1000,1); % --> R2=0.5766(STM),n=55
%net=newrb3b_sp3(pn,tn,0.002,0.1,1000,1,1);
y=sim(net,testn);
yy = mapminmax('reverse',y,ts);

obs=TESTOUT';
est=yy;
mobs = mean(obs);
[nz,ntv] = size(obs);
RMSE=sqrt(mse(yy-TESTOUT'))
MAE=mae(yy-TESTOUT')
R2 = 1 - ( sum( (obs-est).^2 ) / sum( (obs-mobs).^2 ) )
d = 1-(sum((obs-est).^2)/sum((abs(obs-mobs)+abs(obs-mobs)).^2))
%Average=mean(yy)
%[cfun,rmse] = fit(yy-TESTOUT')

i=1:1:length(yy);
figure;
plot(i,yy',i,TESTOUT);
```

E-3 The m-codes for the spatial estimation of ozone concentration

```
function [] = Model_spatial2()
% Spatial estimation of ozone concentration
% ----------------------------------------
%
% newrbfs (FSWLS algorithm)
%% Modelling

clear all;
clc;
f=input('Simulation option: 1=Train new metamodel, 2=Use previous
modelling :');
dataname='Day2004h_NOx3_WDR_06km_domain2.mat';
%dataname='Day2004i_VOC_06km_domain2.mat';

tic;
```

```
if f==1
    disp(' ');
    disp('Modelling in process.  This may take a while, please
wait...');
    disp(' ');
    load(dataname);
    %Normalise
    [pn,ps] = mapminmax(INPUT',0,1);
    [tn,ts] = mapminmax(OUTPUT',0,1);
    [testn,tests] = mapminmax(TESTIN',0,1);

    l=min(min(pn));
    net=newrb1(pn,tn,0.004,0.1,500,1);              % OLS algorithm
    %net=newrbfs2aaa(pn,tn,0.005,0.1,400,1,1);      % FSWLS
algorithm
    % {newrbfs(input,target,goal,sp,maxsize,plotinterval,alfa)}
    y=sim(net,testn);
    yy = mapminmax('reverse',y,ts);
    yyy=yy';
    save metamodel;

elseif f==2
    load metamodel net ts;
    load(dataname);
    [testn,tests] = mapminmax(TESTIN',0,1);
    y=sim(net,testn);
    yy = mapminmax('reverse',y,ts);
    yyy=yy';
end
%save result;
%% Hourly - individual plot

figure;
actual=TESTOUT(:,1);
model=yy(1,:);
i=1:1:length(actual);
plot(i,model','--m',i,actual,'b');
legend('predicted','actual');
xlabel('Day','FontSize',12)
ylabel('Background ozone level(ppb)','FontSize',12)
%title('Predicted ozone concentration for 4 hour maximum average')

disp(' ');
disp('Performance indexes: 1 hour average');
obs=actual';
est=model;
mobs = mean(obs);
[nz,ntv] = size(obs);
RMSE_2=sqrt(mse(yy-TESTOUT'))
MAE_2=mae(yy-TESTOUT')
R2_2 = 1 - ( sum( (obs-est).^2 ) / sum( (obs-mobs).^2 ) )
d_2 = 1-(sum((obs-est).^2)/sum((abs(obs-mobs)+abs(obs-mobs)).^2))

model=model';

toc

% % Spatial plot
%
% tx = 246:2:384;
% ty = 6207:2:6345;
```

```
% [XI,YI] = meshgrid(tx,ty);
% ZII = model';              % actual' or model'
% ZIII = reshape(ZII, size(XI'));
% ZI = ZIII';
%
% % figure;
% % [C,h]=contour(XI,YI,ZI,5);
% % clabel(C,h,'LabelSpacing',300)
%
% figure;
% pcolor(XI,YI,ZI);
% %caxis([0 0.1]);
% caxis auto;
% shading('flat');
% %clabel(C,h,'LabelSpacing',400)
% colorbar;
% save result;
```