C PHD 81

PMD
88 / HAI

SLC
Ref.

MEASUREMENT UNCERTAINTY IN WATER DISTRIBUTION TELEMETRY SYSTEMS

BY

G D HAINSWORTH

A thesis submitted to the Council for National Academic Awards in partial fulfilment of the requirements for the degree of Doctor of Philosophy

COLLABORATING ESTABLISHMENT - SEVERN TRENT WATER AUTHORITY

Trent Polytechnic

July 1988

ACKNOWLEDGEMENTS

The research work which resulted in this thesis was carried out in collaboration with Severn Trent Water Authority. As well as providing sponsorship, Severn Trent were very helpful giving practical guidance and invaluable real data with which to work. In particular, I would like to thank Mr Malcolm Walker for acting as my external supervisor.

At Trent Polytechnic, Dr Andrzej Bargiela has dedicated considerable time and effort to this project and has been a patient and extremely informative supervisor. For his continual guidance I am greatly indebted. I am grateful to all of my colleagues in the Computing Department, for their help and encouragement and for providing me with a most enjoyable three years at the Poly.

ABSTRACT

Efficient control of a complex water distribution system requires accurate information about its current operating state. At present in the water industry, modern telemetry hardware systems are being installed to meet these needs. Unfortunately, due to financial constraints, it is not practical to measure all variables with such a system. Therefore, the information supplied by the telemetry system must be supplemented by pseudomeasurements, such as predictions of consumption at the nodes of the network, before a full picture of its operating state can be calculated. These pseudomeasurements are only estimates and hence, contain a great deal of uncertainty. The real meters linked to the telemetry system, although not completely accurate, provide more reliable data. Effectively, they strengthen the monitoring system by reducing the reliance on less accurate pseudomeasurements.

Measurement uncertainty clearly has an impact on the accuracy to which state estimates can be calculated. The precise nature and level of this impact is investigated in this thesis. A network model which allows for measurement uncertainty, is presented. From this model, algorithms are derived which quantify the effect of measurement uncertainty on the accuracy of the derived state estimates. Rather than a single, deterministic state estimate, the set of all feasible states, given the level of measurement uncertainty, is calculated. This set is presented in the form of upper and lower bounds for the individual variables, and hence provides limits on the potential error of each variable. A water distribution system simulation program, TCLAS, that calculates state estimates in this way, has been developed. This program is also described in the thesis.

The location of meters about the network strongly influences the accuracy of state estimates. By carefully designing the meter placement in the telemetry system, it is possible

to achieve a much higher level of monitoring accuracy. The problem of how best to design this metering - the optimal meter placement problem - is also addressed, and is presented as a mathematical optimisation problem. The mathematical formulation allows a great deal of flexibility in the choice of cost and constraint functions, so that realistic design objectives and telemetry system restrictions can be modelled. Two optimisation algorithms are presented as solutions to this problem. In addition, it is described how TCLAS can be used to experiment with the location of meters and assess the accuracy of these placements.

CONTENTS

CHAPTER 1

INTRODUCTION

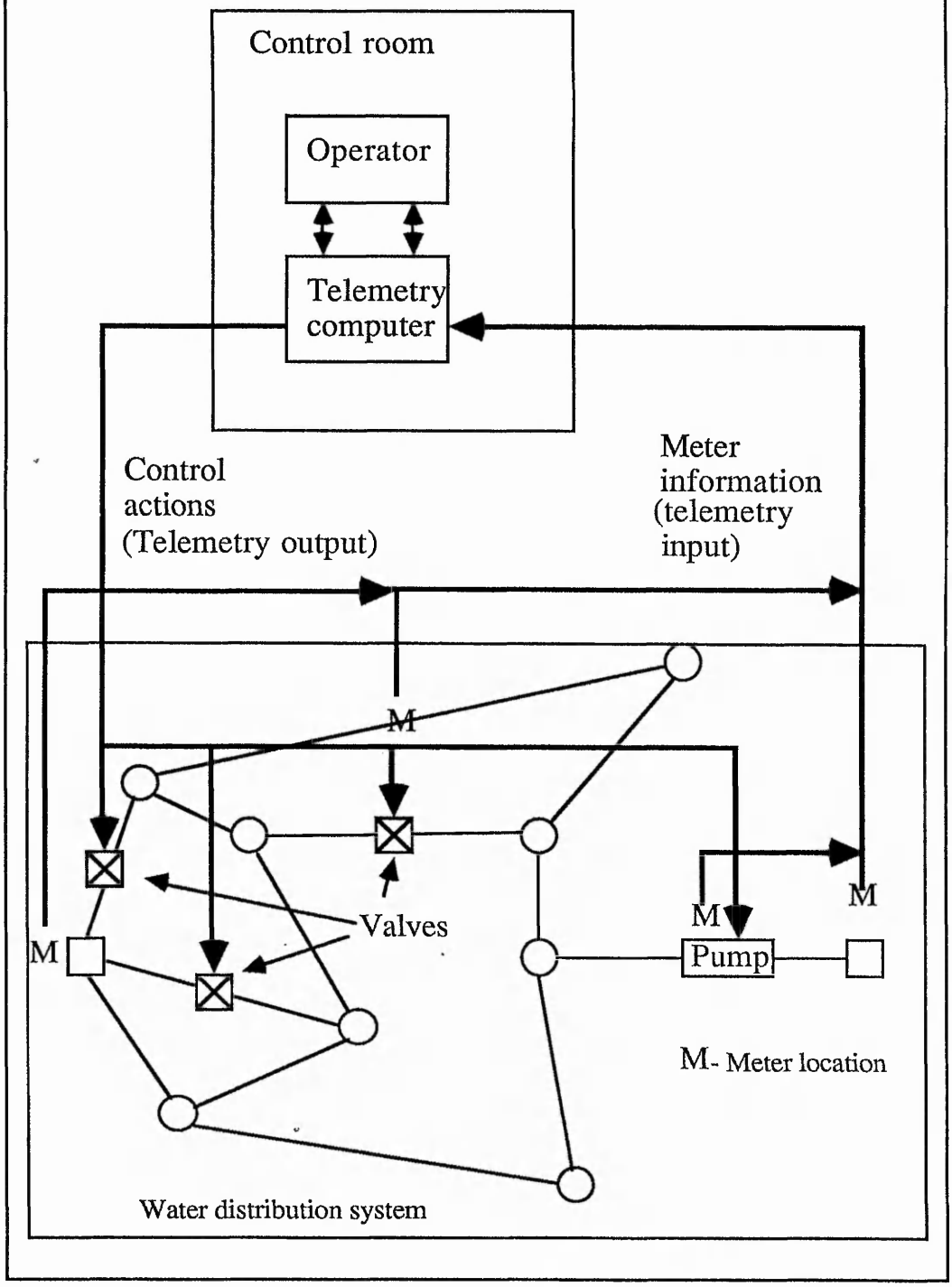
1.1. TELEMETRY SYSTEMS AND TELECONTROL

Efficient and economic control of a water distribution system requires an accurate picture of its current operating state. Pressures and flows throughout the network, pump operation, valve status, reservoir levels, inflow and water use must all be monitored. Similarly, any control strategy must be put into practice by actuating the system's control elements - pumps and valves. The size of the geographical area covered by a water network causes difficulties in gathering and transmitting this kind of information in real-time. Here a telemetry system is of great benefit in water distribution system control.

For monitoring purposes, a water network telemetry system consists of a number of flow and pressure meters placed at various points about the network, see fig 1.1. These relay their readings to a central control room. Once at the control centre, the information can be processed to provide a full picture of the state of the system. Based on this information a control strategy can be derived and messages relaid back through the telemetry system to the pumps and valves of the system so that the strategy can be implemented. The advantages of these telemetry systems are clear. Meter data is instantly available to the operator which gives details about the state of the network as it is now, rather than as it was at some time passed. The need for an engineer to travel to different parts of the network to read meters and for staff to be permanently based at remote stations is reduced.

Many water authorities have already installed limited telemetry systems with results that have been received favourably in the industry. Some of their experiences are

Fig 1.1 : Water distribution telemetry system .



documented in [24, 32, 52, 67, 82, 89, 99, 137, 147]. It is likely that the next few years will see further investment in telemetry hardware and eventually, it could become the norm for water distribution systems to be monitored in this way. Telemetry systems have been developed with the aim of facilitating fully automated control for complex distributed systems. For water distribution systems, this may not be realised in the near future but a much higher level of automation can certainly be expected.

Ideally, a water distribution telemetry system would monitor the flow in every pipe and the consumption and pressure at all nodes of the network. Unfortunately, this is not economically viable. It is not only the cost of the meter itself but its cost of installation and the cost of setting up the communication link with the control room that should be considered. Even in simplified water distribution network models, there may be several hundred potential meter sites. Realistically, only a limited number of flow and pressure meters can be installed into the telemetry system. These must be placed at key points: at reservoirs to monitor water levels, at pumps to monitor operation and at inflow points, for instance. Also a number of meters must be placed about the network itself so that accurate estimates of the system's operating state can be ensured.

1.2. SIMULATION OF WATER SYSTEMS UNDER UNCERTAINTY

The simulation of any complex engineering system will always include a degree of uncertainty. No meters can be fully accurate, no mathematical model can fully reflect the intricacies of a real system's behaviour and no engineer's knowledge is complete. The best endeavours of the modellers and engineers cannot remove these problems, all that can be hoped is that they can be reduced to an acceptable level. Water distribution systems are no exception to this rule. The lack of real meter readings, supplied in real-time by the telemetry system, means that much use is made of inaccurate pseudomeasurements such as nodal consumption predictions. This measurement uncertainty can introduce a large amount of uncertainty into the simulation of the water distribution system's operation. A

large part of the work described in this thesis is concerned with the quantification of the impact of measurement uncertainty on the estimates of the system's behaviour.

At present, there are many simulation packages that are in use within the water industry. Perhaps the most commonly used in this country are GINAS [110], produced by the Water Control Unit at Leicester polytechnic, and WATNET [133, 135], produced by the Water Research Centre. Other packages include: WASMACS [13]; SIMNET [142]; KYPIPE [148]; and NETMAP [112]. Generally, these packages are deterministic in nature. They produce an estimate for the state of the system which is intended to be the best estimate that can be obtained from the uncertain measurement and network data. None of these simulation packages quantify the effects of this uncertainty and provide explicit bounds on the potential state variable error. It is very important that the level of uncertainty present in water system state estimates can be quantified in some way. If these estimates are to be used as the basis for making control decisions it is necessary to know by how much they may be in error. Only then can reliability and safety be ensured. A state estimate without indication of its possible error only paints part of the picture. All it says is that the system is operating in some state close to this estimate. In reality the estimate is unlikely to be met exactly. An operator must know, among other things, whether desired minimum pressures are being met at each node so that demand can be satisfied, where pressures are unacceptably high and hence causing a risk of pipe rupture or leakage, reservoir levels, the status of pressure or flow controlled devices etc. An incorrect estimate for a pressure at a particular node may, for example, indicate that an automatic pressure reducing valve is closed when this valve is actually open. This incorrect information can create a completely false picture of the network's operating state. If the pressure value is presented together with an indication of its possible error or confidence limit, then this possibility becomes apparent. In short, presentation of state estimates, with confidence limits as a quantification of their uncertainty, allows for 'worst-case' control to be implemented, thus ensuring security of operation. The process of quantifying the

uncertainty in the state estimates will be referred to as **confidence limit analysis.**

Presentation of uncertainty also has implications in distribution and monitoring system design. Weaknesses in the monitoring side of telemetry systems can be identified, pointing out where extra metering will be beneficial. When an extension to the network is being designed to meet demand at some future time horizon it is no use imagining a 'typical' demand profile and planning for this. The full range of possible demands must be analysed. This will involve examining the uncertainty associated with future demand predictions, quantifying the resultant uncertainty and designing the network extension with this extra information in mind.

An important part of fully or partly automated real-time control of water distribution is the ability to identify that the system has failed in some way. This may be, for instance, a pipe burst, major leakage or a breakdown of one or more meters. The observed or measured operating state of the system will, inevitably, differ from the predicted or expected state, but does this difference suggest that the system has failed in some way. If a fault has occurred, what form does this fault take, and where in the network is the fault located? Questions such as these can only be answered if there is some knowledge of the uncertainty in the observation and in the prediction.

1.3. IMPROVING SIMULATION ACCURACY

How can the uncertainty in monitoring water distribution systems be reduced? Having accepted that some uncertainty is inevitable, this is one of the questions that must be answered. The first task is to make sure that the model of the network is up-to-date and is as accurate as possible. This will involve checking of the pipe parameters such as C-values, checking that the network is as it is modelled and also ensuring that the records of population distribution and water use are accurate. Even after model errors have been removed, there will be an amount of uncertainty in simulation due to the inaccuracy of the

monitoring system. Fortunately, by carefully designing the metering system - deciding which meters should be used, how accurate they should be and where they should be located in the network - accuracy can be greatly improved.

In 1973, an AWWA research committee sat to examine water distribution research and development needs. One of their conclusions, published in [12], was that "*criteria should be developed for optimum location of field network pressure and flow sensors for design simulation; and, perhaps separately, for adequate indication of field conditions for supervisory monitoring and for automatic control.*" Designing the metering part of a telemetry system will be referred to as **optimal meter placement.** It can be carried out at the initial design stage of the telemetry system, or when the telemetry system is to be enhanced. At some stage in the operation of a water network telemetry system it may become apparent that the results are not accurate enough. If this happens, then an optimal meter placement study can be performed to see what improvements to the telemetry system are required. Optimal meter placement must, in some way, involve confidence limit analysis studies to assess the accuracy of any potential or proposed configuration of meters in the telemetry system. It can be carried out in a partially trial-and-error way, guided by confidence limit analysis results, or can be posed as a mathematical optimisation problem.

CHAPTER 2

REVIEW OF WATER DISTRIBUTION SYSTEM RESEARCH

2.1. INTRODUCTION

In this chapter, some of the published research in the field of water distribution systems is reviewed. Attention is concentrated on those aspects of water systems theory that are of particular relevance to the work in this thesis. The review is divided into four topics: network modelling and calibration; simulation and decision support for water distribution systems; control of water distribution networks; and discussion of experiences in the application of telemetry systems in water systems. There is of course some overlap between some of these categories, but such a division is useful for treating such a wide area of research. This review is intended to be quite general, research that is more specific to the various topics considered in this thesis is reviewed in the appropriate chapters. For instance, a review of state estimation methods is presented in Chapter 3, and uncertainty quantification and optimal meter placement methods are reviewed in Chapters 4 and 5, respectively.

2.2. NETWORK MODELLING AND CALIBRATION

Network modelling is the process of creating a mathematical description of the distribution system. These systems are typically large and complex, made up of many interacting elements. Often, water authority records are incomplete or fuzzy, the precise layout of the network is not always known and it is difficult to assess the state of elements that have been hidden underground for many years. The potential uses of network models are many and varied, as are the systems themselves. These factors mean that the modelling

process is never straightforward. A considerable amount of expertise is demanded of the modellers.

Allen [11] examines the process of building a network model in great detail. Based on the results of this work, she presents a systematic modelling methodology. Three main stages in the process are identified, these are: planning; performance; and practice. The planning stage should involve discussion with experts and potential users. From these discussions, the intended use of the model and its required accuracy will be determined. The performance stage involves the collection of the required data, field testing, calibration and the construction of the model itself. A third stage, often neglected, is the practice stage. As the model is being used the system will change, new elements (pipes, pumps, valves etc) will be added and the demand pattern will alter. It is essential, therefore, that the model is kept up-to-date and recalibrated regularly. Lee [94], presents a similar methodology for network modelling, together with a case study. The issues he discusses include: records and data reviewing; skeletonisation; data input; model calibration; and the assessment of results.

The idea that a network model should be constructed with its intended use in mind is demonstrated by Hamberg and Shamir [70, 71]. They examine the requirements of models for use in the preliminary design of distribution systems. For this purpose, they suggest that simpler models, that can be analysed more efficiently, are appropriate. In the first of these papers [70], a method is presented by which groups of system elements can be combined to form an 'equivalent' individual element. For instance, pipes connected in series or parallel can be combined to form a single pipe element in the model, or a small loop of pipes can be combined to form a simpler pipe junction in the model. In the second paper [71], the distribution system is viewed as a non-linear, horizontal continuum and the concept of the 'flow-field' is developed.

Once the requirements of the model in terms of its use have been determined, the problem of model accuracy must be addressed. To be of practical use, the model must involve a degree of simplification, which, if not done carefully, can introduce inaccuracy. Eggener and Polkowski [54] examine the modelling processes of skeletonisation and load consolidation, and the impact of these in terms of model accuracy. Using a real network as a case study, they conclude that "*Any practical degree of accuracy can be attained in modelling distribution networks if enough effort is put forth to develop the input data*", but qualify this with some warnings: (i) C-values should not be taken directly from tables, but should be adjusted to fit observed data; (ii) Small pipes in the vicinity of the major sources of supply should be included in the model; (iii) There is a "*tremendous need for more basic information on loading and variations in loading*".

A similar comment to (iii) above, was made by an AWWA research committee in a report that outlines water distribution research needs [12]: "*There is an urgent need for more information on loadings such as variations at a point in time; variations of multiple points within a system with time; variations of demand for different classes of users; applicability of findings from one system when used for another (transferability within a particular climatic region); levels of probability of occurrence for extremes, total demands, and serial demand patterns; and related considerations.*"

Once a network model has been constructed, efforts can be made to refine that model to fit observed data. This process is known as calibration and is discussed by several authors [25, 91, 108, 109, 143, 144, 145, 146]. Ormsbee and Wood [109] present an explicit network model calibration algorithm. In this algorithm, the network equations are solved, for a particular observation, to yield headloss coefficients for each pipe. These are then used to alter pipe C-values or to redefine a pipe model to account for the effects of minor losses. Walski also addresses this problem. In [143], he presents formulae which can be used to calculate C-value or water use adjustment factors. Formulae can also be used to decide

whether to adjust water use or pipe function factors. He presents a case study using his methods in [146], and provides some general advice on ensuring accurate network calibration in [145]. Cesario and Davis [25] examine the calibration process, discussing its use, its data requirements, various calibration methods and identify possible sources of error. Lansey [91], and Ormsbee and Chase [108] present network calibration techniques which allow multiple loading conditions to be included. Both methods use non-linear programming methods, with an embedded state estimation routine, to solve the calibration problem. Lansey suggests a gradient non-linear programming method, whereas Ormsbee and Chase use a modified constrained optimisation technique. Coulbeck [34], also describes a method based on optimisation techniques. In this, the discrepancy between observed and calculated flows and pressures is minimised by adjusting the calibration parameters which may include: pipe parameters; pump parameters; nodal consumptions; reservoir flows; and nodal pressures. Clarke et al have published a paper [29] which is related to the topic of model calibration. In this paper, a formula is presented which quantifies the sensitivity of the pressure difference across a pipe which results from the uncertainty of other pipe parameters (diameter, length, discharge, hydraulic gradient and C-values, for instance). From this formula, the accuracy of the pressure meters needed to provide meaningful results is derived.

2.3. SIMULATION AND DECISION SUPPORT

Simulation of the operation of water distribution systems involves the prediction or estimation of its behaviour in response to specified future or current conditions. Simulation may be performed: (i) for design purposes - to assess how effectively a proposed network or system extension will meet its specifications; (ii) In operational planning - to assess whether a particular set of control actions will meet expected demand economically and safely; or (iii) in decision support - providing system operators with information about the current state of the system, upon which decisions about necessary action can be based.

Computers have been used for many years to simulate the behaviour of water distribution systems. They provide the power to perform the many numerical calculations that are required in processing input data and solving the network model. There are several commercially available software packages for analysis and simulation. Perhaps the most comprehensive and widely used in this country are WATNET [133] and GINAS [110]. These two packages incorporate graphic displays, allow both static-state and extended period simulation, and include many different water system elements in their models. Other packages include: NETMAP [112], WASMACS [13], SIMNET [142], KYPIPE [148] and TCLAS [16].

At the core of all computer simulation packages is a state estimation routine. It is this that calculates the state of the system for a given set of inputs. Water systems literature contains many state estimation algorithms [3, 13, 27, 31, 38, 42, 43, 61, 66, 78, 84, 92, 93, 107, 116, 121, 127, 130, 134, 135, 149, 150, 151]. These are reviewed in more detail in the next chapter. Wood and Rayes examine the performance of some of these, their conclusions are reported in [150]. State estimation and network analysis are by nature mathematically intensive processes, and as a result, published work in this topic tends to be very technical. Shamir and Howard, recognising this, present a non-mathematical explanation of water distribution system simulation [128]. Collins [30] discusses some of the problem that can be encountered when using the state estimation routines, these include: non-convergence in the iteration; lack of accuracy of solution; and the lack of a unique solution in some situations. Most of the state estimation techniques described in research papers can be classified as static-state estimators. That is, they provide an estimate of the operating state of the network at a particular instant in time or for a particular set of operating conditions. In [116, 117], Rao et al describe a method for extended period simulation of water systems. This method relies on a series of static-state solutions that are linked by the dynamic aspects of the system. The dynamic linking requires a model of the rate at which reservoirs fill and empty, knowledge of demand variation over the period of

simulation and a schedule of operational changes such as pump and valve switching. Extended period simulation is included in the software packages WATNET, GINAS, KYPIPE and SIMNET.

Recent developments in monitoring system hardware and the availability of cheaper processing power mean that concepts of the role of simulation within the water industry are changing, with operators looking to a more integrated approach. The impact of these developments within water systems management is discussed in [39, 40, 81, 90]. Johnson [81] examines the requirements of decision support systems and knowledge based techniques in water systems management, including a list of example application areas. Cunningham and Amend [39, 40] highlight the importance of interactive simulation, particularly in operator training.

2.4. CONTROL OF WATER DISTRIBUTION SYSTEMS

In [126], Shamir describes the control problem for water distribution systems as "*how to operate the pumps and valves so as to minimise the total cost, while meeting demands and satisfying minimum pressure constraints*". This statement of the problem summerises the principal features involved but other factors must be taken into account, for example: controlling leakage; ensuring that the system elements (pumps and valves, etc) are operating within their safety limits and are not put under undesirable strain; and control under emergency or failure conditions.

Since pumping makes a major contribution to the cost of network control, effort has been directed towards minimisation of pumping costs [33, 35, 36, 37, 56, 82, 99]. Coulbeck [33, 35], and Coulbeck and Sterling [36] examine the problem of optimal pump control. Methods are presented by which pumping costs can be minimised in a restricted class of system. These methods include a dynamic programming and a hierarchical technique. They are extended to cater for more complex systems by simplifying the system

model. Fallside and Perry [56] also describe a hierarchical optimisation technique. This was devised for a specific network - East Worcestershire supply network - and its application to this system is described. Jowitt et al [82] describe an automatic pump scheduling scheme that is to be applied to a real network serving a population of 300,000. The optimiser is to be linked directly to this system's telemetry facilities. The pump optimisation procedure uses a linear programming technique with the constraint on satisfying demand replaced by constraints on reservoir mass balance and bounds on reservoir storage. The problems involved in optimal pump scheduling for water distribution systems are discussed by Creasey in [37]. He identifies size, dimensionality and non-linearity as the principal problems to be overcome in mathematical pump optimisation procedures. Also discussed is the problem of applying pump scheduling to real water networks. Shamir [126] reviews the present state of real-time control of water distribution systems, with a particular emphasis on pump control. In this paper, the hardware requirements of such real-time control systems and their application are also discussed.

Another area of water network control that has been considered by researchers is leakage control [99, 132]. It is estimated that a significant proportion of the water in distribution networks is lost through leakage (20% [99] or 25% [132]). This loss can be important in areas where water resources are scarce and, of course, relates to a financial loss in the processing and pumping of clean water. As high pressure in the network can cause leakage, the leakage reduction techniques concentrate on maintaining an optimal pressure profile. Pressures must be high enough to satisfy demand but not too high so as to cause leakage. In [132], Sterling and Bargiela use a sparse revised Simplex method to determine the valve settings that would maintain an optimal pressure profile. Miyaoka and Funabashi [99] apply a method from network flow theory in a two-level scheme.

The inherent uncertainty in water system monitoring and the mathematical complexity of algorithmic control algorithms means that expert system based control procedures may

be beneficial in some situations. This idea is being investigated by the Water Industry Expert System Club (WIESC) project. Unfortunately, there is little published work on this project. One report, [2], describes the WADNES project, being carried out at Surrey University. WADNES is intended to supervise the control of a water distribution in emergency or failure situations.

2.5. EXPERIENCES IN THE APPLICATION OF TELEMETRY SYSTEMS

The Melbourne and Metropolitan Board of Works have implemented a comprehensive monitoring and control system for its Melbourne water supply network [32]. This system incorporates telemetry facilities, network modelling and interactive computer control. Introduction of this system has resulted in more efficient control and increased understanding of the behaviour of the complex network. This paper describes in detail all aspects of this control system and its operation, including: the telemetry data; input of the demand data; the network simulation and modelling; and the graphic displays. In [147], Williams describes the design of telemetry system to monitor and control water distribution on Hong Kong Island. This discussion concentrates on the hardware of the system and its requirements. Thames Water Authority are installing a real-time pump scheduling system at their Bourne End control centre. To work efficiently, this system will require up-to-date information about water usage. The telemetry system that will supply this is described in [82].

In America, telemetry systems are being used in other areas of water resource management. For instance: the Yakima River remote control system [24]; the Central Arizona aqueduct system [67]; and the Windy Gap diversion dam [52]. In [89], Labadie discusses the architecture and operation of real-time control systems in water resources management and examines a selection of real systems as case studies.

CHAPTER 3

STATE ESTIMATION AND THE NETWORK MODEL

3.1. INTRODUCTION

Before effective computer analysis of a water distribution system can take place, there must be an accurate description of that system. Such a description is called the network model. It must include all of the key elements that make up the system, and must be flexible enough to be of use in a large number of different situations. Water distribution system modelling is discussed in the early sections of this chapter

At any particular instant in time, the system will be in a certain operating state. That is to say, each pipe will have a particular flow rate, each node will be under a particular pressure and water will be consumed or supplied at a particular rate at each of the nodes. State estimation is the process that attempts to calculate explicitly the system's operating state at a particular instant in time. The aim is to calculate the value of the flow in each pipe or the pressure at node. From a state estimate the values of all other variables of interest can be calculated. The process of state estimation in water distribution systems is discussed in section 3.3.

3.2. THE NETWORK MODEL

All water networks are unique but have certain similarities which can be utilised in developing a common model structure. Their purpose is to transport water from one or more sources to consumers distributed over a certain area. They are made up of a collection of pipes, pumps and valves, interconnected to form a network. Other elements commonly found in a water distribution system are: reservoirs; water storage tanks; boreholes; etc. All

of these different elements must be included in the model. Water authorities have records of the networks under their management, it is on these that the modelling is based.

Many authors have addressed the problem of water distribution system modelling [3, 11, 27, 54, 70, 71, 84, 92, 94], most suggesting a network theoretical approach. A water distribution system can be represented as a network consisting of an interconnected set of links and nodes. The links representing the pipes, pumps or valves and the nodes representing the junctions between these links, input (source, reservoir, storage tank etc) points or output (consumption) points. See fig 3.1, which shows the network structure of an example water distribution system. The schematic diagram in fig 3.2 only shows how the various elements of the water distribution system are interconnected, ie the network's connectivity or topology. Before this model can be used there must be some additional data describing the elements themselves. For each of the nodes, information is needed about the population it supplies and the consumption type. Also, an elevation is required for each node. For each pipe, its length, diameter and its roughness must be known and operational parameters of the pumps and valves must also be known. All of this information is basic to the network, when the network model is being used in any way, more information is likely to be required, static state operation data for instance.

A typical water distribution system may serve many thousands of consumers and may consist of many more pipe, pump and valve elements. Even with the computing power that is available today, modelling on this scale is impractical for most applications. Even if this were feasible, it would not be possible to attain sufficient accuracy in the modelling to produce meaningful results. Therefore, modelling on a more manageable scale is required. This can be achieved by lumping together groups of consumers represented by a single node and considering only the most important links. Care must be taken when reducing the network in this way so as not to reduce the potential model accuracy [54].

Fig 3.1 : Example water distribution network.

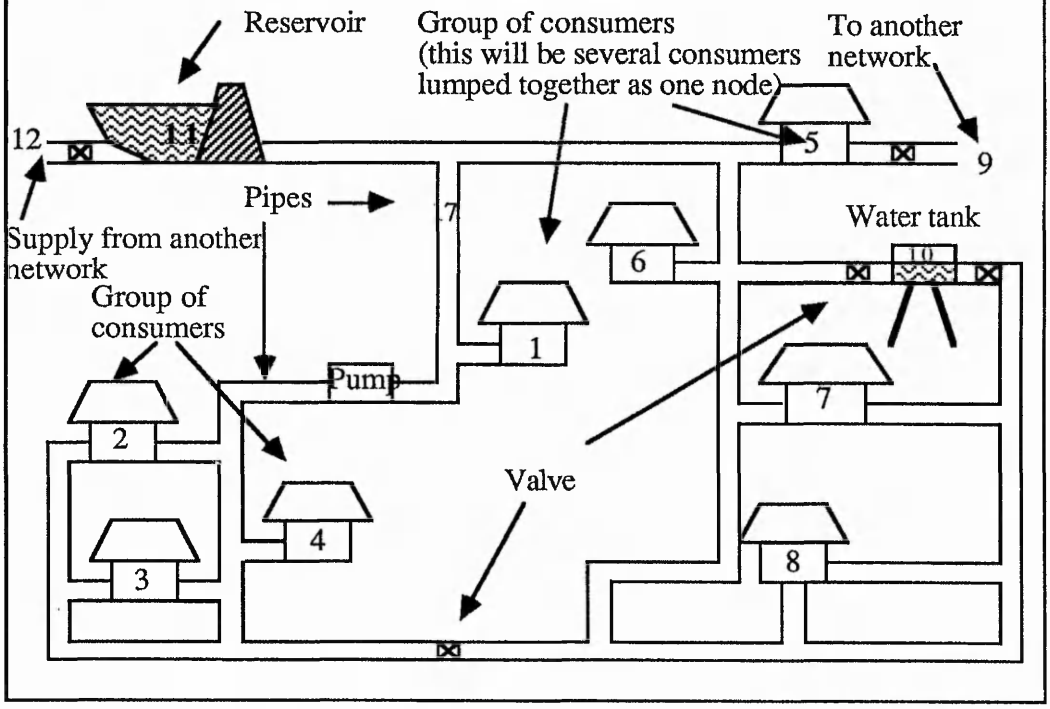
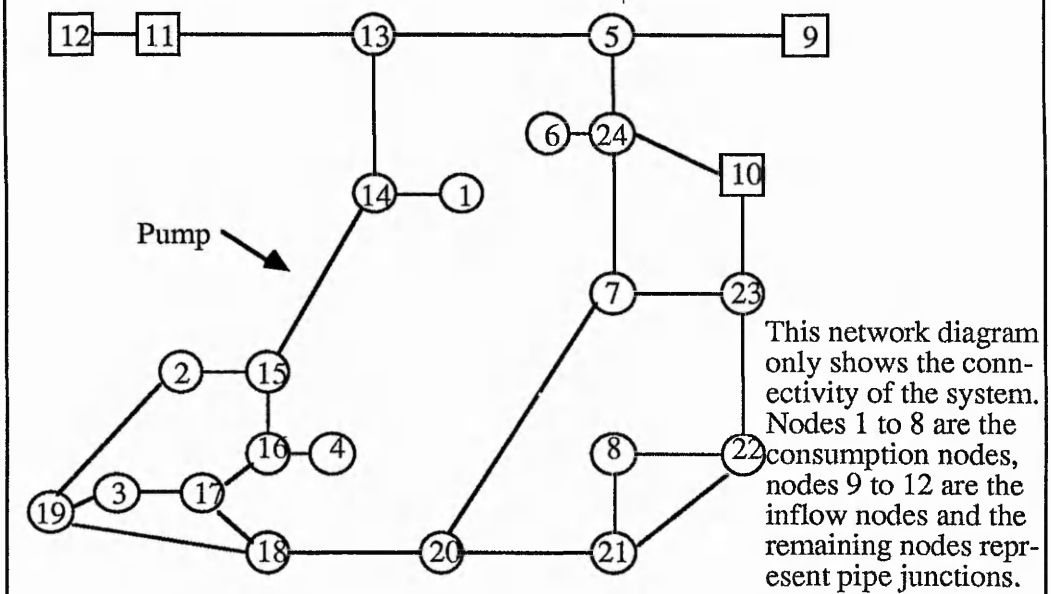


Fig 3.2 : Network schematic diagram for fig 3.1.



This network diagram only shows the connectivity of the system. Nodes 1 to 8 are the consumption nodes, nodes 9 to 12 are the inflow nodes and the remaining nodes represent pipe junctions.

Derivation of the water distribution system model in terms of network theory has several advantages. It provides a general framework within the terms of which, each particular network can be described. Ideas and experiences of engineers and operators working on different systems can be communicated freely. A well-defined modelling structure allows simulation algorithms and computer software to be designed for general rather than specific systems, hence providing a high level of portability. Using mathematical networks to represent water distribution systems means that water systems researchers can benefit from the large amount of existing work in mathematical network theory.

3.2.1. Hydraulic relationships for water networks

Each of the different types of link in the network - pipes, pumps and valves - can be modelled mathematically. Equations can be derived which relate water flow rate to pressure drop and the various element's parameters. The most common element in a water network is a straightforward pipe section. The Hazen-Williams [111, 144] equation:

$$q_{i,j} = 0.27746\, C_{i,j}\, l_{i,j}^{-0.54}\, d_{i,j}^{2.63}\, dp_{i,j}^{0.54} \tag{3.1}$$

relates the flow rate in the pipe section between nodes i and j, $q_{i,j}$, to its diameter, $d_{i,j}$, its length, $l_{i,j}$, and the pressure drop across it, $dp_{i,j}$. The parameter $C_{i,j}$ in this equation is the Hazen-Williams coefficient, or C-value, for the particular pipe. The C-value of a pipe can be regarded as a parameter of roughness, depending on factors such as the pipes material, age, and state of repair. Tables giving typical C-values can be found in [111, 144] but these are generally well known to a water engineer. Equation (3.1) is alternatively called the pipe's head-flow relationship. In this, the direction of flow is from high to low pressure. There are other equations that can be used in place of (3.1), for example the Darcy-Weisbach or the Colebrook-White equations. The relative merits of these and other formulations are discussed in [111].

Other hydraulic elements that may be included in a water network include: fixed-speed pumps; variable-speed pumps and valves of various types. For each of these a hydraulic relationship, similar to the one for a pipe shown in (3.1), can be derived [13, 26, 121, 133, 144].

3.2.2. Model accuracy

The network model, by definition, is only a representation of the real distribution system. Every effort must be made to ensure that it is an accurate representation. Good results cannot be expected from a bad model. There are many areas where inaccuracy can creep in, some of the most important of these are now discussed.

Generally, it is impractical to model every pipe and consumer in a water distribution system - there are far too many. The network must be reduced to a manageable size with its key elements identified. Many small pipes will have to be neglected and some will be combined with others with necessary adjustments to their parameters. Consumers will often be lumped together and treated as a whole rather than individually. This network skeletonisation, when done carefully, will result in an accurate representation of the real network with a reassuring resemblance to the original system. In [54], Eggener and Polkowski examine the impact of some of the common simplifying assumptions. In this paper it is emphasised that skeletonisation is not simply a matter of disregarding all pipes of less than a certain diameter and grouping consumers arbitrarily.

Many authors point to assumed C-values as an important source of model inaccuracy [12, 25, 54, 91, 108, 109, 143, 144, 145, 146]. Tables are often used which give typical C-values for pipes of a certain age and type. While being valuable indicators of C-values, these should not be taken without question. Rarely can a pipe in a network model be considered as typical. A pipe may be exceptionally encrusted or corroded, which will affect its C-value. Minor losses, due to factors such as pipe bends or junctions with smaller (unmodelled) pipes, must also be taken into account. The C-values given in the literature

are for a specific flow rate and so for periods of exceptional flow the assumed values may not be very reliable.

There are many other sources of error in the network model: node elevations, consumer population and demand patterns for instance. When the operation of the network is being analysed, the random variation of water use at nodes throughout the system is a particularly important source of error. This problem is examined in more detail in a later chapter.

Model calibration is a prerequisite to network simulation. At all stages of model development, extensive field tests should be made to ensure that the model is an accurate representation of the real network. Calibration methods, with a particular emphasis on the tuning of C-values, are discussed by several authors in [25, 34, 91, 108, 109, 143, 144, 145, 146]. A water distribution system cannot be regarded as static and unchanging. New pipes are continually being laid to meet new demand, pipes are aging and corroding, the population distribution and flow distributions are ever changing. These changes mean that the network model should be updated and recalibrated at regular intervals.

3.3. STATE ESTIMATION

The pressures and flows in a water distribution system must obey certain physical laws. In addition to the head-flow relationships mentioned in section 3.2.1, are the Flow Conservation Law and the Loop Head Loss Law:

FLOW CONSERVATION LAW: The total flow entering a node must be equal to the total flow leaving that node.

LOOP HEAD LOSS LAW: The sum of the pressure drop around a loop of the network must be equal to zero.

In the first of these, the Flow Conservation Law, the amount of water leaving the node

includes consumer demand and leakage as well as flow through pipes. The second - the Loop Head Loss Law - must also take account of the energy input to the loop or stored within the loop.

The state estimation process is based on a mathematical network model of the water distribution system such as the one discussed in section 3.2. The physical laws governing the system - Flow Conservation Law and Loop Head Loss Law - can be combined with the hydraulic relationships of each element of the system, described in 3.2.1, to construct a set of network equations. These network equations relate either, the network's nodal pressures or the network's flows, to measurement or pseudomeasurement values and are expressed by the following equation:

$$g(\mathbf{x}) = \mathbf{z} \tag{3.2}$$

Here $\mathbf{x}$ is a vector of n state variables, called the **state vector**, which can be either nodal pressures or flows and $\mathbf{z}$ is the **measurement vector** which consists of real measurement values and pseudomeasurements such as predictions of nodal consumption. Equation (3.2) is referred to as the **network equation**. $g(.)$ is the **network function**, which includes information about the connectivity of the network and the parameters of the pipes, pumps and valves. It is a non-linear equation, which means that a direct solution is not possible, instead an iterative solution technique must be used. Solution of (3.2) - calculating $\mathbf{x}$ for a given $\mathbf{z}$ - is called **state estimation**.

The state vector is made up of n independent state variables. It may include nodal pressure variables or flow variables, but must be sufficient to completely specify the operating state of the system. When this is the case, any other system variable can be calculated directly from $\mathbf{x}$. A secondary state, $\mathbf{y}$, is now defined. This vector can consist of any variable that is of interest to the system operator or engineers, suppose that there are N of these. To distinguish it from the state vector $\mathbf{x}$, the vector $\mathbf{y}$ will be referred to as the **derived state vector**. It is introduced to demonstrate that the methods described later in this thesis are not specific to one set of independent state variables, and also to allow an explanation of how these methods can be extended to cover all variables of interest. $\mathbf{y}$ can be calculated directly from the state vector $\mathbf{x}$, with

$$\mathbf{y} = f(\mathbf{x}) \tag{3.3}$$

where $f(.)$ is the **derived state function.**

For on-line state estimation, linked into a telemetry system, the data in the measurement vector comes form two main sources. Firstly, there is the data supplied by the real meters placed in the network, this is referred to as real measurement data. Secondly, there are estimates and predictions of some of the other, unmeasured variables, which are called pseudomeasurements. A particularly common pseudomeasurement is the prediction of the water use at a particular node of the network. In most cases real measurement data is more reliable and much more accurate than pseudomeasurement data but by itself is not sufficient to make the system observable. In other words, it does not contain sufficient information to allow all of the state variables to be calculated. For a precise definition of observability, see [14, 57, 88].

There are many different state estimation techniques, some of which are reviewed in the next section of this chapter. Three particular methods have been selected, variations of which have been implemented as computer programs. These are described in the remainder of the chapter.

3.3.1. Review of state estimation methods

Algorithms for state estimation have been in use since the 1930's when Cross published details of the methods he used for calculating the flow in a pipe network [38]. These methods required much hand calculation but could be applied efficiently to smaller networks and were widely used for many years. In the late 60's and 70's researchers began to look again at this problem [31, 33, 43, 61, 78, 92, 93, 116, 117, 125, 127, 128, 149, 151]. This renewed interest in state estimation coincided with the advent of computers. State estimation is a numerically intensive process, so the benefits of modern computers were quickly recognised. More recently, state estimation methods have been refined.

Exploitation of the special properties of the problem has meant that programs can now run in seconds or fractions of seconds. Improvements have also been made to the accuracy of state estimation programs and their portability to ever smaller machines.

At the present time there are very many different state estimation methods reported in the literature, some of which have been incorporated in comprehensive water network computer simulation packages [13, 110, 112, 133, 135, 142, 148]. Although many of these methods are different in some way, two main types can be classified, these are the node equation formulation and the loop equation formulation. In the first group, the network equation (3.2) is derived form the Flow Conservation Law and the measurements and pseudomeasurements are expressed in terms of the unknown nodal pressures [13, 31, 43, 77, 78, 92, 93, 127, 130, 151]. In the second formulation the unknowns are the pipe flows. To generate the network equation in this case, the Loop Head Loss Law is used as well as the Flow Conservation Law [38, 42, 61, 66, 107, 121, 149, 150]. A formulation which combines the loop and node equations is presented in [134, 135]. Rayes and Wood, in [150], give a comprehensive review of the different types of state estimator.

Many of the methods are based on an iterative procedure of some kind where an initial estimate for the state is repeatedly refined until it satisfies equation (3.2) to within some specified accuracy. Iterative, rather than direct methods, are used because in each of these two formulations - the node equation or the loop equation - the function $g(.)$ consists of a set of simultaneous non-linear equations.

In the loop equation formulation a set of fundamental loops must be identified. There are $p-n+1$ of these, where p is the number of links in the network and n is the number of nodes [63, 150]. There are three common ways of solving these loop equations iteratively. In the first, an adjustment factor for the flow in each of the fundamental loops in turn is calculated, as in the Hardy Cross method. In a second method, $p-n+1$ adjustment factors - one for each of the fundamental loops - are calculated simultaneously. Both of these

methods require an initial estimate for the flows that balance the network, that is a set of flows which satisfy the Flow Conservation Law. In the third method, the two laws are used together so that the measurement values can be expressed directly in terms of the unknown flows. At each iteration in this method, the non-linear equations (the ones resulting from the Loop Head Loss Law) are linearised and a set individual correction values, one for each flow variable, is calculated by solving the set of linearised equations.

The node equation formulation of state estimation is described in the next section.

3.3.2. Node equation format for state estimation

After consideration of many different state estimation methods three hybrid algorithms have been developed. Each of these is designed to cater for a different situation. All three solve the node equations, ie they produce an estimate of the head at each node. From such a state estimate the flows in the network can easily be derived as a secondary calculation, as is shown in (3.3). These algorithms, which have been implemented as computer programs: MINSTEST; ODSTEST and LAVSTEST, are described in detail in later sections. Before this is done the node equation format of state estimation is described in more detail.

Equation (3.2), the network equation, has been described in general terms. In the context of the node equations a more specific description can be given. In this case there are $n-f$ pressure variables - one for each node of the network - and f inflow variables - one for each of the inflow points. n is used to denote the number of independent state variables and therefore $\mathbf{x}$, the state vector, is has n elements. The measurement vector, $\mathbf{z}$, represents a set of m measurements and pseudomeasurements which make the system observable [14, 57, 88]. In view of the observability of the measurement set, one condition on m is that it can be no less than n. The network function, $g(.)$, in reality consists of m simultaneous equations, $g_i(\mathbf{x}) = z_i$, $i=1,...,m$, each one relating one of the measurement or

pseudomeasurement values to a small number of the state variables. These can be divided into four categories depending on the type of measurement (or pseudomeasurement). These are:

(i) Pressure measurements. These are quite straightforward to deal with as they measure directly one of the $n-f$ pressure variables in $\mathbf{x}$. If the i^{th} measurement in $\mathbf{z}$ corresponds to the j^{th} pressure variable in $\mathbf{x}$, then $g_i(\mathbf{x})$ is given by

$$x_j = z_i \tag{3.4}$$

There must be at least one pressure measurement or pseudomeasurement to act as a reference pressure in the state estimation. Pressure measurements may be obtained from pressure meters placed at various nodes in the network or from meters monitoring reservoir levels, for instance.

(ii) Inflow measurements: Again these are quite straightforward as each one directly measures one of the f inflow variables. If the i^{th} measurement corresponds to the j^{th} variable (which, of course, must be one of the inflow variables), then $g_i(\mathbf{x})$ is given by

$$x_j = z_i \tag{3.5}$$

There must be f inflow measurements or pseudomeasurements, one for each of the inflow variables.

(iii) Flow measurements. There may be some flow meters placed in the network, measuring the flow through a particular pipe or pump. If the i^{th} measurement is a flow measurement, then $g_i(\mathbf{x})$ is given by

$$q_{j,k}(x_j,x_k) = z_i \tag{3.6}$$

Here, x_j and x_k are the pressure variables for the nodes at each end of the pipe or pump in the network model and $q_{j,k}(.,.)$ is the hydraulic relationship for this element, as discussed in section 3.2. For example, if the flow through a pipe is being measured, then $q_{j,k}(.,.)$ will

be the Hazen-Williams equation given in equation (3.1).

(iv) Load measurements. The load or consumption at any node in the network can be expressed in terms of the pressures at this and adjacent nodes using the hydraulic head-loss equations of section 3.2. The Flow Conservation law means that the consumption (including leakage) at node 1 in figure 3.3 is equal to the sum of flows from 2 to 1, 3 to 1 and 4 to 1 plus the inflow at this node if the node is an inflow node. This merely says that the consumption at any node must be met by the flow towards that node in the adjacent pipes. More generally, if the i^{th} measurement is a load measurement or pseudomeasurement for a node, then $g_i(\mathbf{x})$ is given by

$$\sum_{j\in\Omega} q_{j,l}(x_j,x_l) + x_k = z_i \tag{3.7}$$

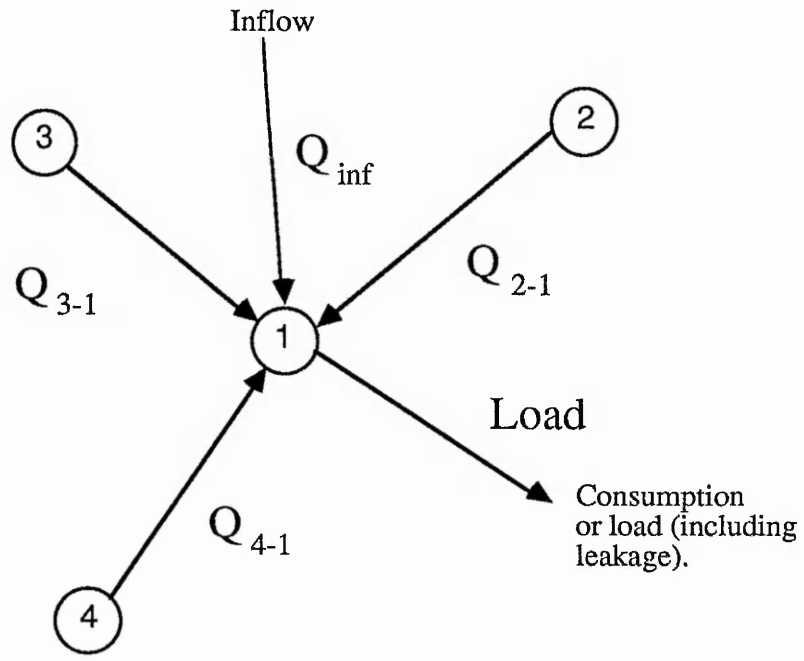
where x_l is the pressure variable for the load node, Ω is the set of pressure variables for the adjacent nodes, x_j for $j\in\Omega$ are the pressure variables for the adjacent nodes, $q_{j,l}(x_j,x_l)$ for $j\in\Omega$ are the head-loss relationships for the adjacent pipes and x_k is the inflow variable for the node (if one exists). If the node is not a consumption node, for instance it may just represent the junction between pipes, then z_i will be zero and equation (3.7) will still hold. This type of measurement or pseudomeasurement usually makes up the bulk of the measurement data. Generally, the loads at each node are not directly measured. In these situations the nodal consumptions must be predicted. So load data are usually pseudomeasurements coming from nodal consumption predictions.

Considering the different measurement types in this way allows for the network equation of (3.2) to be presented in more detail. (3.2) can be replaced by the set of simultaneous equations

$$x_j = z_i, \quad \text{for } i=1,\ldots,m_h$$

$$x_j = z_i, \quad \text{for } i=m_h+1,\ldots,m_h+m_q$$

Fig 3.3 : Flow conservation law for load measurement .



$$Load = Q_{2-1} + Q_{3-1} + Q_{4-1} + Q_{inf}$$

$$q_{j,k}(x_j,x_k) = z_i, \quad \text{for } i=m_h+m_q+1, \ldots, m_h+m_q+m_f$$

$$\sum_{j\in\Omega} q_{j,l}(x_j,x_l) + x_k = z_i, \quad \text{for } i=m_h+m_q+m_f+1, \ldots, m_h+m_q+m_f+m_l \tag{3.8}$$

where m_h, m_q, m_f and m_l are the numbers of pressure, inflow, flow and load measurements and pseudomeasurements, respectively.

Because the second two types of equation, corresponding to the flow and load measurements, are non-linear, a direct solution of (3.8) is not practical. Generally an iterative procedure such as the Newton-Raphson method is used. An initial estimate $\mathbf{x}^0$ is assumed. This is refined in successive iterations until, after a few iterations, an estimate $\hat{\mathbf{x}}$ is found which satisfies equation (3.8) (or equation (3.2)) almost exactly. The iterative procedure requires at the k^{th} step, a linearisation of the function $g(.)$ around the current estimate $\mathbf{x}^k$. This takes the form

$$g(\mathbf{x}^k) + J.\Delta\mathbf{x} = \mathbf{z} \tag{3.9}$$

where k is the index of iterations, $\mathbf{x}^k$ is the current estimate for the state vector, $g(\mathbf{x}^k)$ is the network function evaluated at the current estimate, J is the m by n Jacobian matrix of the current estimate, $\Delta\mathbf{x}$ is the correction vector and $\mathbf{z}$ is the measurement vector. The Jacobian matrix J represents the derivative of the network function $g(.)$ in its linearisation around $\mathbf{x}^k$. This matrix is discussed in more detail in [13]. It is important to note that J shows the same structure as the network equation $g(.)$, ie the $(i,j)^{th}$ element of J is non-zero if and only if measurement i is dependent on variable j in $g(.)$. Equation (3.9) represents a set of linear equations. It must be solved to find the correction vector $\Delta\mathbf{x}$, which can be done using matrix factorisation or linear programming techniques. When the correction vector $\Delta\mathbf{x}$ has been calculated, a new estimate $\mathbf{x}^{k+1}$ is given by

$$\mathbf{x}^{k+1} = \mathbf{x}^k + \Delta\mathbf{x} \tag{3.10}$$

If $\mathbf{x}^{k+1}$ satisfies equation (3.2) to within a predefined convergence accuracy, the iteration procedure stops. Otherwise, a new correction vector is calculated using equation (3.9) with

x^{k+1} instead of x^k. The process repeats until convergence is reached.

All three of the programs - MINSTEST, ODSTEST and LAVSTEST - are based on this formulation of the state estimation problem. The principal difference between them is the way in which they solve equation (3.9), which reflects their intended use.

3.3.3. MINSTEST

The routine MINSTEST produces an estimate for the exact state of a water distribution system using a 'minimal' measurement set. A 'minimal' measurement set is one which has as many elements as there are independent variables in the system and at the same time makes the system observable. In other words, this is the smallest collection of measurements with which it is possible to calculate the state estimate. It usually consists of the consumptions at all but one node, the inflows at each of the inflow points and the pressure at one node that can be used as a reference. For a minimal measurement set, the Jacobian matrix is square and non-singular and so solution of (3.9) presents no problems. The unknown correction vector can be calculated exactly by solving the set of simultaneous linear equations represented by (3.9)

As J is a derivative of the network equation of (3.8), the number of non-zero entries in each of its rows is very small (1 entry for pressure and inflow measurements, 2 entries for each flow measurement and $r+1$ entries for each load pseudomeasurement where r is the number of nodes adjacent to the load node). In other words, J is sparse as it has a large proportion of zero entries. This means that a special sparsity exploiting matrix factorisation method can be used. See Appendix A2 for a more detailed discussion of sparsity and sparsity exploiting methods. The precise method used is a sparse variant of Gaussian elimination which is due to Duff [45]. Special matrix and vector storage schemes have been used which take advantage of the Jacobian's sparse structure. These methods, at the same time, decrease storage requirements, improve numerical stability and increase

computational speed.

Before adding the correction vector to the current estimate each element is scaled. The scaling factor is different for each element and depends on previous correction vectors. For a particular element, i, of the correction vector a scaling factor, α_i, is calculated as follows:

$$\alpha_i = \begin{cases} 1.0 \text{ if } \Delta x_i \text{ and } \Delta x'_i \textit{ have the same sign} \\ 0.5 \text{ if } \Delta x_i \text{ and } \Delta x'_i \textit{ have opposite signs and } |\Delta x_i| > |\Delta x'_i| \\ 1.0 - 0.5(\Delta x_i / \Delta x'_i) \quad \textit{otherwise} \end{cases} \tag{3.11}$$

where $\Delta \mathbf{x}'$ is the previous correction vector and $\Delta \mathbf{x}$ is the current correction vector. This procedure is designed to reduce the number of iterations required in solution. If the corresponding elements of two successive correction vectors have opposite signs, then this points to oscillation in the convergence of this element. When such situations are identified, the scaling factor of (3.11) has the effect of damping out the oscillation

MINSTEST checks for convergence by testing how large the elements of the correction vector are at each iteration as well as calculating the values the current estimate would give for the measurements and comparing these against the observed measurement values.

This program is intended for use when speed of execution is an important factor or when there are no extra measurements.

3.3.4. ODSTEST

The routine ODSTEST produces a weighted least squares estimate for the state of a water distribution system, in cases when there is an over-determined measurement set. An over-determined measurement set is one that contains more measurements than are absolutely necessary, ie one that is observable and contains more measurements and pseudomeasurements than there are independent state variables. It uses an iterative

procedure and is intended for use when the accuracy of the state estimate is a priority, rather than speed.

The algorithm for ODSTEST is similar to MINSTEST in that an iterative procedure is used which is again based on the nodal format network equations. At each iteration the Jacobian matrix of the linearised network equations is used to calculate the correction vector. The essential difference between these two state estimators is that ODSTEST is designed to deal with an over-determined measurement set. This, combined with the fact that there will always be an amount of measurement error or noise in each equation, means that there will be no state vector that can satisfy (3.8) exactly. To take account of this measurement noise, an error vector $\mathbf{v}$ is introduced to give a new network equation

$$g(\mathbf{x}) = \mathbf{z} + \mathbf{v} \tag{3.12}$$

The aim is to find a state vector $\mathbf{x}$ that minimises $\mathbf{v}$ in some way. In ODSTEST, this is done by applying a weighted least squares technique [13, 130]. This method is employed at the stage when the correction vector, $\Delta\mathbf{x}$, is calculated at each iteration. With the uncertainty incorporated, equation (3.9) becomes

$$g(\mathbf{x}^k) + J.\Delta\mathbf{x} = \mathbf{z} + \mathbf{v} \tag{3.13}$$

with $\mathbf{v}$ the introduced error vector. A correction vector is found which minimises the weighted sum of squares of the elements of the error vector $\mathbf{v}$. This problem can be solved by solving the following augmented matrix problem:

$$\begin{bmatrix} 0 & I & J \\ -I & R^{-1} & 0 \\ J^T & 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{s} \\ \mathbf{r} \\ \Delta\mathbf{x} \end{bmatrix} = \begin{bmatrix} \Delta\mathbf{z} \\ 0 \\ 0 \end{bmatrix} \tag{3.14}$$

where R is a diagonal measurement noise covariance matrix, I is the m by m identity matrix and $\Delta\mathbf{z} = g(\mathbf{x}^k) - \mathbf{z}$. Again this augmented matrix is highly sparse and can be factorised using the same sparse matrix techniques as used in MINSTEST. Convergence tests are carried out on the correction vector and on the current estimate satisfying the

measurement values.

ODSTEST is a numerically stable algorithm that can handle any number of measurements efficiently, producing the most accurate state estimate that is possible from the noise affected measurement data.

3.3.5. LAVSTEST

The routine LAVSTEST calculates a state estimate for an over-determined measurement set. It is intended as an alternative to ODSTEST, producing a weighted least absolute values estimate rather than a weighted least squares state estimate. Like ODSTEST, LAVSTEST uses an iterative procedure with the correction vector, at each iteration, calculated from the current Jacobian and current measurement discrepancy. The essential difference between these two methods is that LAVSTEST uses a linear programming technique to calculate this correction vector [13, 77], whereas ODSTEST uses a matrix inversion method.

At each iteration of LAVSTEST a solution, $\Delta\mathbf{x}$, to equation (3.13) must be found. In LAVSTEST, a solution is found which minimises the weighted sum of absolute values of the elements of $\boldsymbol{\nu}$. This problem can be formulated as the following linear programming problem

$$\begin{aligned} &\textit{Minimise} \quad \mathbf{w}^T\boldsymbol{\nu}_1 + \mathbf{w}^T\boldsymbol{\nu}_2 \\ &\textit{Subject to} \quad J.\Delta\mathbf{x} + \boldsymbol{\nu}_1 - \boldsymbol{\nu}_2 = g(\mathbf{x}^k) - \mathbf{z} \\ &\qquad\qquad \boldsymbol{\nu}_1, \boldsymbol{\nu}_2 \geq 0 \end{aligned} \tag{3.15}$$

Where $\mathbf{w}$ is a weighting vector of dimension n, $\boldsymbol{\nu}_1$ and $\boldsymbol{\nu}_2$ are two error vectors representing the positive and negative components of $\boldsymbol{\nu}$ respectively. It can easily be shown that the corresponding elements of $\boldsymbol{\nu}_1$ and $\boldsymbol{\nu}_2$ will not both be non-zero in the optimal solution.

The principal advantage of a least absolute values state estimator is that at each iteration the correction vector is determined from just n measurements (or n constraints in

(3.15)) which it will satisfy exactly. The remaining $m-n$ measurements will have been rejected. Of course, the selection of the n accepted measurements is done so that the errors in the rejected measurements are minimal. In this way, a least absolute values estimator can filter out gross measurement errors of the type that may be caused by meter malfunction or system failure.

3.4. CONCLUSIONS

Network modelling and state estimation are techniques that are long established for water distribution systems. Both topics have been extensively covered in the literature. The discussion of network modelling in section 3.2 merely collects together and reviews some of the work in this field. Despite its long history, water system modelling is not a routine procedure. Modellers must expend a great deal of care and effort in ensuring that their results accurately reflect the system and that they suit their intended purpose.

In the literature there are many reports of efficient state estimation techniques that have been implemented successfully as computer programs. The three routines presented in this chapter - MINSTEST, ODSTEST and LAVSTEST - are based on hybrid algorithms. Each has been designed to be efficient in different circumstances. By only considering a minimal measurement set and exploiting matrix sparsity, MINSTEST benefits in terms of speed of execution. ODSTEST also uses efficient sparse matrix methods but considers all measurements that are available which makes this routine slightly slower but more accurate. LAVSTEST, by using linear programming techniques, has good bad-data rejection properties.

CHAPTER 4

UNCERTAINTY IN WATER DISTRIBUTION SYSTEMS

4.1. INTRODUCTION

For a water distribution system the three main causes of uncertainty are [12, 16, 17, 54, 143, 144, 145] an inaccurate network model; inaccurate predictions of consumption for a particular instant in time at a particular node; and noise or systematic errors in measurement values. These three factors will all be discussed in more detail in later sections. A smaller contribution to simulation uncertainty comes from the inaccuracy of the mathematical solution techniques and from the precision limits of the computers used. The input uncertainty, from all of these sources, is transferred, through the state estimation process, and results in estimates of the operating state that are also uncertain. The precise way in which the input uncertainty effects the output accuracies is not at all clear, many interrelated factors are involved. The distribution of the meters throughout the network, the network's topology and the operating state of the network all play an important part.

Confidence limit analysis - the process of quantifying the effects of this uncertainty on estimates of state and derived variables - has implications in many areas of water distribution system simulation and operation. Some of the potential application areas are discussed in Chapter 1. These fall into the following categories: real-time control; decision support; operational planning; distribution system design; telemetry system design and operator training. The package TCLAS, described in Chapter 6, has been developed around the concepts of confidence limit analysis. Also, the program OPTMP - an automatic optimal meter placement design program, makes use of confidence limit analysis techniques in assessing the value of each meter configuration. Confidence limit analysis is, therefore,

very important in many areas of water distribution system operation.

In this chapter the uncertainty associated with the monitoring and state estimation of water distribution systems is examined. Firstly, the potential sources of uncertainty and inaccuracy are discussed. A mathematical model of water distribution system uncertainty is presented. This model is used to develop confidence limit algorithms to quantify state estimate uncertainty. Results are presented for a realistic test network.

4.2. REVIEW OF PREVIOUS RESEARCH

Several papers have been published which deal with the problems of state estimation under measurement uncertainty. Some of these papers fall within the bounds of Identification theory and treat the problem in general mathematical terms. Others are concerned with the more specific problem of uncertainty in water or power systems.

In [102], Norton describes Identification as "*the process of constructing a mathematical model of a dynamical system from observations and prior knowledge.*" When noise or errors in the observations are considered, it is easy to see how this fits in with state estimation and confidence limit analysis. Schweppe, in [124], introduced the concept of unknown-but-bounded errors to Identification. His work was followed up by others in [18, 20, 59, 98, 100, 102, 103, 104]. In this work, the following parameter-bounding model, derived from the bounded-noise measurements, is used.

$$\mathbf{z} = g(\mathbf{x}) + \mathbf{v}, \quad |v_i| \leq |e_i^z|, \quad i=1,...,m \tag{4.1}$$

where $\mathbf{v}$ is a noise vector for the observations and $\mathbf{e^z}$ is the measurement error vector. Equation (4.1) is simply saying that the measurement values are inexact and have errors that are unknown but fall within a range bounded by $\mathbf{e^z}$. Identification procedures based on (4.1) are presented. These aim to determine the set of all state estimates that are feasible according to (4.1). An ellipsoidal-bounding method was suggested by Schweppe [124]. This method is further examined in [18, 20, 59, 100, 102, 103, 104]. Alternative parameter-

parameter-bounding methods based on linear programming are presented in [18, 98, 100]

The parameter-bounding methods aim to find all feasible state estimates and do not identify any particular one as better than the others or the most likely as the true state. A more common approach to the problem of state estimation under uncertainty is to try and find the state that best fits the measurement data. This is called deterministic state estimation and relies on statistical or probabilistic assumptions about the behaviour of the measurement errors. Similar assumptions about the behaviour of measurement errors can be used to provide an indication of the reliability or accuracy of these deterministic state estimates. Uncertainty methods based on this idea are now described.

One group of methods is based on the deterministic weighted least squares approach to state estimation [44, 87, 136]. In this form of state estimation, the statistical properties of measurement errors are assumed to be known in advance. In particular, the expected measurement values and their variance is assumed. When this is the case, a value for the variance of each state variable is given by the corresponding element on the leading diagonal of the state covariance matrix, $(J^T R^{-1} J)^{-1}$, where R is the measurement covariance matrix and J is the Jacobian matrix calculated at the state estimate. Stuart and Herget in [136] and Koglin in [87] suggest this as a measure of the potential accuracy of the state variables. In [44], Dopazo, Kiltin and Sasson go further and suggest that, for each state variable, the interval enclosed by its estimated value plus or minus 3 times its standard deviation has a 99% probability of containing the true value of the variable. The standard deviation for each variable is the square root of its variance.

Methods for defining the probabilistic behaviour of state estimate errors, given assumptions about the probabilistic distribution of measurement errors, are described in [4, 5, 6, 7, 8, 9, 10, 22, 86, 95, 129]. These methods allow a probability density function to be defined for each state variable. From these functions it is possible to calculate the probability that a given state variable is above or below a certain limit. In order to be able

to use these methods, accurate probability density functions must be available that describe the random and systematic measurement and pseudomeasurement error variations. These functions are not available in enough detail at present for water distribution system monitoring.

A method based on matrix condition number is proposed by Edelmann in [53]. The condition number (see [76] for more detail on matrix condition numbers) of the Jacobian matrix J, when this is square and non-singular, is suggested as a measure relating the error in the state estimates to the error in the measurements. For the case when J is non-square a different relationship to the one used in [53] must be derived, [76, 79] provide details of how this can be done. In this sense, the condition number gives a measure for the quality of a particular measurement set. As only one value for the quality of this measurement set is provided by this method it cannot be used to produce absolute bounds for each individual state variable as is required by confidence limit analysis.

4.3. SOURCES OF UNCERTAINTY

4.3.1. Model inaccuracy

For a water distribution system with its scale and complexity, a simplified model must be used. Otherwise, real-time simulation becomes impractical. This simplification inevitably leads to inaccuracy. Many authors have tackled the problem of ensuring that the network model accurately reflects the real distribution system. Some of their conclusions are reported below.

Allen, in [11], provides a valuable insight into the process of network model construction, from initial conception to model use and calibration. The suggestions and conclusions made in this paper are based on experience in modelling a real network. The modelling process is broken down into three stages - planning, performance and practice - with the methodology of each examined closely. Important points made in this paper

include: the need for discussion with experts, setting of objectives for the model is required, it must be ensured that records are correct, models must be calibrated and recalibrated regularly and the model requires updating as it is being used.

In 1973 an AWWA committee was set up to examine research and development needs for water distribution. Its conclusions, published in [12], identified that one area where further research was required was in ensuring accuracy of network models and modelling techniques. This request prompted considerable research activity. Eggener and Polkowski, in [54], examined the the impact of modelling assumptions such as skeletonisation, load consolidation and assumed values for pipe friction coefficients. They identified C-values as the weakest piece of information in water network models but suggested that "*any practical degree of accuracy can be attained in modelling distribution networks if enough effort is put forth to develop input data.*" Other authors [25, 33, 91, 108, 109, 128, 143, 144, 145, 146] also examine the accuracy of network modelling. Shamir and Howard [128] look at model accuracy and discuss this in terms of requirements for engineering analysis. Walski in [143, 144, 145, 146], Ormsbee and Wood in [109], Lansey in [91], Ormsbee and Chase in [108], Coulbeck in [34] and Cesario and Davis in [25], explore the topic of network calibration. This is the process of adjusting model parameters, usually pipe C-values or friction factors, so that they fit observed data. In these papers the amount and type of data that is required for a calibration study is set out and formulae for C-value adjustments are presented. Walski [143] also acknowledges the influence of uncertainty about assumed loadings in calibration and suggests how this should be accounted for in his procedure.

The research work discussed above provides very useful methods for reducing model inaccuracy. Once the methods described have been employed, the network model has been constructed accurately and calibrated to fit observed data, there will still be a small amount of residual inaccuracy. This is unavoidable but must be acknowledged. A distinction must

be made between the uncertainty caused by model inaccuracy and that caused by errors in measurements and in predictions of nodal consumptions. Model accuracy is to some extent constant, whereas measurement and pseudomeasurement errors vary randomly with time and with the state of the system. In general terms, model inaccuracy introduces systematic errors and measurement and pseudomeasurement errors introduce random noise. Of course, systematic metering errors play a part but the random noise, particularly in the nodal consumption errors, is dominant. For this reason it is more informative to treat the two sources of uncertainty - model and measurement - separately.

The problems of network calibration and model accuracy have received much attention in the literature. This research work has not been matched by researchers examining the problems of measurement and pseudomeasurement uncertainty. It is this other side of the coin that is now addressed.

4.3.2. Uncertainty in nodal consumption predictions

Predictions of the consumption at each node in the network make up the bulk of the measurement data used in water distribution system simulation. The unpredictability of water use on an individual or nodal basis makes this type of data very unreliable. As a result much of the uncertainty is introduced from this source. Estimation of total water usage for the network as a whole or for a large number of consumers can be carried out quite accurately [124, 131]. The effects of random variations of individual use are evened out when predicting at this scale. When the scale of prediction is reduced to a nodal level, with smaller numbers of consumers considered, accuracy suffers. Typically, an individual consumer will take its water in just a few short spells throughout the day. This statement may not be true for industrial users but is valid for domestic users who account for the bulk of demand. This extreme unpredictability makes estimation of the demand at a node for a particular instant of time very difficult.

There are many suggestions for how nodal consumptions should be predicted ([3, 133] for example). Typically, these involve splitting usage into two or more classes, domestic and industrial for instance. A daily demand profile for each type must be given and this multiplied by the amount of each class' consumption. For example, multiplying average individual use by the nodal population for domestic consumption classes. On top of this consumption is an amount of unaccounted for usage, through leakage for example. Adding each of these terms together will result in a predicted nodal consumption. The sum of all of these, over all nodes in the network, should equal the total water input to make the system balanced. These nodal predictions only give an average expected demand. The real figures may be considerably different. For a small node the demand may be 100% in error. Generally speaking, the larger the demand the more accurate, in relative terms, such a prediction will be. Although, even for the largest nodes an accuracy of only 30% would not be unreasonable.

4.3.3. Measurement uncertainty

A greater monitoring accuracy for water distribution systems can be achieved by supplementing the nodal prediction estimates with real metered values. Meters can be installed in the network and linked to the telemetry system to provide real measurement data in real-time. Meters themselves are never fully accurate but are generally more reliable than are predictions of nodal consumption and other pseudomeasurements. Pressure and flow meters are supplied with the manufacturer's specifications on accuracy and, when calibrated properly, should operate within these limits. These accuracy specifications prove very useful when defining the meters' accuracy. In [74], Hayward gives a useful account of the different types of meter that may be used in water system monitoring and the accuracies that can be expected with these.

Generally, the addition of a meter to the network will increase the accuracy of the state estimates. As an example, consider the effect of adding a pressure meter at a node in the network. It is clear that the pressure at this node can now be estimated more accurately, provided, of course, the meter is accurate enough. The pressure estimates in the neighbouring nodes will also show an improvement, but the amount of improvement will decrease as distance from the meter increases. The exact extent of the meter's region of influence will be dependent on, among other things, the accuracy of the nodal consumption estimates in the neighbouring nodes, the accuracy of the flow estimates in this area, pipe sizes and the presence of any other meters near by. The flow variables close to the new pressure meter may also show an improvement in accuracy, although this will not be as marked. In a similar way, the addition of a flow meter to the network will have an effect in reducing the potential error in the neighbouring flow and pressure variables. Although the region of influence may have a different shape and the flow and pressure variables will be affected to a different degree.

The number of real meters in the measurement system and their accuracy are not the only factors that effect the improvement in state estimate accuracy. The distribution of these meters throughout the network is also very important. If all of the meters are placed in one region of the network then it will be possible to obtain accurate state estimates for variables in this region, but the accuracy of variables in other regions may be poor. This distribution effect is complicated further by the network's topology. For example, a meter placed in a weakly connected region of the network will have little influence on the accuracy of state variables elsewhere in the network. Similarly, a flow meter may have a greater impact if it is placed in a pipe which has a large flow, as this will mean that a larger proportion of the water flowing through the network is being measured. A combination of meter distribution, meter accuracy and network topology may mean that certain meters are redundant. That is to say, the accuracy of system variables will not be adversely affected by the removal of such a meter. This situation occurs when a meter is measuring a variable

that can be estimated more accurately using data from other meters in its vicinity.

The operating state of the system can have a large effect on the way in which the measurement uncertainty is passed on to the state estimates. In many water networks the flow pattern can change considerably throughout a day's operation. The accuracy of nodal consumption estimates may similarly alter as a result of these changes. Also some flow meters have an accuracy that is dependent, in absolute terms, on the size of the flow they are measuring. Consequently, one particular measurement configuration may provide differing levels of accuracy under different operating conditions or at different times of the day.

4.4. THE UNCERTAIN NETWORK MODEL

In this section, the problem of introducing uncertainty into the deterministic network equation, (3.2), is addressed. The first problem to be solved is how should the uncertainty in the measurement data and in the control or state variables be presented. The accuracy of each variable must be assessed independently as a particular meter configuration will mean that some variables can be estimated accurately and others poorly. Similarly, the accuracy of each measurement or a pseudomeasurement value should be expressed independently to reflect the difference in accuracy between different types of meter and between metered values and pseudomeasurement values. So, it is not possible to describe how accurate a meter configuration is just by one value.

The method chosen for representing the measurement uncertainty is to specify upper and lower bounds for each measurement and pseudomeasurement value. This uses the unknown-but-bounded concept for observation error introduced by Schweppe in [124]. If $\mathbf{z}^o$ is the **observed measurement vector**, made up of the meter readings and the estimated pseudomeasurement values. then a lower limit, $\mathbf{z}^l$, and an upper limit, $\mathbf{z}^u$, can be specified. Generally, $\mathbf{z}^l$ and $\mathbf{z}^u$ are calculated by subtracting and adding the **measurement error vector**,

$\mathbf{e}^z$, from and to $\mathbf{z}^o$. So, $\mathbf{z}^l = \mathbf{z}^o - \mathbf{e}^z$ and $\mathbf{z}^u = \mathbf{z}^o + \mathbf{e}^z$, where $\mathbf{e}^z$ is a non-negative, m-dimensional vector. This formulation of uncertainty derives from the recognition that the observed measurement vector is not necessarily the true measurement vector (the vector that would be obtained with completely accurate meters and pseudomeasurement predictions), but the true vector is contained within a region bounded by the accuracy of the measurements and pseudomeasurements which is specified by $\mathbf{e}^z$. This is summarised by the following inequalities:

$$z_i^t \leq z_i^u = z_i^o + e_i^z, \quad i = 1,...,m$$
$$z_i^t \geq z_i^l = z_i^o - e_i^z, \quad i = 1,...,m \tag{4.2}$$

where $\mathbf{z}^t \in \mathbf{R}^m$ is the **true measurement vector.**

Elements of the $\mathbf{e}^z$ vector can be derived from the manufacturer's accuracy specifications in the case of real measurements. For nodal consumption predictions, a thorough investigation must be made into water use patterns for each class of node and at each time of day. The nodes being split into classes according to type of user (industrial or domestic etc) and according to population or amount of consumption. These studies, as well as providing the traditional demand profiles, will give values for the variance of demand. This information can be used to derive the upper and lower demand profiles for each node. For pseudomeasurements other than nodal consumption estimates a similar procedure will yield the desired values and bounds.

The deterministic network equation, (3.2), must be altered to take account of the measurement uncertainty. In the deterministic situation it is assumed that the true measurement vector, $\mathbf{z}^t$, can be approximated by $\mathbf{z}^o$, the observed measurement vector. In the non-deterministic or uncertain model, all that it is assumed is that the true measurement vector is contained in the region bounded by $\mathbf{z}^l$ and $\mathbf{z}^u$. A measurement set M is defined as a collection of variables in the system for which real metered values or predicted pseudomeasurements are available. This set will include loads for which nodal

consumption predictions are available, and pressure and flows that are metered. A distinction is made between the measurement set M and the collection of values that this set would produce for a particular operating state. This measurement set at a particular instant in time, or for a particular assumed operating state, will produce a measurement vector $\mathbf{z}^o \in \mathbf{R}^m$, where m is the cardinality of M. With the measurement set M producing measurement vector $\mathbf{z}^o$, the set of feasible measurement vectors is given by:

$$Z(M,\mathbf{z}^o) := \{\mathbf{z} \in \mathbf{R}^m : z_i^l \leq z_i \leq z_i^u,\ i=1,\ldots,m\} \tag{4.3}$$

where m is the cardinality of the measurement set M and $\mathbf{z}^l$ and $\mathbf{z}^u$ are defined as above. $Z(M,\mathbf{z}^o)$ just defines a region of $\mathbf{R}^m$ in which the true measurement vector is contained. This region is the smallest that can be obtained within the limits of accuracy of the measurement set. In this format, the network equation of (3.2) is replaced by the following set inclusion:

$$g(\mathbf{x}) \in Z(M,\mathbf{z}^o) \tag{4.4}$$

following from the assumption that the true measurement vector is unknown but contained in $Z(M,\mathbf{z}^o)$. This gives the set of feasible state vectors, $X(M,\mathbf{z}^o)$, for measurement set M and measurement vector $\mathbf{z}^o$, as:

$$X(M,\mathbf{z}^o) := \{\mathbf{x} \in \mathbf{R}^n : g(\mathbf{x}) \in Z(M,\mathbf{z}^o)\} \tag{4.5}$$

Equation (4.4) will be referred to as the **uncertain network equation** with $X(M,\mathbf{z}^o)$, of (4.5), representing the state uncertainty set. For the uncertain network equation there is no unique operating state that can be calculated. All that can be defined is a set of possible operating states resulting from the set of possible measurement vectors. No preference is placed on any of these, all are assumed to be equally likely. This reflects the lack of preference for a particular measurement vector in $Z(M,\mathbf{z}^o)$. In [102], Norton justifies this position in the following way: "*Lack of a unique estimate of* θ **(x in the notation used above)** *is at first worrying, but we can reassure ourselves by reflecting that engineering design is largely a*

matter of tolerancing for adequate performance in the worst case." Also, this method does not make unrealistic assumptions about the probabilistic properties of the measurement data, its expected values or its probabilistic variation. The unknown-but-bounded treatment of measurement uncertainty leads to this simple and flexible presentation of state estimate uncertainty, without making unrealistic measurement assumptions. Feasible state estimates are specified by a sharply defined set, $X(M, \mathbf{z}^0)$. This fits neatly with many of the intended uses of state estimates, in worst-case control for instance. Is the system operating in an acceptable range? Has the system failed? If so, where is the fault located? Questions such as these can be answered more easily and categorically when the range of possible operating states of the system can be clearly defined.

When faced with measurement uncertainty in state estimation the most common response of engineers and researchers has been to try and produce estimates that best fit the measurement data in some way. Some algorithms and techniques for deterministic state estimation under measurement noise are presented in [44, 77, 86, 125, 129, 130, 136]. The advantages of deterministic over uncertain state estimation - principally that only one estimate is presented - must be balanced by the lack of indication of how accurate this estimate is. Inherent in some of these methods [86, 129, 136] are assumptions, similar to those explicitly formulated by Schweppe and Wildes in [125], about the statistical properties of the measurement errors. For instance, Schweppe and Wildes assume that measurement error is random with a zero mean and can be modelled by a positive definite symmetric matrix. Under assumptions like these, or similar implicit assumptions, it can be claimed that these deterministic state estimates are 'better' than any other feasible state. If, on the contrary, such assumptions are not valid then no particular feasible state can be identified as a better estimate than any other. This point is made in [16, 59, 98, 100, 102, 103] where the validity of some of these statistical assumptions is questioned. Some examples that show how the statistical assumptions made by Schweppe and Wildes and others may be invalid are now given: (i) A meter may have a systematic error that causes it to give

consistently over or under estimated readings. This will invalidate the zero mean assumption about measurement error. (ii) Leakage at a node may go unnoticed which will result in under estimated consumption predictions for this node. This will again invalidate the zero mean assumption about measurement error. (iii) An incorrect prediction for nodal population may result in consumption predictions to be consistently high or low. (iv) Some unconsidered external factors, such as weather conditions, may be influencing the water use throughout the whole network, resulting in erroneous nodal consumption estimates. This type of factor will invalidate assumptions about the statistical distribution of measurement errors.

The probabilistic approach, suggested in [4, 5, 6, 7, 8, 9, 10, 22, 86, 95, 129] suffers from a similar drawback. This work is very thorough but requires much information about the probabilistic nature of the measurement uncertainty. As is observed by Allan et al in [4, 6, 9], it cannot be assumed that measurement errors have a simple statistical distribution pattern such as the normal distribution. Examples given in these papers show distributions that have several peaks. Also, it is made clear that error distribution from different sources are strongly correlated. These two factors mean that such a technique cannot be applied effectively to real-time monitoring of water distribution systems because the probability density functions for measurement error can only be generated by sampling measurements over a considerable period.

To make the uncertainty in state estimates more accessible, uncertainty intervals or confidence limits, similar to those for measurement values, can be derived in the following way. Let

$$x_i^l := \min_{\mathbf{x}\in X(M,\mathbf{z}^o)} x_i, \quad i=1,\ldots,n$$

$$x_i^u := \max_{\mathbf{x}\in X(M,\mathbf{z}^o)} x_i, \quad i=1,\ldots,n \tag{4.6}$$

The vectors $\mathbf{x}^l$ and $\mathbf{x}^u$ will provide lower and upper bounds on the state vector $\mathbf{x}$ in the

same way that $\mathbf{z}^l$ and $\mathbf{z}^u$ did for the measurement vector. For each individual variable, the interval (x_i^l, x_i^u) is referred to as the **uncertainty interval** for the i^{th} variable and x_i^l and x_i^u are referred to as its **confidence limits**. These uncertainty intervals or confidence limits are as tight as can be achieved from the measurement uncertainty. Calculating these bounds - the process referred to as confidence limit analysis or uncertainty quantification - is dealt with in much more detail later in this chapter. If $X^*(M, \mathbf{z}^o)$ is the set defined by these bounds, ie

$$X^*(M, \mathbf{z}^o) := \{\mathbf{x} \in \mathbf{R}^n : x_i^l \leq x_i \leq x_i^u,\ i = 1,...,n\} \tag{4.7}$$

then it must be noted that $X^*(M, \mathbf{z}^o)$ may not be the same as $X(M, \mathbf{z}^o)$. Clearly, $X(M, \mathbf{z}^o) \subseteq X^*(M, \mathbf{z}^o)$, but not every combination of values that are each feasible for the individual state variables form a feasible state vector. Let $Z^*(M, \mathbf{z}^o)$ be the subset of $\mathbf{R}^m$ onto which $X(M, \mathbf{z}^o)$ is mapped by $g(.)$ the network function. Then

$$Z^*(M, \mathbf{z}^o) := \{\mathbf{z} \in \mathbf{R}^m : \mathbf{z} = g(\mathbf{x}),\ \mathbf{x} \in X(M, \mathbf{z}^o)\} \tag{4.8}$$

$Z^*(M, \mathbf{z}^o) \subseteq Z(M, \mathbf{z}^o)$, but these two sets are unlikely to be equal. There may be a $\mathbf{z} \in Z(M, \mathbf{z}^o)$ for which there is no $\mathbf{x}$ (neither in $X(M, \mathbf{z}^o)$ nor $\mathbf{R}^n$) for which $g(\mathbf{x}) = \mathbf{z}$. In other words, there may be vectors in $Z(M, \mathbf{z}^o)$ that are inconsistent for $g(.)$. These two remarks are demonstrated in fig 4.1 for the 2-dimensional case.

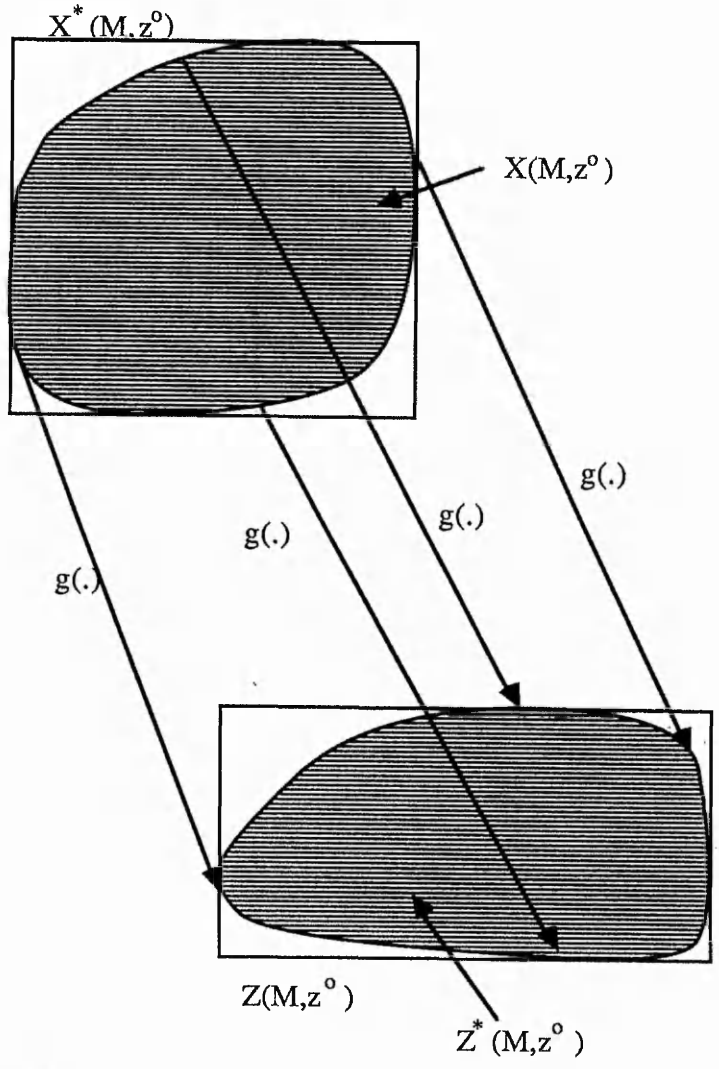
Uncertainty bounds or confidence limits for the derived state variables, defined by (3.3), can also be constructed. Let $Y(M, \mathbf{z}^o)$ be defined as

$$Y(M, \mathbf{z}^o) := \{\mathbf{y} \in \mathbf{R}^N : \mathbf{y} = f(\mathbf{x}),\ \mathbf{x} \in X(M, \mathbf{z}^o)\} \tag{4.9}$$

$Y(M, \mathbf{z}^o)$ will be referred to as the derived state uncertainty set and corresponds to $X(M, \mathbf{z}^o)$ which is the state uncertainty for the independent state variables. Upper and lower bounding vectors, $\mathbf{y}^l$ and $\mathbf{y}^u$, can be defined for $Y(M, \mathbf{z}^o)$, where:

$$y_i^l := \min_{\mathbf{y} \in Y(M, \mathbf{z}^o)} y_i,\quad i = 1,...,N$$

Fig 4.1. Relationship between X^* and X, and between Z^* and Z.



$X(M,z^o)$ - The state uncertainty set.
$X^*(M,z^o)$ - The smallest box containing the state uncertainty set.
$Z(M,z^o)$ - The measurement uncertainty set.
$Z^*(M,z^o)$ - The image of the state uncertainty set when mapped by the network function g(.)

$$y_i^u := \max_{y \in Y(M,z^0)} y_i, \quad i=1,...,N \tag{4.10}$$

(y_i^l, y_i^u) is called the uncertainty interval for the i^{th} derived variable and y_i^l and y_i^u are referred to as its confidence limits. A further set can be defined:

$$Y^*(M,z^0) := \{y \in \mathbf{R}^N : y_i^l \leq y_i \leq y_i^u, \ i=1,...,N\} \tag{4.11}$$

The set $Y^*(M,z^0)$ is the 'smallest box' containing $Y(M,z^0)$. The definitions (4.9), (4.10) and (4.11) correspond directly to the definitions (4.5), (4.6) and (4.7), and represent state uncertainty in terms of the derived state vectors rather than the independent state vectors.

The differences between the two sets $X(M,z^0)$ and $X^*(M,z^0)$ and between the two sets $Y(M,z^0)$ and $Y^*(M,z^0)$ should always be recognised. The uncertainty intervals or confidence limits described by $X^*(M,z^0)$ and $Y^*(M,z^0)$ provide a simple and convenient way of presenting state uncertainty at the level of the individual variables. Generally, the sets $X(M,z^0)$ and $Y(M,z^0)$ will be highly complex subsets of $\mathbf{R}^n$ and $\mathbf{R}^N$ respectively, which have no simple or concise characterisation. Conversely, a pair of vectors is all that is required to describe $X^*(M,z^0)$ and $Y^*(M,z^0)$ completely, ie the upper and lower bounding vectors in each case. This presentation of state uncertainty is also more accessible for possible applications such as worst-case control or design.

4.5. CONFIDENCE LIMIT ANALYSIS

The process of calculating uncertainty bounds for the state estimates, which result from the measurement and pseudomeasurement uncertainty, is referred to as confidence limit analysis. Based on the model of uncertainty, described in the previous section, mathematical methods for calculating these confidence limits are now presented.

It can now be seen how confidence limit analysis can be formulated as a series of mathematical optimisation problems. For each of the independent state variables, $i=1,...,n$

$$
\begin{aligned}
&x_i^l = \min\ x_i \\
&\textit{subject to}\ \mathbf{x} \in X(M, \mathbf{z}^o), \quad \text{and} \\
&x_i^u = \max\ x_i \\
&\textit{subject to}\ \mathbf{x} \in X(M, \mathbf{z}^o)
\end{aligned}
\tag{4.12}
$$

This follows from equation (4.6). Confidence limits for the derived state variables can be found by a similar set of optimisations. For $i = 1,\ldots,N$

$$
\begin{aligned}
&y_i^l = \min\ y_i \\
&\textit{subject to}\ \mathbf{y} \in Y(M, \mathbf{z}^o), \quad \text{and} \\
&y_i^u = \max\ y_i \\
&\textit{subject to}\ \mathbf{y} \in Y(M, \mathbf{z}^o)
\end{aligned}
\tag{4.13}
$$

In equation (4.13), the condition that $\mathbf{y} \in Y(M, \mathbf{z}^o)$ is equivalent to the condition that $\mathbf{y} = f(\mathbf{x})$ for some $\mathbf{x} \in X(M, \mathbf{z}^o)$.

The nature of the network equations in $g(.)$ means that the optimisation problems of (4.12) and (4.13) are non-linear. Therefore, the choice of optimisation technique to be used is not at all clear. With n, the number of independent state variables, N, the number of derived state variables, and m, the number of measurements, confidence limit analysis requires $2n+2N$ non-linear optimisations each subject to $2m$ constraints (the $2m$ constraints are supplied by the lower and upper bounds on the measurement uncertainty). For a realistic water distribution system there may be several hundred independent state variables and several hundred measurements and pseudomeasurements. Therefore, confidence limit analysis is a highly computationally intensive task. For this reason, any optimisation method used must be both fast and efficient. Savings in computational run-time can be made by exploiting the special structure of the network model, which results in network equations that are sparse (ie each individual equation is dependent on only a relatively small number of state variables). Also, with problems of this size, numerical instability in the solution algorithms can cause problems. Several different confidence limit algorithms have been developed with these factors in mind.

In the remainder of this section the confidence limit algorithms are presented. These fall into two categories: non-linear and linearised methods.

4.5.1. Monte Carlo method

In normal use, deterministic state estimators produce one state estimate for one measurement vector. Used in this way they give no indication of how a state estimate may vary in response to variations in the measurement values. Alternatively, if a deterministic state estimator is used repeatedly for a whole range of measurement vectors then some indication of state estimate variance is provided. It is this idea that forms the basis to the Monte Carlo approach to confidence limit analysis. A large number of feasible state estimates are generated, as randomly as possible, and from these the state estimate confidence limits are estimated. The larger the number of random feasible state estimates the more reliable the confidence limits.

Let $\mathbf{z}^l$ be a measurement vector, selected randomly from the set $Z(M,\mathbf{z}^o)$ of all feasible measurement vectors, and let $\mathbf{x}^l$ be a deterministic state estimate calculated from $\mathbf{z}^l$. $\mathbf{x}^l$ is a feasible state estimate if $g(\mathbf{x}^l)\in Z(M,\mathbf{z}^o)$. This follows from the definition of feasible state vectors given in equation (4.5). It must be noted that $g(\mathbf{x}^l)$ is not necessarily equal to $\mathbf{z}^l$. If $\mathbf{z}^l$ is not a consistent vector, then there is no state vector $\mathbf{x}$ for which $g(\mathbf{x})=\mathbf{z}^l$. In fact, if $Z^*(M,\mathbf{z}^o)$ is defined as in (4.8), then $Z^*(M,\mathbf{z}^o)=Z(M,\mathbf{z}^o)$ only when M is a minimal measurement set (ie if M is an observable measurement set and has no observable subset). For a sequence, $\mathbf{z}^1,....,\mathbf{z}^k$, of measurement vectors selected randomly from $Z(M,\mathbf{z}^o)$, a sequence of sets $X^1, \ldots, X^k$ can be defined, with

$$X^j := \{\mathbf{x}^l\in\mathbf{R}^n : g(\mathbf{x}^l)\in Z(M,\mathbf{z}^o) \textit{ for some } i\in\{1,...,j\}\},\ j=1,...,k \tag{4.14}$$

where $\mathbf{x}^l$ is the state estimate calculated from $\mathbf{z}^l$. X^j is the set of feasible state estimates generated by the sequence of measurement vectors $\mathbf{z}^1,....,\mathbf{z}^j$. This sequence of sets is such that $X^j\subseteq X^k\subseteq X(M,\mathbf{z}^o)$ for all $j=1,....,k-1$, as only feasible state estimates are contained in

X^j. For a large number, k, of randomly selected measurement vectors it can be assumed that X^k is approximately equal to $X(M,\mathbf{z}^o)$. In other words, as $k\to\infty$, $X^k\to X(M,\mathbf{z}^o)$. This assumption suggests a Monte Carlo simulation approach to confidence limit analysis. A similar sequence of sets, $Y^1,...,Y^k$, for the derived state variables can be defined, with

$$Y^j := \{\mathbf{y}\in\mathbf{R}^N : \mathbf{y}=f(\mathbf{x}),\ \mathbf{x}\in X^j\},\quad j=1,...,k \tag{4.15}$$

Y^j, for j= 1,...,k, is the set of feasible derived vectors generated by the sequence of measurement vectors $\mathbf{z}^1,\ldots,\mathbf{z}^j$. It can also be claimed that as $k\to\infty$, $Y^k\to Y(M,\mathbf{z}^o)$.

Before an algorithm description is given, some implementation practicalities must be explained. Firstly, the choice of state estimator must be consistent throughout. The actual estimator used is of no importance, provided that it is unbiased and that it can guarantee convergence in a high proportion of cases, because all state estimates are checked for feasibility before being used to update X^j or Y^j. A sequence of random measurement vectors can be selected from $Z(M,\mathbf{z}^o)$ by using a random number generator. For example, a sequence of random numbers, $r_1^j,\ldots,r_m^j$, scaled to be between 0.0 and 1.0, can be generated (where m is the number of elements in the measurement set M). From this sequence a random measurement vector $\mathbf{z}^j$ can be constructed with

$$z_i^j = z_i^l + r_i^j.(z_i^u - z_i^l),\quad i=1,...,m \tag{4.16}$$

where $\mathbf{z}^l$ and $\mathbf{z}^u$ are the lower and upper bounds for $Z(M,\mathbf{z}^o)$. $\mathbf{z}^{j+1}$ can be constructed in a similar way from a new sequence of random numbers. A final consideration is the storage of the sets X^j and Y^j of feasible state and derived state vectors at the j^{th} stage. The large number of these vectors means that it is impractical to store them all, fortunately, this is not necessary. For X^j, only two vectors need be stored, these are $\mathbf{x}^{jl}$ and $\mathbf{x}^{ju}$, the lower and upper bounding vectors for the current set of feasible state estimates, X^j. These vectors will be updated whenever a new feasible vector, not contained in any of the X^j 's, is found. Similarly, only two vectors are required to store the set Y^j, $j=1,...,k$. The Monte Carlo confidence limit algorithm can now be described.

Monte Carlo confidence limit algorithm

1. Select a large number, k. This will limit the number of simulations to be carried out. Advice on how large k should be is given below. Set $i=0$.

2. Set $i=i+1$.

3. Select a sequence of m random numbers, $r_1^i, \ldots, r_m^i$, and use these to construct a random measurement vector $\mathbf{z}^i$ from $Z(M, \mathbf{z}^o)$ as described in (4.16)

4. Calculate a state estimate, $\mathbf{x}^i$, for $\mathbf{z}^i$. If $g(\mathbf{x}^i) \in Z(M, \mathbf{z}^o)$, then use $\mathbf{x}^i$ to update $\mathbf{x}^{ll}, \mathbf{x}^{lu}, \mathbf{y}^{ll}$ and $\mathbf{y}^{lu}$. Otherwise reject $\mathbf{x}^i$ as infeasible.

5. If $i < k$, go back to step 2. Otherwise stop.

In this algorithm description no advice is given on the number of simulations that are required to make X^k an accurate approximation to $X(M, \mathbf{z}^o)$. The short answer is as many as possible. A more realistic suggestion is to select an upper limit on the amount of computing time that is available, many hours will be required for large networks, and set k accordingly. The very large computation times that are required, means that the Monte Carlo confidence limit algorithm is not suitable for routine use. When there is a significant degree of measurement redundancy a large proportion of the state estimates have to be rejected as infeasible. This extra complication further adds to computation time. It is however, possible to refine the algorithm to improve its run-time performance. After selecting a random measurement vector from $Z(M, \mathbf{z}^o)$, it can be tested against the balance constraint before being passed to the state estimator. The balance constraint ensures that the amount of water leaving the network, as defined by the nodal consumption predictions, is equal to the amount of water entering the network, as defined by the inflow measurements or predictions. If a measurement vector fails the balance constraint then it is pointless trying to find a state estimate for it as this may be rejected at a later stage. The

balance constraint pre-check has the effect of reducing the number of unnecessary calls to the state estimator.

As state estimation accounts for the majority of the computation time in this algorithm, a significant improvement to run-time can be made by using a fast state estimator. One such estimator is MINSTEST, described in Chapter 3. This makes use of only a minimal number measurements and runs approximately four times faster than an equivalent state estimator designed for an over-determined measurement set. To make use of this estimator, a minimal measurement subset M' of M must be selected. This will have n elements (one for each independent state variable) and must be observable. Then, in step 3 of the algorithm, $Z(M,\mathbf{z}^o)$ must be replaced by $Z(M',\mathbf{z}^o)$. The rest of the algorithm will remain unchanged. (Note: the network function $g(.)$ in step 4 will be the one consistent with the full measurement set M and that $Z(M,\mathbf{z}^o)$ cannot be replaced by $Z(M',\mathbf{z}^o)$). The observation that the final bounds for X^k and Y^k are reached for measurement vectors that have individual measurements on the bounds of $Z(M,\mathbf{z}^o)$ for a large number of the measurements, leads to a final suggestion for improvement. If measurement vectors are selected so that n of their m elements are on the bounds of $Z(M,\mathbf{z}^o)$, then a higher proportion of the feasible state estimates will be extremal or closer to the bounds in $X(M,\mathbf{z}^o)$. Care must be taken to ensure randomness of measurement vectors if this modification is adopted. In each simulation the n bounded measurements can be selected randomly from the m members of M, this safeguard will improve the randomness of the measurement vectors and hence the reliability of the algorithm.

Monte Carlo simulation is obviously slow computationally, but despite this is useful in some situations. The condition that only feasible state estimates are used to update $\mathbf{x}^{ll}$,$\mathbf{x}^{lu}$,$\mathbf{y}^{ll}$ and $\mathbf{y}^{lu}$ makes the procedure mathematically reliable and ensures that these bounds can be attained. It has been used to provide a yardstick, against which the accuracy of all other confidence limit algorithms can be compared. Unfortunately, it is impractical in

many applications.

Other more practical methods are described in the next sections. Firstly, the problem is linearised and a linear version of the state uncertainty model is presented. This appears in section 4.5.2. In sections 4.5.3, 4.5.4 and 4.5.5, three confidence limit algorithms, based on this linear uncertainty model, are presented. These three methods are: the linear programming method; the sensitivity matrix method and the ellipsoid method. In section 4.5.6, the linear confidence limit algorithms are extended so that they can be used for the derived state variables.

4.5.2. Linearised confidence limit algorithms

For real-time and routine design applications, confidence limit algorithms must be able to run in a few seconds. To achieve this level of performance approximate methods must be considered. In this section the uncertainty problem is linearised. Based on this linearisation, some fast and efficient confidence limit algorithms have been designed. These are described in sections 4.5.3, 4.5.4 and 4.5.5. Initially in this section, the mathematical formulae and the algorithms refer only to the uncertainty in the independent state vector, $\mathbf{x}$, and do not mention the derived state vectors. In section 4.5.6 these omissions are rectified, with the algorithms extended to allow confidence limit analysis for the derived state variables.

Suppose that $\hat{\mathbf{x}}$ is the state estimate calculated from the measurement vector $\mathbf{z}^o$, where $\mathbf{z}^o$ is defined as in section 4.4. The non-linear network function, $g(.)$, can be linearised around $\hat{\mathbf{x}}$ using a first order Taylor approximation [13, 65] to give:

$$g(\mathbf{x}) \simeq J.(\mathbf{x}-\hat{\mathbf{x}}) + g(\hat{\mathbf{x}}) \tag{4.17}$$

for all state vectors close to $\hat{\mathbf{x}}$. In (4.17), J is the Jacobian matrix evaluated at $\hat{\mathbf{x}}$. $g(.)$ can be linearised around any state vector, $\hat{\mathbf{x}}$, this need not necessarily be the state estimate for $\mathbf{z}^o$. It is better, however, to use a value for $\hat{\mathbf{x}}$ that is in some way central to the set of

feasible state vectors. This is because, the approximation used in (4.17) is more accurate for values of $\mathbf{x}$ for which $\|\mathbf{x}-\hat{\mathbf{x}}\|$ is small. The best estimate available for the centre of $X(M,\mathbf{z}^o)$ is the state estimate calculated from $\mathbf{z}^o$. In section 4.4, a feasible state vector was described as one for which $g(\mathbf{x})\in Z(M,\mathbf{z}^o)$. This condition can be linearised using (4.17) to give a linear approximation, $X^1(M,\mathbf{z}^o)$, of the state uncertainty set $X(M,\mathbf{z}^o)$. This is defined as follows:

$$X^1(M,\mathbf{z}^o) := \{\mathbf{x}\in\mathbf{R}^n : J.(\mathbf{x}-\hat{\mathbf{x}}) + g(\hat{\mathbf{x}})\in Z(M,\mathbf{z}^o)\} \tag{4.18}$$

$X^1(M,\mathbf{z}^o)$ will be referred to as the linearised state uncertainty set. Replacing $\mathbf{x}-\hat{\mathbf{x}}$ by $\mathbf{dx}$ and using the definition of $Z(M,\mathbf{z}^o)$ given in (4.3), $X^1(M,\mathbf{z}^o)$ can be redefined as

$$X^1(M,\mathbf{z}^o) := \{\mathbf{x}\in\mathbf{R}^n : \mathbf{x}=\hat{\mathbf{x}}+\mathbf{dx},\ \mathbf{z}^l-g(\hat{\mathbf{x}})\le J.\mathbf{dx}\le \mathbf{z}^u-g(\hat{\mathbf{x}})\} \tag{4.19}$$

It is easy to see that these two definitions, (4.18) and (4.19), of $X^1(M,\mathbf{z}^o)$ are equivalent. For the reasons given in section 4.4, the set $X^1(M,\mathbf{z}^o)$ will not be calculated explicitly. Rather, the smallest 'box' or orthotope containing $X^1(M,\mathbf{z}^o)$ is sought. This set will be denoted by $X^{1*}(M,\mathbf{z}^o)$ and referred to as the linearised state uncertainty box. Following the definition of $\mathbf{x}^l$ and $\mathbf{x}^u$ in (4.6), lower and upper limits for $X^1(M,\mathbf{z}^o)$ can be defined as follows:

$$\begin{aligned} x_i^{1l} &:= \min_{\mathbf{x}\in X^1(M,\mathbf{z}^o)} x_i,\quad i=1,\ldots,n \\ x_i^{1u} &:= \max_{\mathbf{x}\in X^1(M,\mathbf{z}^o)} x_i,\quad i=1,\ldots,n \end{aligned} \tag{4.20}$$

The definition of state uncertainty has been linearised in this way to allow confidence limit algorithms based on linear programming methods. Calculating the bounding vectors of $X^1(M,\mathbf{z}^o)$ can easily be formulated as a linear programming problem. To allow this some new notation is introduced:

$$\mathbf{dz}^l := \mathbf{z}^l-g(\hat{\mathbf{x}}) \tag{4.21}$$

$$\mathbf{dz}^u := \mathbf{z}^u-g(\hat{\mathbf{x}}) \tag{4.22}$$

$$DZ(M,\mathbf{z}^o) := \{\mathbf{dz}\in\mathbf{R}^m : g(\hat{\mathbf{x}})+\mathbf{dz}\in Z(M,\mathbf{z}^o)\} \tag{4.23}$$

$$DX^1(M,\mathbf{z}^o) := \{\mathbf{dx}\in\mathbf{R}^n : \hat{\mathbf{x}}+\mathbf{dx}\in X^1(M,\mathbf{z}^o)\} \tag{4.24}$$

$DX^1(M,\mathbf{z}^o)$ is just the set $X^1(M,\mathbf{z}^o)$ shifted by $\hat{\mathbf{x}}$, $DZ(M,\mathbf{z}^o)$ is the measurement uncertainty set shifted by $g(\hat{\mathbf{x}})$ and $\mathbf{dz}^l$ and $\mathbf{dz}^u$ are the bounding vectors for this set. Let $\mathbf{dx}^l$ and $\mathbf{dx}^u$ represent the 'tightest' lower and upper bounds for the set $DX^1(M,\mathbf{z}^o)$. Then, the i^{th} element of $\mathbf{dx}^l$ can be found by solving the linear programming problem

$$\begin{aligned} &\textit{minimise } dx_i \\ &\textit{subject to } \mathbf{dz}^l \leq J.\,\mathbf{dx} \leq \mathbf{dz}^u \end{aligned} \tag{4.25}$$

Similarly, the i^{th} element of $\mathbf{dx}^u$ can be found by solving the corresponding linear programming problem

$$\begin{aligned} &\textit{maximise } dx_i \\ &\textit{subject to } \mathbf{dz}^l \leq J.\,\mathbf{dx} \leq \mathbf{dz}^u \end{aligned} \tag{4.26}$$

Hence by performing $2n$ linear programs, the vectors $\mathbf{dx}^l$ and $\mathbf{dx}^u$ can be constructed. Once $\mathbf{dx}^l$ and $\mathbf{dx}^u$ have been calculated, it is a simple matter to construct the bounds, $\mathbf{x}^{1l}$ and $\mathbf{x}^{1u}$ for $X^1(M,\mathbf{z}^o)$ as

$$\mathbf{x}^{1l} = \hat{\mathbf{x}} + \mathbf{dx}^l \tag{4.27}$$

$$\mathbf{x}^{1u} = \hat{\mathbf{x}} + \mathbf{dx}^u \tag{4.28}$$

Three special cases can be identified:

(i) $g(\hat{\mathbf{x}}) = \mathbf{z}^o$ where $\mathbf{z}^o$ is the measurement vector from which $\hat{\mathbf{x}}$ was calculated as a state estimate. When this situation occurs, $\mathbf{dz}^u = -\mathbf{dz}^l = \mathbf{e}^z$, in other words, $Z(M,\mathbf{z}^o)$ is symmetric about $g(\hat{\mathbf{x}})$ This symmetry is carried over to the linearised state uncertainty set $X^1(M,\mathbf{z}^o)$, hence $\mathbf{dx}^u = -\mathbf{dx}^l$. This means that only n linear programming problems need be solved, those in (4.25) say. It must be noted that a general measurement vector $\mathbf{z}^o$ will suffer from inconsistency, which means that there will be no state vector, $\mathbf{x}$, in $\mathbf{R}^n$ for which $g(\mathbf{x})$ is equal to $\mathbf{z}^o$.

(ii) M is an observable measurement and no subset of M is observable. In other words, M is a minimal measurement set. When this is the case, all $\mathbf{z}$ in $Z(M,\mathbf{z}^o)$ are consistent and so $\mathbf{z}^o$, in particular, is consistent. Hence there is an $\hat{\mathbf{x}}\in\mathbf{R}^n$ for which $g(\hat{\mathbf{x}})=\mathbf{z}^o$. If $\hat{\mathbf{x}}$ can be found, only n linear programs need be performed (this follows from the previous special case). Furthermore, when J is the Jacobian matrix for M, evaluated at $\hat{\mathbf{x}}$, and M is a minimal measurement set, J is square and non-singular. Hence there exists an inverse, J^{-1}, for J. It will now be shown how the upper and lower bounds for $DX^1(M,\mathbf{z}^o)$, and hence those for $X^1(M,\mathbf{z}^o)$, can be calculated from the individual rows of J^{-1}, without help from linear programming methods. Before this can be done, two lemmas must be proved:

Lemma 4.1 Let M be a minimal observable measurement set, $\mathbf{z}^o$ be an observed measurement vector from M, $\hat{\mathbf{x}}$ be the state estimate for $\mathbf{z}^o$ and J be the non-singular Jacobian matrix defined by M and $\hat{\mathbf{x}}$. Then, for $DX^1(M,\mathbf{z}^o)$, defined as in (4.24),

$$DX^1(M,\mathbf{z}^o) = \{\mathbf{dx}\in\mathbf{R}^n : \mathbf{dx}=J^{-1}.\mathbf{dz},\ \mathbf{dz}^l\leq \mathbf{dz}\leq \mathbf{dz}^u\} \tag{4.29}$$

Proof By definitions (4.21), (4.22), (4.24) and (4.19), $\mathbf{dx}\in DX^1(M,\mathbf{z}^o)$ if and only if $\mathbf{dz}^l\leq J.\mathbf{dx}\leq \mathbf{dz}^u$. Putting $\mathbf{dz}=J.\mathbf{dx}$, $\mathbf{dz}^l\leq J.\mathbf{dx}\leq \mathbf{dz}^u$ when and only when $J^{-1}.\mathbf{dz}=\mathbf{dx}$ and $\mathbf{dz}^l\leq \mathbf{dz}\leq \mathbf{dz}^u$. Hence, $DX^1(M,\mathbf{z}^o)$ is equal to the set $\{\mathbf{dx}\in\mathbf{R}^n : \mathbf{dx}=J^{-1}.\mathbf{dz},\ \mathbf{dz}^l\leq \mathbf{dz}\leq \mathbf{dz}^u\}$ and the lemma is proved.

Lemma 4.2 Let M be a minimal observable measurement set, $\mathbf{z}^o$ be an observed measurement vector from M, $\hat{\mathbf{x}}$ be a state estimate for $\mathbf{z}^o$ and let J be the non-singular Jacobian matrix defined by M and $\hat{\mathbf{x}}$. For $i=1,....,n$, the i^{th} element of the lower bounding vector for $DX^1(M,\mathbf{z}^o)$, dx_i^l, is given by

$$dx_i^l = \mathbf{a}^l.\mathbf{dz}^*, \quad \textit{where}$$

$$dz_j^* = \begin{cases} dz_j^u \text{ if } a_j^i < 0.0 \\ dz_j^l \textit{ otherwise} \end{cases} \tag{4.30}$$

where $\mathbf{a}^i$ is the i^{th} row of the matrix J^{-1}.

Proof By definition (4.25), dx_i^l is the minimum value for dx_i subject to $\mathbf{dx}$ being a member of $DX^1(M,\mathbf{z}^0)$. Using lemma 4.1, this is the same as the minimum value of $(J^{-1}.\mathbf{dz})_i$ subject to $\mathbf{dz}^l \leq \mathbf{dz} \leq \mathbf{dz}^u$. For any $\mathbf{dz}$, $(J^{-1}.\mathbf{dz})_i = (\mathbf{a}^i)^T.\mathbf{dz}$, where $\mathbf{a}^i$ is the i^{th} row of the matrix J^{-1}. As $\mathbf{dz}^l \leq \mathbf{dz} \leq \mathbf{dz}^u$ is the only constraint on the elements of the vector $\mathbf{dz}$, the minimum value of $(\mathbf{a}^i)^T.\mathbf{dz}$ is equal to the sum of the minimum values of $a_j^i.dz_j$ for $j=1,....,n$. The minimum value of $a_j^i.dz_j$, subject to $dz_j^l \leq dz_j \leq dz_j^u$ is just $a_j^i.dz_j^u$ or $a_j^i.dz_j^l$, depending on whether a_j^i is less than or greater than 0.0, respectively. When $a_j^i = 0.0$, $a_j^i.dz_j = 0.0$ as well, so such elements do not contribute to the value of dx_i^l. The lemma now follows.

Once the inverse of J has been calculated, lemma 4.2 can be applied to each element of $\mathbf{dx}^l$ in turn, providing a straightforward way of calculating this vector that does not rely on optimisation methods. Because of the symmetry in this situation, as discussed in an earlier part of this special case, the upper bound for $DX^1(M,\mathbf{z}^0)$, $\mathbf{dx}^u$, is equal to the negative of the lower bound. So, Lemma 4.2 need only be applied n times. The bounding vectors, $\mathbf{x}^{1l}$ and $\mathbf{x}^{1u}$, are found by adding $\hat{\mathbf{x}}$ to $\mathbf{dx}^l$ and $\mathbf{dx}^u$, by equations (4.27) and (4.28).

(iii) $DX^1(M,\mathbf{z}^0)$, and hence $X^1(M,\mathbf{z}^0)$, may be empty, even when $Z(M,\mathbf{z}^0)$ is non-empty. If this situation occurs then $Z(M,\mathbf{z}^0)$ is said to be inconsistent. This is reflected in the bounding vectors, by $x_i^{1u} \leq x_i^{1l}$ for at least one i in $1,....,n$.

The linearised state uncertainty set, $X^1(M,\mathbf{z}^0)$, is only an approximation to the true state uncertainty set, $X(M,\mathbf{z}^0)$. The question - how good an approximation is it? - now arises. To answer this question, an upper bound on the difference between x_i^u and x_i^{1u} - the upper limits on the feasible values for the i^{th} state variable in $X(M,\mathbf{z}^0)$ and $X^1(M,\mathbf{z}^0)$, respectively - is derived. This is supplied by the next lemma:

Lemma 4.3 For a measurement uncertainty set $Z(M,\mathbf{z}^o)$, where M is a minimal measurement set and $\mathbf{z}^o$ is an observed measurement vector derived from M. Let $\hat{\mathbf{x}}$ be the state estimate calculated from $\mathbf{z}^o$ and let J be the Jacobian matrix calculated at $\hat{\mathbf{x}}$. For the i^{th} state variable, $i \in \{1,...,n\}$,

$$|x_i^u - x_i^{1u}| \leq \|\mathbf{a}^i\| . \|\boldsymbol{\varepsilon}\| \tag{4.31}$$

where x_i^u and x_i^{1u} are the upper limits for the i^{th} state variable in the true and linearised state uncertainty sets respectively, $\mathbf{a}^i$ is the i^{th} row of J^{-1} and $\boldsymbol{\varepsilon}$ is a vector for which:

$$\|\boldsymbol{\varepsilon}\| = O(\|\mathbf{x} - \hat{\mathbf{x}}\|^2) \tag{4.32}$$

In (4.32), $\mathbf{x}$ is a feasible state vector in $X(M,\mathbf{z}^o)$.

Proof The first point to note is that as M is a minimal measurement set, J^{-1} and $\mathbf{a}^i$ are well defined. The lemma will be proved in two cases. In the first, it will be supposed that $x_i^u \geq x_i^{1u}$ and in the second it will be supposed that $x_i^u < x_i^{1u}$.

Case (i): $x_i^u \geq x_i^{1u}$. Let $\mathbf{x}^*$ be a vector in $X(M,\mathbf{z}^o)$ for which the i^{th} state variable attains its upper bound, ie $\mathbf{x}^*$ is a vector for which $x_i^* = x_i^u$. As $\mathbf{x}^*$ is a feasible member of $X(M,\mathbf{z}^o)$, there is a measurement vector, $\mathbf{z}^*$, in $Z(M,\mathbf{z}^o)$ for which $g(\mathbf{x}^*) = \mathbf{z}^*$. A vector $\mathbf{x}^{**}$ in $X^1(M,\mathbf{z}^o)$ can be defined equal to $J^{-1}(\mathbf{z}^* - g(\hat{\mathbf{x}})) + \hat{\mathbf{x}}$. This is just the vector in $X^1(M,\mathbf{z}^o)$ associated with $\mathbf{z}^*$ under the relationship given in the definition of $X^1(M,\mathbf{z}^o)$ by (4.18). The vector $\mathbf{x}^{**}$ is a member of $X^1(M,\mathbf{z}^o)$, so $x_i^{**} \leq x_i^{1u}$, as x_i^{1u} is the maximum value that the i^{th} state variable can take in $X^1(M,\mathbf{z}^o)$. Hence

$$|x_i^u - x_i^{1u}| \leq |x_i^u - x_i^{**}| = |x_i^* - x_i^{**}| \tag{4.33}$$

Attention is now focussed on the difference $|x_i^* - x_i^{**}|$. From the definition of $\mathbf{x}^{**}$,

$$|x_i^* - x_i^{**}| = |x_i^* - [J^{-1}.(\mathbf{z}^* - g(\hat{\mathbf{x}}))]_i + \hat{x}_i| \tag{4.34}$$

In the rhs of this equation, the vector $\mathbf{z}^*$ appears. $\mathbf{z}^*$ was defined as equal to $g(\mathbf{x}^*)$.

The Taylor approximation of (4.17), gives $g(\mathbf{x}^*) \approx g(\hat{\mathbf{x}}) + J.(\mathbf{x}^*-\hat{\mathbf{x}})$. The '$\approx$' sign in this relation can be replaced by an '$=$' sign if a small vector is included. That is, using a Taylor expansion

$$\mathbf{z}^* = g(\mathbf{x}^*) = g(\hat{\mathbf{x}}) + J.(\mathbf{x}^*-\hat{\mathbf{x}}) + \boldsymbol{\varepsilon} \tag{4.35}$$

for a vector $\boldsymbol{\varepsilon}$ of order $O(\|\mathbf{x}^*-\hat{\mathbf{x}}\|^2)$ [65]. Substituting for $\mathbf{z}^*$ in (4.34) and cancelling gives

$$|x_i^*-x_i^{**}| = |[J^{-1}\boldsymbol{\varepsilon}]_i| \tag{4.36}$$

The i^{th} element of $J^{-1}\boldsymbol{\varepsilon}$ is simply $(\mathbf{a}^i)^T.\boldsymbol{\varepsilon}$, where $\mathbf{a}^i$ is the i^{th} row of J^{-1}. Combining (4.33), (4.36) with this last remark, and using the *Cauchy-Schwartz* inequality [101] it follows that $|x_i^u-x_i^{1u}| \leq \|\mathbf{a}^i\|.\|\boldsymbol{\varepsilon}\|$, which proves the lemma in this case.

Case (ii): $x_i^u < x_i^{1u}$. Let $\mathbf{x}^\dagger$ be the vector in $X^1(M,\mathbf{z}^o)$ for which the i^{th} variable attains its upper limit. If $\mathbf{z}^\dagger$ is defined as equal to $g(\hat{\mathbf{x}}) + J.(\mathbf{x}^\dagger-\hat{\mathbf{x}})$ and $\mathbf{x}^{\dagger\dagger}$ is the state vector in $X(M,\mathbf{z}^o)$ for which $g(\mathbf{x}^{\dagger\dagger}) = \mathbf{z}^\dagger$ (this exists as M is a minimal measurement set and so $\mathbf{z}^\dagger$ must be consistent), then a similar argument to that in case (i) gives:

$$|x_i^u-x_i^{1u}| \leq |x_i^\dagger-x_i^{\dagger\dagger}| \leq \|\mathbf{a}^i\|.\|\boldsymbol{\varepsilon}\| \tag{4.37}$$

where $\mathbf{a}^i$ is the i^{th} row of J^{-1} and $\|\boldsymbol{\varepsilon}\|$ is of order $O(\|\mathbf{x}^{\dagger\dagger}-\hat{\mathbf{x}}\|^2)$. The result now follows.

It can be assumed, without loss of generality, that the Jacobian matrix is scaled so that $\|J\|$ is of order unity. When this is the case, and when J is not ill-conditioned, $\|\mathbf{a}^i\|$ will also be of order unity. This means that the maximum discrepancy between the upper limits for any of the state variables in $X(M,\mathbf{z}^o)$ and $X^1(M,\mathbf{z}^o)$, respectively, is of order $O(\|\mathbf{x}-\hat{\mathbf{x}}\|^2)$, where $\mathbf{x}$ is a feasible vector in $X(M,\mathbf{z}^o)$. In other words, the accuracy of the linearised state uncertainty set is of the same magnitude as that of the Taylor approximation in (4.17) and so the discrepancy between $\mathbf{x}^u$ and $\mathbf{x}^{1u}$ will not rise significantly when the confidence limit analysis problem is treated in this linearised form. Table 4.3, presented

later in this Chapter, gives an idea of the actual accuracy of the linearisation of the state uncertainty set for a realistic test network in a typical state of operation. The results of Lemma 4.3 only provide an upper bound for the discrepancy between $X(M,\mathbf{z}^o)$ and $X^1(M,\mathbf{z}^o)$, the true magnitude of this discrepancy will, in most cases, be much less than this bound.

Lemma 4.3 provides bounds on the accuracy of the upper limits for the true state and the linearised state uncertainty sets. In a similar way bounds can be derived for the lower limits of these sets. Also, Lemma 4.3 was stated for a minimal measurement set. This is in fact the extreme case. For an over-determined measurement set the state uncertainty sets will be smaller. So, the bounds of Lemma 4.3 can also be applied in the over-determined case.

In the general case, $2n$ linear programs are required, each involving $2m$ constraints. There are many texts covering the topic of linear programming ([28, 123] for example), and many different linear programming algorithms. Perhaps the most common algorithms for this size of problem are variations of the revised simplex method, which can be applied here. Unfortunately, a direct application of the revised simplex, or any similar linear programming algorithm, to this problem will be too time consuming. In the next section the linear programming problem involved in determination of the bounding vectors for the linearised uncertainty set is presented in a different format, allowing a more efficient implementation of the revised simplex method. After that, alternative approximate methods, not based on linear programming are presented.

4.5.3. Linear programming method

In (4.25), (4.26), (4.27) and (4.28), the confidence limit analysis problem was presented in a linear programming format. This presentation, whilst being theoretically correct, is not amenable to practical implementation. In this section an improved format for this problem is presented. It will be assumed that the measurement set M is both observable and over-determined. If M is not observable, then the state uncertainty set is unbounded. If M is a minimal observable measurement set, then confidence limit analysis can be performed more efficiently using the method given in special case (ii) of the previous section.

Without loss of generality, it can be assumed that the elements of the measurement set M are ordered so that the first n elements correspond to an observable set of measurements. For any measurement vector, $\mathbf{z} \in Z(M, \mathbf{z}^0)$, two new vectors can be defined. These are $\mathbf{z}^n \in \mathbf{R}^n$, the vector consisting of these first n elements of $\mathbf{z}$ and $\mathbf{z}^{m-n} \in \mathbf{R}^{m-n}$, the vector consisting of the remaining $m-n$ elements. In the same way define $\mathbf{dz}^n$, $(\mathbf{dz}^n)^l$, $(\mathbf{dz}^n)^u \in \mathbf{R}^n$, and $\mathbf{dz}^{m-n}$,$(\mathbf{dz}^{m-n})^l$ and $(\mathbf{dz}^{m-n})^u \in \mathbf{R}^{m-n}$. New matrices J^n and J^{m-n} can also be defined, J^n consisting of the first n rows of J and J^{m-n} the remaining rows.

Lemma 4.4 The maximisation of (4.26) is equivalent to

$$\begin{aligned} &\textit{maximise} \quad (\mathbf{a}^l)^T \cdot \mathbf{dz}^n \\ &\textit{subject to} \quad \begin{matrix} (\mathbf{dz}^n)^l \le \mathbf{dz}^n \le (\mathbf{dz}^n)^u \\ (\mathbf{dz}^{m-n})^l \le J^{m-n}(J^n)^{-1}.\mathbf{dz}^n \le (\mathbf{dz}^{m-n})^u \end{matrix} \end{aligned} \tag{4.38}$$

where $\mathbf{a}^l$ is the i^{th} row of $(J^n)^{-1}$

Proof The first point to note is that the observability of the first n measurements ensures that J^n is non-singular. So $\mathbf{a}^l$ and $J^{m-n}(J^n)^{-1}$ are well defined. Let $\mathbf{dz}^n \in \mathbf{R}^n$ be a feasible vector by the conditions in (4.38). That is $(\mathbf{dz}^n)^l \le \mathbf{dz}^n \le (\mathbf{dz}^n)^u$ and $(\mathbf{dz}^{m-n})^l \le J^{m-n}(J^n)^{-1}.\mathbf{dz}^n \le (\mathbf{dz}^{m-n})^u$. As J^n is non-singular, there is a unique

$\mathbf{dx}' \in \mathbf{R}^n$ such that $J^n.\mathbf{dx}' = \mathbf{dz}^n$. It is clear, by the first constraint of (4.38), that $(\mathbf{dz}^n)^l \leq J^n.\mathbf{dx}' \leq (\mathbf{dz}^n)^u$. Also, $J^{m-n}.\mathbf{dx}' = J^{m-n}(J^n)^{-1}.\mathbf{dz}^n$ which means, by the second constraint of (4.38), that $(\mathbf{dz}^{m-n})^l \leq J^{m-n}.\mathbf{dx}' \leq (\mathbf{dz}^{m-n})^u$. It is now clear that $\mathbf{dx}'$ satisfies the constraints of (4.26). Conversely, let $\mathbf{dx} \in \mathbf{R}^n$ be a vector satisfying the constraints of (4.26), and let $(\mathbf{dz}^n)' = J^n.\mathbf{dx}$. As $\mathbf{dx}$ is feasible by the constraints in (4.26), $(\mathbf{dz}^n)^l \leq (\mathbf{dz}^n)' \leq (\mathbf{dz}^n)^u$. Also $(\mathbf{dz}^{m-n})^l \leq J^{m-n}(J^n)^{-1}(\mathbf{dz}^n)' \leq (\mathbf{dz}^{m-n})^u$. That is $(\mathbf{dz}^n)'$ is feasible in (4.38). It has now been shown that there is a one-to-one correspondence between the feasible $\mathbf{dx}$ in (4.26) and the feasible $\mathbf{dz}^n$ in (4.38). More precisely, $\mathbf{dx}$ is feasible in (4.26) if and only if $J^n.\mathbf{dx}$ is feasible in (4.38). To complete the proof, all that remains is to show that for a feasible $\mathbf{dx}$ by (4.26) the two cost functions are the same. Let $\mathbf{dz}^n$ be feasible by (4.38) and $\mathbf{dx}' = (J^n)^{-1}\mathbf{dz}^n$. It is easy to see that $dx'_i = (\mathbf{a}^i)^T.\mathbf{dz}^n$, where $\mathbf{a}^i$ is the i^{th} row of $(J^n)^{-1}$. The proof is now complete.

In just the same way, it can be shown that the minimisation

$$\begin{aligned} &\textit{minimise} \quad (\mathbf{a}^l)^T.\mathbf{dz}^n \\ &\textit{subject to} \quad \begin{matrix} (\mathbf{dz}^n)^l \leq \mathbf{dz}^n \leq (\mathbf{dz}^n)^u \\ (\mathbf{dz}^{m-n})^l \leq J^{m-n}(J^n)^{-1}.\mathbf{dz}^n \leq (\mathbf{dz}^{m-n})^u \end{matrix} \end{aligned} \tag{4.39}$$

is equivalent to the minimisation given in (4.25). As before, the bounds $\mathbf{dx}^l$ and $\mathbf{dx}^u$ can be constructed for $DX^1(M, \mathbf{z}^o)$ by performing $2n$ maximisations and minimisations of this form. Also, the bounds $\mathbf{x}^{1l}$ and $\mathbf{x}^{1u}$ for $X^1(M, \mathbf{z}^o)$ can be calculated by adding $\hat{\mathbf{x}}$ to $\mathbf{dx}^l$ and $\mathbf{dx}^u$.

This formulation of the problem has an important advantage over the formulation of (4.26). That is, if (4.26) is implemented using linear programming methods it has $2m$ constraints whereas (4.38) has only $2(m-n)$. In water distribution monitoring systems, there is generally only a small amount of measurement redundancy, which means that $m-n << m$. A disadvantage of the second formulation is that it requires the inversion of

the matrix J^n. In confidence limit analysis, maximisations and minimisations such as these must be carried out $2n$ times (one maximisation and one minimisation for each variable) and in the second formulation J^n need only be inverted once. So if efficient sparse matrix methods are used to invert J^n (see Appendix A2 for a description of sparse matrix factorisation methods) this disadvantage quickly disappears.

The following confidence limit algorithm is based on this formulation.

Linear programming confidence limit algorithm

1. Select an observable subset of M consisting n measurements. This will be the minimal measurement set and denoted by M'. Order M with the elements of M' appearing first.

2. Re-order $\mathbf{dz}^{\mathrm{l}}$ and $\mathbf{dz}^{\mathrm{u}}$ according to the new ordering of M. Assemble $(\mathbf{dz}^{\mathrm{n}})^{\mathrm{l}}$, $(\mathbf{dz}^{\mathrm{n}})^{\mathrm{u}}$, $(\mathbf{dz}^{\mathrm{m-n}})^{\mathrm{l}}$, $(\mathbf{dz}^{\mathrm{m-n}})^{\mathrm{u}}$, J^n and J^{m-n}

3. Factorise J^n and calculate $J^{m-n}(J^n)^{-1}$ (efficient methods for calculating $J^{m-n}(J^n)^{-1}$ are described below) using sparse matrix techniques.

4. For each variable, $i=1,...,n$, calculate $\mathbf{a}^{\mathrm{l}}$, the i^{th} row of $(J^n)^{-1}$ and carry out the maximisation in (4.38) using a linear programming method. The resultant value of $(\mathbf{a}^{\mathrm{l}})^T.\mathbf{dz}^{\mathrm{n}}$ is the i^{th} element of $\mathbf{dx}^{\mathrm{u}}$. Similarly, carry out the minimisation in (4.39), to obtain the i^{th} element of $\mathbf{dx}^{\mathrm{l}}$.

5. Add $\mathbf{dx}^{\mathrm{l}}$ and $\mathbf{dx}^{\mathrm{u}}$ to $\hat{\mathbf{x}}$ to obtain $\mathbf{x}^{\mathrm{ll}}$ and $\mathbf{x}^{\mathrm{lu}}$

A few notes on the implementation of the algorithm are now given. Selecting the observable set of n measurements presents no difficulties. The set consisting of one reference pressure measurement, an inflow measurement for each inflow point and a load pseudomeasurement for each node except the reference head node will suffice. This will

leave mainly head and flow measurements in the remaining $m-n$ measurements. The linear programming step may benefit from another selection of the n element observable measurement set, for instance putting the most restrictive measurements in the first n, but this would require more time being spent on checking observability.

As J^n is a sparse matrix, a lot of computation time can be saved by employing sparse matrix techniques in the factorisation such as those described in Appendix A2. This will also reduce the amount of storage space required and increase numerical stability. Also $(J^n)^{-1}$ need not be calculated explicitly. It will be seen that $(J^n)^{-1}$ is required one row at a time. This can be done by factorising $(J^n)^{-1}$ and then back-substituting to solve

$$\mathbf{i}^T = (\mathbf{a}^l)^T . J^n \tag{4.40}$$

where $\mathbf{i} \in \mathbf{R}^n$ is the vector which has 1.0 in its i^{th} position and 0.0 everywhere else, $\mathbf{a}^l$ is then the i^{th} row of $(J^n)^{-1}$. The same factorisation of $(J^n)^{-1}$ can be used to calculate $J^{m-n}(J^n)^{-1}$. J^{m-n} is sparse, so it can be assumed that there are only a small number of non-zero entries in its i^{th} row, $b(i,j_1),...,b(i,j_s)$, for a small integer s, say. It is easy to check that the i^{th} row of $J^{m-n}(J^n)^{-1}$, $\mathbf{c}^l$, is given by

$$\mathbf{c}^l = \sum_{k=1}^{s} b(i,j_k).\mathbf{a}^{jk} \tag{4.41}$$

where $\mathbf{a}^{jk}$ is the $j^{k\,th}$ row of $(J^n)^{-1}$. It has already been shown how the $j^{k\,th}$ row of $(J^n)^{-1}$ can be obtained from the factorised version of $(J^n)^{-1}$. Due to the way the observable n element measurement set is chosen, most of the measurements which correspond to the rows of J^{m-n} are pressure or flow measurements. This means that many of the rows of J^{m-n} will have only one or two entries which makes the calculation of $J^{m-n}(J^n)^{-1}$, by this method, very efficient computationally.

Each of the maximisations and minimisations in step 4 of the algorithm requires an initial feasible vector, $\mathbf{dz}^n$, if a linear programming method such as the revised simplex method is to be used. Finding a vector, $\mathbf{dz}^n$, that satisfies the constraints in (4.38) and

(4.39) is equivalent to finding a vector $\mathbf{dx}'$ for which $\mathbf{dz}^{l} \leq J.\mathbf{dx}' \leq \mathbf{dz}^{u}$. This problem can be posed as the following linear programming problem:

$$\begin{aligned} &\textit{Minimise} \sum_{i=1}^{m} (e_{1i} + e_{2i}) \\ &\qquad J.\,\mathbf{dx} + \mathbf{e}_1 - \mathbf{f}_1 = \mathbf{dz}^{l} \\ &\textit{subject to} \quad -J.\,\mathbf{dx} + \mathbf{e}_2 - \mathbf{f}_2 = -\mathbf{dz}^{u} \\ &\qquad \mathbf{e}_1,\ \mathbf{e}_2,\ \mathbf{f}_1,\ \mathbf{f}_2 \geq 0 \end{aligned} \tag{4.42}$$

where $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{f}_1$ and $\mathbf{f}_2$ are m-dimensional slack variable vectors. There is a feasible $\mathbf{dx}'$ vector if and only if there is a solution to this problem with a zero value for the cost function. In this situation the feasible $\mathbf{dx}'$ is just $\mathbf{dx}$ from the solution and the initial feasible $\mathbf{dz}^{n}$ vector is given by $J^{n}.\mathbf{dx}'$. When the cost of the minimal solution to this problem is less than zero it can be said that there is no feasible $\mathbf{dx}'$ vector and so $DX^{1}(M, \mathbf{z}^{o})$ is empty.

Within the linear programming procedure used to perform the maximisations and minimisations given in (4.38) and (4.39), many iterations must be performed. For the test network shown in appendix A1 several hundred iterations were required when a general initial feasible vector such as the $\mathbf{dz}^{n}$, described above, was used. Careful selection of the initial feasible vector at each maximisation or minimisation will lead to a considerable reduction in the number of iterations that need be performed. Let $\mathbf{dz}^{n}$ be the feasible vector in $DZ(M, \mathbf{z}^{o})$ which maximises the value $(\mathbf{a}^{l})^{T}.\mathbf{dz}^{n}$ in (4.38) and so provides the upper bound on the i^{th} state variable. Suppose further that the i^{th} state variable is a pressure variable at node j. Generally speaking, this vector $\mathbf{dz}^{n}$ will also provide near-maximal values for the pressure variables at nodes adjacent to j. A similar observation holds for flow variables in adjacent pipes. These observations provide a way of reducing the number of iterations in each maximisation and so improve the run-time performance. If the set of independent state variables being used are the pressure and inflow variables, then the nodes of the network can be ordered so that, in as many cases as possible, each node is adjacent in the network to the one that precedes it in the ordering. The state variables can then be

ordered correspondingly. If at each maximisation (or minimisation) the initial feasible vector used is the vector that was the maximal solution vector in the previous maximisation (or minimisation), then the number of iterations required is cut considerably. A similar method can be used when the state variables are the flow variables. Implementing a method based on this idea results in approximately 10-fold improvement in run-time for the test network shown in Appendix A1.

4.5.4. Sensitivity matrix method

Some applications, semi-automated control or on-line decision support for instance, need a confidence limit analysis procedure that can produce uncertainty bounds in just a few seconds. The linear programming method described in section 4.5.3 may be considered a little too slow. In this section an alternative method is considered. This is is referred to as the sensitivity matrix method.

When the measurement set M is minimal (ie it is observable and contains no observable subset), the linearised uncertainty bounds can be calculated without recourse to a linear programming procedure. It is explained how this can be done in the discussion of special case (ii) of section 4.5.2. So in these circumstances, confidence limit analysis can be carried out much more rapidly than in the general case when the linear programming algorithm of the previous section has to be used. The short cut can be taken (when M is minimal) because the Jacobian matrix J is square and invertible. So, any $\mathbf{dx} \in DX^1(M, \mathbf{z}^o)$ is given by $J^{-1}.\mathbf{dz}$ for some $\mathbf{dz} \in DZ(M, \mathbf{z}^o)$. In general, M is over-determined and so J is an m by n matrix of rank n. When J has this form it has no inverse. The lack of inverse is not due to a shortage of information, rather, there is a surfeit of measurement data. As there is no proper inverse for J, a pseudo-inverse must be used. Let $\mathbf{dx} \in \mathbf{R}^n$ and $\mathbf{dz} \in \mathbf{R}^m$, for which $J.\mathbf{dx} = \mathbf{dz}$, then $J^T J.\mathbf{dx} = J^T.\mathbf{dz}$ and so

$$\mathbf{dx} = (J^T J)^{-1} J^T.\mathbf{dz} \tag{4.43}$$

This equation is well defined because when J is of rank n, J^TJ is both square and invertible. The matrix $(J^TJ)^{-1}J^T$ is the psuedo-inverse that will be used when J^{-1} is not well defined. In fact, when J^{-1} is well defined $(J^TJ)^{-1}J^T$ is equal to J^{-1}. $(J^TJ)^{-1}J^T$ will be referred to as the **sensitivity matrix** as its $(i,j)^{th}$ element relates the sensitivity of the i^{th} element of the state vector to changes in the j^{th} element of the measurement vector.

A new approximate linearised state uncertainty set, $X^2(M,\mathbf{z}^o)$, can be defined as follows:

$$X^2(M,\mathbf{z}^o) := \{\mathbf{x}\in\mathbf{R}^n : \mathbf{x}=\hat{\mathbf{x}}+\mathbf{dx},\ \mathbf{dx}=(J^TJ)^{-1}J^T.\mathbf{dz},\ \mathbf{dz}\in DZ(M,\mathbf{z}^o)\} \tag{4.44}$$

This set can be seen as an approximation to the true linearised state uncertainty set, $X^1(M,\mathbf{z}^o)$. The next lemma provides more detail about the relationship between $X^1(M,\mathbf{z}^o)$ and $X^2(M,\mathbf{z}^o)$.

Lemma 4.5 For $X^2(M,\mathbf{z}^o)$ defined as in (4.44) and $X^1(M,\mathbf{z}^o)$, defined as in (4.18),

$$X^1(M,\mathbf{z}^o) \subseteq X^2(M,\mathbf{z}^o) \tag{4.45}$$

and when M is a minimal observable measurement set

$$X^1(M,\mathbf{z}^o) = X^2(M,\mathbf{z}^o) \tag{4.46}$$

Proof Let $\mathbf{dx}+\hat{\mathbf{x}}\in X^1(M,\mathbf{z}^o)$, then $J.\mathbf{dx}\in DZ(M,\mathbf{z}^o)$ by (4.19) and (4.23). Put $\mathbf{dz}=J.\mathbf{dx}$. The element of $X^2(M,\mathbf{z}^o)$ generated by $\mathbf{dz}$ is just $\hat{\mathbf{x}}+(J^TJ)^{-1}J^T.\mathbf{dz} = \hat{\mathbf{x}}+(J^TJ)^{-1}J^TJ.\mathbf{dx} = \hat{\mathbf{x}}+\mathbf{dx}$. Hence, $\hat{\mathbf{x}}+\mathbf{dx}\in X^2(M,\mathbf{z}^o)$ and (4.45) follows. Proof of the second part, (4.46), is straightforward, following directly from the observation that when M is minimal, $(J^TJ)^{-1}J^T$ degenerates to J.

When M is over-determined there may be vectors $\mathbf{dz}\in Z(M,\mathbf{z}^o)$ that are inconsistent. For such a vector there can be no $\mathbf{dx}\in\mathbf{R}^n$ with $J.\mathbf{dx}=\mathbf{dz}$. So, in particular $J(J^TJ)^{-1}J^T.\mathbf{dz}$ is not equal to $\mathbf{dz}$. It may even be the case that $J(J^TJ)^{-1}J^T.\mathbf{dz}$ is not contained $DZ(M,\mathbf{z}^o)$. It is these vectors that account for the difference between $X^2(M,\mathbf{z}^o)$ and $X^1(M,\mathbf{z}^o)$. That is,

$\hat{\mathbf{x}}+\mathbf{dx}\in X^2(M,\mathbf{z}^o)-X^1(M,\mathbf{z}^o)$ if and only if $\mathbf{dx}=(J^TJ)^{-1}J^T.\mathbf{dz}$ for some $\mathbf{dz}\in DZ(M,\mathbf{z}^o)$ with $J(J^TJ)^{-1}J^T.\mathbf{dz}$ not a member of $DZ(M,\mathbf{z}^o)$. Although $X^2(M,\mathbf{z}^o)$ is not identical to $X^1(M,\mathbf{z}^o)$ it can be used to form bounds for the linearised state uncertainty set. It will not be the case that these bounds are the tightest that can be obtained but at least they will not rule out any feasible state vector from the state uncertainty box.

Bounding vectors for $X^2(M,\mathbf{z}^o)$, denoted $\mathbf{x}^{2l}$ and $\mathbf{x}^{2u}$, can be defined just as $\mathbf{x}^{1l}$ and $\mathbf{x}^{1u}$ were for the true linearised state uncertainty set $X^1(M,\mathbf{z}^o)$. The following algorithm provides a way of calculating these vectors.

Sensitivity matrix confidence limit algorithm

1. Set $i=0$.

2. Factorise the matrix J^TJ. (This can be done using an augmented matrix).

3. Set $i=i+1$.

4. Calculate $\mathbf{b}^i$, the i^{th} row of the sensitivity matrix $(J^TJ)^{-1}J^T$. (In the implementation details, following the description of the algorithm, it is explained how this can be done efficiently, taking account of the sparsity of J and using the factorisation of the augmented matrix mentioned in step 2.)

5. Put $x_i^{2u}=(\mathbf{b}^i)^T.\mathbf{dz}^+ + \hat{x}_i$, where

$$dz_j^+ = \begin{cases} dz_j^u & \text{if } b_j^i > 0.0 \\ dz_j^l & \text{otherwise} \end{cases} \tag{4.47}$$

Put $x_i^{2l}=(\mathbf{b}^i)^T\mathbf{dz}^- + \hat{x}_i$, where

$$dz_j^- = \begin{cases} dz_j^l & \text{if } b_j^i > 0.0 \\ dz_j^u & \text{otherwise} \end{cases} \tag{4.48}$$

6. If $i<n$, go back to step 3. Otherwise stop.

This algorithm bears a striking resemblance to the method of calculating the bounds for $X^1(M, \mathbf{z}^o)$ described in special case (ii) in section 4.5.2. In fact, justification of equations (4.47) and (4.48) follows as a corollary to lemma 4.2. Although, in this case, there is not necessarily symmetry about $\hat{\mathbf{x}}$ as $g(\hat{\mathbf{x}})$ may not equal $\mathbf{z}^o$)

As with all algorithms that involve the manipulation of the Jacobian matrix J, sparse matrix methods may be used. (see Appendix A2). These will reduce run-time, reduce storage requirements and increase numerical stability. Because of storage limitations, it is not practical to store each row of the sensitivity matrix $(J^TJ)^{-1}J^T$. So each row is calculated only when it is required. First of all, the matrix J^TJ must be factorised. The j^{th} element, b^i_j, of the i^{th} row of $(J^TJ)^{-1}J^T$ is given by

$$b^i_j = (\mathbf{c}^i)^T.\mathbf{d}^j \tag{4.49}$$

where $\mathbf{c}^i$ is the vector corresponding to the i^{th} row of $(J^TJ)^{-1}$ and $\mathbf{d}^j$ is the j^{th} column of J^T As J is sparse, only a small number of the entries in $\mathbf{d}^j$ are non-zero. Let these be denoted by the set $\Omega_j := \{k \in \{1,...,n\} : d^j_k \neq 0.0\}$. Then

$$b^i_j = \sum_{k \in \Omega_j} c^i_k.d^j_k \tag{4.50}$$

For (4.49) and (4.50), the vector $\mathbf{c}^i$ (representing the i^{th} row of $(J^TJ)^{-1}$) must be calculated. The matrix $(J^TJ)^{-1}$ is not calculated directly. Each row of $(J^TJ)^{-1}$ is calculated only when it is required (again this is due to storage practicalities) and this is done using an augmented matrix format. It is easy to see that $\mathbf{c}^i$ is given by solving the equation

$$\mathbf{i}^T = (\mathbf{c}^i)^T.(J^TJ) \tag{4.51}$$

where $\mathbf{i}$ is the vector that has 1.0 in its i^{th} position and 0.0 in all other positions. Solution of (4.51) can again be carried out using sparse matrix techniques in an augmented matrix format.

4.5.5. Ellipsoid method

In [124], Schweppe suggests a method for confidence limit analysis for linear systems with unknown-but-bounded observation errors. This method is based on the iterative shrinking of ellipsoids, and will be referred to as the **ellipsoid method.** The procedure has also been considered by other authors in [18, 20, 59, 100, 102, 103, 104].

Mathematically, an ellipsoid, E^t, is a region of space defined as follows:

$$E^t := \{\mathbf{x} \in \mathbf{R}^n : (\mathbf{x}-\mathbf{x}^t)P_t^{-1}(\mathbf{x}-\mathbf{x}^t) \leq 1.0\} \tag{4.52}$$

for some $\mathbf{x}^t \in \mathbf{R}^n$ and some symmetric and positive-definite matrix P_t, of dimension n by n. E^t is therefore, a region of $\mathbf{R}^n$ centred on $\mathbf{x}^t$. The aim of the ellipsoid method is to start with a large ellipsoid (usually a n-dimensional sphere) that contains the whole state uncertainty set, and then to generate a sequence of ellipsoids, decreasing in size, leading to one that fits the state uncertainty set as tightly as possible. Using an ellipsoid as an approximation to the state uncertainty set provides a simple and concise description of what can be a very complicated set. The ellipsoid algorithm itself, has the advantages of being sequential, mathematically and conceptually simple and can be very fast computationally.

Application of the ellipsoid algorithm to confidence limit analysis in water distribution systems is now described. The first point to note is that it is a linear method and so the state uncertainty set to be approximated is the set $X^1(M, \mathbf{z}^o)$, for a measurement set M and measurement vector $\mathbf{z}^o$. An ellipsoid that certainly contains $X^1(M, \mathbf{z}^o)$ is used as the starting ellipsoid E^t. This ellipsoid may be the n-dimensional sphere centred at the state estimate $\hat{\mathbf{x}}$ (this is the state estimate generated by $\mathbf{z}^o$) with a suitably large diameter, α. In this case $P_0 = \alpha.I$, where I is the n by n identity matrix. The 'observations' in confidence limit analysis are the linearised measurement constraints provided by (4.19), which can be rewritten as

$$\mathbf{z}^l - g(\hat{\mathbf{x}}) + J.\hat{\mathbf{x}} \leq J.\mathbf{x} \leq \mathbf{z}^u - g(\hat{\mathbf{x}}) + J.\hat{\mathbf{x}} \tag{4.53}$$

for all $\mathbf{x}$ in $X^1(M, \mathbf{z}^0)$. In this equation, $\mathbf{z}^l - g(\hat{\mathbf{x}}) + J.\hat{\mathbf{x}}$ and $\mathbf{z}^u - g(\hat{\mathbf{x}}) + J.\hat{\mathbf{x}}$ are constant vectors and so can be pre-calculated. (4.53) represents m constraints, bounding $J.\mathbf{x}$ above and below. Each of these is taken in turn and used to modify the current ellipsoid. Suppose that the t^{th} constraint is being used to update the $t-1^{st}$ ellipsoid, $t \in \{2,...,m\}$, and that E^{t-1} contains $X^1(M, \mathbf{z}^0)$. The region bounded by the this constraint also contains the uncertainty set $X^1(M, \mathbf{z}^0)$. So $X^1(M, \mathbf{z}^0)$ is contained in the intersection of these two regions as is shown in fig 4.2. A new ellipsoid, E^t, can be produced which contains the intersection of E^{t-1} and the region bounded by the constraint's hyperplanes. E^t is the ellipsoid $\{\mathbf{x} \in \mathbf{R}^n : (\mathbf{x}-\mathbf{x}^t)P_t^{-1}(\mathbf{x}-\mathbf{x}^t) \leq 1.0\}$, where

$$\mathbf{x}^t = \mathbf{x}^{t-1} + (\rho_t \nu_t / (e_t^z)^2) P'_{t-1}.\mathbf{a}^t \tag{4.54}$$

$$P_t = (1 + \rho_t - (\rho_t \nu_t / ((e_t^z)^2 + \rho_t g_t))) P'_{t-1} \tag{4.55}$$

$$P'_{t-1} = (I - (\rho_t / ((e_t^z)^2 + \rho_t g_t)) P_{t-1} \mathbf{a}^t . (\mathbf{a}^t)^T) P_{t-1} \tag{4.56}$$

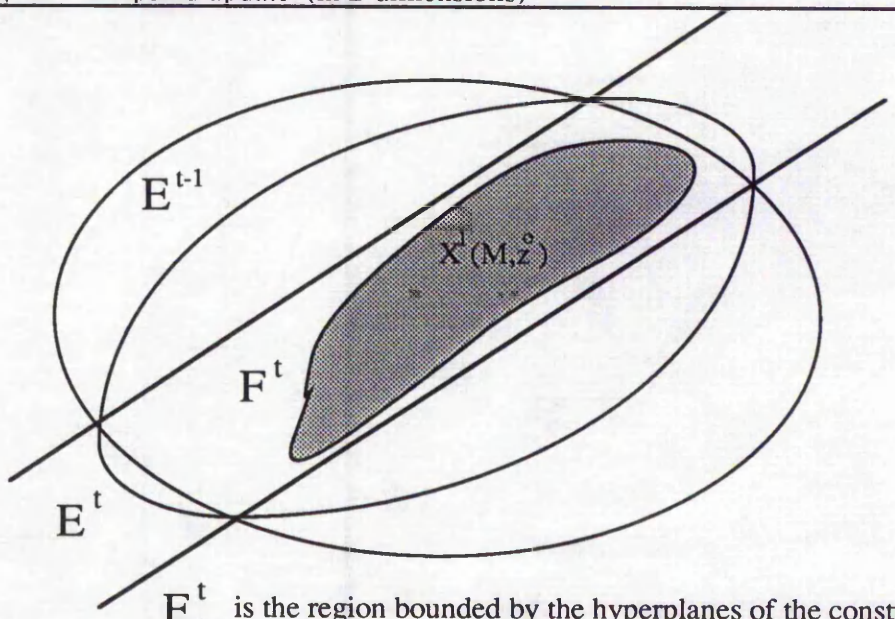
$$g_t = (\mathbf{a}^t)^T P_{t-1} \mathbf{a}^t \tag{4.57}$$

$$\nu_t = 0.5(z_t^u - z_t^l) - (g(\mathbf{x}^t))_t + (J.\mathbf{x}^t)_t - (\mathbf{a}^t)^T . \mathbf{x}^{t-1} \tag{4.58}$$

In these equations, P_{t-1} and $\mathbf{x}^{t-1}$ are the positive definite matrix and centre vector, respectively, for the previous ellipsoid, E^{t-1}, and ρ_t can be any non-negative real value. The value, e_t^z used in these equations, is the t^{th} element of the measurement error vector introduced in section 4.4. Derivation of these formulae is contained in [102]. It should be noted that, despite the fact that (4.52) refers to the P_t^{-1} matrix in its inverted form and (4.54) to (4.58) do not, no matrix inversion is involved in the algorithm. In fact, the matrix P_t^{-1} need never be known as all updating is performed using matrices P_{t-1} and P_t

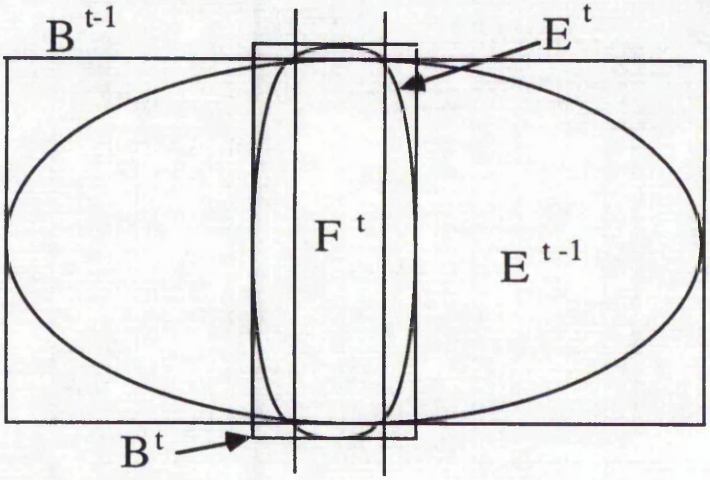
The parameter ρ_t, which appears in several of the updating formulae, has not yet been fixed. For any non-negative, real ρ_t, the updated ellipsoid, E^t, will contain the intersection of E^{t-1} with the region bounded by the t^{th} measurement constraints. For $\rho_t = 0.0$, it is easily seen that E^t is just the same as E^{t-1}. So this choice of ρ_t leads to no improvement. A more reasonable choice for ρ_t would be the value that minimises the size

Fig 4.2 : Ellipsoid update (in 2-dimensions)



F^t is the region bounded by the hyperplanes of the constraint
E^t is the updated ellipsoid.

Fig 4.3: Ellipsoid update does not always lead to improvement in all variables.



The new bounds, marked by B^t, are tighter in the horizontal direction than the old bounds, marked by B^{t-1}, but are not as tight in the vertical direction

of E^t in some way. Such a selection strategy will give the algorithm better convergence properties. In [59], Fogel and Huang make two suggestions. The first involves the solution of a quadratic equation in ρ_t and produces the ellipsoid of minimum volume. The second requires the solution of a cubic equation and minimises the sum of squares of the semi-axis lengths in E^t.

On termination of the algorithm, the confidence limits for each variable are easily calculated from the final positive-definite matrix, P^t, and the final centre vector, $\mathbf{x}^t$. These are given by Norton in [103]:

$$x_i^{1u} = x_i^t + \sqrt{P_t(i,i,)}$$
$$x_i^{1l} = x_i^t - \sqrt{P_t(i,i,)} \qquad (4.59)$$

In some situations, tight bounds can be found by processing each measurement constraint only once, in which case only m steps are required. Although, in [18] Bona and Belforte suggest that a further reduction in the bounds is often possible by reprocessing some or all of the constraints. Reprocessing may be carried out until no further improvement is made. Another comment made by Bona and Belforte in [18] is that the order in which the constraints are processed has an effect on the rate of convergence of the algorithm. Further comments and suggestions for algorithm modification are made by Mo and Norton in [100].

A version of the ellipsoid algorithm has been implemented as a routine for confidence limit analysis in water distribution systems. Although the run-time of this routine is compares favourably with other linear confidence limit routines, its accuracy in bounding the state uncertainty set is poor. Other approximate methods, that can bound $X^1(M,\mathbf{z}^o)$ more accurately with similar computational cost, can be easily be derived. In [18, 100, 102, 103, 104], it is reported that the performance of the ellipsoid algorithm is good, especially when computation time is compared with that of the linear programming method for instance. The poor performance of the ellipsoid algorithm in this particular application is due to the nature of the Jacobian matrix and hence the structure of the measurement

constraints. Since J is sparse, each measurement constraint only bounds a few of the variables. In the ellipsoid algorithm, constraints are considered individually and so can only improve confidence limits on the few variables that they bound explicitly. As an example, consider the ellipsoid update represented in fig 4.3. This is a 2-dimensional example in which the observation hyperplanes only bound one of the variable (ie, variables are bound in the horizontal direction by the hyperplanes but not in the vertical direction). As a result, the new ellipsoid, E^t, produces tighter confidence limits than E^{t-1} in the horizontal direction but looser ones in the vertical direction. When this idea is extended to many dimensions, only a few of which are bound by each constraint, it is easy to see that at each iteration the majority of the variables will have their confidence limits increased and only a minority will have them reduced.

Because of the emphasis placed on the ellipsoid method as an efficient confidence limit algorithm in [18, 20, 59, 100, 102, 103, 104, 124] it was considered important to include a section on the ellipsoid method even though its results proved negative.

4.5.6. Linearised confidence limit methods for the derived state variables

In sections 4.5.3 and 4.5.4, a linear programming method and a sensitivity matrix method were presented as linearised confidence limit algorithms. These algorithms were described in terms of the set of independent state variables only. By initially restricting attention in this way, the basic concepts and ideas behind the methods could be explained more clearly and informatively. The extension of these two methods, to include the derived state variables, is explained in this section. This extension adds only a little to the complexity of the algorithms and is included for completeness.

In the same way that the network function, $g(.)$, was linearised in equation (4.17), the derived state function, $f(.)$, can be linearised around a state vector, $\hat{\mathbf{x}}$. This linearisation takes the form

$$f(\mathbf{x}) \approx F.(\mathbf{x}-\hat{\mathbf{x}}) + f(\hat{\mathbf{x}}) \tag{4.60}$$

where F is the N by n Jacobian matrix for the vector function $f(.)$, evaluated at $\hat{\mathbf{x}}$. Again, $\hat{\mathbf{x}}$ is chosen as the state estimate generated by $\mathbf{z}^0$. Using the linearised approximation and the definition of the derived state uncertainty set (4.9), a new, approximate derived state uncertainty set, $Y^1(M,\mathbf{z}^0)$, can be defined as

$$Y^1(M,\mathbf{z}^0) := \{\mathbf{y}\in\mathbf{R}^N : \mathbf{y}=F.(\mathbf{x}-\hat{\mathbf{x}})+f(\hat{\mathbf{x}}),\ \mathbf{x}\in X(M,\mathbf{z}^0)\} \tag{4.61}$$

A further approximation can be made by replacing the condition (in (4.61)) that $\mathbf{x}\in X(M,\mathbf{z}^0)$ by the condition that $\mathbf{x}\in X^1(M,\mathbf{z}^0)$, where $X^1(M,\mathbf{z}^0)$ is the linearised state uncertainty set of equations (4.18) and (4.19). Changing notation, replacing $\mathbf{x}-\hat{\mathbf{x}}$ by $\mathbf{dx}$, and using definition (4.24) gives

$$Y^1(M,\mathbf{z}^0) \approx \{\mathbf{y}\in\mathbf{R}^N : \mathbf{y}=F.\mathbf{dx}+f(\hat{\mathbf{x}}),\ \mathbf{dx}\in DX^1(M,\mathbf{z}^0)\} \tag{4.62}$$

Equation (4.62) gives the form of the linearised, derived state uncertainty set that will be used.

Firstly, the extension of the linear programming confidence limit algorithm will be considered. For a vector $\mathbf{dx}\in\mathbf{R}^n$, it was shown in the proof of Lemma 4.4, that the condition that $\mathbf{dz}^l \le J.\mathbf{dx} \le \mathbf{dz}^u$ is equivalent to the condition that $\mathbf{dx}=(J^n)^{-1}\mathbf{dz}^n$ for some $\mathbf{dz}^n\in\mathbf{R}^n$ with $(\mathbf{dz}^n)^l \le \mathbf{dz}^n \le (\mathbf{dz}^n)^u$ and $(\mathbf{dz}^{m-n})^l \le J^{m-n}(J^n)^{-1}\mathbf{dz}^n \le (\mathbf{dz}^{m-n})^u$. (The notation used here is defined immediately before Lemma 4.4). Suppose that $\mathbf{dz}^n\in\mathbf{R}^n$ satisfies these conditions. Clearly, $\mathbf{dx}$, defined as $(J^n)^{-1}\mathbf{dz}^n$, is a member of $DX^1(M,\mathbf{z}^0)$ and so $F.\mathbf{dx}+f(\hat{\mathbf{x}})$ is a member of $Y^1(M,\mathbf{z}^0)$. $F.\mathbf{dx}$ is equal to $F(J^n)^{-1}\mathbf{dz}^n$. For a particular derived state variable, y_i say for $i\in\{1,...,N\}$, finding its upper confidence limit is simply a matter of performing the following linear program and adding $f(\hat{\mathbf{x}})$ to the result.

$$\begin{aligned} &\textit{maximise}\ \ (\mathbf{d}^i)^T.\mathbf{dz}^n \\ &\textit{subject to}\ \ \begin{matrix}(\mathbf{dz}^n)^l \le \mathbf{dz}^n \le (\mathbf{dz}^n)^u, \text{ and} \\ (\mathbf{dz}^{m-n})^l \le J^{m-n}(J^n)^{-1}.\mathbf{dz}^n \le (\mathbf{dz}^{m-n})^u\end{matrix} \end{aligned} \tag{4.63}$$

where $\mathbf{d}^i$ is the i^{th} row of the matrix $F(J^n)^{-1}$. The lower confidence limit for y_i can be found by minimising the same cost function, subject to the same conditions. To modify the linear programming confidence limit algorithm so that lower and upper bounds, $\mathbf{y}^{1l}$ and $\mathbf{y}^{1u}$, for $Y^1(M,\mathbf{z}^o)$ can be calculated, all that needs to be done is to insert an extra step between step 4 and step 5. This step, step 4a say, is as follows:

4a. For each derived state variable, $i=1,...,N$, calculate $\mathbf{d}^i$, the i^{th} row of $F.(J^n)^{-1}$ and perform the maximisation of (4.63). Adding $f(\hat{\mathbf{x}})$ to the result gives the upper confidence limit for y_i. The lower confidence limit for y_i is found by performing the corresponding minimisation, and again adding $f(\hat{\mathbf{x}})$ to the result.

Extension of the sensitivity matrix method is now considered. In section 4.5.4, a set $X^2(M,\mathbf{z}^o)$ was defined as an approximation to the state uncertainty set, $X(M,\mathbf{z}^o)$. The same can be done for the derived state uncertainty set. Define

$$Y^2(M,\mathbf{z}^o) := \{\mathbf{y}\in\mathbf{R}^N : \mathbf{y}=F.\,\mathbf{dx}+f(\hat{\mathbf{x}}),\ \mathbf{dx}=(J^TJ)^{-1}J^T\mathbf{dz},\ \mathbf{dz}\in DZ(M,\mathbf{z}^o)\} \tag{4.64}$$

As before, $Y^2(M,\mathbf{z}^o)$ approximates the derived state uncertainty set $Y(M,\mathbf{z}^o)$. Definition (4.64) can be rewritten to give

$$Y^2(M,\mathbf{z}^o) := \{\mathbf{y}\in\mathbf{R}^N : \mathbf{y}=F(J^TJ)^{-1}J^T\mathbf{dz}+f(\hat{\mathbf{x}}),\ \mathbf{dz}\in DZ(M,\mathbf{z}^o)\} \tag{4.65}$$

In this form it can be seen that by replacing $(J^TJ)^{-1}J^T$ by $F(J^TJ)^{-1}J^T$ the sensitivity matrix confidence limit algorithm can be modified to produce bounding vectors, $\mathbf{y}^{2l}$ and $\mathbf{y}^{2u}$, for $Y^2(M,\mathbf{z}^o)$. All that needs to be done is to replace $\mathbf{b}^i$ by the i^{th} row of $F(J^TJ)^{-1}J^T$ in step 4, replace $\mathbf{x}^{2l}$, $\mathbf{x}^{2u}$ and $\hat{\mathbf{x}}$ by $\mathbf{y}^{2l}$, $\mathbf{y}^{2u}$ and $f(\hat{\mathbf{x}})$, respectively, in step 5 and to replace n by N in step 6.

4.6. RESULTS

Using the test network shown in Appendix A1, results for the Monte Carlo, the linear programming and the sensitivity matrix confidence limit algorithms have been calculated. These are presented and discussed in this section.

Two different measurement sets are considered, both for the same operating state shown in table 4.1. The first measurement set, M1, is a minimal measurement set, consisting of a nodal consumption estimate for all but one of the nodes, an inflow measurement for each of the inflow nodes and one reference pressure measurement at node 160. The actual measurement and pseudomeasurement values used are shown in table 4.2. Monte Carlo and linear programming confidence limits for M1 are shown in table 4.3. These results can be used to assess the accuracy of the linearisation which forms the basis of the linear confidence limit algorithms. The second measurement set, M2, consists of the same measurements and pseudomeasurements as in M1 together with an additional set of pressure and flow measurements from meters distributed around the network. These measurement values are shown in table 4.2, and the distribution of the meters on fig 4.4. Linear programming and sensitivity matrix confidence limits for measurement set M2 are shown in table 4.4. This table can be used to compare the performance of these two algorithms as well as to assess the impact of adding meters.

As has been explained earlier in this Chapter, the true measurement vector rarely coincides with the observed measurement vector. This discrepancy is caused by meter noise or meter error in the case of real metered values, and because of the difficulty in predicting demand in the case of nodal consumption estimates. In view of this, the measurement values used - the observed measurement values in table 4.2 - are not the same as the true measurement values - those that would be expected for the true operating state. The observed measurement values were created in the following way. Firstly, a true operating state, $\mathbf{x}^t$, for the system was assumed, this is shown in column 2 of table 4.1.

Table 4.1: True and observed state estimates

Pressures (m):

Node	True state	State estimate
1	140.11	140.04
2	140.23	140.17
6	140.20	140.14
7	140.15	140.08
8	140.02	139.96
10	139.94	139.89
11	140.02	139.95
12	140.38	140.21
13	140.41	140.23
14	139.91	139.84
15	139.93	139.85
16	140.05	139.95
17	141.81	141.58
18	140.36	140.18
19	140.36	140.18
22	140.30	140.13
23	140.36	140.18
24	140.40	140.22
25	140.24	140.09
61	140.23	140.10
34	139.94	139.74
35	140.17	140.00
36	140.33	140.15
37	140.32	140.14
38	140.33	140.15
39	140.33	140.15
40	140.06	139.84
102	140.07	139.86
42	140.33	140.15
43	140.07	139.85
44	140.06	139.85
45	140.02	139.80
46	140.45	140.27
47	139.96	139.75
48	139.99	139.78
53	140.92	140.65
54	140.21	139.99
55	140.05	139.85
56	140.03	139.83
57	140.71	140.56
58	141.11	141.00
59	143.48	143.39
62	142.84	142.66
63	141.79	141.53
64	141.11	141.20

Table 4.1: (Continued)

Node	True state	State estimate
65	141.32	141.01
66	141.71	141.42
67	141.25	140.97
69	140.31	140.14
70	143.88	143.90
72	140.25	140.10
73	141.78	141.87
75	140.73	140.76
76	139.97	139.95
77	140.36	140.34
78	140.35	140.32
79	140.27	140.11
80	140.24	140.10
81	139.97	139.95
106	140.34	140.13
26	144.37	144.18
3	140.34	140.28
60	144.82	144.81
160	144.77	144.75
68	141.88	141.59
Inflows (l/s):		
Node	True state	State estimate
26	65.00	65.54
3	31.00	31.43
60	34.00	33.46
160	45.00	45.85
68	31.00	30.93

Table 4.2. Measurement data

Minimal measurement set			
Node	True	Observed	Error
Reference pressure (m):			
160	144.77	144.75	0.03
Inflows (l/s):			
26	65.00	65.54	1.29
3	31.00	31.43	0.63
60	34.00	33.46	0.69
160	45.00	45.85	0.90
68	31.00	30.93	0.61
Loads (l/s):			
1	-4.85	-5.00	1.50
2	-6.77	-5.65	1.70
6	-2.09	-1.94	0.97
7	-1.64	-0.98	0.68
8	-1.16	-0.78	0.55
10	-9.64	-8.56	2.57
11	-0.34	-0.29	0.29
12	-0.35	-0.53	0.37
13	0.50	0.79	0.55
14	-6.54	-5.38	1.61
15	-3.18	-4.40	1.76
16	-2.01	-1.46	0.73
17	-8.51	-10.27	3.08
18	-8.71	-9.69	2.91
19	0.00	0.00	0.00
22	2.35	2.36	0.94
23	-0.47	-0.49	0.49
24	-1.73	-2.01	0.80
25	-2.71	-2.21	0.89
61	-0.42	-0.22	0.22
34	-7.40	-5.96	1.79
35	-2.58	-2.48	0.99
36	1.92	2.44	0.97
37	-2.98	-2.20	0.88
38	2.36	1.64	0.82
39	-0.65	-1.00	0.50
40	-6.77	-7.48	2.25
102	-2.13	-2.14	0.86
42	-8.03	-7.68	2.30
43	-3.51	-4.56	1.82
44	-1.89	-1.97	0.99
45	-1.10	-1.62	0.81
46	-2.73	-3.19	1.28

Table 4.2. (continued)

47	-10.80	-12.82	3.85
48	-2.95	-2.47	0.99
53	-0.67	-0.84	0.59
54	-4.54	-4.79	1.92
55	-10.83	-9.47	2.84
56	0.78	0.85	0.60
57	-0.16	-0.12	0.12
58	-5.68	-4.38	1.75
59	-2.88	-2.92	1.17
62	-2.94	-3.66	1.46
63	-10.46	-10.54	3.16
64	-3.75	-2.99	1.20
65	-3.84	-3.99	1.60
66	-2.15	-2.76	1.11
67	-1.87	-2.45	0.98
69	-4.52	-6.21	1.86
70	-2.18	-1.74	0.87
72	-11.51	-11.70	3.51
73	-2.77	-2.47	0.99
75	-1.32	-1.39	0.69
76	-5.37	-4.83	1.93
77	-1.16	-0.93	0.65
78	-1.35	-1.27	0.64
79	-1.91	-2.52	1.01
80	-2.64	-2.56	1.02
81	-2.79	-2.21	0.88
106	-1.74	-2.55	1.02
26	-0.26	-0.31	0.31
3	-7.95	-8.91	2.67
60	0.58	0.79	0.55
68	-2.46	-2.55	1.02

Additional measurements

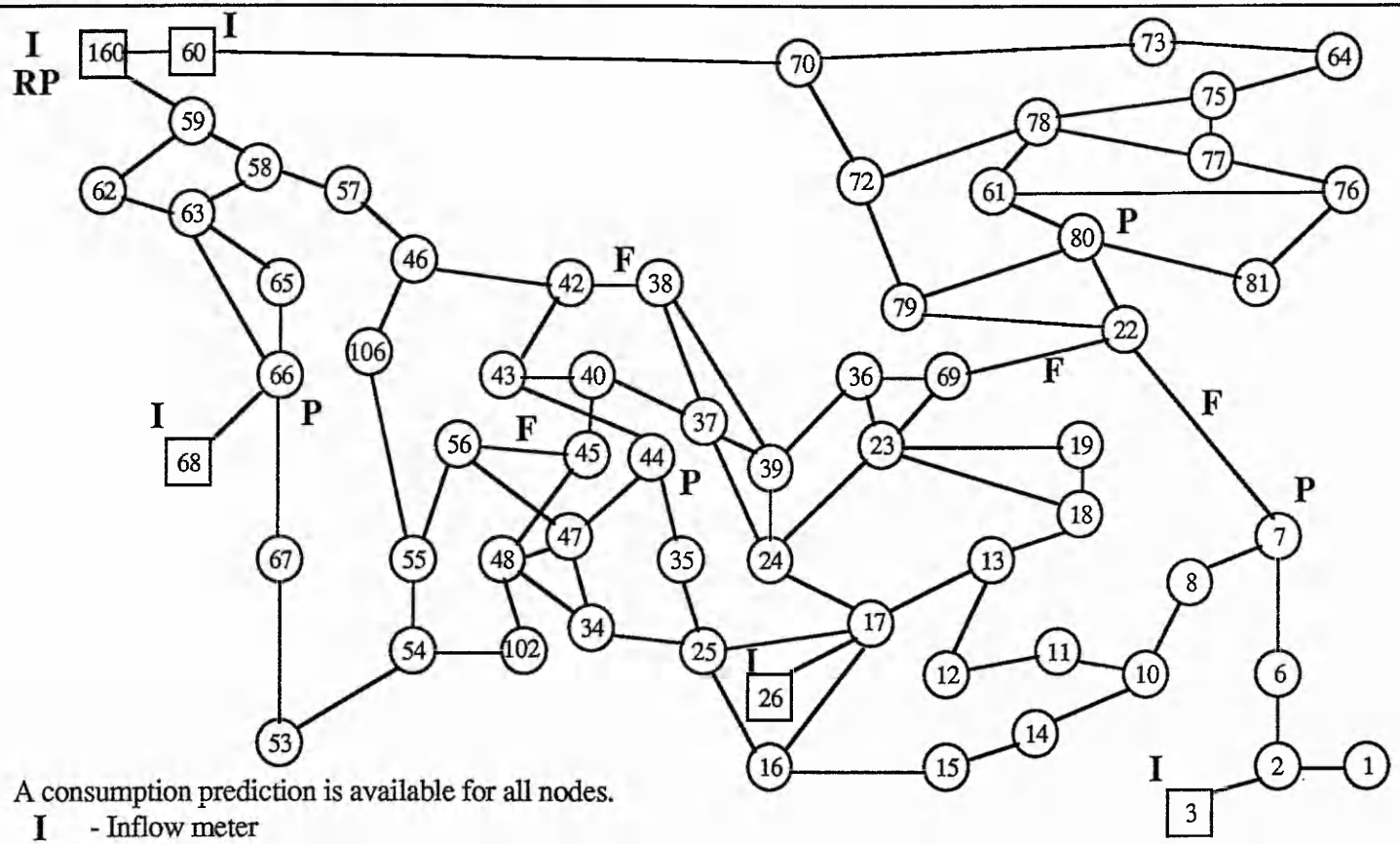
Pressures (m):

7	140.08	140.08	0.10
44	139.85	139.85	0.10
66	141.42	141.42	0.10
80	140.10	140.10	0.10

Flows (l/s)

Pipe	true	Observed	Error
22-69	-7.13	-7.13	0.50
42-38	-0.16	-0.16	0.50
7-22	1.94	1.94	0.50
56-45	0.30	0.30	0.50

Fig 4.4 : Location of real meters for test network.



A consumption prediction is available for all nodes.

I - Inflow meter

P - Pressure meter

RP - Reference pressure

F - Flow meter

Table 4.3. Confidence limits for minimal measurement set (M1)								
	Monte Carlo				Linear programming			
Node	Lower	Upper	Ave.	Error	Lower	Upper	Ave.	Error
Pressure (m):								
1	137.48	142.13	139.80	2.33	138.05	142.04	140.04	2.00
2	137.68	142.17	139.93	2.24	138.25	142.10	140.17	1.93
6	137.68	142.10	139.89	2.21	138.25	142.03	140.14	1.89
7	137.67	141.94	139.80	2.13	138.27	141.90	140.08	1.82
8	137.56	141.82	139.69	2.13	138.12	141.80	139.96	1.84
10	137.48	141.75	139.61	2.13	138.03	141.74	139.89	1.86
11	137.62	141.75	139.68	2.06	138.14	141.77	139.95	1.81
12	138.54	141.69	140.12	1.58	138.71	141.70	140.21	1.49
13	138.63	141.69	140.16	1.53	138.77	141.69	140.23	1.46
14	137.47	141.70	139.58	2.12	138.00	141.68	139.84	1.84
15	137.50	141.70	139.60	2.10	138.01	141.68	139.85	1.83
16	137.79	141.72	139.76	1.97	138.19	141.70	139.95	1.75
17	139.71	143.30	141.50	1.80	139.89	143.26	141.58	1.69
18	138.61	141.57	140.09	1.48	138.75	141.60	140.18	1.43
19	138.62	141.56	140.09	1.47	138.76	141.60	140.18	1.42
22	138.62	141.51	140.07	1.45	138.71	141.55	140.13	1.42
23	138.64	141.55	140.10	1.46	138.76	141.60	140.18	1.42
24	138.67	141.59	140.13	1.46	138.79	141.64	140.22	1.42
25	138.25	141.74	140.00	1.75	138.45	141.73	140.09	1.64
61	138.53	141.51	140.02	1.49	138.68	141.51	140.10	1.42
34	138.04	141.15	139.59	1.55	138.22	141.27	139.74	1.52
35	138.24	141.58	139.91	1.67	138.40	141.60	140.00	1.60
36	138.64	141.51	140.07	1.43	138.76	141.54	140.15	1.39
37	138.63	141.46	140.04	1.42	138.76	141.52	140.14	1.38
38	138.65	141.46	140.06	1.40	138.77	141.53	140.15	1.38
39	138.65	141.47	140.06	1.41	138.77	141.53	140.15	1.38
40	138.22	141.19	139.71	1.48	138.42	141.26	139.84	1.42
102	138.26	141.20	139.73	1.47	138.41	141.30	139.86	1.44
42	138.69	141.45	140.07	1.38	138.78	141.53	140.15	1.38
43	138.24	141.21	139.72	1.49	138.42	141.29	139.85	1.43
44	138.20	141.21	139.71	1.50	138.39	141.31	139.85	1.46
45	138.14	141.18	139.66	1.52	138.33	141.26	139.80	1.47
46	138.99	141.47	140.23	1.24	139.03	141.52	140.27	1.25
47	138.06	141.15	139.60	1.55	138.24	141.26	139.75	1.51
48	138.12	141.16	139.64	1.52	138.29	141.27	139.78	1.49
53	139.14	141.81	140.48	1.34	139.31	141.99	140.65	1.34
54	138.49	141.25	139.87	1.38	138.62	141.36	139.99	1.37
55	138.25	141.18	139.72	1.47	138.41	141.28	139.85	1.43
56	138.21	141.18	139.69	1.49	138.37	141.28	139.83	1.45
57	139.44	141.61	140.53	1.08	139.47	141.65	140.56	1.09
58	140.13	141.82	140.98	0.85	140.06	141.94	141.00	0.94
59	143.07	143.70	143.38	0.31	143.10	143.67	143.39	0.28
62	141.97	143.28	142.63	0.65	142.01	143.31	142.66	0.65
63	140.26	142.62	141.44	1.18	140.28	142.78	141.53	1.25
64	139.85	142.38	141.12	1.26	139.96	142.44	141.20	1.24

Table 4.3. (Continued)								
65	139.25	142.41	140.83	1.58	139.34	142.68	141.01	1.67
66	139.96	142.57	141.27	1.31	140.04	142.80	141.42	1.38
67	139.46	142.14	140.80	1.34	139.59	142.34	140.97	1.37
69	138.63	141.51	140.07	1.44	138.73	141.54	140.14	1.41
70	143.49	144.31	143.90	0.41	143.54	144.26	143.90	0.36
72	138.51	141.54	140.02	1.51	138.67	141.53	140.10	1.43
73	140.71	142.89	141.80	1.09	140.83	142.91	141.87	1.04
75	139.35	142.00	140.68	1.32	139.44	142.09	140.76	1.32
76	138.21	141.47	139.84	1.63	138.42	141.47	139.95	1.52
77	138.85	141.67	140.26	1.41	138.97	141.71	140.34	1.37
78	138.82	141.65	140.24	1.42	138.96	141.68	140.32	1.36
79	138.56	141.51	140.04	1.48	138.68	141.53	140.11	1.43
80	138.54	141.51	140.02	1.49	138.67	141.52	140.10	1.42
81	138.21	141.47	139.84	1.63	138.42	141.47	139.95	1.52
106	138.80	141.30	140.05	1.25	138.96	141.30	140.13	1.17
26	142.19	146.01	144.10	1.91	142.37	145.98	144.18	1.80
3	137.76	142.32	140.04	2.28	138.32	142.23	140.28	1.95
60	144.72	144.94	144.83	0.11	144.71	144.90	144.81	0.10
160	144.72	144.78	144.75	0.03	144.72	144.78	144.75	0.03
68	140.11	142.76	141.43	1.32	140.19	142.99	141.59	1.40
Inflows (l/s):								
26	64.20	66.80	65.50	1.30	64.24	66.83	65.54	1.29
3	30.80	32.00	31.40	0.60	30.80	32.05	31.43	0.63
60	32.80	34.20	33.50	0.70	32.77	34.15	33.46	0.69
160	44.90	46.70	45.80	0.90	44.95	46.75	45.85	0.90
68	30.30	31.50	30.90	0.60	30.32	31.54	30.93	0.61

Table 4.4. Confidence limits for augmented measurement set (M2)

	Linear programming				Sensitivity matrix			
Node	Lower	Upper	Ave.	Error	Lower	Upper	Ave.	Error
Pressure (m):								
1	139.61	140.48	140.04	0.43	139.60	140.49	140.04	0.45
2	139.81	140.53	140.17	0.36	139.79	140.55	140.17	0.38
6	139.81	140.47	140.14	0.33	139.79	140.48	140.14	0.34
7	139.83	140.33	140.08	0.25	139.82	140.35	140.08	0.27
8	139.86	140.06	139.96	0.10	139.86	140.06	139.96	0.10
10	139.74	140.04	139.89	0.15	139.69	140.09	139.89	0.20
11	139.80	140.11	139.95	0.15	139.79	140.11	139.95	0.16
12	139.83	140.58	140.21	0.37	139.73	140.68	140.21	0.47
13	139.84	140.62	140.23	0.39	139.73	140.73	140.23	0.50
14	139.53	140.15	139.84	0.31	139.51	140.18	139.84	0.34
15	139.51	140.19	139.85	0.34	139.49	140.21	139.85	0.36
16	139.52	140.38	139.95	0.43	139.50	140.40	139.95	0.45
17	140.95	142.21	141.58	0.63	140.89	142.27	141.58	0.69
18	139.78	140.57	140.18	0.40	139.67	140.68	140.18	0.51
19	139.78	140.57	140.18	0.40	139.67	140.68	140.18	0.51
22	139.75	140.52	140.13	0.38	139.63	140.63	140.13	0.50
23	139.78	140.57	140.18	0.39	139.67	140.68	140.18	0.51
24	139.82	140.61	140.22	0.40	139.71	140.72	140.22	0.51
25	139.55	140.63	140.09	0.54	139.53	140.65	140.09	0.56
61	139.72	140.48	140.10	0.38	139.58	140.62	140.10	0.52
34	139.11	140.38	139.74	0.64	139.07	140.42	139.74	0.67
35	139.44	140.57	140.00	0.56	139.43	140.58	140.00	0.58
36	139.77	140.54	140.15	0.38	139.65	140.65	140.15	0.50
37	139.76	140.52	140.14	0.38	139.64	140.64	140.14	0.50
38	139.77	140.54	140.15	0.38	139.65	140.65	140.15	0.50
39	139.77	140.54	140.15	0.38	139.65	140.65	140.15	0.50
40	139.27	140.41	139.84	0.57	139.23	140.45	139.84	0.61
102	139.38	140.33	139.86	0.48	139.25	140.46	139.86	0.61
42	139.77	140.54	140.15	0.38	139.65	140.65	140.15	0.50
43	139.29	140.42	139.85	0.57	139.25	140.46	139.85	0.60
44	139.27	140.43	139.85	0.58	139.24	140.46	139.85	0.61
45	139.19	140.40	139.80	0.60	139.15	140.44	139.80	0.64
46	139.87	140.68	140.27	0.40	139.78	140.77	140.27	0.49
47	139.12	140.38	139.75	0.63	139.08	140.42	139.75	0.67
48	139.17	140.39	139.78	0.61	139.13	140.43	139.78	0.65
53	140.55	140.75	140.65	0.10	140.55	140.75	140.65	0.10
54	139.88	140.10	139.99	0.11	139.47	140.51	139.99	0.52
55	139.29	140.41	139.85	0.56	139.24	140.46	139.85	0.61
56	139.25	140.41	139.83	0.58	139.20	140.45	139.83	0.63
57	140.15	140.97	140.56	0.41	140.07	141.05	140.56	0.49
58	140.58	141.42	141.00	0.42	140.51	141.50	141.00	0.50
59	143.17	143.60	143.39	0.22	143.14	143.63	143.39	0.24
62	142.28	143.04	142.66	0.38	142.26	143.06	142.66	0.40
63	141.01	142.04	141.53	0.51	140.90	142.16	141.53	0.63
64	140.83	141.57	141.20	0.37	140.54	141.86	141.20	0.66

Table 4.4. (Continued)

65	140.22	141.80	141.01	0.79	139.90	142.11	141.01	1.11
66	140.99	141.85	141.42	0.43	140.86	141.98	141.42	0.56
67	140.71	141.22	140.97	0.26	140.65	141.28	140.97	0.31
69	139.75	140.52	140.14	0.38	139.64	140.64	140.14	0.50
70	143.72	144.08	143.90	0.18	143.69	144.10	143.90	0.21
72	139.76	140.44	140.10	0.34	139.55	140.66	140.10	0.56
73	141.49	142.25	141.87	0.38	141.21	142.52	141.87	0.66
75	140.49	141.04	140.76	0.27	140.29	141.24	140.76	0.48
76	139.51	140.38	139.95	0.44	139.38	140.52	139.95	0.57
77	140.24	140.44	140.34	0.10	140.24	140.44	140.34	0.10
78	140.18	140.46	140.32	0.14	140.17	140.47	140.32	0.15
79	139.70	140.51	140.11	0.41	139.58	140.64	140.11	0.53
80	139.71	140.49	140.10	0.39	139.57	140.62	140.10	0.53
81	139.51	140.39	139.95	0.44	139.37	140.52	139.95	0.58
106	139.67	140.59	140.13	0.46	139.62	140.64	140.13	0.51
26	143.43	144.92	144.18	0.75	143.37	144.98	144.18	0.81
3	139.89	140.67	140.28	0.39	139.87	140.68	140.28	0.41
60	144.73	144.88	144.81	0.07	144.73	144.88	144.81	0.08
160	144.72	144.78	144.75	0.03	144.72	144.78	144.75	0.03
68	141.14	142.04	141.59	0.45	141.01	142.17	141.59	0.58
Inflows (l/s):								
26	64.24	66.83	65.54	1.29	64.24	66.83	65.54	1.29
3	30.80	32.05	31.43	0.63	30.80	32.05	31.43	0.63
60	32.77	34.15	33.46	0.69	32.77	34.15	33.46	0.69
160	44.95	46.75	45.85	0.90	44.95	46.75	45.85	0.90
68	30.32	31.54	30.93	0.61	30.32	31.54	30.93	0.61

From this vector a true or expected measurement vector was calculated using the network function, that is $\mathbf{z}^t$ was calculated as $g(\mathbf{x}^t)$. Measurement values for the additional meters in M2 were calculated in a similar way. These are shown in column 2 of table 4.2. Following the explanation of the measurement error vector in section 4.4, a vector $\mathbf{e}^z$ was calculated for this measurement vector, this is shown in column 4 of table 4.2. The observed measurement values, $\mathbf{z}^o$, shown in column 3 of table 4.2, were selected randomly from within the range specified by $\mathbf{z}^l = \mathbf{z}^t - \mathbf{e}^z$ and $\mathbf{z}^u = \mathbf{z}^t + \mathbf{e}^z$. In this way, $\mathbf{z}^o - \mathbf{e}^z \leq \mathbf{z}^t \leq \mathbf{z}^o + \mathbf{e}^z$, which corresponds to the real-life situation where measurement values are not exact but are contained in a range specified by the accuracy of the meters used and the accuracy of the prediction of the pseudomeasurement values. The state vector shown in column 3 of table 4.1 is the state estimate calculated from the observed measurement vector using the state estimator ODSTEST. The difference between this state estimate and the true state should be noted. It is caused solely by the addition of the simulated measurement errors and shows how corrupted measurement data can affect deterministic state estimates.

Table 4.3 shows the state uncertainty sets or confidence limits for the measurement set M1 (the minimal measurement set) as calculated by the Monte Carlo confidence limit algorithm and the linear programming confidence limit algorithm. (Results for the sensitivity matrix confidence limit algorithm are not included because for a minimal measurement set these will be the same as those for the linear programming algorithm, as is explained in Lemma 4.5). In each set of results, the confidence limits are shown in two ways. Firstly, the lower and upper limits of each variable are shown (in columns 2 and 3 for the Monte Carlo results and in columns 6 and 7 for the linear programming results). These correspond to the values x_i^{1l} and x_i^{1u} for each variable $i \in \{1,...,n\}$. Alternatively, an average value for each variable, together with an error bound is shown (columns 4 and 5 for Monte Carlo results and columns 8 and 9 for the linear programming results). The average value corresponds to $(x_i^{1u} + x_i^{1l})/2.0$ and the error bound to $(x_i^{1u} - x_i^{1l})/2.0$ for each $i \in \{1,...,n\}$. The second method of presentation is included as this is perhaps the more traditional way

of presenting uncertainty in engineering systems, with the value for a particular state variable being read as the average value ± the error bound. The Monte Carlo results demonstrate the scale of the potential error in state estimates for a system with no measurement redundancy. Pressure errors are in excess of 2.0m in the region of the network most distant from the reference pressure at node 160. In fact, the majority of pressure errors are over 1.0m with only those nodes close to node 160 having reasonably accurate estimates. These results demonstrate the comment made in section 4.3.3, that pressure errors increase with distance from the accurate pressure meters. Overall, the confidence limits produced by the linear programming method compare well with those of the Monte Carlo method. (It was explained in section 4.5.1, that the Monte Carlo results are very reliable mathematically, and can be used as a yardstick against which other algorithms can be compared). In the linear programming results, the increase in error with distance from the reference pressure is again very noticeable. In this feature there is a very strong correlation between the two sets of results. In most cases, the error bounds produced by the linear programming method vary by no more than 10% when compared with the Monte Carlo error bounds. The main exceptions to this rule appear in the pressure variables in nodes 1, 2, 6, 7, 8, 10 and 11, which are the most remote from the reference pressure reading at node 160. Also, there is no apparent shift in the uncertainty ranges produced by the two algorithms. These observations lead to the conclusion that no significant accuracy is lost in linearising the uncertainty model and justify the use of linearised confidence limit algorithms.

In table 4.4, linear programming and sensitivity matrix confidence limits are given for the augmented measurement set M2. The same convention is used as in table 4.3, with the confidence limits presented in two ways: lower and upper limits and average values together with error bounds. For all state variables, the uncertainty bounds are tighter when calculated by the linear programming method than they are when calculated by the sensitivity method. This reflects the observation that $X^1(M, \mathbf{z}^o) \subseteq X^2(M, \mathbf{z}^o)$, contained in

Lemma 4.5. Again, it can be seen that the pressure errors are smallest in the nodes closest to the accurate pressure meters. With this measurement set, the error bounds for the pressure variables average at 0.4m to 0.5m. This shows that by adding this small set of meters greatly improves the accuracy to which the state estimates can be calculated. Adding a different set of meters may not result in such an improvement in accuracy as this set of meters was chosen to demonstrate the value of adding meters. The results of the sensitivity matrix method, although not as tight as those of the linear programming method, compare favourably in most cases. To calculate these results the computation time taken was about 2 seconds for the sensitivity matrix method, compared with about 10 seconds for the linear programming method (time measured on a VAX8700). Therefore, the sensitivity matrix method can be considered as an adequate substitute for the linear programming method, particularly when fast computation is required.

4.7. CONCLUSIONS

In this Chapter, the problem of uncertainty in the monitoring of water distribution systems is examined.

Present day deterministic state estimation techniques are very efficient, having small computational requirements and producing results of an acceptable level of accuracy. But no state estimator can give accurate results from inaccurate data. Due to the cost of metering, the water industry is, and will be in the near future, making use of relatively inaccurate pseudomeasurements. So state estimates are bound to be subject to uncertainty. Monte Carlo simulation has shown that this uncertainty can be very significant. The degree of confidence that can be put in these results must be calculated and presented with the state estimates themselves. Only then can safe and reliable operation of the distribution system be ensured.

The principal causes of this uncertainty are investigated. Of these, measurement and pseudomeasurement inaccuracy is identified as having a major impact. To assess this impact, an uncertain network model is formulated. From this, the concept of confidence limit analysis - quantifying the effects of measurement and pseudomeasurement uncertainty on the accuracy of state and derived state estimates - is introduced. This model implicitly incorporates all of the complex and interrelated factors - network topology, distribution of meters and operational state of the system, for instance. Four confidence limit algorithms are presented, these are: the Monte Carlo method; the linear programming method; the sensitivity matrix method; and the ellipsoid method.

The Monte Carlo technique generates a set of feasible state estimates and from these calculates upper and lower bounds for each state variable. To guarantee the validity of the Monte Carlo results, a massive number of state estimates must be used. For this reason the Monte Carlo method is an unrealistic proposition for real-time application. Conversely, the results it produces are very reliable mathematically. This is because any error bound or confidence limit produced by the Monte Carlo method is attainable, ie there is a feasible state estimate that will reach this bound. This mathematical reliability means that this approach provides a yardstick against which other algorithms can be tested. The linear programming and sensitivity matrix methods are both based on an accurate linearisation of the system model. However, the results produced by these methods compare very well with those of the Monte Carlo method. The computational efficiency of the sensitivity matrix method, in particular, makes this method suitable for real-time and on-line applications. The ellipsoid method, although recommended for this type of problem by other authors, proved unsuitable for application to confidence limit analysis in water distribution systems.

CHAPTER 5

OPTIMAL METER PLACEMENT

5.1. INTRODUCTION

Ideally, a water distribution telemetry system would monitor the flow in every pipe and the consumption and pressure at all nodes of the network. Unfortunately, this is not economically viable. Even in simplified water distribution network models, there may be several hundred potential meter sites. Realistically, only a limited number of flow and pressure meters can be installed into the telemetry system. This means that there will always be a significant level of uncertainty present. Fortunately, by carefully selecting sites for the real meters in the network it is possible to significantly reduce this uncertainty. It is this problem that is addressed in this chapter - how should the telemetry system meters be positioned about the network to achieve the greatest level of accuracy? Whenever a new telemetry system is being designed or an existing one altered it is important to optimise the design of this meter configuration. As metering resources are limited, the optimisation will be a complicated procedure, balancing the potential accuracy against the cost of the metering.

One way in which a good meter configuration can be determined is by using the software package TCLAS. TCLAS - Telemetry Confidence Limit Analysis Software - is an interactive program that can be used in meter placement studies and in water distribution decision support [16] (TCLAS is described more fully in Chapter 6). It is the first of these uses that is of interest here. Using TCLAS, an operator can input a proposed meter configuration and then call on TCLAS to calculate the uncertainty in state estimates that would be produced by this metering. TCLAS presents the accuracy of a metering

graphically, allowing the operator to see which parts of the network require further metering. The proposed meter placement can then be modified accordingly. By experimenting with meter configuration in this way TCLAS operators can use their experience and system knowledge, to find a balance between accuracy and cost of metering, eventually designing a satisfactory meter placement. Although TCLAS provides a useful insight into the effects of meter placement and meter accuracy on the accuracy of state estimates, it cannot guarantee that the resultant meter placement is the best. Also, its use can be very time consuming for large networks. What is required is a technique that will automatically provide the optimal meter placement design for any network, under all operating conditions.

The optimal meter placement problem for water network telemetry systems is addressed in this chapter. The next section includes a review of the existing literature on this subject. In section 5.3, the various considerations involved in water network telemetry system metering design are discussed. These concepts are formulated mathematically in section 5.4 and the problem is described as a mathematical optimisation problem. In section 5.5, two solution techniques are suggested. Optimisation is dependent on the particular operating state under which the study is carried out. So, in section 5.6, a method is suggested by which the problem can be extended to include different operating states. Section 5.7 presents some results for a realistic test network and conclusions are reported in section 5.8.

5.2. REVIEW OF PREVIOUS RESEARCH

The optimal meter placement problem in power system monitoring has been addressed by Koglin, details of this approach are given in [87]. In this paper, a sub-optimal method is suggested which uses a cost function based on the weighted mean variance of all state variables and a constraint limiting the variance of each state variable. Initially, meters

are considered in all possible sites in the network and the process systematically eliminates meters, until a desired number are left. At each step the meter discarded is the one whose removal causes the smallest increase in cost value. This requires, at each step, the cost value for the current meter set, without each of the meters in turn, to be calculated. To assess the value of each of these meter sets, Koglin uses the weighted mean variance of the set of derived state variables. These variances can be obtained from the leading diagonal of the derived state covariance matrix.

Aam et al, in [1], extend Koglin's method, to obtain a more robust meter placement. Specificly, they consider situations where the availability of readings from the meters placed in the system is unreliable. Their algorithm is split into three phases, the first being the same as Koglin's algorithm. In the second phase, a pre-specified number, r say, of meters are removed from the phase 1 meter set. The same elimination method as is used in Koglin's method is used here. Finally, in phase 3, r meters are added to the current meter set. These meters are chosen so as to reduce the sensitivity of Koglin's cost function to loss of measurements or bad data. Rather than calculating derived variable variance by using the covariance matrix, as suggested by Koglin, Aam et al use a Monte Carlo method.

Phua and Dillon [113] proposed method of optimal meter selection based on an 'entropy' function. The entropy function is used to indicate the level of the lack of knowledge or uncertainty in the state variables due to the measurement uncertainty. A non-linear programming method is proposed to maximise mutual information between measurement and state vectors.

5.3. OPTIMAL METERING: CONSIDERATIONS

Optimisation of the meter configuration for a water distribution system can have two aims. Minimisation of the cost of metering or maximisation of the accuracy of system monitoring. These two are usually conflicting aims, so a careful balancing is required.

Generally speaking, the more meters used, the more accurately system variables can be estimated. This is because the telemetry system's meters can be seen as replacing the highly inaccurate nodal consumption predictions with more accurate data. Optimisation can therefore be approached in two ways. One is to specify the cost of metering and then find the most accurate meter placement that does not exceed these spending limits. Conversely, a level of accuracy can be specified and the least expensive metering that achieves this level found.

The cost of a particular metering is easily calculated as this involves mainly hardware considerations. Such as the number of meters to be used, their individual price, cost of communication links, cost of installation and maintenance, etc. Quantification of the value of a metering in terms of accuracy is more difficult. For instance the aim may be to ensure that each state variable can be calculated to within a specified accuracy. Alternatively, an average accuracy over all variables may be required. As in explained in Chapter 4, quantification of the accuracy a particular meter placement is itself a complicated task. Any method intended to automatically design an optimal meter placement must be applicable in each of these situations

A mathematical optimisation problem of this type requires a cost function and usually has some constraints. The cost function is needed to assess the value of each configuration and the constraints are used to model the system's restrictions. Optimisation is then a matter of finding the meter placement that minimises the cost function while satisfying all of the constraints. In this case, three basic types of cost function have been identified. These are:

COST1:- Minimise metering cost. This assesses the cost, in financial or pseudo-financial terms, of each configuration and picks the most economical one. Of course, this type of cost function must be considered together with constraints which will primarily be concerned with the monitoring system's accuracy.

COST2:- Minimise maximum state variable error. In this type of optimisation the accuracy of the system variables must be analysed using techniques such as those of Chapter 4. The best meter configuration being the one which reduces the potential error in all of the variables to a minimum. Here, constraints specifying limits on the cost of metering must be used.

COST3:- Minimise average state variable error. This is similar to the optimisation type above except that average state variable error is considered. Again constraints on metering cost must be used.

By varying the coefficients in any of these three types of cost function the different elements can be weighted. For example, if a particular variable is required to a greater degree of accuracy, this can be specified by weighting the appropriate element of the type 2 or type 3 cost function. Alternatively, different meter sites can be weighted, according to level of preference or cost, in a type 1 cost function. Zero weights may also be used, this has the effect of limiting consideration to a particular subset of meter sites or state variables (ie that subset with non-zero weights). It is the weighting system that will provide the required flexibility. Hopefully all design priorities can be modelled using one of these three weighted cost functions.

As well as being able to select the cost function, it must be possible to select any number of additional constraints. These may be economic constraints, operational constraints, water network constraints, telemetry system constraints, etc. Weighted functions similar to the cost functions can be used to model these constraints. Together with these functions a value which represents the maximum value the constraint can take must also be supplied. A few possible examples for the type of constraint that may be required are shown below, but the list is by no means exhaustive:

1. The cost of metering is limited. This may specify limitations on the cost of metering in a

specific region of the network or on the cost of a certain type of meter, flow or pressure meters for instance. Alternatively, a simple limit on the number of meters may be used instead of the cost of metering in each of these cases.

2. Certain meter sites must or must not be included in the optimal configuration. This is particularly useful if the telemetry system already has some meters in place.

3. A limit for the accuracy of a particular flow or pressure variable is specified.

4. A limit for the weighted accuracy (average or minimum) for any subset of pressure and flow variables is specified. In this type of constraint, the subset of variables chosen may be, for example, all flow variables, all pressure variables, all variables in a certain region of the network, all variables that are used to trigger control devices or indeed any set of variables that an operator considers important.

Any number of such constraints, with weighted coefficients, in any combination, may be needed to fully describe the metering requirements as well as the operational characteristics and peculiarities of the system.

A typical water network may have several hundred nodes and pipes, creating several hundred variables and several hundred possible meter sites. This combined with the computational complexity of the methods to quantify pressure and flow variable errors for a given meter configuration dictate that the problem of optimising meter placement for water networks is by no means a trivial one. Any methods used must be extremely powerful and their computer implementations will be very time consuming. In the rest of this chapter an attempt is made to tackle these difficulties.

5.4. MATHEMATICAL FORMULATION

In the previous section the problem of designing an optimal meter placement for a water distribution system has been outlined. The problem is now formulated mathematically, with cost and constraint functions modelled.

The aim of optimising meter placement for a given water network is to determine a set of meters that can be added to the telemetry system that is optimal according to one of the three types of cost function while, at the same time, meeting a set of pre-specified constraints. It must be the case that either, the cost function, or one of the constraints will involve the potential accuracy of the state variables that can be achieved by adding each possible meter set. The potential accuracy of the estimated state variables is dependent on the operating state of the system as well as the measurement set. There are other factors involved, network connectivity and pipe parameters, but these can be considered as constant for a given system as they are unlikely to change during the study period. A general statement of the optimal meter placement problem would require the optimisation to be carried for all possible meter placements and over all operating states. As there are an infinite number of operating states it is completely impractical to optimise over both variables at the same time. Instead, the problem can be split into a two level optimisation problem. The bottom level requires a method that can determine an optimal meter placement for a particular operating state. The first sections of this chapter deal with this problem. At the top level of the optimisation a method is required which can analyse optimal meter placements for a whole range of operating states, found by the lower level optimisation method, and find the meter placement that performs the best overall. Section 5.6 tackles this top level optimisation. In the remainder of this section and in section 5.5 attention will be restricted to the lower level optimisation. For this, a particular operating state is assumed. The operating state selected for the optimisation is referred to as the base operating state for the study and is denoted by $\mathbf{x}^b$. $\mathbf{x}^b$ will be an n-dimensional state

vector, having one element for each independent state variable, that can fully specify the system's operating state.

For state estimation of water distribution systems it is necessary, in order to make the system observable, to have a certain minimal set of measurements [14]. These measurements generally come from meters that are already in place in the system. Otherwise predictions of their values have to be made. For this reason a minimal measurement set is assumed present before the meter placement study is carried out. In reality this is not a restricting assumption as the values of the minimal measurement set can be assigned suitably large measurement error bounds if no corresponding meters or pseudomeasurements are actually available. Real meters at the minimal measurement set sites can also be added to the set of possible meter sites. This minimal measurement set will be denoted by M'.

In a water network, R different sites can be chosen as possible sites for installing meters, these being additional to the ones already included in the minimal measurement set. The meters should have a specified accuracy, type and cost. For example, meter site 1 might be for a flow meter in pipe $i-j$, with accuracy $\pm\, a\,\%$ costing $\$b$. Other meter sites may be intended for different types of meters in different parts of the network. The same node or pipe may appear more than once in the list of possible meter sites, signifying that different types of meter are being considered for the same location. A vector $\mathbf{m} = (m_1, \ldots, m_R)$ can now be defined.This will be referred to as the location vector and has one element for each proposed meter site. Each element $m_1, \ldots, m_R$ can take a value of either 1 or 0. If $m_i=0$, $(i \in \{1,....,R\})$, then no meter is to be placed at site i, in the meter set specified by $\mathbf{m}$, and if $m_i=1$ then there is to be a meter placed at site i. The meter location vector will become the decision vector of the optimisation, the object being to find a vector $\mathbf{m} \in \{0,1\}_1 \times \cdots \times \{0,1\}_R$ satisfying all constraints whilst optimising the cost function. Let $M(\mathbf{m})$ denote the meter set specified by the location vector $\mathbf{m}$.

The meter location vector specifies the proposed meter set that is to be added to the minimal measurement set M' (assumed to be already present). So $M' \cup M(\mathbf{m})$ specifies the full measurement set proposed for the system. With this measurement set and with the assumed base operating state, a measurement uncertainty set $Z(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ can be derived. This uncertainty set is derived in same way as was described in Chapter 4 for $Z(M, \mathbf{z}^o)$ with M a general measurement set. The only difference being that in Chapter 4 it was assumed that the measurement data ($\mathbf{z}^o$, $\mathbf{z}^l$ and $\mathbf{z}^u$ in particular) were available. For the purposes of optimal meter placement studies this is not the case as the meter sets are only proposed. To solve this problem, $\mathbf{z}^o$ can be replaced by $g(\mathbf{x}^b)$, where $g(.)$ is the network function for the meter set $M' \cup M(\mathbf{m})$, and $\mathbf{z}^l$ and $\mathbf{z}^u$ can be derived from $\mathbf{x}^b$, estimates of pseudomeasurement variance and the manufacturer's specifications on the accuracy of the meters, as was explained in Chapter 4. Because of this definition a distinction must be drawn between the measurement uncertainty set $Z(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ and the measurement uncertainty sets discussed in the previous chapters. $Z(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ is the uncertainty set that can be expected from a measurement set $M' \cup M(\mathbf{m})$ and the uncertainty set, $Z(M, \mathbf{z}^o)$ discussed before is the actual uncertainty set derived from existing measurement data. All references to a measurement uncertainty set in this chapter will be to an expected uncertainty set of the type $Z(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$. The notation will include a the base operating state $\mathbf{x}^b$ to emphasise this point.

5.4.1. Cost functions

COST1: A weighting vector $\mathbf{a} = (a_1, ..., a_R)$ can be defined. Each element a_i, $(i \in \{1, ..., R\})$, of $\mathbf{a}$ will take the value of the weight applied to the individual meter site i in a type 1 cost function. The weights can be any non-negative real number and should reflect the cost of installing and maintaining a particular type of meter at site i. The cost of metering of any particular meter configuration is now given by

$$\textit{metering cost} = \mathbf{a}^{\mathrm{T}}.\mathbf{m} \tag{5.1}$$

where $\mathbf{m}$ is the meter location vector for that configuration. With these definitions, minimising metering costs is a matter of minimising equation (5.1) over all meter configurations satisfying all of the constraints.

COST2: In an optimal meter placement study, any state variables can be considered, not just the set of independent state variables. It will be assumed that there is a set of N of these. This set may include all of the n independent state variables as well as any other derived state variable. For the i^{th} derived state variable, $i \in \{1,...,N\}$, its potential uncertainty is dependent on both the measurement set $M' \cup M(\mathbf{m})$ and on the base operating state selected for the study. This dependency is denoted by

$$e_i = e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^{\mathrm{b}})) \tag{5.2}$$

where e_i is the potential error associated with the i^{th} state variable. The base operating state is, at this stage, also considered as a constant but is included in (5.2) to emphasise the point that a particular base state must be chosen for the study. An N-dimensional cost vector $\mathbf{b}$ can be defined which has one element for each derived state variable. These elements represent the weights for the individual state variables and should reflect how accurately that variable is required to be known. With these definitions the cost function is given by

$$\textit{maximum state variable error} = \max_{i=1}^{N} b_i.e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^{\mathrm{b}})) \tag{5.3}$$

where b_i is the i^{th} element of the vector $\mathbf{b}$. The maximisation is carried out with the particular value of $\mathbf{x}^{\mathrm{b}}$ chosen as the base state for the study.

COST3: The third cost function is very similar to the second but differs from it in that the average state variable error for each metering is considered rather than the maximum error. Firstly a weighting vector, $\mathbf{c}$, consisting of N elements must be constructed. $\mathbf{c}$ is defined in

a similar way to **b** in COST2. With this weighting the cost function is given by

$$\textit{average state variable error} = \frac{\sum_{i=1}^{N} c_i.e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))}{N} \tag{5.4}$$

In this cost function, the average state variable error is calculated for the base operating state, $\mathbf{x}^b$.

5.4.2. Constraints

For this optimisation problem to have non-trivial results, it must be subject to some constraints. The ideas behind these were discussed in section 5.3. Close examination of the constraints that are required by the optimal meter placement problem reveals that these fall into three main categories. These are:

1. Weighted cost of metering must be less than a specified limit.

2. Maximum weighted state variable error must be below a specified limit.

3. Average weighted state variable error must be below a specified limit.

Any number of these three types of constraint, combined in any order, together with a suitable weighting can be used to allow all real constraints to be adequately modelled.

The three types of constraint function that are used resemble the three types of cost function. They may have different weighting coefficients associated with them and must have a specified limit, but they can be modelled in a similar way. The optimal meter placement problem will have a number, C say, of constraints. Taking the j^{th} of these, it will have a weighting vector $\mathbf{a}^j$, $\mathbf{b}^j$ or $\mathbf{c}^j$, depending on whether it is a type 1, 2 or 3 constraint. The constraint will also have a limit, ϕ^j say. Mathematically, the j^{th} constraint, ($j \in \{1,...,C\}$), will take one of the following forms.

$$(\mathbf{a}^j)^T.\mathbf{m} \leq \phi^j \tag{5.5}$$

$$\max_{i=1}^{N} b_i^j.e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b)) \leq \phi^j \tag{5.6}$$

$$\frac{\sum_{i=1}^{N} c_i^j.e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))}{N} \leq \phi^j \tag{5.7}$$

depending on whether it is a type 1, 2 or 3 constraint. Again, $\mathbf{x}^b$ is the base state chosen for the study.

It can now be seen how the optimal meter placement problem can be formulated as a mathematical minimisation problem. The cost function will take the form of one of the equations (5.1), (5.3) or (5.4) and the minimisation will be subject to C constraints of the form shown in equations (5.5), (5.6) and (5.7).

5.4.3. Calculating potential errors - the e_i function

$e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ is defined as the potential error for a particular state variable as a result of the measurement uncertainty set $Z(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$. The values of e_i can therefore, be calculated using one of the confidence limit algorithms of Chapter 4. The Monte Carlo method is clearly out of the question as this problem requires that the values of e_i be calculated for each i and for many different meter location vectors. So, one of the linearised methods must be used. For the i^{th} derived state variable, its potential error for the uncertainty set, $Z(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$, is defined as

$$e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b)) := (y_i^{1u} - y_i^{1l})/2.0 \tag{5.8}$$

where y_i^{1u} and y_i^{1l} are the upper and lower bounds for this variable in the linearised, derived state uncertainty set $Y^1(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$. As $Z(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ is symmetric about $g(\mathbf{x}^b)$, $Y^1(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ is symmetric about $f(\mathbf{x}^b)$, where $f(.)$ is the derived state function of equation (3.3). This statement is just a corollary to the special case (i) in section 4.5.2. Hence, the potential error for a variable, defined in (5.8), is the

maximum value that $(y_i - y_i^b)$ can take for a y in $Y^1(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$, where $\mathbf{y}^b = f(\mathbf{x}^b)$. In this way, the definition of potential error in the variables corresponds to the concepts of error in worst-case analysis.

Calculating $e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ still presents a major problem as there are a very large number of possible location vectors, $\mathbf{m}$, to be assessed. The sensitivity matrix method described in Chapter 4 is too time consuming to be of direct use here. What is required is an accurate approximation to this function that can be calculated quickly. Let $\mathbf{m}^k$ be the location vector for which $m_i^k = 0$ unless $i=k$, ie $\mathbf{m}^K$ is the location vector for the meter set consisting of one, and only one, meter which is the k^{th} in the list of possible meter sites. A new error function can be defined as follows:

$$e_i^*(M' \cup M(\mathbf{m}), g(\mathbf{x}^b)) := \min_{k:m_k=1} e_i(M' \cup M(\mathbf{m}^k), g(\mathbf{x}^b)) \tag{5.9}$$

where the minimisation is carried out over all those k for which $m_k = 1$. In other words, for a given set of meters, the function $e_i^*(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ gives a value for the error in variable i equal to the smallest error for that variable when each of the meters is considered individually. Although there is no guarantee of equality, it is the case that

$$e_i^*(M' \cup M(\mathbf{m}), g(\mathbf{x}^b)) \approx e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b)) \tag{5.10}$$

particularly when the meters are well spaced and there is little significant interaction between them. This approximation, e_i^*, is motivated by the following lemma

Lemma 5.1 Let $\mathbf{m}$ and $\mathbf{m}^k$ be defined as above. Then for all k for which $m_k = 1$, that is, for all meter sites specified by $\mathbf{m}$,

$$e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b)) \leq e_i(M' \cup M(\mathbf{m}^k), g(\mathbf{x}^b)), \quad i=1,...,N \tag{5.11}$$

Proof Let $\mathbf{y}$ be the derived state vector in $Y^1(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ for which $y_i - y_i^b$ is maximal, where $\mathbf{y}^b = f(\mathbf{x}^b)$, and let $\mathbf{x}$ be the state vector in $X^1(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$

for which $\mathbf{y} = f(\mathbf{x})$. Then $e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b)) = y_i - y_i^b$. As $\mathbf{x} \in X^1(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$, $\mathbf{x}$ satisfies all of the constraints specified by the measurement uncertainty set $Z(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$. Since $M' \cup M(\mathbf{m}^k) \subseteq M' \cup M(\mathbf{m})$ for all k with $m_k = 1$, $\mathbf{x}$ will satisfy all of the constraints specified by $Z(M' \cup M(\mathbf{m}^k), g(\mathbf{x}^b))$ for all k with $m_k = 1$. This means that $\mathbf{x} \in X^1(M' \cup M(\mathbf{m}^k), g(\mathbf{x}^b))$ for all such k. Hence, $e_i(M' \cup M(\mathbf{m}^k), g(\mathbf{x}^b)) \geq y_i - y_i^b = e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ for all k such that $m_k = 1$. The result now follows.

It follows directly from this lemma that

$$e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b)) \leq \min_{k:m_k=1} e_i(M' \cup M(\mathbf{m}^k), g(\mathbf{x}^b)) = e_i^*(M' \cup M(\mathbf{m}), g(\mathbf{x}^b)) \quad (5.12)$$

The simplified function $e_i^*(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$ is introduced because its evaluation can be carried out much more quickly than $e_i(M' \cup M(\mathbf{m}), g(\mathbf{x}^b))$. The procedure will therefore be to evaluate $e_i(M' \cup M(\mathbf{m}^k), g(\mathbf{x}^b))$ for all $1 \leq k \leq R$ and all $1 \leq i \leq N$, where R is the number of possible meter sites. The results can be stored in a matrix E, where

$$E(k, i) = e_i(M' \cup M(\mathbf{m}^k), g(\mathbf{x}^b)) \quad (5.13)$$

This means that the element $E(k, i)$ of the matrix E holds the value of the error bound for the i^{th} state variable after the k^{th} meter from the list of possible meter sites has been added to the minimal measurement set (with no other meter in system). With the matrix E defined as in (5.13), the values of the e_i^* function are given by

$$e_i^*(M' \cup M(\mathbf{m}), g(\mathbf{x}^b)) = \min_{k:m_k=1} E(k, i) \quad (5.14)$$

5.5. SOLUTION TECHNIQUES

The methods reviewed in section 5.2 have been examined for application to the problem of optimal meter placement in water distribution systems. Each of these methods relies on a particular, somewhat arbitrary, choice of cost function and constraints. As a

result, these methods are not flexible enough to be applied directly to this problem as it is formulated. Furthermore, calculating the derived variable variance using the covariance matrix in Koglin's method [87] or by using the Monte Carlo method suggested by Aam et al [1] for each proposed meter set will be impractical for a large network. Also, these two methods provide no guarantee that the resulting meter placement is optimal according to their chosen cost functions. A solution technique based on the more flexible uncertainty formulation of Chapter 4 and the E matrix overcomes these problems. This is the approach adopted.

A careful study of the mathematical optimisation methods available identified those used in the field of locational analysis [60] as the most promising. Within this field the optimal meter placement problem can be classified as a set covering problem. One of the methods of solution presented here is based on a set covering algorithm due to Roy [120]. This can guarantee an optimal solution to the problem as formulated in section 5.4. Alternatively, the problem can be viewed as a mixed integer programming problem. Solutions, based on enumeration techniques will be far too time consuming in this situation, so a restricted search method has been developed. This can produce near optimal results much more rapidly. The restricted search method is, in some respects, similar to Koglin's method but has a greater level of flexibility.

In the remainder of this section the chosen solution techniques are examined more closely, starting with the calculation of the E matrix in section 5.5.1. In sub-sections 5.5.2 and 5.5.3, the set covering and restricted search methods are presented.

5.5.1. The *E* matrix

In Chapter 4 several methods for calculating the error bounds of the state variables for a given state and measurement configuration were presented. Of these the sensitivity matrix method seems the most suitable for these purposes. Other methods for quantifying

the quality of a meter placement have been assessed. An assessment based on the condition number of a matrix, proposed by Edelmann in [53], was rejected as this provides just one number for the value of a particular meter configuration. This means that Edelmann's method can provide no information about the effects of the meter placement on each individual variable and so is not flexible enough to be used here. Koglin's analytic method, in [87], was also considered but this proved to be too slow. The application of the sensitivity method is quite straightforward in this situation. For each meter in the list of possible meter sites, the error bounds for each variable that would be obtained when this meter added to the minimal measurement set, are calculated. The results can be inserted directly into the appropriate row of the E matrix. Thus, with R applications of the sensitivity matrix method, the E matrix can be assembled.

5.5.2. Set covering algorithm

The set covering problem can be stated as follows. Let $U := \{u_i : 1 \leq i \leq K\}$ be a set of K elements and let $V := \{v_i : v_i \subseteq U, 1 \leq i \leq L\}$ be a collection of L subsets of U. The set V will be referred to as the set of blocks of U. A cover W of U is a subset of V such that each element of U is contained in at least one element of W. The set covering problem is to find a cover W, of U, that is optimal in some way.

In [120], Roy presents "*An algorithm for a general constrained set covering problem*". This has many features that can be exploited for the purposes of the optimal meter placement problem. The most important of these being the facility to specify a cost function and constraints. The format that these functions take is very similar to that used in the previous section. So, the set covering problem considered in Roy's paper is an extension of the true set covering problem as defined in the previous paragraph. Roy introduces the terms "configuration" and "partial solution", these are now described. A configuration is a partition of the set of blocks V, into two subsets - the set of those blocks said to be "in" the

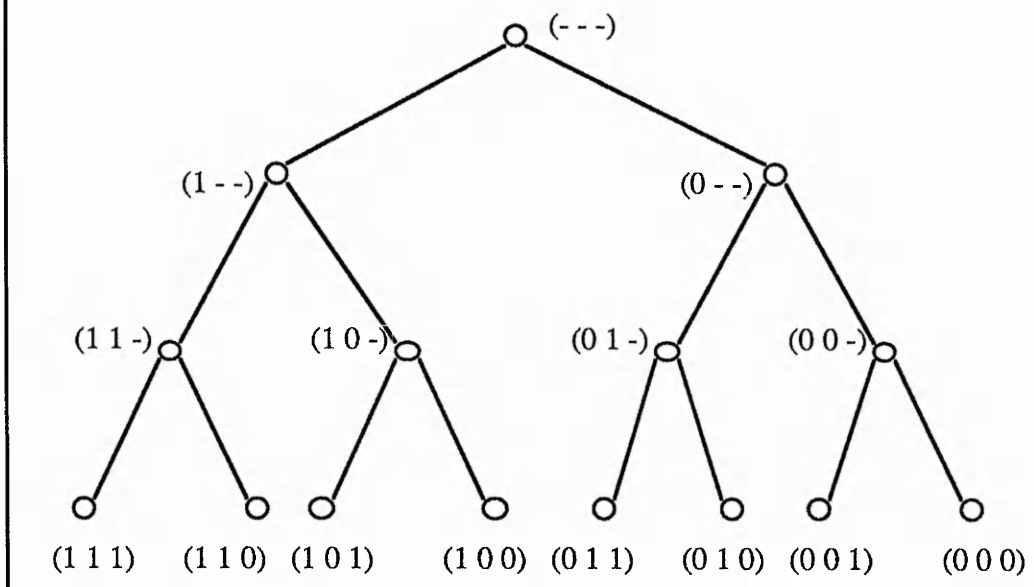
configuration and the set of those blocks "not in" the configuration. A partial solution is a partly specified configuration. It has some blocks specified to be "in" all configurations that it may lead to and some blocks specified to be "not in" these configurations. Together, these blocks will be referred to as forced blocks. The partial solution will also have a number of blocks that are, as yet, unspecified, these are referred to as free blocks. If a partial solution has F free blocks, then it may give rise to 2^F different configurations once these are specified, since each of these can be specified as "in" or "not in". The algorithm is based on a branch and bound method for which a decision tree must be constructed similar to the one in fig 5.1. Fig 5.1 shows the decision tree for a three block problem. The nodes of this tree will represent the possible partial solutions. A path, starting from the root, is followed through the tree. At each branch point, that partial solution is assessed to see whether any of the configurations it may lead to, will produce the optimal cover (see fig 5.2). In order to do this, three things must be checked, (i) whether the partial solution may lead to a cover, (ii) whether the partial solution may lead to a configuration that has a better value for the cost function than the best potential value so far, and (iii) whether the partial solution may lead to a configuration that will satisfy all of the constraints. If any of these three are not satisfied, then the node is declared a bound node (ie it is not worth pursuing the path any further than this point) and the path backtracks to the last branch node. Eventually a path will be found that includes the optimal cover, if such a configuration exists.

A detailed description of the set cover algorithm is now given. Fig 5.3 shows a flow diagram for this algorithm.

Set covering algorithm

STEP I. INITIALISATION: Introduce an empty stack. Let Φ be the value of the cost function for the current best configuration, this is initially set at $+\infty$ as no

Fig 5.1 : Decision tree for three block problem.



(. . .) denotes a partial solution. "1" denotes a block that is to be contained in configuration, "0" denotes a block not to be contained and "-" denotes a free block.

Fig 5.2 : Typical path through decision tree.

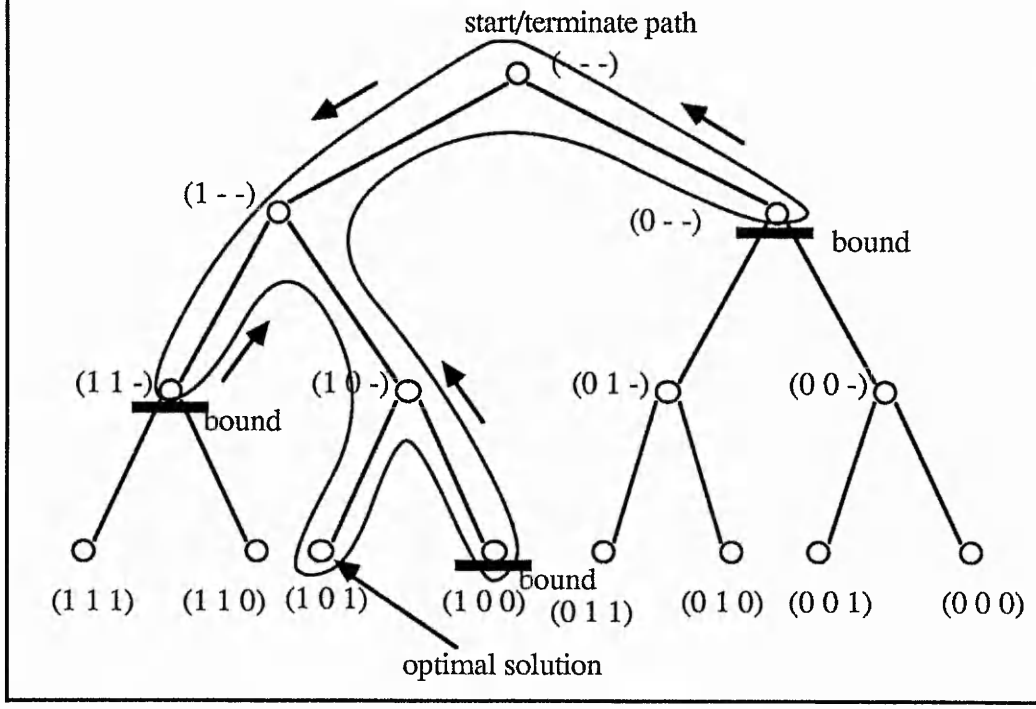
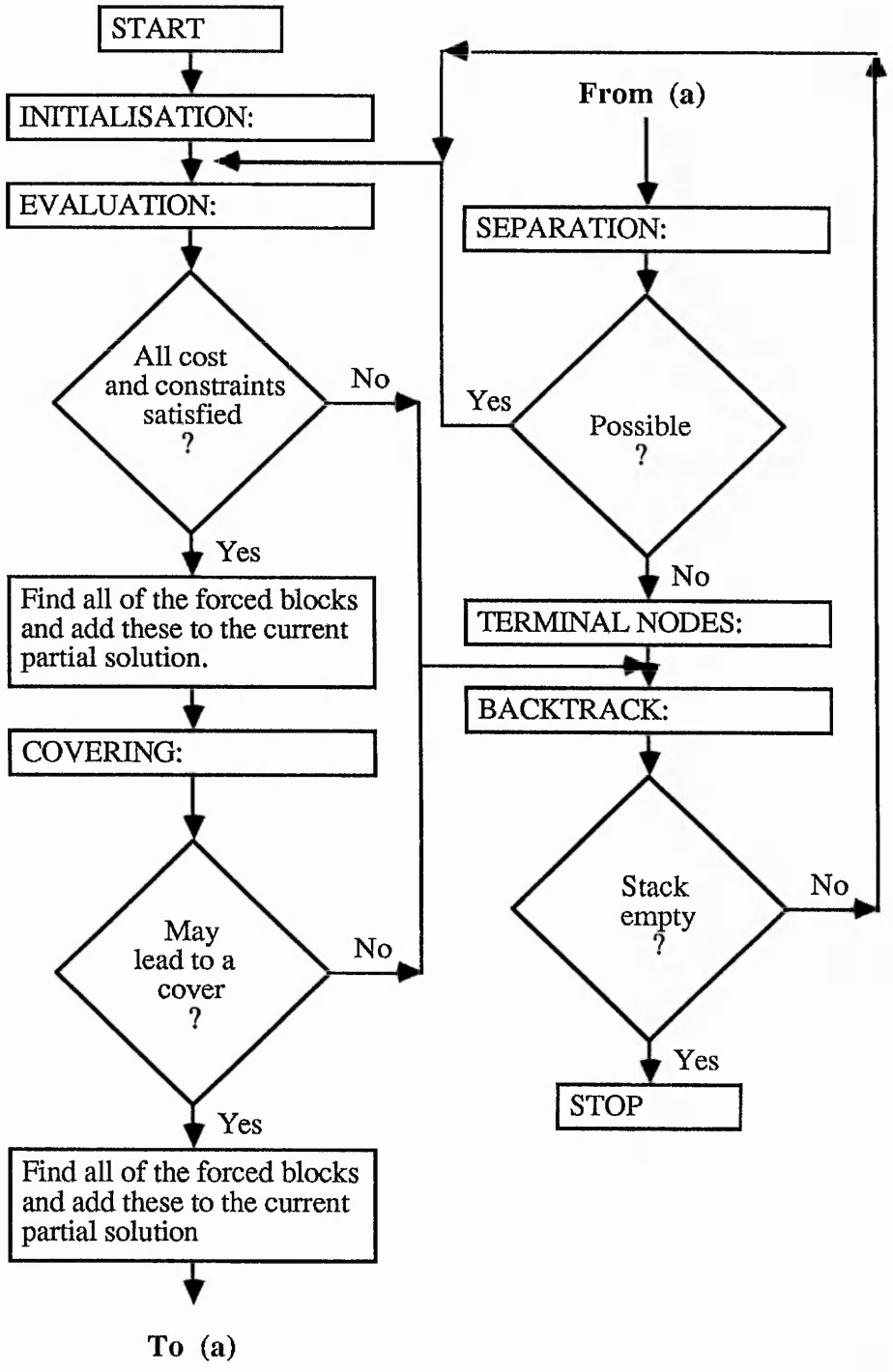


Fig 5.3 : Flow diagram for the setcovering algorithm.



configurations have been examined. Set the initial partial solution as the one for all blocks are free blocks.

STEP E. EVALUATION: Calculate the optimistic estimate (the optimistic estimate functions will be described in a later paragraph) for the value of the cost function for the current partial solution. This involves calculating a lower bound for the value of the cost function over the set of possible configurations that may lead from the current partial solution. If this optimistic estimate will not improve on the current best value, Φ, then go to step B. Otherwise calculate the optimistic estimates for each of the constraints for the current partial solution. If it becomes clear that one or more of the constraints will not be satisfied by any of the configurations that may lead from this partial solution, then go to step B. On evaluation of the optimistic estimates it may become clear that if this partial solution is to lead to any feasible or optimal configuration then some of the free blocks must become specified. That is to say it may be apparent that all feasible/optimal configurations leading from this partial solution must include certain of the free blocks and must not include others. Any such blocks are called forced blocks. These should be identified if possible and the partial solution amended. Go to step C.

STEP C. COVERING: Check whether any of the configurations that can be obtained from this partial solution cover U. If none do, go to step B. If all do, go to step S. If some of these configurations do but others don't then it should be possible to identify some forced blocks. Add these forced blocks to the partial solution and go back to the beginning of step C.

STEP S. SEPARATION: If there are no free blocks left in the partial solution (ie the partial solution uniquely specifies a configuration), then go to step T. Otherwise select one of the remaining free blocks. Two new partial solutions can be obtained, the first

by adding this free block to list of blocks to be included in the configuration and the second by adding it to the list not to be included. The latter of these is put onto the stack and the former is considered as the new current partial solution. Go to step E.

STEP T. TERMINAL NODES: The current partial solution now uniquely specifies a configuration. Firstly it must be checked whether this configuration is feasible (in order to have reached this far it must be a cover), if so the cost function is calculated. If this value is better than the current best then this cover is the best encountered so far and so must stored and Φ must be updated. Otherwise, the old value of Φ and the old current best cover are kept. Go to step B.

STEP B. BACKTRACKING: This stage looks back through the decision tree to find the last feasible partial solution that has not been fully explored. This can be done by picking the last partial solution from the stack, then taking this to step E as the current partial solution. If the stack is empty then terminate the algorithm. If Φ is still equal to $+\infty$ then there is no solution, otherwise the optimal, feasible cover is the one which is stored as the current best.

In the EVALUATION stage of this algorithm, the "optimistic estimate functions" were mentioned. These are now described. The problem with these functions is that the they are defined on the partial solutions rather than on the completely specified configurations. To deal with this, some more terminology must be introduced. In a partial solution some blocks (or possible meter sites) are specified to be included in the configurations leading from it, some blocks are specified to not be included in these configurations and the remaining blocks are free. The idea of optimistic estimates for the cost and constraint functions is to provide a way of estimating the minimal value these functions may take if all possible configurations leading from the partial solution are considered. For this reason it is sensible to consider two configurations for any partial

solution. These are the minimal and maximal configurations that can be obtained from the partial solution. The minimal configuration for a partial solution will consist of only those blocks specified to be included, and the maximal configuration will consist of these blocks together with the free blocks. They will be referred to by their location vectors $\mathbf{m}^-$ and $\mathbf{m}^+$, respectively. It is now easy to see how the optimistic estimate functions can be defined. That is

$$\textit{optimistic cost of metering} := \mathbf{a}^T.\mathbf{m}^- \tag{5.15}$$

$$\textit{optimistic maximum variable error} := \max_{i=1}^{N} \; b_i.\,(\min_{j:m_j^+=1} E(j,i)) \tag{5.16}$$

$$\textit{optimistic average variable error} := \frac{\sum_{i=1}^{N} c_i.(\min_{j:m_j^+=1} E(j,i))}{N} \tag{5.17}$$

The optimistic estimates for the constraint functions can be similarly defined.

The set covering algorithm, detailed above, can be used directly to solve all three of the different types of optimisation of meter configuration in water systems. This is done in the following way. In the optimal meter placement problem the set that needs to be covered is the set of N derived state variables. These need to be covered by meter sites which will guarantee their error is below a specified limit. For each meter site the corresponding row of the E matrix gives an indication of how much influence that meter has on the accuracy of each variable. A cutoff value, α, can be defined. The "region of influence" of a meter site i is defined as the set of variables which can be estimated with error less than α with just meter i added to the minimal measurement set. It is now possible to describe the optimal meter placement problem in set covering terms. Let U be the set of N state variables. For any meter site i, $1 \leq i \leq R$, a block v_i can be defined as follows

$$v_i := \{j \in U : E(i,j) \leq \alpha\} \tag{5.18}$$

The collection of blocks, V, is then the collection of all v_i for each possible meter site i.

That is

$$V := \{v_i : i \in \{1,...,R\}\} \tag{5.19}$$

In optimal meter placement terms, a cover for U (the set of all state variables) is a subset of V for which every state variable is contained in at least one block (or region of influence) of that subset. Alternatively, a set of possible meter sites is said to cover all of the state variables if each variable is contained in the region of influence of at least one meter in that set.

The general constrained set covering algorithm can guarantee an optimal solution to the problem defined by the cost function (either equation (5.1), (5.3) or (5.4)) and a list of constraints (equations (5.5), (5.6) and (5.7)), when these are calculated using the E matrix. But for large water networks, with a large number of possible meter sites, it can be time consuming. Since the problem, so defined, is only an approximate model of the real optimal meter placement problem, an approximate solution method has been designed. One that produces a near optimal solution in a fraction of the time of the set covering algorithm. This method is presented in 5.5.3.

5.5.3. Restricted search method

This method is based on the strategy an operator using TCLAS [16] might use in designing an optimal meter placement (see Chapter 6 for a fuller description of the software package TCLAS). That is, firstly try to find a meter placement that satisfies all of the constraints. Once this has been done, the operator will try to improve on this placement by either removing a meter, adding a meter or replacing a small set of meters in the current best placement by a small set of meters not yet in the current best placement. Repeating this process until no further improvement can be found. If, at each step of the improvement, it is ensured that the new configuration still satisfies all of the constraints, this method will terminate with a good meter configuration. The restricted search method

can be regarded as an extension of Koglin's method with improved optimality. Use of the E matrix means that cost values can be calculated much more quickly and so more meter sets can be assessed. The initial phase of the two methods is very similar, meters are removed (or added in the restricted search, depending on the cost function used) when this can be done without violating a constraint. Once this phase has been completed, the restricted search goes on to improve the current meter set by finding meters that are not included that can be swapped with meters that are, when this will improve the cost value and not violate a constraint. In this way, the restricted search can guarantee a result closer to the optimum. In addition, restricted search has the flexibility required for this problem. The details of the algorithm are now given:

Restricted search algorithm

STEP 1. INITIAL SOLUTION: An initial solution is one that satisfies all of the constraints. It does not matter at this stage what its cost is. If the optimisation has a type one cost function - minimise metering cost - as in equation (5.1). Then its constraints will be ones limiting the errors of the state variables. The configuration that uses all of the possible meter sites is the one for which the accuracy of the state variables is the greatest. So, if this configuration does not satisfy all of the constraints, none will. Therefore, a suitable initial solution for this type of optimisation would be the configuration consisting of all possible meter sites.

On the other hand, if the cost function for the optimisation is given by a (5.3) or (5.4) type of function, its constraints will ones be limiting the metering cost. So in these cases, a suitable choice for an initial solution will be the null configuration, the one that contains no possible meter sites.

If no initial solution can be found, terminate the algorithm with no solution, otherwise go to step 2.

STEP 2. IMPROVEMENT: The case when the cost function is as in (5.1) will be considered first. The aim here is to reduce the metering costs. Initially each meter site included in the current best configuration is tested. If any meter site can be removed without violating a constraint, then this is done. Removing meters can only have the effect of improving the value of this type of cost function. This follows as a corollary to Lemma 5.1. This process goes on until all the meter sites that can be removed from the current best configuration have been removed. When this happens each pair consisting of one meter site in the current best configuration and one meter site not in the current best configuration is considered. As soon as one of these pairs is found that can be swapped without violating a constraint and at the same time improving on the current best cost, the pair are swapped. The improved configuration is then examined again to see if any meter site can be removed. When all of these have been removed the pairs are examined again. This process, removing meters followed by swapping meters, is repeated until no further improvement can be achieved. At this point the algorithm is terminated with the current best configuration as the solution.

The case where the cost function is specified by an equation like (5.3) or (5.4) can be treated in a very similar way. The difference being, where meters are removed in the previous case, meters are added here, when this can be done without violating a constraint. This is because the aim is to improve accuracy rather than reduce metering costs and the addition of meters must result in such an improvement (when assessed by a type 2 or type 3 cost function). The process of swapping meters is also used in this case.

5.6. EXTENSION TO INCLUDE DIFFERENT OPERATING STATES

So far, optimisation has been restricted to a study considering only one operating state. In many cases this may prove satisfactory. However, to get a more reliable optimal meter design it is worthwhile considering several different operating states. For instance, a

set of states can be selected which highlight a 'typical' days operation, one state for each hour perhaps. In this section, this approach is discussed and a method for combining the results for different operating states is presented.

If a number of optimisation studies are carried out for the same network under different operating states, a whole range of 'optimal' meter configurations will be produced. It may be that these 'optimals' have very similar cost values but significantly different location vectors. For instance, two meter sets may have no sites in common, but can produce the same level of accuracy. Which one of these should be installed? Preliminary results have shown that this is often what happens in water networks. It could be argued, that if they have the same cost value, then each configuration is equally valid. Nevertheless, these observations are disconcerting, particularly when real meters have to be installed into the telemetry system on such a recommendation.

What is required is a meter placement that is 'good' for all possible operating states, rather than one that is 'optimal' for one state. This requires a higher level of optimisation. The approach suggested, is based on an evaluation of a number, k say, of 'optimal locations. From these, the one which performs the best overall is selected. For example, 24 optimisations can be conducted, each one for a state typical of a particular hour of the day. Each one of the 'optimal' locations can then be costed at all other times of day. All of these results can be assembled to form a cost matrix C, with $C(i,j)$ (the $(i,j)^{th}$ element in the matrix) equal to the cost value for the i^{th} optimal metering under the conditions of the j^{th} state. Table 5.5 shows an example of a cost matrix. Examination of the C matrix should yield a meter placement that is the best overall, ie the optimal of the 'optimals'. For instance, this may be the 'optimal' whose cost differs the least from the optimal cost at each of the states. With this criteria, the cost difference can be the maximal cost difference or the average cost difference. In both cases the differences should be normalised so that the different cost values for different operating states can be accounted for. There is plenty

of room for more specific criteria to judge the best overall metering. Any one used should reflect the requirements of the particular telemetry system and should be able to select a metering that performs well in all situations.

5.7. RESULTS

A program OPTMP has been developed which incorporates the algorithms presented in this chapter. It can use either the set covering or the restricted search optimisation procedure. This allows the same examples to be tested using both techniques and the results compared. The following test problems have been designed to illustrate the flexibility of the optimal meter placement model.

The test network for this study is described in Appendix A1. With this network, 50 possible meter sites have been selected - 25 flow meter sites and 25 pressure meter sites - which are shown in table 5.1. An accuracy of ± 0.1m was assumed for the pressure meters and ± 0.5l/s for the flow meters. Solutions for the following test problems, determined using both the set covering and restricted search procedures, are summarised in table 5.2 and are shown graphically on figs 5.4a to 5.4e.

1. Minimise the maximum nodal pressure error. Using only three pressure meters and no flow meters.

2. Minimise the average nodal pressure error. Using only four pressure meters and no flow meters.

3. Minimise the maximum flow error. Using only four flow meters and no pressure meters.

4. Minimise the average flow error. Using only four flow meters and no pressure meters.

5. Minimise the number of meters that are needed to achieve a pressure error of less than 0.5m in each node.

Table 5.1. List of possible meter sites.

Pressure meters			Flow meters		
Number	Node	Accuracy(m)	Number	Pipe	Accuracy(l/s)
1	1	0.1	1	22-69	0.5
2	7	0.1	2	7-8	0.5
3	8	0.1	3	23-36	0.5
4	10	0.1	4	66-67	0.5
5	17	0.1	5	58-57	0.5
6	18	0.1	6	64-75	0.5
7	22	0.1	7	2-6	0.5
8	24	0.1	8	23-18	0.5
9	25	0.1	9	63-65	0.5
10	26	0.1	10	8-10	0.5
11	38	0.1	11	66-63	0.5
12	44	0.1	12	22-7	0.5
13	46	0.1	13	38-42	0.5
14	54	0.1	14	53-54	0.5
15	56	0.1	15	11-12	0.5
16	58	0.1	16	16-15	0.5
17	63	0.1	17	34-25	0.5
18	65	0.1	18	56-45	0.5
19	66	0.1	19	160-59	0.5
20	69	0.1	20	61-76	0.5
21	70	0.1	21	72-79	0.5
22	72	0.1	22	24-17	0.5
23	75	0.1	23	38-39	0.5
24	80	0.1	24	60-70	0.5
25	81	0.1	25	24-37	0.5

Table 5.2. Results for test problems.

Problem	True optimal	Set covering method	Restricted search method
1	Cost: 0.78m Meters: 7, 46, 65	Time: 15 secs Cost: 0.79m Meters: 10, 46, 66	Time: 3 secs Cost: 0.83m Meters: 10, 18, 54
2	Cost: 0.29m Meters: 8, 18, 56, 66	Time: 90 secs Cost: 0.30m Meters: 8, 69, 56, 66	Time: 5 secs Cost: 0.30m Meters: 8, 69, 56, 66
3	Cost: 7.45l/s Meters: 22-69, 38-39, 22-7, 54-53	Time: 6 secs Cost: 7.74l/s Meters: 22-69, 38-39, 22-7, 54-53	Time: 4 secs Cost: 7.74l/s Meters: 22-69, 38-39, 22-7, 67-66
4	Cost: 2.97l/s Meters: 22-69, 22-7, 38-39, 54-53	Time: 70 secs Cost: 3.28l/s Meters: 22-69, 22-7, 38-39, 54-53	Time: 5 secs Cost: 3.28l/s Meters: 22-69, 22-7 38-39, 54-53
5	Cost: six meters Meters: 75, 65, 8, 17, 80, 44	Time: 2 secs Cost: seven meters Meters: 75, 65, 8, 17, 80, 44, 66	Time: 1 sec Cost: seven meters Meters: 75, 65, 8, 26, 80, 44, 66

Fig 5.4a : Meter locations for solution to problem 1.

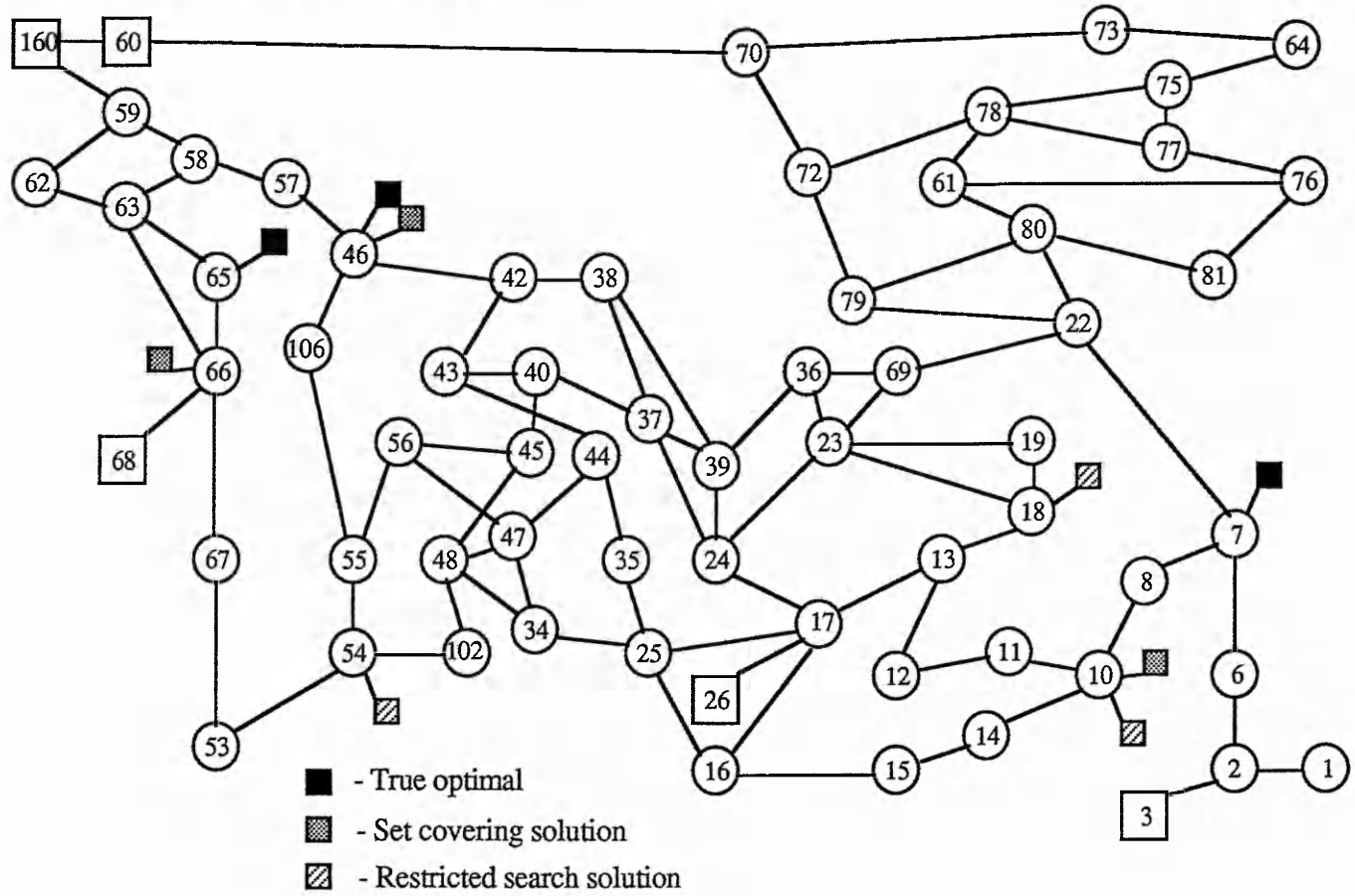


Fig 5.4b : Meter locations for solution to problem 2

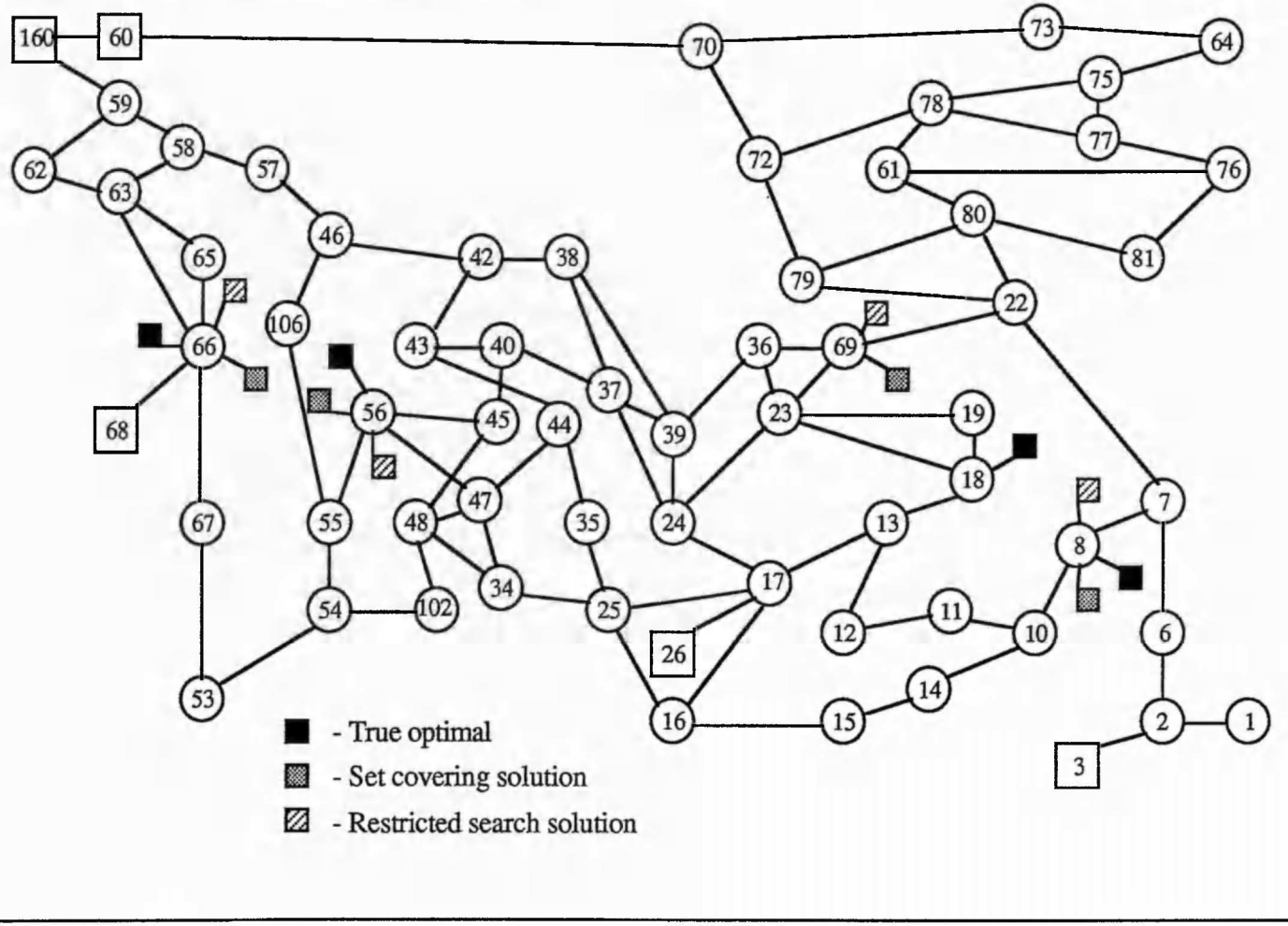
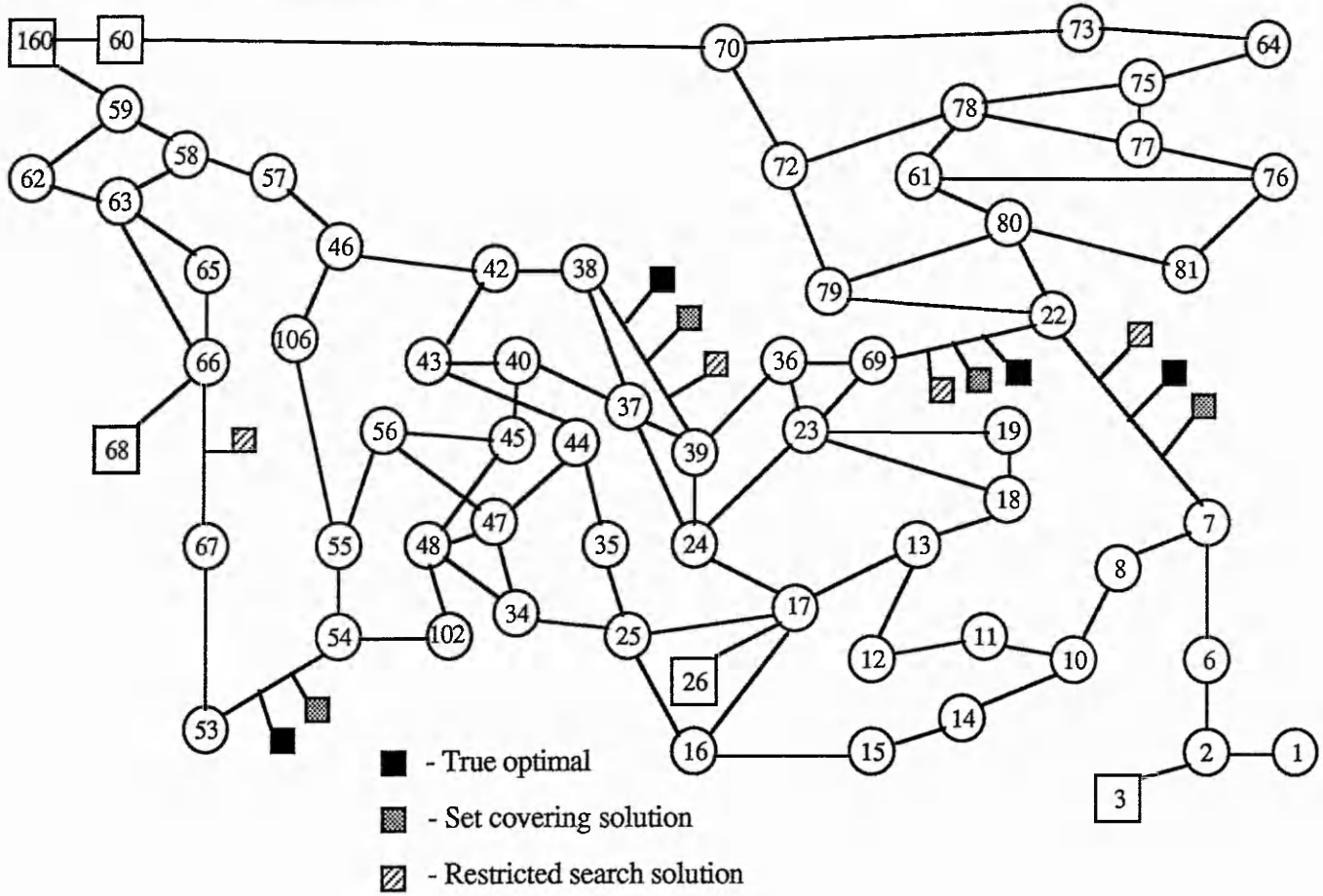


Fig 5.4c : Meter locations for solutions to problem 3.



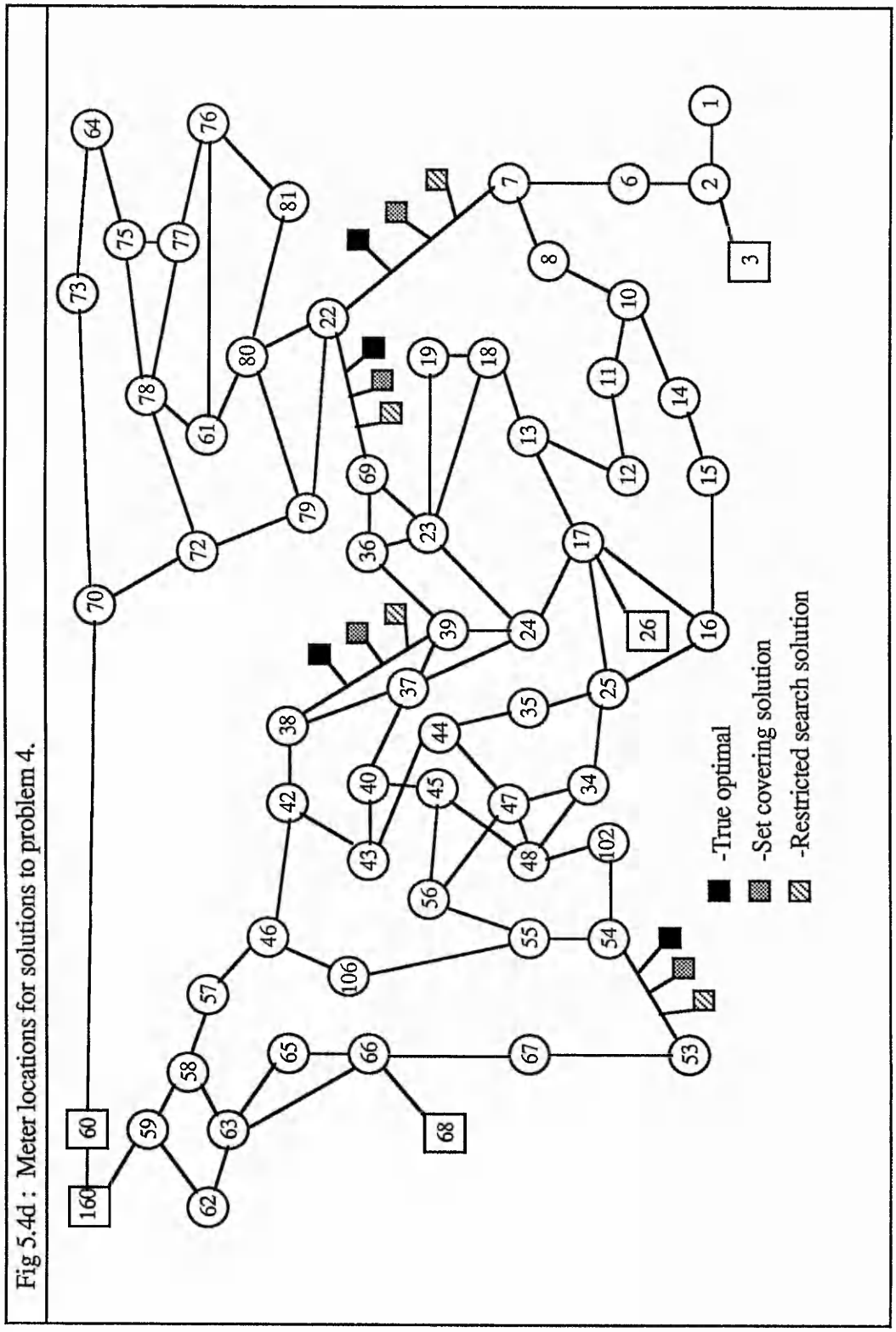
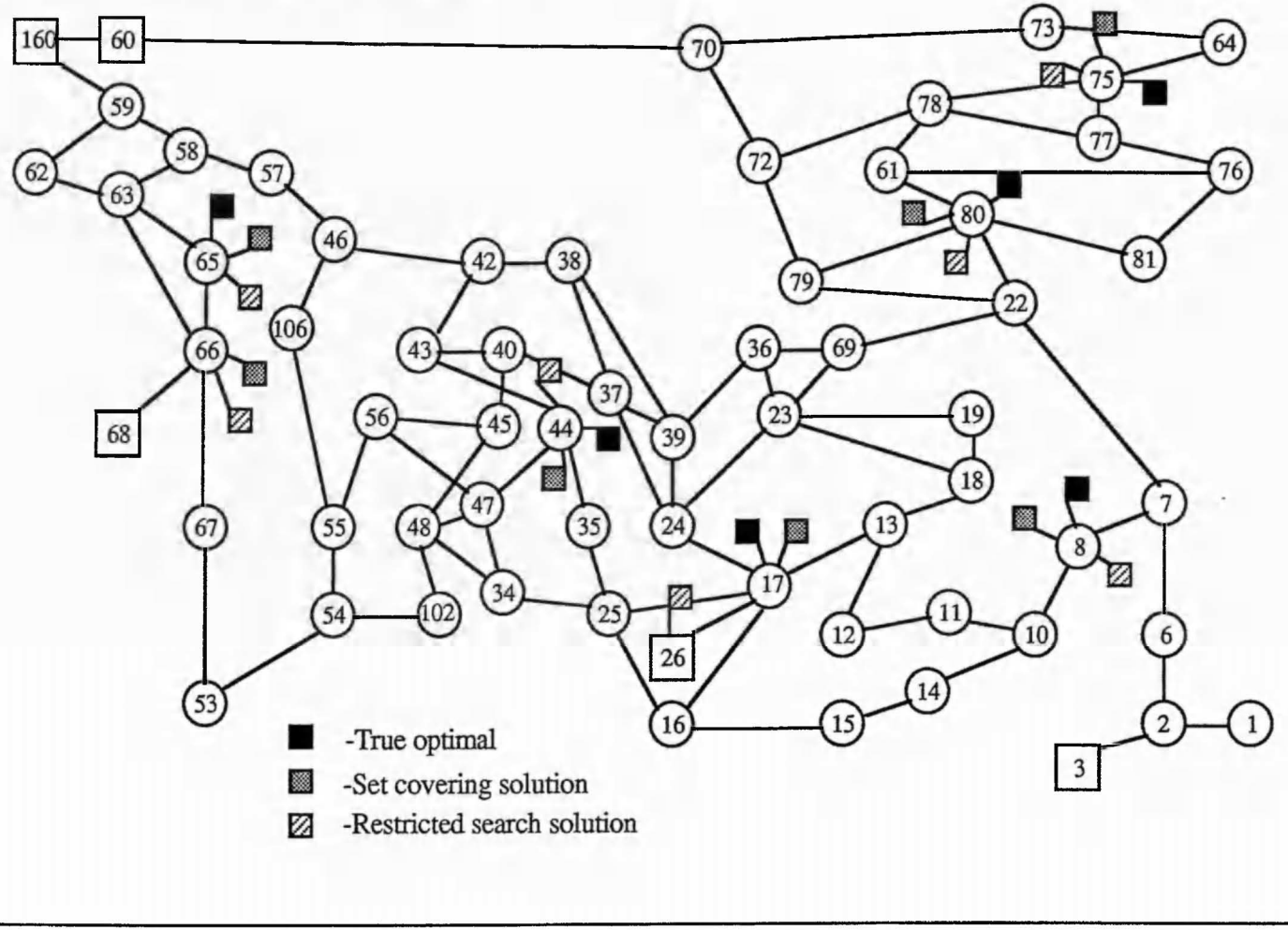


Fig 5.4d : Meter locations for solutions to problem 4.

Fig 5.4e : Meter locations for solutions to problem 5.



The base state assumed in these optimisations is the one corresponding to the period of peak demand - about 12.00 with this network.

A set of 'true optimal' results for these problems (ie ones that do not rely on the approximate e_i^* function) were generated. These are also shown in table 5.2. To obtain these 'true optimal' results, a full enumeration of all possible meter placements was performed. To evaluate the accuracy of each meter placement, the function e_i (using the sensitivity matrix confidence limit algorithm) was used. These results, when compared with those of the set covering method, allow the accuracy lost through using the approximate e_i^* function instead of e_i, to be assessed. This method cannot be considered as a feasible alternative to the set covering or restricted search procedures as it requires several hours of computing time to generate solutions to even the smallest of problems. In the three sets of solutions shown in table 5.2, there are two levels of approximation. The first is the use of e_i^* rather than e_i to calculate the accuracy of a particular meter placement. This is used in the set covering method. The second approximation is the restricted search procedure which is itself sub-optimal. The optimal meter placements for problems 3 and 4 are identical for both the 'true optimal' and set covering methods. In problem 2, two very similar meter sets, with nearly identical cost functions, are produced. A meter at node 69 in the set covering solution, replacing one at node 18 in the 'true optimal', is the only difference. For problem 1, the two procedures produce meter configurations that differ by two meters - node 10 in the set covering result replacing node 7 in the 'true optimal' and node 66 replacing node 65. However, nodes 65 and 66 are adjacent in the test network and nodes 10 and 7 are only separated by one node, so these two placements are again very similar. Also, the cost values of these two placements are nearly the same. In the final problem, test 5, the 'true optimal' procedure found a meter placement that can achieve a pressure error of less than 0.5m in each node, which contains only six meters. The set covering meter placement has seven meters to achieve this accuracy. However, for the two meter sets produced, the six meter set is a subset of the seven meter set. As the function

$e_i \leq e_i^*$ for all i (as is given by equation (5.12)), it can be expected that an optimisation of this type, using e_i, will produce a smaller set of meters than one using e_i^*. This observation also explains why, in each of the other problems (1 to 4) the cost values in the 'true optimal' column are smaller than those in the set covering column. This does not suggest that the set covering method is sub-optimal, rather it is explained by the fact that two different cost functions are being used. From these comparisons, it can be concluded that the use of the approximate e_i^* function, introduces only a small degree of inaccuracy into the results. Also, in view of the significant improvement in execution time which results from the use of the e_i^* function, its introduction can be justified.

By comparing the restricted search results with the set covering results, the accuracy lost by using the sub-optimal procedure can be assessed. (Both of these methods use the approximate function e_i^*). In two out of five of these examples the restricted search produced exactly the same result as the set covering method which generates a true optimal under the assumptions outlined in section 5.4 and when e_i^* is used in place of e_i. It did this in a fraction of the computing time. In example 3, the two meterings produced only differed by one meter site - a flow meter in pipe 67-66 replacing one in pipe 54-53 - and the cost functions were nearly identical. So in this case, although the restricted search method did not produce the optimal meter placement, it generated one that was very nearly optimal. In problem 5, the two solutions can, in some respects, be considered equivalent. The cost function used in this case was the number of meters used. Both the set covering method and the restricted search method produced optimal meterings with seven meters, ie meterings that are equally as good as each other. The fact that the two meter sets produced were different in composition - a meter in node 26 replacing one in node 17 - merely reflects the fact that there may be more than one combination of seven meters which measure the network to the required accuracy. Overall, the restricted search method performed favourably in comparison with the set covering method.

To demonstrate the method presented in section 5.6 - to extend the optimisation to cover a number of different states - the following optimisation

Minimise the average pressure error using only four pressure meters

was carried out for twelve different operating states using the set covering method. These states corresponded to the operating state of the test network at two-hourly intervals (0:00,2:00,....,22:00), for a typical day. The actual inflow figures used are shown in table 5.3. The optimal meter sets were chosen from the list of possible meter locations shown in table 5.1. The first step of the analysis was to find the optimal meter configuration for each of these time steps. This produced ten different 'optimal' meter sets (two sets were 'optimal' for more than one time). These are shown in table 5.4, together with their cost values. It should be noted that although ten different 'optimal' meter sets were produced, only thirteen from the possible twenty five pressure meter sites were used. Also, of these thirteen meter sites used some appeared very frequently, for instance node 56 appeared nine times, node 7 appeared seven times, node 66 appeared six times and nodes 24, 75 and 69 appeared five times each. Examination of table 5.4 and fig 5.5 reveals a strong pattern of distribution for the individual meter sites in the 'optimal' meter sets. For all of the ten 'optimal' meter sets, those that do not contain node 56 as a site contain node 44 instead. Similarly, in all but one case the second meter site in each of the 'optimals' is either node 7, 8 or 10. These three nodes are adjacent in the test network. The third meter site is either at node 66 or node 24 in all but one case and the fourth site is either at node 69, 75 or 81 in all but two cases. This distribution is shown more clearly in fig 5.5. The majority of the 'optimal' meter sets are made up of one meter from group 1, one from group 2, one from group 3 and one from group 4. Only four of the individual meter sites do not fit this pattern.

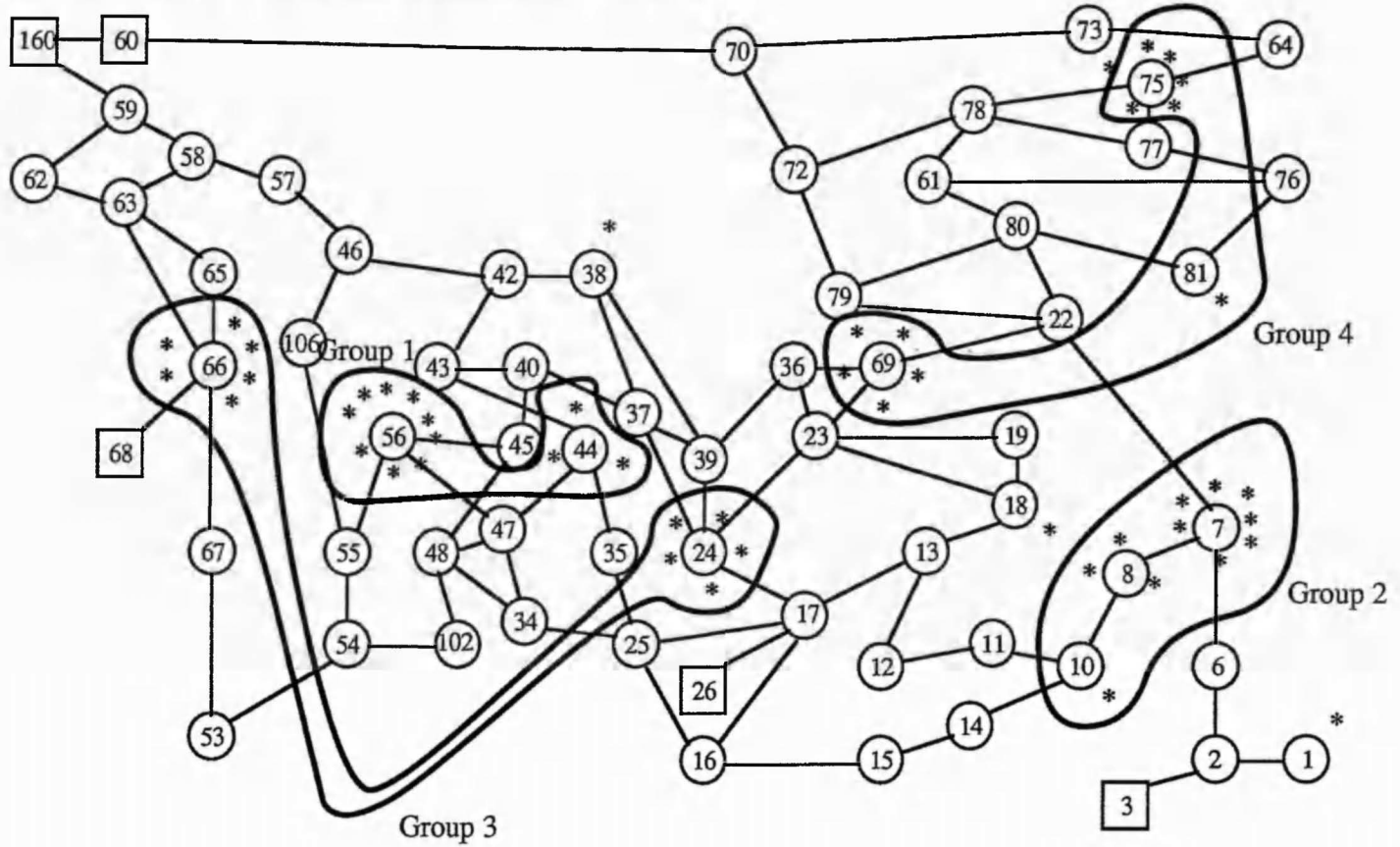
Each of these 'optimal' sets was tested at all of the twelve different times, their cost values calculated at each, to see if they are at all times 'good' configurations. In each row

Table 5.3. Typical inflow rates at 2-hourly intervals (in l/s).					
Time	Node 26	Node 3	Node 60	Node 160	Node 68
0:00	33.6	31.0	3.4	-41.3	28.3
2:00	18.6	30.6	-4.0	-14.7	26.9
4:00	18.6	30.3	-4.6	8.3	27.3
6:00	18.6	30.0	-4.6	32.2	27.6
8:00	46.6	30.6	19.2	44.5	30.3
10:00	63.0	30.3	30.0	44.0	30.3
12:00	62.0	29.3	25.2	40.0	29.7
14:00	50.4	30.8	18.5	32.5	29.3
16:00	45.6	32.6	19.3	33.0	26.7
18:00	44.6	33.3	29.4	45.0	25.7
20:00	46.6	32.6	23.4	-3.0	27.2
22:00	48.6	32.0	19.6	-51.1	28.2

Table 5.4 Optimal solutions for 2-hourly operating states.			
Time	Meter set	Cost	Meter set no.
0:00	56, 7, 24, 81	0.20 m	1
2:00	56, 10, 24, 1	0.17 m	2
4:00	56, 7, 18, 75	0.16 m	3
6:00	44, 7, 24, 75	0.18 m	4
8:00	44, 75, 66, 69	0.25 m	5
10:00	56, 8, 66, 38	0.32 m	6
12:00	56, 8, 66, 69	0.30 m	7
14:00	56, 8, 66, 69	0.26 m	7
16:00	56, 7, 66, 69	0.25 m	8
18:00	44, 7, 66, 69	0.29 m	9
20:00	56, 7, 24, 75	0.21 m	10
22:00	56, 7, 24, 75	0.23 m	10

Table 5.5. Average pressure errors (in m) for all optimal pressure meter sets for all times. C Matrix.										
Set Time	1	2	3	4	5	6	7	8	9	10
0:00	**0.20**	0.20	0.20	0.21	0.27	0.21	0.21	0.21	0.21	0.20
2:00	0.17	**0.17**	0.18	0.18	0.24	0.18	0.17	0.17	0.17	0.17
4:00	0.16	0.17	**0.16**	0.16	0.23	0.18	0.17	0.17	0.17	0.16
6:00	0.18	0.19	0.18	**0.18**	0.22	0.20	0.20	0.19	0.19	0.18
8:00	0.28	0.30	0.28	0.29	**0.26**	0.27	0.27	0.28	0.27	0.28
10:00	0.34	0.35	0.33	0.36	0.40	**0.32**	0.32	0.32	0.33	0.33
12:00	0.31	0.32	0.31	0.33	0.36	0.30	**0.30**	0.30	0.31	0.31
14:00	0.26	0.27	0.27	0.28	0.30	0.26	**0.26**	0.26	0.26	0.26
16:00	0.26	0.26	0.26	0.28	0.31	0.26	0.25	**0.25**	0.26	0.26
18:00	0.31	0.31	0.30	0.32	0.30	0.29	0.29	0.29	**0.29**	0.31
20:00	0.22	0.23	0.22	0.22	0.29	0.25	0.24	0.24	0.25	**0.21**
22:00	0.23	0.23	0.23	0.24	0.31	0.25	0.24	0.24	0.24	**0.23**
TD	0.09	0.17	0.09	0.22	0.65	0.10	0.09	0.09	0.12	0.07
MD	0.02	0.04	0.02	0.04	0.08	0.04	0.03	0.03	0.04	0.02

Fig 5.5 : Distribution of meter sites in ten 'optimal' meter sets.



of table 5.5 the bold entry corresponds to the cost value of the optimal meter set for that time. Table 5.5 must be used to decide which of the nine proposed meter sets is the best overall. The first thing that is noticed is that the optimal values vary throughout the day. This reflects the fact that the errors are largest when the water demand is high. For this reason, values in different rows should not be directly compared. Comparisons should be restricted to one row at a time. The "TD" and "MD" rows of this table are useful in deciding which of the sets is best overall. When table 5.5 is regarded as a C-matrix, with $C(i,j)$ representing the cost of the j^{th} meter set (j=1,...,10) at time i (i=0:00,...,22:00). Let j_i represent the meter set that is optimal for the i^{th} time, then $C(i,j_i)$ is the minimal entry in the i^{th} row of C. With this notation the "TD" and "MD" rows can be expressed as follows:

$$TD(j) := \sum_i (C(i,j) - C(i,j_i)) \tag{5.20}$$

$$MD(j) := \max_i (C(i,j) - C(i,j_i)) \tag{5.21}$$

The "TD" row gives the total discrepancy for each meter set and the "MD" row gives the maximum discrepancy for each meter set. Meter set 10, the 'optimal' meter set for times 20:00 and 22:00, has a maximum discrepancy of 0.02m and a total discrepancy of only 0.07m. This meter set, therefore, emerges as the best overall. Meter sets 1 and 2 - with TD's of 0.09 and MD's of 0.02 - and meter sets 7 and 8 all perform very well overall, so they can be regarded as adequate alternatives. It is noticeable that most of the 'optimal' meter sets have good accuracy at all times, with only set 5 as a significant exception. This emphasizes the importance of carrying out this second level of optimisation. If only a single study had been carried out with the base state corresponding to that of time 8:00, then meter set 5 would have been selected as the 'optimal'. This is clearly not the best choice.

5.8. CONCLUSIONS

In this chapter the problem of finding an optimal meter placement for a water network telemetry system is addressed.

All water distribution networks are unique and highly complex systems. A design problem such as this, therefore, can have no straightforward solution. Before the problem can be solved it must be described precisely in mathematical terms. A formulation to the problem based on the uncertainty model of Chapter 4 has been used as the basis for this. A cost function can be selected from one of three types. In this function, its coefficients can be varied to tailor it to the precise needs of the optimisation. In addition, any number of constraints can be used. Each of these is again selected from one of three types and has variable coefficients to allow the metering system's restrictions to be modelled. Two solution methods have been developed. One is based on an existing optimisation technique and can produce an optimal metering subject to the assumptions of the mathematical formulation. The second method uses a new procedure. This is heuristic in nature and can only guarantee a near-optimal solution. The optimisation methods are then extended to cover a range of operating states.

Results are presented which go only part of the way to demonstrate the applicability of this work. For the problem as formulated in section 5.4 with e_i^* used to calculted meter set accuracy, the set covering method produces accurate results with a guarantee of optimality but can be slow computationally. Alternatively, the restricted search method is much faster but may not have the same accuracy. In the examples given, the performance of the restricted search method compared very favourably with that of the set covering method, producing meter sets almost identical in cost but in a fraction of the time. Although the set covering method is the slower of the two it should not be discounted. If computer time is not a high priority, as is often the case in design applications such as this, this is the method that should be used as it provides optimal results with this simplified formulation.

The benefits of meter placement study for a water network telemetry system are clear. The state estimates calculated from these optimal meter sets can be significantly more accurate than those of a random placement of the same number of meters. This improvement in accuracy can then be utilised to achieve more efficient control. The cost of telemonitoring a water network can be very large. The methods of this chapter provide a way of ensuring that these resources are invested effectively.

CHAPTER 6

TCLAS AND DATACON

6.1. INTRODUCTION

It has been shown that the monitoring system's uncertainty and the random fluctuations of nodal consumption have a significant effect on the accuracy to which the system's variables can be estimated. Confidence limit analysis techniques can be used both in designing telemetry systems so as to minimise this impact and in quantifying its effects. The software packages TCLAS and DATACON, described in this chapter, have been developed to enable water system operators to take advantage of these techniques. Both packages are interactive and incorporate a full colour graphical display. (The Graphical display was generated using FLIB - a general purpose graphics library [15]). They have been designed for ease of use and present their results as informatively as possible. It is envisaged that these programs will be used within the water industry for a wide range of applications, including: telemetry system design; distribution system design; real-time control; decision support and operator training.

6.2. TCLAS

TCLAS - Telemetry Confidence Limit Analysis Software - can be used as an interactive meter placement design tool for telemetry systems as well as in decision support for the control and design of distribution systems. At its core it has a routine for confidence limit analysis which quantifies the uncertainty in pressure and flow estimates that is caused by the uncertainty of an input measurement set when the system is in a particular operating state. The current version of TCLAS uses an implementation of the sensitivity matrix

confidence limit algorithm (see section 4.5.4), although the modular design of the package allows an alternative routine, such as one based on the linear programming method (see section 4.5.3) for instance, to be easily incorporated.

TCLAS drives three graphics screens, pages 1, 2 and 3, to display the flow and pressure variables and their uncertainties. Page 1 displays a network diagram showing all load nodes, inflow nodes, pipes, pumps and valves simultaneously. Over this screen the pressure variables' uncertainty is shown. Page 2 displays the corresponding information for the flows in the network. Measurement set data, meter positions and accuracy can be requested on both of these pages. This is displayed as an overlay to the network diagrams. The right hand side of both pages 1 and 2 is reserved for written displays. Menus, keys and requested numerical data can be written here. Page 3 can be used to obtain more detailed information about the values of the variables in the region of a particular node. This screen has a small network diagram displayed in the bottom right hand corner, on which the cursor can be used to select a particular node. The region around this node, and its adjacent nodes and pipes, is then redrawn to fill the remainder of the screen. Included in this diagram are values for the pressure, flow and load variables in this region and their associated uncertainty. Also displayed are details of the measurements in this region.

A combination of colour graphic and written displays adds a flexibility to the presentation of the results and data. For instance the confidence limits for the state variables can be accessed in several different ways:

1. By means of colour coded error blocks. The range of possible errors for each type of variable is divided into three smaller ranges. Each of these has its own colour. Once the confidence limits for a particular variable have been calculated a rectangle is drawn by the appropriate node or pipe in the network in one of these three colours. The rectangle is filled in by a certain amount, representing the error bound of this variable. This presentation can be taken in by just one glance and gives a good picture of how errors vary

throughout the network.

2. For a more accurate numerical value the complete data for a particular node or pipe can be requested. This is displayed to the right hand side of the screen and includes all the associated node and pipe parameters. This presentation is useful when the errors need to put into context with the other parameters.

3. A particular node in the network can be 'zoomed' in on. This node together with all adjacent nodes and pipes are then drawn on the full screen together with the values and error bounds of all of the associated variables.

4. To keep a full record of the meter set and state estimate accuracy for later reference, there is an option to write such data to file.

6.2.1. Meter placement design with TCLAS

By selecting the appropriate options on the menu an operator can input a proposed meter configuration and then call on TCLAS to calculate the uncertainty in state estimates that would be produced by this configuration. The resultant confidence limits, which are presented in various formats, can be examined by the operator, who will then decide whether the proposed configuration is satisfactory. If the accuracy of the proposed meter placement is not acceptable in some way then it can be altered by either adding more meters, replacing some meters by more accurate ones or by moving some meters to different parts of the network. The accuracy of this new metering can be assessed and the process repeated if necessary. By experimenting with meter configuration in this way an operator can find a balance between accuracy and cost of metering.

At any one time TCLAS has stored a certain meter configuration, called the current measurement set. Initially, the current measurement set consists of a consumption prediction for each node, a prediction for the inflow at each inflow node and a prediction of

the pressure at one node (the reference pressure, this is usually a water level at a reservoir) with, in each case, an uncertainty bound. These pseudomeasurements are assumed present as they can be easily estimated. This measurement set will be referred to as the base measurement set. If any of these cannot be reliably estimated, a suitably large uncertainty bound can be used to reflect this. Without a minimal measurement set of a similar form to this, estimates for the network's flows and pressures cannot be calculated. So, this assumption is a necessary one. At any stage in the TCLAS session, one of the pseudomeasurements in the base measurement set can be replaced by a real, more accurate meter.

If, by calculating the error bounds, it has been shown that the current measurement set is not satisfactory, then it can be altered in several ways. If the accuracy is not sufficient then a new meter can be added. This is done by moving the input cursor to the appropriate place in the network diagram and selecting the add meter option. A meter must be added together with an error bound given in absolute terms. Alternatively it may be beneficial to change the accuracy of a meter already included in the current measurement set, or take a meter from the current meter set with the intention of placing it somewhere else in the network. The colour coded error blocks, discussed earlier, are invaluable in deciding where to place meters or which meters to remove. A glance is all that is needed to see which parts of the network are weakly measured and require extra meters. Also, when a meter is being added or when the accuracy of a meter is being changed, TCLAS suggests an appropriate range of accuracies. In some situations it may become apparent that the current meter set is too expensive or will not lead to an optimal meter configuration. In these circumstances meters may have to be removed, again the colour coded error blocks show where this can be done. The facilities for changing a measurement set (ie 'add meter', 'delete meter' or 'change accuracy' of a meter) make TCLAS flexible enough to cope with all situations and mean that an optimal meter configuration can be found quickly, simply and accurately.

An important factor to be considered when designing an optimal meter configuration is the operating state of the network. When TCLAS is first used it contains data typical of daytime operation. Since the operating state of the system has a large effect on the way meter accuracy relates to state estimate accuracy, it is important that simulation includes variation of this state. This can be done by changing the inflow pattern. Alternatively, there is a facility to model the operation of control valves by changing the C-values of the pipes. Whenever one of these features is used the state estimate is updated and new error bounds can be calculated.

At any time TCLAS can be called upon to show where each of the meters is located, as it is easy to lose track of the current measurement set. The accuracies of the meters placed in the network can also be accessed by requesting node or pipe data for the appropriate node or pipe.

6.2.2. Decision support with TCLAS

In order to make well informed control decisions it is necessary to have reliable estimates of all system variables. An operator needs to be able to access the values of all flows and all pressures. The inherent uncertainty in water network monitoring, and its effect on state estimate accuracy, has already been discussed. This problem means that a traditional, deterministic state estimator may provide misleading information. It can provide a single estimate for the state of the system, given the measurement data, but this data may be inaccurate. No reflection of this uncertainty is given by a deterministic state estimator. It has been shown [17] that even for a medium size network, with a shallow hydraulic gradient and a poor metering configuration, pressures can be in error by as much as 2.0m and flows by as much as 15.0l/s. TCLAS can provide this vital information for any measurement configuration. This is done by presenting an error bound with each flow or pressure value, thus specifying a range of feasible values or an uncertainty interval for

these variables.

In order to be able to calculate the state variable uncertainties, the meter location and measurement uncertainties are required. These can be input using the menu options described in the previous section. In the current version of TCLAS, the operating state of the distribution system can be altered by using the 'change inflow' option. Also, the pipes' C-values can be altered which has the effect of changing the state. These facilities are satisfactory when TCLAS is being used as a tool for distribution system design or for operator training. They allow a hypothetical state or network to be input. This can then be investigated to see whether proposed control changes or system design is acceptable in uncertainty terms. For decision support in real-time control, it is envisaged that TCLAS will be put on-line by linking it directly to the telemetry system. Meter readings will then be input directly and combined with demand predictions stored in a data file to produce state estimates and confidence limits in real-time.

By monitoring variables in this way, an operator can get an extra dimension of information. Below some examples of the use of this uncertainty data are given:

1. It is possible to check whether pre-set safety limits are being or are likely to be exceeded. In order to maintain a reliable distribution of water, certain operational limits can be specified. For instance it is necessary, in some parts of the network, to keep pressure above a certain level, ensuring that consumption requirements can be met. Alternatively, it is important to keep pressure and flows below a certain level so as to reduce the risk of burst pipes and leakage. A deterministic state estimator may predict that a pressure or flow variable is within the pre-specified safety limits. Whereas, it may be the case that the true value of the variable exceeds these limits. This possibility can easily be checked by comparing the variables' uncertainty interval with the safety limits.

2. In a similar way, limits for a variable can be set for economic reasons. For instance,

pumping costs can be minimised for a particular pressure distribution. It can easily be checked whether these limits are satisfied. Also an operator can see whether there is any scope for further cost reduction by identifying those variables that can be more strictly controlled.

3. Some pumps and valves in a network operate differently depending on the state of their control variables. The on/off nature of some of these elements makes it vital for an operator to be able to tell what state they are in. A deterministic state estimator can be particularly misleading in this situation, indicating the wrong operational state for one of these pumps or valves, and giving no guide as to how reliable that information is. Using TCLAS it is possible to see when these elements are definitely in one state or another and when there is some doubt.

It is easy to see how factors such as these above are important in helping an operator to make judgements and decisions towards effective control

In order to highlight the importance of such information, a specific example, based on a real occurrence, is given. Consider a network supplied via a pressure reducing valve (PRV) from a high pressure zone. This PRV will have a threshold pressure value, above which the valve is closed and below which the valve is open. When the reference pressure for the PRV is close to the threshold value, the valve may oscillate between open and closed states. This is because a pressure slightly above the threshold will close the valve and hence reduce pressure. This reduction in pressure will then cause the valve to open, hence increasing the pressure again. This process may repeat once started in motion, causing an undesirable on/off valve cycle. A deterministic simulator, because of pseudomeasurement error, may predict this valve to be definitely open or definitely closed when the opposite is in fact true. Furthermore, it will give no indication that there is any doubt as to which state the PRV is in. This can obviously be very misleading to an operator as it gives a completely inaccurate view of the system's input at that particular time. TCLAS, because it accounts

for measurement uncertainty, is able to point out that this undesirable valve behaviour is likely to be happening, prompting the operator to take an appropriate control action.

6.2.3. Further features of TCLAS

The two most important features of TCLAS are its graphical display and its interactive input. Much of the information is presented as an overlay to a full schematic diagram of the network or after the interactive cursor has been moved to a particular point of the network and data requested. It is believed that by highlighting this locational aspect the results are more accessible and easier to interpret. In particular, patterns of variation of accuracies throughout the network are easily noticeable and reference to pipes and nodes is clearer. Both interactive cursor and the keyboard are used to enter commands and data into TCLAS. The cursor is used to identify nodes and pipes and the keyboard is used to select options from a menu and enter numerical values.

TCLAS contains many other features which are designed for ease of use, some of which are mentioned below. At each stage in the use of TCLAS the operator is aided by prompts and menus. Other safety features include internal consistency checks. In these, each input, be it a command or a piece of data, is checked to see whether it is as TCLAS expects and consistent with all previous inputs. This reduces the risk of a mistaken entry going unnoticed. Already discussed is the ability to change pipe C-values, previously this facility was used to model valve operation, but it can also be used to tune the mathematical model. TCLAS presents the values and accuracies for each of the following variable types: flows; pressures; consumptions and inflows.

6.3. DATACON

To aid the creation and updating of water network data files, the program DATACON has been developed. This is again an interactive and graphic program that produces general

purpose water network data files as well as TCLAS specific data files.

The process of creating and updating network data files can be time consuming and is prone to error. Mistakes can easily be made, especially when dealing with numerical data. These mistakes may then go unnoticed in the data file, possibly leading to costly computational errors when the file is used. Even when a mistake has been identified it is not always a simple matter to locate the source of the error in the data file. DATACON has been developed to simplify the data input process.

At the start of a session, DATACON calls the data for the network that is to be amended and then draws a diagram for this network, along with a list of editing options, on the screen. Obviously, only the editing menu appears when a new network is to be created. The operator is invited to select one of the options which include adding or deleting a node and adding or deleting pipes. Positions of nodes and pipes in the network are identified by positioning the cursor over the appropriate spot in the diagram. When the add pipe or add node option is selected a series of pipe or node parameters are prompted for. At each stage the network diagram is updated on the screen so that the operator can check that the correct network is being constructed.

DATACON has several in built safety features which reduce the possibility of input error. It is believed that the graphical display makes certain errors easier to pick out and helps the operator keep track of what has been done. For instance, it is much easier to check that the network has the correct connectivity and that there are no isolated nodes by examining a network display rather than searching through a long list of figures. To prevent data being input out of sequence, each value is prompted for when it is required. For each input value DATACON performs a series of tests to check that the data is reasonable, within expected bounds and consistent with previous entries. The graphical display and cursor entry speed up the input process and provide a framework that is realistic and readily understandable for the operator.

On leaving DATACON, two data files are produced, both containing the complete data but in different formats. The first is to be used by later DATACON sessions and can be easily read by the operator. The second data file is intended for use by TCLAS.

CHAPTER 7

CONCLUSIONS AND FURTHER RESEARCH

7.1. CONCLUSIONS

Recent developments in telemetry and monitoring system hardware are bringing great advantages to the operation and control of water distribution systems. These systems, combined with the power of modern computers, mean that a much clearer picture of the operation of water systems is available and a higher level of automation is now practical. However, before these benefits can be fully exploited, some basic problems must be solved. One of the most important of these is the problem of uncertainty in the monitoring system and its impact on state estimate accuracy. This thesis presents the results a thorough investigation into this problem.

Two principal causes of uncertainty have been identified. These are: inaccuracies in the distribution system model and errors in the measurement and pseudomeasurement values used. In the latter type of uncertainty, the unpredictability of nodal consumption estimates makes the most important contribution, but this is amplified by weaknesses in the metering system used. The inaccuracy of network models has received much attention in the literature [11, 12, 25, 34, 91, 108, 109, 143, 144, 145, 146]: its causes have been identified; its impact assessed and methods proposed by which the models can be improved. By taking account of the experiences and suggestions made in these papers, it is possible to attain a very high level of accuracy for the network model, provided of course, that sufficient effort is put into modelling, calibration and recalibration. Unfortunately, research into the other main cause of uncertainty and its impact is not so far advanced.

To allow consideration of monitoring system uncertainty in water distribution system simulation, an uncertain network model is proposed. From this formulation, uncertainty intervals or confidence limits can be derived for the system variables. In this way, the results of traditional, deterministic state estimators can be extended to provide a more reliable picture of the operation of the system. Four confidence limit algorithms are presented, these are the Monte Carlo method, the linear programming method, the sensitivity matrix method and the ellipsoid method. The Monte Carlo method provides the most reliable quantification of state and derived state variable uncertainty, but at considerable computational cost. The accuracy of this method derives from its full treatment of network model non-linearity and the fact that the bounds it produces are attainable. It is used as a yardstick against which the performance of the other algorithms can be assessed. The remaining algorithms are based on a linearisation of the uncertain network model. With this linearisation, the linear programming method provides the most accurate results, although its execution time makes it more suitable for design or off-line applications such as operational planning or decision support. Comparison of the linear programming results with the Monte Carlo results show that in linearising the uncertain network model no significant accuracy is lost. The sensitivity matrix algorithm produces confidence limits that compare satisfactorily with those of the linear programming and Monte Carlo algorithms. It does this in just a fraction of the time - about two seconds for the test network shown in Appendix A1 using a VAX8700 computer. This computational speed together with its acceptable accuracy, make the sensitivity matrix algorithm the most suitable for real-time or on-line application. It is envisaged that this algorithm can be linked into a telemetry system to provide confidence limits for the system variables in real-time. Hence providing the uncertainty information for partially or fully automated control, when this becomes a realistic proposition. The ellipsoid algorithm has been suggested by other researchers, as an efficient method for quantifying uncertainty in more general problems [18, 20, 59, 100, 102, 103, 104, 124]. Unfortunately, it was found to perform badly in

water distribution system confidence limit analysis. Its computational speed compares favourably with the sensitivity matrix method but the accuracy of its results is poor.

It has been shown that the accuracy of state and derived state estimates depends strongly on the location and accuracy of meters that provide the measurement values from which they are calculated. This relationship means that by carefully designing the meter positioning in the network's telemetry system, it is possible to increase the accuracy of the monitoring system and hence the efficiency of control. From this basis, a mathematical optimisation problem has been formulated. This is referred to as the optimal meter placement problem and its derivation is consistent with the uncertain network model and confidence limit analysis. The flexibility of the cost and constraint functions in this optimisation problem, mean that many realistic metering priorities and restrictions can be modelled and accounted for in the solution. To solve this problem, two solution techniques are proposed. These are the set covering method and the restricted search method. The set covering method is based on a branch and bound solution to this discrete location problem and can guarantee an optimal solution to the problem as formulated. The restricted search method is a near-optimal procedure that provides results that are, in most cases, identical or equivalent to the solution of the set covering method. The computational speed for the restricted search method is much faster than for the set covering method, but both are suitable for the optimal meter placement problem, which is a design application and need not be carried out in real-time.

All of the algorithms presented in this thesis have been implemented as computer programs. In particular, a software package TCLAS has been developed. This incorporates a confidence limit algorithm together with a full graphic and interactive interface. Together with its sister program - DATACON - TCLAS can be used for meter placement and telemetry system design, for decision support, for network design, for operational planning or as an educational tool.

7.2. SUGGESTIONS FOR FURTHER RESEARCH

(i) In this thesis, an uncertain network model is proposed. This concept needs to be explored in all other areas of water distribution system design, simulation and control. In particular, the question of how simulation uncertainty affects existing control or operational procedures should be examined. If a higher level of automation in water distribution is to be achieved, a much higher demand will be placed on simulation results. Part of this demand is met by the quantification of uncertainty through confidence limit analysis proposed in this thesis, but also, more research into methods to cope with the uncertainty is required.

(ii) The inherent uncertainty in water distribution simulation means that expert system control may be more appropriate than traditional, algorithmic control methods. This possibility should be explored. Work in this area has already started with the WIESC project at the University of Surrey [2], but as yet has produced little published work. A prerequisite for expert system control is an adequate method for dealing with uncertainty. Confidence limit analysis algorithms provide this information in a particularly accessible way. For instance, by comparing observed confidence limits with predicted confidence limits it may be possible to indicate when and where a fault has occurred or when the monitoring system has failed in some way.

(iii) Uncertainty in network modelling and measurement uncertainty have, because of their inherent differences, been examined separately. It may prove beneficial to investigate the possible interaction between these two sources of error.

APPENDIX 1

TEST NETWORK

The network described below is presented as a test network for which example results can be calculated. This network has 65 nodes, 92 pipe and 5 inflow points. The 5 inflow points are at nodes 160 and 60 which represent reservoirs, node 68 which represents a pumping station, node 3 where water is supplied from a zone of higher pressure through a pressure-reducing-valve (PRV) and at node 26. Pipe data - lengths, diameters and C-values - are contained in table A1.1 and the layout of the network is shown in fig A1.1. The network is based on an operational supply network.

The algorithms and computer programs have been tested on many other networks, some larger and more complex. But this one was chosen as it is large and realistic enough to demonstrate the efficiency and potential of the tested algorithms, while not being so large that an excessive amount of data would be required to describe it. In results calculated for larger networks similar patterns of behaviour are apparent and similar conclusions can be drawn.

Fig A1.1 : Test network.

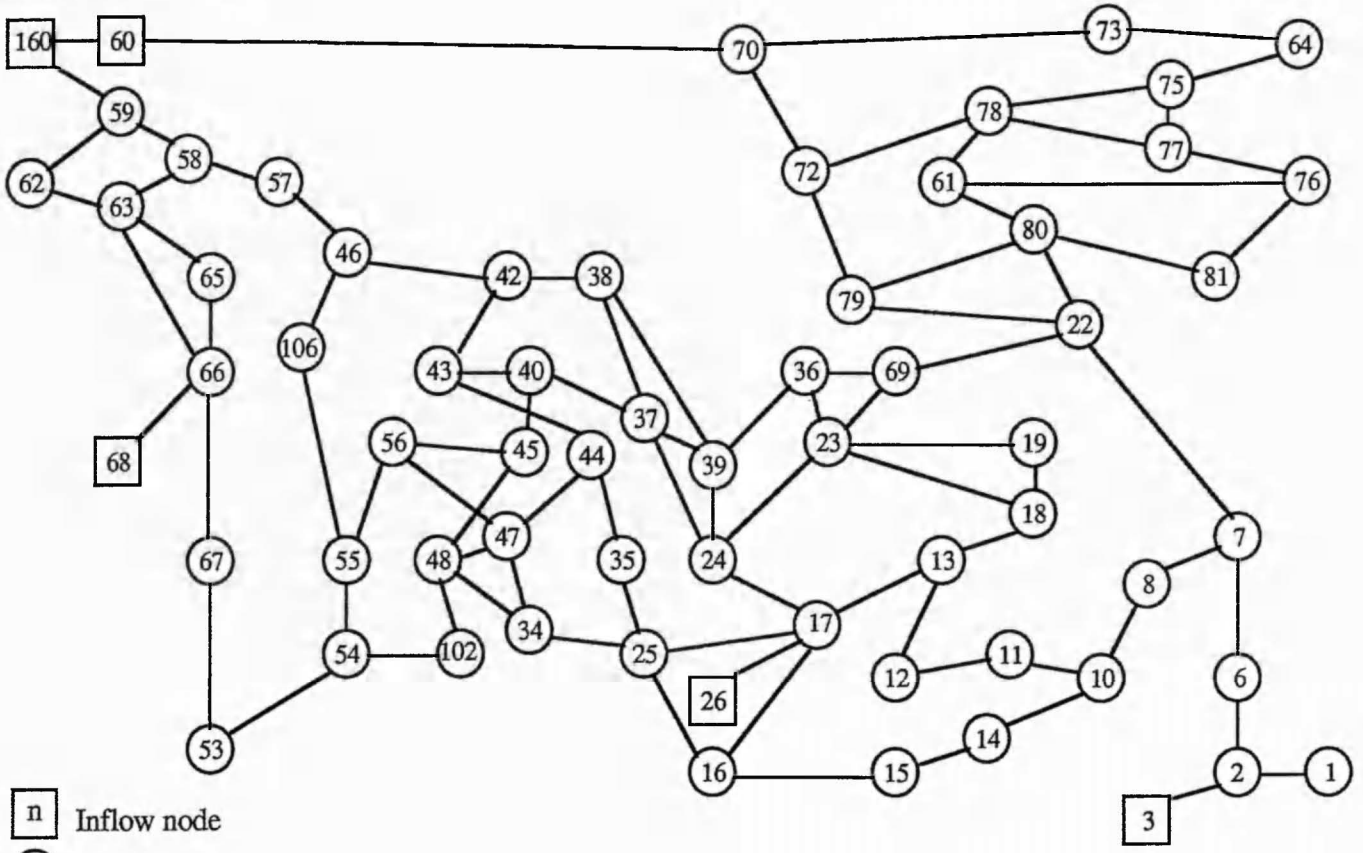


Table A1.1: Test network pipe data

Pipe	Length (m)	Diameter (m)	C-value
1-2	800.0	0.200	140.0
2-3	400.0	0.300	165.0
2-6	400.0	0.300	165.0
6-7	970.0	0.300	165.0
7-8	300.0	0.225	135.0
8-10	350.0	0.225	171.1
8-11	350.0	0.125	60.0
11-10	360.0	0.150	105.0
11-12	180.0	0.125	60.0
12-13	200.0	0.175	114.8
10-14	710.0	0.225	110.0
14-15	225.0	0.225	110.0
15-16	310.0	0.225	90.0
16-17	590.0	0.094	80.0
17-13	740.0	0.175	114.8
13-18	250.0	0.225	157.8
18-19	330.0	0.300	145.0
22-7	1510.0	0.225	95.7
22-69	120.0	0.300	145.0
69-23	420.0	0.150	90.0
23-24	300.0	0.300	145.0
24-17	650.0	0.200	158.4
17-25	330.0	0.175	127.1
25-16	630.0	0.225	104.4
26-17	360.0	0.300	80.0
25-34	780.0	0.175	80.0
25-35	320.0	0.225	119.3
35-44	710.0	0.225	90.0
44-47	520.0	0.250	70.0
47-34	610.0	0.225	145.0
47-48	540.0	0.250	70.0
48-34	900.0	0.175	80.0
48-102	310.0	0.250	70.0
102-54	660.0	0.225	140.0
54-53	480.0	0.225	157.8
54-55	380.0	0.225	159.3
55-56	190.0	0.225	145.0
56-45	610.0	0.125	55.0
45-48	1060.0	0.175	80.0
44-43	230.0	0.250	80.0
43-40	380.0	0.250	80.0
40-45	580.0	0.168	120.0
40-37	320.0	0.168	120.0
37-24	390.0	0.200	145.0
37-39	280.0	0.168	120.0
39-24	300.0	0.150	90.0
37-38	270.0	0.200	145.0
38-39	550.0	0.300	229.0
39-36	210.0	0.300	126.9

Table A1.1 (continued)			
Pipe	Length (m)	Diameter (m)	C-value
36-23	180.0	0.300	112.0
36-69	147.0	0.300	145.0
22-79	160.0	0.225	80.0
79-80	340.0	0.150	90.0
80-22	390.0	0.200	145.0
80-81	1220.0	0.150	139.0
81-76	600.0	0.150	145.0
76-77	670.0	0.150	116.0
77-78	150.0	0.150	116.0
78-61	460.0	0.094	170.0
61-80	530.0	0.150	145.0
78-72	1100.0	0.142	105.0
72-79	600.0	0.225	60.0
72-70	1770.0	0.225	47.0
70-73	3090.0	0.356	46.2
73-64	410.0	0.225	80.0
64-75	420.0	0.225	80.0
75-78	350.0	0.094	170.0
75-77	400.0	0.150	145.0
61-76	1470.0	0.150	81.0
70-60	2200.0	0.356	100.0
160-59	370.0	0.381	50.0
59-58	630.0	0.300	50.0
58-57	730.0	0.300	118.4
57-46	260.0	0.300	85.0
46-42	250.0	0.300	85.0
42-38	430.0	0.300	145.0
42-43	720.0	0.250	80.0
46-106	720.0	0.200	145.0
106-55	700.0	0.142	137.0
56-47	550.0	0.225	145.0
53-67	220.0	0.225	160.7
67-66	270.0	0.225	160.7
66-63	800.0	0.225	110.0
66-65	210.0	0.125	60.0
65-63	590.0	0.125	60.0
63-58	2050.0	0.150	40.0
63-62	770.0	0.225	110.0
62-59	350.0	0.225	110.0
23-19	430.0	0.300	145.0
23-18	760.0	0.117	60.0
66-68	440.0	0.300	170.0
60-160	270.0	0.381	32.0

APPENDIX 2

SPARSITY EXPLOITING TECHNIQUES

A2.1. INTRODUCTION

A sparse matrix is a matrix which has a large proportion of zero elements. This type of matrix arises in many applications. In particular, the Jacobian matrix used in many of the algorithms presented in preceding chapters is highly sparse. The sparse structure of the Jacobian matrix results from the nature of the network equation (3.2). Equation (3.2) represents a set of measurement equations, each one dependent on only a small number of independent state variables. Of all of these measurement equations, the ones derived from nodal consumption predictions and load measurements are the ones which depend on the largest number of state variables. Typically, this type of equation gives rise to only 5 or 6 entries in the Jacobian matrix, when the nodal equation format is used. When this is compared with the total number of independent state variables, which may be several hundred, it is clear that the Jacobian is very sparse.

In many applications of water network analysis, manipulation of the Jacobian matrix involves substantial computational effort. In water systems research, and in the analogous field of power systems research, it has long been recognised that significant reductions in the run-time of simulation algorithms can be achieved by exploiting the sparsity of the matrices involved [13, 19, 27, 42, 84]. Sparse matrix methods have also received considerable attention from a more general mathematical viewpoint [41, 45, 46, 48, 50, 51, 62, 64, 72, 73, 80, 85, 96, 97, 105, 106, 114, 118,119, 139, 140, 141, 152].

Motivated by the requirements of the work described in this thesis, two sparse matrix problems have been identified. These are:

(i) Solution of a set of simultaneous linear equations taking the form:

$$A.\mathbf{x} = \mathbf{b} \qquad \text{(A2.1)}$$

where A is a sparse, non-singular and square matrix. The state estimation routine, MINSTEST (section 3.3.3), requires at each iteration the solution of (3.9) which represents a set of linear equations in the form of (A2.1). In this case A represents the matrix J (square and non-singular), $\mathbf{b}$ represents the vector $\mathbf{z} - g(\mathbf{x}^k)$ and $\mathbf{x}$ represents the unknown correction vector. For the over-determined case, ODSTEST requires at each iteration the solution of (3.14). This can again be represented by (A2.1) with A used to represent the augmented matrix, $\mathbf{b}$ used to represent the vector $(\Delta z,0,0)^T$ and the unknown $\mathbf{x}$ used to represent $(\mathbf{s},\mathbf{r},\Delta\mathbf{x})^T$

(ii) Inversion of a sparse matrix A. Need for this type of technique arises in the sensitivity matrix algorithm of section 4.5.4 where the sparse matrix J^TJ must be inverted. In the implementation notes following the algorithm it is explained how J^TJ can be inverted by solving equation (4.51) for a sequence of right-hand-side vectors. Equation (4.51) has the form of (A2.1) and so inversion of J^TJ can be achieved by repeated application of type (i) techniques. Also, the linear programming confidence limit algorithm of section 4.5.3 requires the partial inversion of the sparse square matrix J^n. Again it is explained how the necessary rows of $(J^n)^{-1}$ can be obtained by solution of a series of equations (4.40) which have the form of (A2.1). In both of the examples, inversion of a sparse matrix is replaced by solution of a series of equations of form (A2.1). For this reason, sparse matrix inversion is not treated as a separate problem. In both cases the matrix is the same throughout the series of solutions, it is only the right-hand-side vector that is changing. This means that the matrix in each case, J^TJ or J^n, need only be factorised once, with the unknowns being determined by repeated back-substitution into the factorised matrix.

In many other areas, efficient design of matrix handling techniques will bring advantages. For example, in matrix multiplication and matrix-vector multiplication routines, care must be taken to ensure that no unnecessary mathematical operations are performed. Some suggestions for improvements are given in the implementation notes that follow the algorithms. Other, more general techniques are discussed in [50, 114].

Some methods for the solution of (A2.1) with sparse matrices are now reviewed. Details of the method used in this work will also be given.

A2.2. SOLUTION OF A SYSTEM OF SPARSE LINEAR EQUATIONS

Methods to obtain a solution of $A.\mathbf{x} = \mathbf{b}$ (equation (A2.1)) are commonly based on Gaussian elimination or one of its variants. In these procedures, the matrix A and the right-hand-side vector $\mathbf{b}$ are updated by a sequence of elementary operations. This leads eventually - after n steps when n is the rank of A - to a new matrix A^n and a new right-hand-side vector $\mathbf{b}^n$, where A^n is a triangular matrix. In practice, A and $\mathbf{b}$ may not be updated simultaneously. To allow solution of (A2.1) for the same matrix but several right-hand-side vectors, it is usual for A to be factorised first, with a record kept of this process so that each $\mathbf{b}$ can be updated correspondingly. This means that factorisation of A need only take place once. After the updating or factorising process has been performed, (A2.1) can be replaced by

$$A^n.\mathbf{x} = \mathbf{b}^n \qquad \text{(A2.2)}$$

This equation is equivalent to (A2.1) and from it the solution can be found by forward-substitution. Most texts on numerical analysis or linear algebra will include a description of this algorithm, see [58, 101, 114] for example.

In these elementary operations, the current (partially factorised) matrix A^k $(k < n)$ is updated to form a new matrix A^{k+1} and $\mathbf{b}^k$ is updated to form $\mathbf{b}^{k+1}$, where

$$A^{k+1}(i,j) = A^k(i,j) - \frac{A^k(i,k).A^k(k,j)}{A^k(k,k)}, \quad i=1,...,n \quad j=k+1,...,n \tag{A2.3}$$

$$b_i^{k+1} = b_i^k - \frac{A^k(i,k).b_k^k}{A^k(k,k)}, \quad i=k+1,...,n \tag{A2.4}$$

The aim of each of these k steps is to remove the non-zero entries in the k^{th} row that are to the right of the leading diagonal. Examination of (A2.3) and (A2.4) gives some idea of why it is beneficial to design Gaussian elimination techniques that account for sparsity. If, in (A2.3) a large value for $A^k(i,k).A^k(k,j)/A^k(k,k)$ were subtracted from a small value for $A^k(i,j)$, or vice versa. Then the information present in the smaller number would be lost. Similarly for large and small values in equation (A2.4). This problem is referred to as numerical instability and is particularly troublesome when very small values of $A^k(k,k)$ - the pivot - are used. Once computational inaccuracy has been introduced into one of the A^k matrices it is likely to be magnified by the successive updating steps and can lead to significant errors in the final matrix A^n. Stability of Gaussian elimination can be partially controlled by selecting suitable pivots. One suggestion [48, 129] is to choose an element $A^k(k,k)$ as pivot which satisfies one of the inequalities

$$A^k(k,k) \geq u. \max_{k \leq i \leq n} |A^k(i,k)| \tag{A2.5}$$

$$A^k(k,k) \geq u. \max_{k \leq j \leq n} |A^k(k,j)| \tag{A2.6}$$

for a pre-set parameter, u, in the range $0.0 \leq u \leq 1.0$.

If sparsity has been preserved through to the matrix A^k, many of the updating calculations of (A2.3) and (A2.4) are not necessary. For instance, when either $A^k(i,k)$ or $A^k(k,j)$ is zero, then $A^{k+1}(i,j) = A^k(i,j)$. Similarly, when either $A^k(i,k)$ or b_k^k is zero, $b_i^{k+1} = b_i^k$. A pivotal strategy which selects as a pivot an element $A^k(k,k)$ which will result in a small number of updates will clearly be advantageous. When a pivot is chosen for which $A^k(i,j)$ is zero but $A^k(i,k).A^k(k,j)$ is not, for some $i \in \{1,...,n\}$, then fill-in is said to occur in the $A^{k+1}(i,j)$ element. Excessive fill-in is undesirable as it means that the A^{k+1}

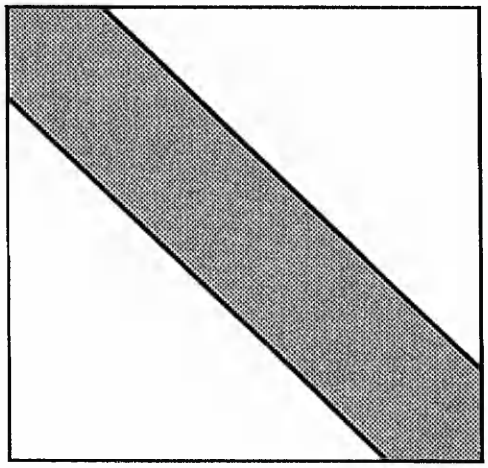
matrix is less sparse than it may have been and so more updating calculations are required in future steps. Selecting an appropriate pivot can therefore have a significant effect on the number of updates that are required in the current and the following steps of the elimination. Markowitz [97] suggests that the selected pivot should be the one for which the product of the other non-zeros in its column and the other non-zeros in its row is minimal. Berry, in [19], suggests that the pivot chosen should be the one which minimises local fill-in. This strategy involves many more comparisons and Duff and Reid, in [49], argue that its results may not be much better than those of Markowitz's method.

The preceding paragraphs have shown how a careful pivot selection strategy, applied locally, can be beneficial in improving numerical stability or in reducing the number of mathematical calculations that are required in the Gaussian elimination process. The same or similar considerations apply to other methods for solving simultaneous linear equations.

In the pivotal strategies described above, the pivot selection is dependent on the current partially factorised matrix A^k. Another class of sparsity exploiting techniques rely on a pre-processing of the matrix A. Usually these involve permuting the rows and columns of A (and the corresponding permutations of $\mathbf{x}$ and $\mathbf{b}$), to obtain some desirable form for the the matrix. Row and column permutations have no effect on the final solution but can significantly improve the the performance of the algorithm. The amount of fill-in and the number of update calculations can be reduced greatly. The reduction in the number of calculations performed may in turn increase the numerical stability of the algorithm. Duff, in [48], gives a thorough survey of these methods.

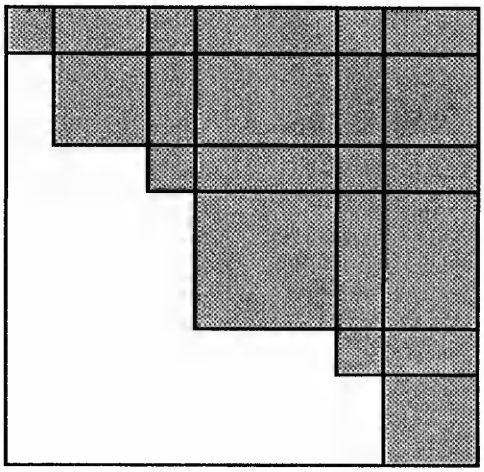
In [139, 140, 141], Tewarson describes some of the desirable forms for sparse matrices. These require the non-zero elements of the sparse matrix to be contained within specific areas of the matrix. Also included in [139, 140, 141] is an explanation of why these desirable forms are advantageous and methods by which some of them can be obtained. Two of these desirable forms are now examined. These are: the banded form

Fig A2.1 : Sparse matrix in banded form.



All of the non-zero entries are contained in the shaded region.

Fig A2.2 : Sparse matrix in block-triangular form.



All of the non-zero entries are contained in the shaded region.

(see fig A2.1) and the block-triangular form (see fig A2.2). Before describing these forms and methods by which they can be obtained, some associated graph-theoretic terminology is introduced. The distribution of the non-zero entries of a square matrix can be represented by a directed graph. Suppose that the matrix A has n rows (and hence n columns) and let $V := \{1,...,n\}$. A subset E of $V \times V$ can be defined as

$$E := \{(i,j) \in V \times V : A(i,j) \neq 0.0\} \qquad \text{(A2.7)}$$

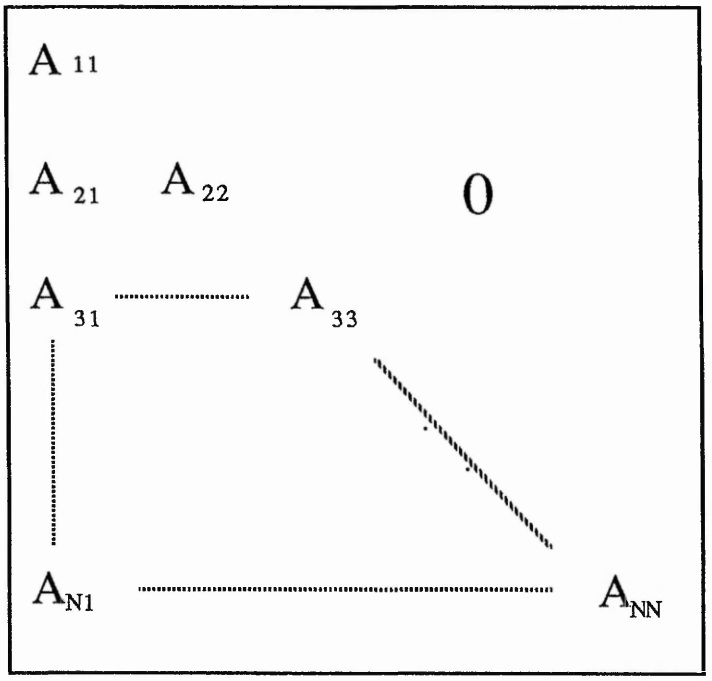
A directed graph which has vertex set V and edge set E, can be defined for the matrix A. This will be denoted by $G(A) = (V,E)$ and will be referred to as the matrix graph for A. In other words, $G(A)$ is the directed graph that has one node for each row (or column) of A and a directed edge for each non-zero entry in A. The graph-theoretical properties of sparse matrices are examined in [72, 73, 105, 119]

In some cases it is possible to re-order the rows and columns of the matrix so that all of the non-zero entries fall within a narrow band about the leading diagonal. Such a matrix is called a banded matrix, see fig A2.1. The advantage of such an ordering is that fill-in during the elimination is restricted to this band of the matrix. Hence, the narrower the band the smaller the amount of fill-in. The principal bandwidth minimisation techniques are based on an ordering of the vertices of the graph $G(A)$, usually this ordering is derived from the graph's adjacency structure. This vertex ordering will induce a symmetric permutation of the matrix. Cuthill and McKee [41] propose such a method. One of the graph's vertices is selected and numbered by 1. All of the vertices adjacent to this vertex are numbered 2. The ordering continues, with each new vertex being ordered by i when it is adjacent to a vertex numbered by $i-1$. This ordering scheme is equivalent to an ordering based on a breadth-first search of the graph. The choice of the starting vertex in this scheme can be important. Cuthill and McKee select the starting vertex after testing several vertices of near-minimal degree (the degree of a vertex is the number vertices adjacent to it). Other researchers make suggestions for the starting vertex, for example, Gibbs et al

[64] and George and Liu [62]. A slightly different ordering scheme is due to King [85]. This method again requires a starting vertex. At the k^{th} $(k=1,...,n)$ stage of the ordering, the subset V^k of V, which consists those vertices that are not yet numbered but which are adjacent to a vertex that is, is considered. From this set, that vertex whose numbering will result in the fewest unnumbered vertices being added to V^{k+1}, is selected and numbered k.

For a matrix in block-triangular form, the blocks can be numbered as in fig A2.3. Each of the diagonal blocks, $A_{11}, \ldots, A_{NN}$ (N will be less than n, if $N = n$, the matrix is in triangular form), is square and non-singular. When this is the case, each of the diagonal blocks can be factorised separately, using Gaussian elimination or one of its variants, and the unknown vector calculated using block-forward (or back) substitution. Permuting a matrix to this form has the advantage of reducing the problem to one of solving a collection of smaller problems. The non-zeros are also restricted to the existing non-zero blocks during factorisation and so fill-in is controlled. Duff and Reid [45, 46, 47, 51] define the problem of permuting to block-triangular form as a two part one. Firstly, unsymmetric permutations are performed to ensure that the leading diagonal is zero-free. If this is not possible then the matrix is structurally-singular. After this has been done, symmetric permutations are carried out to provide the block structure. The first part of this method requires a maximal transversal to be found for the collection of sets $\{V_i \subseteq V : i \in \{1,...,n\}\}$, where $V_i := \{j \in \{1,...,n\} : (i,j) \in E\}$. The set V_i, $i \in \{1,...,n\}$, is just the set of vertices adjacent to vertex i in $G(A)$. Finding a maximal transversal is a matter of associating a unique column, j_i, with each row, i, so that $A(i,j_i) \neq 0.0$. When the columns of A are ordered with j_1 first, followed by j_2 and so on, the permuted matrix has a zero-free leading diagonal. Algorithms for obtaining a maximal transversal can be found in [23, 69, 75, 83]. Two important algorithms for determining the symmetric permutation into the final block-triangular form are presented in [122] and in [138]. Duff and Reid explain how these can be used in [51].

Fig A2.3 : Numbering of blocks in a block triangular matrix .



After considering many of these sparsity exploiting techniques, the precise method chosen is due to Duff and is implemented in the package of sparse matrix routines, MA28 [45, 46, 50, 51]. In this method, the matrix A of (A2.1) is first ordered so that it is in lower block-triangular form. Each of the diagonal blocks, $A_{11}, A_{22}, \ldots, A_{NN}$ are square and non-singular, as is shown in [46]. The diagonal blocks are each factorised into lower triangular form separately, again using sparsity exploiting methods. In the factorisation of each block the Markowitz pivotal strategy is used in conjunction with the strategy that limits the size of the pivot (pre-set parameter u is selected so that $0.0 \leq u \leq 1.0$). Once each diagonal block has been factorised, equation (A2.1) can be solved by block-forward-substitution.

REFERENCES

1. S AAM, L HOLTEN and O GJERDE, "Design of the measurement system for state estimation in the Norwegian high-voltage transmission network", IEEE Trans PAS, Vol PAS-102, No 12, Dec 1983.

2. K AHMAD, P R HOLMES-HIGGIN, A J LANGDON and M WAREING, "The water distribution network expert system", Computing Unit, University of Surrey, 1987.

3. H ALEGRA, "Use of simulation models of water supply systems - processing of input and output data", presented at the international conference "Computer applications for water supply and distribution", Leicester, Sept 1987.

4. R N ALLAN and M R G AL-SHAKARCHI, "Probabilistic a.c. load flow", Proc IEE, Vol 123, No 6, June 1976.

5. R N ALLAN and M R G AL-SHAKARCHI, "Probabilistic techniques in a.c. load-flow analysis", Proc IEE, Vol 124, No 2, Feb 1977.

6. R N ALLAN and M R G AL-SHAKARCHI, "Linear dependence between nodal powers in probabilistic a.c. load flow", Proc IEE, Vol 124, no 6, June 1977.

7. R N ALLAN, B BORKOWSKA and C H GRIGG, "Probabilistic analysis of power flows", Proc IEE, Vol 121, No 12, Dec 1974.

8. R N ALLAN, C H GRIGG and M R G AL-SHAKARCHI, "Numerical techniques in probabilistic load flow problems", Int J for Numerical Methods in Engineering, Vol 10, pp 853-860, 1976

9. R N ALLAN, C H GRIGG, D A NEWEY and R F SIMMONS, "Probabilistic power-flow techniques extended and applied to operational decision making", Proc IEE, Vol 123, No 12, Dec 1976.

10. R N ALLAN, A M LEITE DA SILVA and R C BURCHETT, "Evaluation methods and accuracy in probabilistic load flow solutions", IEEE Trans PAS, Vol PAS-100, No 5, May 1981.

11. R ALLEN, "Network analysis - The real story", presented at the international conference "Computer applications for water supply and distribution", Leicester, 1987.

12. AWWA Committee report, "Water distribution and applied development needs", Journal of AWWA, Vol 66, No 6, June 1974.

13. A BARGIELA, "On-line monitoring of water distribution networks", PhD Thesis, University of Durham, May 1984.

14. A BARGIELA, "An algorithm for observability determination in water-system state estimation", Proc IEE, Vol 132, Pt D, No 6, Nov 1985.

15. A BARGIELA, "FLIB - Frontier graphics library", Department of Computing, Trent Polytechnic, Internal report.

16. A BARGIELA and G D HAINSWORTH, "Telemetry system design and on-line decision support with 'TCLAS' software", presented at the international conference "Computer applications for water supply and distribution", Leicester, 1987.

17. A BARGIELA and G D HAINSWORTH, "Confidence limit analysis in water systems.", presented at the international Conference on "Computer applications for water supply and distribution", Leicester, Sept 1987.

18. G BELFORTE and B BONA, "An improved parameter identification algorithm for signals with unknown-but-bounded errors", Conf Proc of IFAC/IFORS Symposium on Identification and System Parameter Estimation, York, July 1985.

19. R D BERRY, "Optimal ordering of electronic circuit equations for sparse matrix solution", IEEE Trans Circuit Theory, Vol CT-18, pp 139-145, 1971.

20. D BERTSEKAS and I RHODES, "Recursive state estimation for a set-membership description of uncertainty", IEEE Trans AC, Vol AC-16, No 2, April 1971.

21. D I BLOCKLEY, "Uncertainty analysis in expert systems", Civ Eng Systems, Vol 4, March 1987.

22. B BORKOWSKA, "Probabilistic load flow", IEEE Trans PAS, Vol PAS-93, No 3, May-June 1974.

23. V BRYANT and H PERFECT, "**Independence theory in Combinatorics**", Chapman and Hall, 1980.

24. W H CASOLA, O J PERALA and D W FARRELL, "Yakima supervisory real-time monitoring and control system", Proc ASCE speciality conference "Computer applications in water resources", Buffalo, New York, 1985.

25. A L CESARIO and J O DAVIS, "Calibrating water system models", Journal of AWWA, Vol 76, No 2, July 1984.

26. M CHANDRASHEKAR, "Extended set of components in pipe networks", Journal of the Hydraulics Division, ASCE, Vol 106, No HY1, Jan 1980.

27. M CHANDRASHEKAR and K H STEWART, "Sparsity orientated analysis of large pipe networks", ASCE, Journal of the Structural Division, Vol 101, HY4, April 1974.

28. V CHVATAL, "**Linear programming**", Freeman, 1983.

29. D A CLARKE, E A McBEAN and S A AL-NASSRI, "Uncertainties in water distribution systems", Journal of the Hydraulics Division, ASCE, Vol 107, No HY10, Oct 1981.

30. M A COLLINS, "Pitfalls in pipe network analysis techniques", Transportation Engineering Journal, ASCE, Vol 106, No TE5, Sept 1980.

31. M COLLINS, L COOPER, R HELGASON, J KENNINGTON and LEBLANC, Solving the pipe network analysis problem using optimisation techniques", Management Science, Vol 24, March 1978.

32. G O COSGRIFF, P E FORTE, M A KENNEDY, J V RUSSELL, R D SMITH and A K

WEST, "Interactive computer modeling, monitoring and control of Melbourne's water supply system", Water Resources Research, Vol 21, No 2, Feb 1985.

33. B COULBECK, "Optimisation and modelling techniques in dynamic control of water distribution systems", PhD Thesis, University of Sheffield, July 1977.

34. B COULBECK, "A computer program for calibration of water distribution systems", Journal IWES, Vol 38, No 1, 1984.

35. B COULBECK, "Optimisation of water networks", Trans Int MC, Vol 6, No 5, Oct 1984.

36. B COULBECK and M J H STERLING, "Optimised control of water distribution systems", IEE Proc, Vol 125, No 9, Oct 1978.

37. J D CREASEY, "Pump scheduling in water supply: More than a mathematical problem", presented at the international conference on "Computer applications in water supply and distribution", Leicester, Sept 1987.

38. H CROSS, "Analysis of flow in networks of conduits or conductors", Bulletin No 286, Univ of Illinois Engr Experiment Station, 1934.

39. A B CUNNINGHAM and J R AMEND, "Water management using interactive simulation, Journal of Water Resources Planning and Management, ASCE, Vol 110, No 3, July 1984.

40. A B CUNNINGHAM and J R AMEND, "Interactive simulation of water resource systems", Journal of Water Resources Planning and Management, ASCE, Vol 112, No 3, July 1986.

41. E CUTHILL and J MCKEE, "Reducing the bandwidth of sparse symmetric matrices", Proc 24th national conf ACM, 1969.

42. A O DEMUREN and F J K IDERIAH, "Pipe network analysis by partial pivoting method", ASCE, Journal of Hydraulic Engineering, Vol 112, No 5, May 1986.

43. R P DONACHIE, "Digital program for water network analysis", Journal of the Hydraulic Division, ASCE, Vol 100, No HY3, March 1974.

44. J F DOPAZO, O A KILTIN and A M SASSON, "Stochastic load flows", IEEE Trans PAS, Vol PAS-94, No 2, March/April 1975.

45. I S DUFF, "MA28 - A set of Fortran subroutines for sparse unsymmetric linear equations", Report R 8730, AERE Harwell, Nov 1980.

46. I S DUFF, "On permutations to block triangular form", J Inst Maths Applics, Vol 19, pp339-342, 1979.

47. I S DUFF, "On algorithms for obtaining a maximal transversal", Report CSS 49, AERE Harwell, Oct 1978

48. I S DUFF, " A survey of sparse matrix research", Proc IEEE, Vol 65, No 4, April 1977.

49. I S DUFF and J K REID, "A comparison of sparsity orderings for obtaining a pivotal sequence in Gaussian elimination", J Inst Math Appl, Vol 14, pp 281-291, 1974.

50. I S DUFF and J K REID, "Some design features of a sparse matrix code", ACM Trans on Math Software, Vol 5, No 1, March 1979.

51. I S DUFF and J K REID, "An implementation of Tarjan's algorithm for the block triangularisation of a matrix", ACM Trans on Math Software, Vol 4, No 2, June 1978. Trans PAS, Vol PAS-94, No 2, March 1975.

52. J ECKHARDT, "The Windy Gap project SCADA system", Journal of Water Resources Planning and Management, ASCE, Vol 112, No 3, July 1986.

53. H EDELMANN, "A universal assessment for the superior quality of distribution of measuring points for the state estimation if high-voltage networks", Proc Conf PSCC, Cambridge, 1975.

54. C L EGGENER and L B POLKOWSKI, "Network models and the impact of modeling assumptions", Journal of AWWA, Vol 68, No 4, April 1976.

55. A EL-BAHRAWY and A A SMITH, "Application of MINOS to water collection and distribution networks", Civ Eng Systems, Vol 2, March 1985.

56. F FALLSIDE and P F PERRY, "Hierarchical optimisation of a water supply network", IEE Proc, Vol 122, No 2, Feb 1975.

57. E E FETZER and P M ANDERSON, "Observability in the state estimation of power systems", IEEE Trans PAS, Vol PAS-94, No 6, Nov/Dec 1975.

58. D T FINKBEINER, "**Linear Algebra**", Freeman, 1972

59. E FOGEL and Y F HUANG, "On the value of information system identification - bounded noise case", Automatica, Vol 18, No 2, 1982.

60. R L FRANCIS, L F McGINNIS and J A WHITE, "Locational analysis", European Journal of Operational Research, Vol. 12, pp 220-252, 1983

61. C R GAGNON and L S JACOBY, "Computer simulation of water distribution networks", Transportation Engineering Journal, ASCE, Vol 101, No TE3, Aug 1975.

62. J A GEORGE and J W H LIU, "An automatic partitioning and solution scheme for solving large sparse positive definite systems of linear algebraic equations", Dept Computer Science, Univ Waterloo, Ontario, Canada, CS-75-17, 1975.

63. A GIBBONS, "**Algorithmic graph theory**", CUP, 1985.

64. N E GIBBS, W G POOLE and P K STOCKMEYER, "An algorithm for reducing the bandwidth and profile of a sparse matrix", SIAM J Numer Anal, Vol 13, pp236-250, 1976.

65. P E GILL, W MURRAY and M H WRIGHT, "**Practical optimization**", Academic Press, 1981.

66. E GOFMAN and M RODEH, "Loop equations with unknown pipe characteristics", Journal of the Hydraulics Division, ASCE, Vol 107, No HY9, Sept 1981.

67. R S GOOCH and A L GRAVES, "Central Arizona project supervisory control system",

Journal of Water Resources Planning and Management, ASCE, Vol 112, No 3, July 1986.

68. G D HAINSWORTH and A BARGIELA, "Optimal telemetry system design for water networks", Proc Int Symp on Computer Modelling of Water Distribution Systems, Kentucky, May 1988.

69. P HALL, "On representatives of subsets", Journal of the London Mathematical Society, Vol 10, No 37, Pt 1, 1935.

70. D HAMBERG and U SHAMIR, "Schematic models for distribution systems design. I: Combination concept", Journal of Water Resources Planning and Management, ASCE, Vol 114, No 2, March 1988.

71. D HAMBERG and U SHAMIR, "Schematic models for distribution systems design. II: Continuum approach", Journal of Water Resources Planning and Management, ASCE, Vol 114, No 2, March 1988.

72. F HARARY, "Sparse matrices and graph theory", in "**Large sparse sets of linear equations**", Ed J K REID, Academic Press, 1971.

73. L HASKINS and D J ROSE, "Toward characterisation of perfect elimination digraphs", SIAM J Comput, Vol 2, No 4, Dec 1973

74. A T F HAYWARD, "**Flowmeters**", Macmillan, 1979.

75. J E HOPCROFT and R M KARP, "An $n^{5/2}$ algorithm for maximum matchings in a bipartite graph", SIAM J Comput, Vol 2, pp225-231, 1973.

76. R A HORN and C A JOHNSON, "**Matrix analysis**", CUP, 1985.

77. M R IRVING and M J H STERLING, "Implementation of state estimation and data validation in electric power systems", in "**Comparative models for electric load forecasting**", Ed D W BUNN and E D FARMER, John Wiley, 1985.

78. L T ISAACS and K G MILLS, "Linear theory methods for pipe network analysis", Journal of the Hydraulics Division, ASCE, Vol 106, No HY7, July 1980.

79. L S JENNINGS and M R OSBORNE, "A direct error analysis for least squares", Numer Math, Vol 22, pp 325-332, 1974.

80. A JENNINGS and A D TUFF, "A direct method for the solution of large sparse symmetric simultaneous equations", in "**Large sparse sets of linear equations**", Ed J K REID, Academic Press, 1971.

81. L E JOHNSON, "Water resource management decision support systems", Journal of Water Resources Planning and Management, ASCE, Vol 112, No 3, July 1986.

82. P W JOWITT, R T GARRETT, S C COOK and G GERMANOPOULOS, "Real time forecasting and control for water distribution", presented at the international conference on "Computer applications in water supply and distribution", Leicester, Sept 1987.

83. R M KARP and A WIGDERSON, "A fast parallel algorithm for the maximal independent set problem", Journal ACM, Vol 32, No 4, Oct 1985.

84. H K KESAVAN and M CHANDRASHEKAR, "Graph-theoretic models for pipe network analysis", Journal of the Hydraulics division, ASCE, Vol 98, No HY2, Feb 1972.

85. I P KING, "An automatic reordering scheme for simultaneous equations derived from network systems", Int Journal of Numerical Methods in Engineering, Vol 2, pp 523-533, 1970.

86. K KINSNER, A SERWIN and M SOBIERAJSKI, "Practical aspects of load flow calculations", Archiv fur Elektrotechnik, Vol 60, 1978.

87. H-J KOGLIN, "Optimal measuring system for state estimation", Proc of PSCC Conf, Cambridge, 1975.

88. G R KRUMPHOLZ, K A CLEMENTS and P W DAVIS, "Power system observability, a practical algorithm using network topology", IEEE Trans PAS, Vol PAS-99, No 4, July/Aug 1980.

89. J W LABADIE, "Real-time control of water resource systems", Proc ASCE speciality conference "Computer applications in water resources", Buffalo, New York, 1985.

90. J W LABADIE and C H SULLIVAN, "Computerized decision support systems for water managers", Journal of Water Resources Planning and Management, ASCE, Vol 112, No 3, July 1986.

91. K E LANSEY, "A procedure for water distribution network calibration considering multiple loading conditions", Proc Int Symp on Computer Modelling of Water Distribution Systems, Kentucky, May 1988.

92. C F LAM and M L WOLLA, "Computer analysis of water distribution systems: Part I - Formulation of equations", Journal of the Hydraulics division, ASCE, Vol 98, No HY2, Feb 72.

93. C F LAM and M L WOLLA, "Computer analysis of water distribution systems: Part II - Numerical solution", Journal of the Hydraulic Division, ASCE, ASCE, Vol 98, No HY3, March 1972.

94. J H LEE, "Case study of a water distribution system by computer modelling", Proc Int Symp on Computer Modelling of Water Distribution Systems, Kentucky, May 1988.

95. A M LEITE DA SILVA, V L ARIENTI and R N ALLAN, "Probabilistic load flow considering dependence between input nodal powers", IEEE Trans PAS, Vol PAS-103, No 6, June 1984.

96. J W H LIU, "Modification of the minimum-degree algorithm by multiple elimination", ACM Trans on Math Software, Vol 11, No 2, June 1985.

97. H M MARKOWITZ, "The elimination form of the inverse and its application to linear programming", Management Science, Vol 3, pp255-269, 1957

98. M MILANESE and G BELFORTE, "Estimation theory and uncertainty intervals evaluation in the presence of unknown but bounded errors: linear families of models and estimators", IEEE trans-AC, Vol AC-27, No 2, April 1982.

99. S MIYAOKA and M FUNABASHI, "Optimal control of water distribution systems by network flow theory", IEEE Trans AC, Vol AC-29, No 4, April 1984.

100 S H MO and J P NORTON, "Parameter bounding identification algorithms for bounded-noise records", Proc IEE, Vol 135, Pt D, No 2, March 1988.

101. A O MORRIS, "**Linear Algebra**", VNR, 1978.

102. J P NORTON, "**An introduction to identification**", Academic Press, 1986.

103. J P NORTON, "Identification and application of bounded-parameter models", Automatica, Vol 23, No 4, 1987.

104. J P NORTON, "Identification of parameter bounds for ARMAX models from records with bounded noise", Int J Control, Vol 45, No 2, 1987.

105. E C OGBUOBIRI, W F TINNEY and J W WALKER, "Sparsity-directed decomposition for Gaussian elimination on matrices", IEEE Trans PAS, Vol PAS-89, No 1, Jan 1970.

106. T OHTSUKI, "A fast algorithm for finding an optimal ordering for vertex elimination on a graph", SIAM J Comput, Vol 5, No 1, March 1976.

107. K ONIZUKA, "System dynamics approach to pipe network analysis", Journal of Hydraulic Engineering, ASCE, Vol 112, No 8, Aug 1986.

108. L E ORMSBEE and D V CHASE, "Hydraulic network calibration using non-linear programming", Proc Int Symp on Computer Modelling of Water Distribution Systems, Kentucky, May 1988.

109. L E ORMSBEE and D J WOOD, "Explicit pipe network calibration", Journal of Water Resources Planning and Management, ASCE, Vol 112, No 2, April 1986

110. C-H ORR and B COULBECK, "Water supply application programs in an operations environment", presented at the international conference on "Computer applications for water supply and distribution", Leicester, Sept 1987.

111. H M ORTH, "**Model-based design of water distribution and sewage systems**", John Wiley, 1986.

112. D F J OSBORN, "Distribution network analysis: NETMAP - a graphical approach", Journal IWES, Vol 38, No 4, Aug 1984.

113. K PHUA and T S DILLON, "Optimal choice of measurements for state estimation", PICA Conference, Toronto, 1977.

114. S PISSANETSKY, "**Sparse matrix technology**", Academic Press, 1984

115. J QUEVEDO and G CEAMBRANO, "Time series modelling of water demand in large networks. A study on short-term and long-term predictions", presented at the international Conference on "Computer applications for water supply and distribution", Leicester, 1987

116. H S RAO and D W BREE JR, "Extended period simulation of water systems - Part A", Journal of the Hydraulics Division, ASCE, Vol 103, No HY2, Feb 1977.

117. H S RAO, L C MARKEL and D W BREE JR, "Extended period simulation of water systems - Part B", Journal of the Hydraulics Division, ASCE, Vol 103, No HY3, March 1977.

118. J K REID, "A survey of sparse matrix computation", Electric Power Problems, The Mathematical Challenge, SIAM 1980.

119. D J ROSE, "A graph-theoretic study of the numerical solution of sparse positive definite systems of linear equations", in "**Graph theory and computing**", Ed R C READ, Academic Press, 1972.

120. B ROY, "An algorithm for a general constrained set covering problem", in "**Graph theory and computing**", Ed. R C READ, Academic Press, 1972.

121. R SALGADO, E TODINI and P E O'CONNELL, "Extending the gradient method to include pressure regulating valves in pipe networks", Proc Int Symp on Computer Modelling of Water Distribution Systems, Kentucky, May 1988.

122. R W H SARGENT and A W WESTERBERG, " 'Speed-up' in chemical engineering design", Trans Inst Chem Eng, Vol 42, pp 190-197, 1964.

123. A SCHRIJVER, "**Theory of linear and integer programming**", John Wiley, 1986.

124. F C SCHWEPPE, "Recursive state estimation: Unknown but bounded errors and system inputs", IEEE Trans AC, Vol AC-13, No 1, Feb 1968.

125. F C SCHWEPPE and J WILDES, "Power system static state estimation, Part I: exact model", IEEE Trans PAS, Vol PAS-89, No 1, Jan 1970.

126. U SHAMIR, "Computer applications for real-time operation of water distribution systems", Proc ASCE speciality conference "Computer applications in water resources", Buffalo, New York, 1985

127. U SHAMIR and D D HOWARD, "Water distribution system analysis", Journal of the Hydraulics Division, ASCE, Vol 94, No HY1, Jan 1968.

128. U SHAMIR and D D HOWARD, "Engineering analysis of water distribution systems", Journal of AWWA, Vol 69, No 9, Sept 1977.

129. M SOBIERAJSKI, "Optimal stochastic load flows", Electrical Power Systems Research, Vol 2, 1979.

130. M J H STERLING and A BARGIELA, "Minimum norm state estimation for computer control of water distribution systems", Proc IEE, Vol 131, Pt D, No 2, March 1984.

131. M J H STERLING and A BARGIELA, "Adaptive forecasting of daily water demand", in "**Comparative models for electrical load forecasting**", Ed D W BUNN and E D FARMER, John Wiley, 1985

132. M J H STERLING and A BARGIELA, "Leakage reduction by optimised control of valves in water networks", Trans Int MC, Vol 6, No 6, Oct-Dec 1984.

133. K R STIMSON, "Analysis and simulation of water networks. A guide to the Watnet simulation computer package", WRc report Engineering TR 237, March 1986.

134. K R STIMSON and A BRAMELLER, "An integrated mesh-nodal method for steady-state water distribution network analysis", Journal IWES, Vol 35, No 2, March 1981.

135. K R STIMSON and A BRAMELLER, "Water distribution network analysis using interactive graphics", Journal IWES, Vol 36, No 2, April 1982.

136. T A STUART and C J HERGET, "A sensitivity analysis of weighted least squares state estimation for power systems", IEEE PES Winter meeting, New York, Jan-Feb 1973.

137. R J SUTTER, R D CARLSON and D LUTE, "Data automation for water supply management", Journal of Water Resources Planning and Management, ASCE, Vol 109, No 3, July 1983.

138. R TARJAN, "Depth first search and linear graph algorithms", SIAM J Comput, Vol 1, pp146-160, 1972.

139. R P TEWARSON, "Sorting and ordering sparse linear systems", in "**Large sparse sets of linear equations**", Ed J K REID, Academic Press, 1971

140. R P TEWARSON, "**Sparse matrices**", Academic Press, 1973

141. R P TEWARSON, "Row column permutation of sparse matrices", Comput J, Vol 10, pp 300-305, 1967

142. A R D THORLEY and D J WOOD, "Pipe network analysis, steady flow and extended period simulations with SIMNET", The City University, Thermo-fluids engineering research centre, Nov 1986.

143. T M WALSKI, "Technique for calibrating network models", Journal of Water Resources Planning and Management, ASCE, Vol 109, No 4, Oct 1983.

144. T M WALSKI, "**Analysis of water distribution systems**", VNR New York, 1984.

145. T M WALSKI, "Assuring accurate model calibration", Journal of AWWA, Vol 77, No 12, Dec 1985.

146. T M WALSKI, "Case study: Pipe network model calibration issues", Journal of Water Resources Planning and Management, ASCE, Vol 112, No 2, April 1986.

147. H J WILLIAMS, "The application of telemetry systems in the Hong Kong water supplies department", Journal of the Hong Kong Inst of Engineers, Vol 13, No 10, Oct 1985.

148. D J WOOD, "Computer analysis of flow in pipe networks including extended period simulations", University of Kentucky, Lexington, Kentucky, 1980

149. D J WOOD and C O A CHARLES, "Hydraulic network analysis using linear theory", Journal of the Hydraulics Division, ASCE, Vol 98, No HY7, July 1972.

150. D J WOOD and A G RAYES, "Reliability of algorithms for pipe network analysis", Journal of the Hydraulics Division, ASCE, Vol 107, No HY10, Oct 1981.

151. M S ZARGHAMEE, "Mathematical model for water distribution systems", Journal of the Hydraulics Division, ASCE, Vol 97, No HY1, Jan 1971.

152. K ZOLLENKOPF, "Bi-factorisation - Basic computational algorithm and programming techniques", in "**Large sparse sets of linear equations**", Ed J K REID, Academic Press, 1971.