AN INTEGRATED NEURAL BASED SYSTEM FOR STATE ESTIMATION AND CONFIDENCE LIMIT ANALYSIS IN WATER NETWORKS

B. Gabrys and A. Bargiela
Real Time Telemetry Systems
Department of Computing, The Nottingham Trent University
Burton Street, Nottingham NG1 4BU
Tel: +44 (0) 115 9418418 ext 2162, Fax: +44 (0) 115 9486518
E-mail: bcz@doc.ntu.ac.uk, andre@doc.ntu.ac.uk,

ABSTRACT

In this paper a simple recurrent neural network (NN) is used as a basis for constructing an integrated system capable of finding the state estimates with corresponding confidence limits for water distribution systems. In the first phase of calculations a neural linear equations solver is combined with a Newton-Raphson iterations to find a solution to an overdetermined set of nonlinear equations describing water networks.

The mathematical model of the water system is derived using measurements and pseudomeasurements consisting certain amount of uncertainty. This uncertainty has an impact on the accuracy to which the state estimates can be calculated. The second phase of calculations, using the same NN, is carried out in order to quantify the effect of measurement uncertainty on accuracy of the derived state estimates. Rather than a single deterministic state estimate, the set of all feasible states corresponding to a given level of measurement uncertainty is calculated. The 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.

The simulations have been carried out and results are presented for a realistic 34-node water distribution network.

INTRODUCTION

In the monitoring of water distribution systems, the inaccuracy of input data contributes greatly to the inaccuracy of system state estimates calculated from them. It is important, therefore, that the system operators are given not only the values of flows and pressures in the network at any instant of time but also that they have some indication of how reliable these values are. The quantification of the inaccuracy of calculated state estimates caused by the input data uncertainty is called confidence limit analysis (Bargiela and Hainsworth 1988).

Some applications, semi-automated or on-line decision support for instance, need a confidence limit analysis procedure that can produce uncertainty bounds in real time. This is, however, the task requiring much of computational effort.

Artificial Neural Networks (ANNs), considering their known properties like massively parallel structure, fault tolerance etc., are seen as a means of overcoming the computational complexity.

In the previous paper (Gabrys and Bargiela 1995) we presented the neural network based technique for the solution of a water system state estimation problem. Since the state estimation forms the basis of the confidence limit analysis, in the next two section we briefly report the estimation method and its neural network implementation.

In the following section the confidence limit analysis using the same neural network is introduced. Next the implementation of the integrated system for estimation and CLA is described. This is followed by results of computer simulations for the realistic 34-node water network. Both the state estimates and associated with them boundaries resulting from the confidence limit analysis are presented. Discussion of the results is also given in this section. And finally conclusions are reported.

WATER SYSTEM MODEL AND ESTIMATION METHOD

The state estimation process is based on a mathematical network model of the water distribution system. The physical laws governing the system can be combined with the hydraulic relationship of each element of the system to construct a set of network equations. These nonlinear network equations relate either, the network's nodal pressures or the network's flows to measurement or pseudumeasurement values and are expressed by the following equation:

$$\mathbf{z}=g(\mathbf{x})+\omega \quad (1)$$

where $\mathbf{z}$ is a measurement vector; $g(\mathbf{x})$ are nonlinear functions describing system; ω is a vector of measurement inconsistency.

The state estimation can be expressed as a problem of minimization of discrepancies between the actual measurements and the values calculated from the mathematical model.

Using the least squares criterion the state estimation problem can be expressed as:

$$\min_{\mathbf{x}} \; E_2(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{z}-\boldsymbol{g}(\boldsymbol{x}))^T \boldsymbol{W}(\boldsymbol{z}-\boldsymbol{g}(\boldsymbol{x})) \quad (2)$$

where: $\boldsymbol{W} = diag[w_1, w_2, \ldots, w_m]$ is a measurement weight matrix.

The proposed solution of the state estimation problem (2) is based on the Newton-Raphson method. Expanding $\mathbf{g}(\mathbf{x})$ by an initial guess of the state vector $\boldsymbol{x}^{(0)}$, using a first-order Taylor series and defining $\boldsymbol{z}^{(0)} = \boldsymbol{g}\left(\boldsymbol{x}^{(0)}\right)$, we obtain

$$\boldsymbol{z} = \boldsymbol{z}^{(0)} + \Delta\boldsymbol{z} \quad (3)$$

$$\boldsymbol{g}(\boldsymbol{x}) = \boldsymbol{g}\left(\boldsymbol{x}^{(0)}\right) + \boldsymbol{J}^{(0)}\Delta\boldsymbol{x} \quad (4)$$

After this linearisation we obtain the following set of equations:

$$\boldsymbol{J}^{(k)}\Delta\boldsymbol{x} = \boldsymbol{z} - \boldsymbol{g}\left(\boldsymbol{x}^{(k)}\right) \quad (5)$$

where:

$\boldsymbol{J}^{(k)} = \left.\frac{\partial g}{\partial \boldsymbol{x}}\right|_{\boldsymbol{x}=\boldsymbol{x}^{(k)}} \in \boldsymbol{R}^{mxn}$ - Jacobian matrix evaluated at $\boldsymbol{x}^{(k)}$

k=0,1,... - step of the estimation process

Equations (2) can be therefore expressed as

$$\min_{\Delta x} \; E_2(\Delta \boldsymbol{x}) = \frac{1}{2}\left(\Delta \boldsymbol{z} - \boldsymbol{J}^{(k)}\Delta \boldsymbol{x}\right)^T \boldsymbol{W}\left(\Delta \boldsymbol{z} - \boldsymbol{J}^{(k)}\Delta \boldsymbol{x}\right) \qquad (6)$$

The overdetermined set of linear equations (5) form the basis for the construction of a neural network which is presented in the following section.

Since the measurement equations (1) are nonlinear, the solution to (2) is an iterative process with the consecutive state estimates calculated by under-relaxation of the linear solution

$$\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \gamma \Delta \boldsymbol{x}^{(k)}, \text{ k=0,1,...} \qquad (7)$$

If all elements of $\Delta \boldsymbol{x}$ in k-th iteration are lower or equal to a predefined convergence accuracy, the iteration procedure stops. Otherwise, a new correction vector is calculated using equation (5) with $\boldsymbol{x}^{(k+1)}$ instead of $\boldsymbol{x}^{(k)}$ and suitable neural network.

NEURAL NETWORK SOLVING SYSTEM OF LINEAR EQUATIONS

The minimisation problems described by (6) can be generalised as follows:

$$\min_{\Delta x} \; E(\Delta \boldsymbol{x}) = \sum_{i=1}^{m} \sigma_i [r_i(\Delta \boldsymbol{x})] \qquad (8)$$

where: E is a general cost (energy) function; $\boldsymbol{A} = \boldsymbol{J}^{(k)}$, $\boldsymbol{b} = \boldsymbol{z} - \boldsymbol{g}\left(\boldsymbol{x}^{(k)}\right)$; $r_i(\Delta \boldsymbol{x}) = \boldsymbol{a}_i^T \Delta \boldsymbol{x} - b_i$ is the i-th residual; $\sigma_i[r_i]$ represents a suitably chosen convex functions.

In a special case when $\sigma_i(r_i) = r_i^2/2$ we obtain the standard least-squares criterion (6). More information on other criterions (least absolute values or Chebyshev criterions) and their neural network applications can be found in (Cichocki and Unbehauen 1992a, 1992b; Gabrys and Bargiela 1995; Cichocki and Bargiela 1996).

The minimization of the energy function described by eq. (8) by a standard gradient descent method leads to the following system of nonlinear differential equations:

$$\frac{d\Delta x_j}{dt} = -\mu_j \sum_{i=1}^{m} a_{ij}\left(f_i\left(\sum_{k=1}^{n} a_{ik}\Delta x_k - b_i\right)\right) \qquad (9)$$

or in compact matrix form

$$\frac{d\Delta \boldsymbol{x}}{dt} = -\mu \boldsymbol{A}^T f(\boldsymbol{r}(\Delta \boldsymbol{x})) \qquad (10)$$

where: $\mu_j > 0$ is the learning parameter; $f_i(r_i(\Delta \boldsymbol{x})) = \frac{\partial \sigma_i(r_i)}{\partial r_i}$ is the activation function.

For $\sigma_i(r_i) = r_i^2/2$ we have $f_i(r_i(\Delta \boldsymbol{x})) = r_i(\Delta \boldsymbol{x})$ - linear activation function.

The system of differential equations (10) has been implemented as an artificial neural network (ANN) shown in Figure 2, using the SIMULINK software.

CONFIDENCE LIMIT ANALYSIS (CLA)

Confidence limit analysis is a process of calculating uncertainty bounds for state estimates which are caused by the inaccuracies of input data, and in particular, inaccuracies of pseudomeasurements. The algorithm used here is based on a linear approximation of the water network model and it has been first introduced in (Bargiela and Hainsworth 1988).

Let $\boldsymbol{b}^{opt}$ represent the difference $\boldsymbol{z} - \boldsymbol{g}(\hat{\boldsymbol{x}})$ and $\boldsymbol{J}^{opt} = \left.\frac{\partial g}{\partial \boldsymbol{x}}\right|_{x = \hat{x}}$. If $\boldsymbol{x}^{(k)}$ in equation (5) is replaced by the true state vector, $\hat{\boldsymbol{x}}$, for the system, $\boldsymbol{b}^{opt}$ will then represent the difference between the measured vector and the true values of the measured variables.

With the symbols introduced above the linear equation (5) takes form:

$$\boldsymbol{J}^{opt}\Delta \boldsymbol{x} = \boldsymbol{b}^{opt} \qquad (11)$$

So it is of interest to see how the solution is affected by perturbation in vector of measurements $\boldsymbol{b}^{opt}$ (effectively in vector **z** since $\boldsymbol{g}(\hat{\boldsymbol{x}})$ is constant for a fixed state vector).

If we introduce the vector of perturbations δb into (11) we have

$$\boldsymbol{J}^{opt}(\Delta \boldsymbol{x} + \Delta \boldsymbol{x}_p) = \boldsymbol{b}^{opt} + \delta \boldsymbol{b} \qquad (12)$$

Utilizing the fact that the set of equations (12) is linear and putting (11) into (12) we obtain the following:

$$\boldsymbol{J}^{opt}\Delta \boldsymbol{x}_p = \delta \boldsymbol{b} \qquad (13)$$

where: δb - vector of perturbations; $\Delta \boldsymbol{x}_p$ - vector of changes in state estimates caused by perturbations δb.

Vector δb in (13) consists of values that are within the range: $\pm$variability of consumption or $\pm$accuracy of meter in case of a measurement. The underlying principle of the CLA is the consideration of the worst possible case. It means that the maximum variabilities of consumptions and inaccuracies of meters are assumed during the calculations.

The proposed method of finding the confidence limits utilizes the neural network presented in Figure 2.

Making the vector δb of the equation $\boldsymbol{J}^{opt}\Delta \boldsymbol{x}_{cl}^{(i)} = \Delta \boldsymbol{b}^{(i)}$ successively $\left[\Delta b_1 \; 0 \; \dots \; 0\right]^T$, $\left[0 \; \Delta b_2 \; \dots \; 0\right]^T$,..., $\left[0 \; 0 \; \dots \; \Delta b_m\right]^T$ the confidence vector $\boldsymbol{x}_{cl}$ is found by summing up the absolute values of $\Delta \boldsymbol{x}_{cl}^{(i)}$ for each source of inaccuracy Δb_i, i=1,...,m.

$$\boldsymbol{x}_{cl} = \sum_{i=1}^{m} \left|\Delta \boldsymbol{x}_{cl}^{(i)}\right| \qquad (14)$$

THE INTEGRATED SYSTEM FOR ESTIMATION AND CLA

The system depicted in Figure 1 is an implementation of a Newton-Raphson method for solving systems of nonlinear equations. The description of this method for water distribution network has been given in previous sections.

Apart from the state estimation achieved by means of implementing N-R method the system also incorporates the necessary logic and other elements (integrators, absolute value block etc.) to obtain confidence limits for the state estimates.

The functionality of this system can be summarised as follows:

1) For the initial guess $\boldsymbol{x}^{(0)}$ calculate the Jacobian $\boldsymbol{J}^{(0)}$ and the right hand side of the linearised system of equations $\boldsymbol{z}-\boldsymbol{g}\left(\boldsymbol{x}^{(0)}\right)$.

2) Using the analog neural network for solving systems of linear equations ("Neural Estimator") solve $\boldsymbol{J}^{(0)}\Delta\boldsymbol{x}^{(0)} = \boldsymbol{z}-\boldsymbol{g}\left(\boldsymbol{x}^{(0)}\right)$.

3) If all elements of $\Delta\boldsymbol{x}^{(k)}$ (k=0,1,2,... - number of iteration) are lower or equal to predefined accuracy, go to point 7, otherwise go to point 4.

4) Adjust the current state estimate values according to formula: $\boldsymbol{x}^{(k+1)} = \boldsymbol{x}^{(k)} + \gamma\Delta\boldsymbol{x}^{(k)}$.

5) For $\boldsymbol{x}^{(k+1)}$ calculate the Jacobian $\boldsymbol{J}^{(k+1)}$ and the right hand side of the linearised system of equations $\boldsymbol{z}-\boldsymbol{g}\left(\boldsymbol{x}^{(k+1)}\right)$.

6) Using neural based state estimator find the solution to the system of linear equations $\boldsymbol{J}^{(k+1)}\Delta\boldsymbol{x}^{(k+1)} = \boldsymbol{z}-\boldsymbol{g}\left(\boldsymbol{x}^{(k+1)}\right)$ and go to point 3.

7) For the Jacobian calculated for the optimal state estimate vector $\hat{\boldsymbol{x}}$, calculate the confidence limits making the right hand side vector $\Delta\boldsymbol{b}$ of equation $\boldsymbol{J}^{opt}\Delta\boldsymbol{x}_{cl}^{(i)} = \Delta\boldsymbol{b}^{(i)}$ successively $\left[\Delta b_1\ 0\ \dots\ 0\right]^T$, $\left[0\ \Delta b_2\ \dots\ 0\right]^T$,..., $\left[0\ 0\ \dots\ \Delta b_m\right]^T$ and summing $\boldsymbol{x}_{cl} = \sum_{i=1}^{m}\left|\Delta\boldsymbol{x}_{cl}^{(i)}\right|$, where $\boldsymbol{x}_{cl}$ is the vector of confidence limits of $\hat{\boldsymbol{x}}$ calculated for the vector of disturbances $\Delta\boldsymbol{b}$ and the state vector $\hat{\boldsymbol{x}}$.

Detailed description of the system and subsystems

There are two output signals from the "Neural Estimator" block. First - LEC (Linear Estimation Control) is the control signal normally set to 0 (zero). It changes from 0 (zero) to 1 (one) (an impulse is generated) every time when the convergence criterion (accuracy condition) of the neural estimator is met. Only then the second signal $\boldsymbol{x}_l$, which is a vector of current estimates of the neural estimator, is allowed to be processed.

Following the signals LEC and $\boldsymbol{x}_l$ we now go to the block "NR Control". The function of this subsystem is to produce control signals enabling us to distinguish between the state estimation an the confidence limit analysis stages of computation.

Four control signals C1, C2, C3 and C4 of the "NR Control" block decide if the signal y=LEC*$\boldsymbol{x}_l$ is directed to: the "Newton-Raphson method integrator" block - the state estimation stage or the "Confidence Limits Integrator" block - the confidence limits finding stage.

Referring to figure 3 we can see that C1 is equal to LEC when C2 is 1 (one) and 0 (zero) otherwise. C2 is 1 (one) as long as the convergence criteria of the Newton-Raphson method is NOT satisfied, namely when there is even one value of the vector y (at the state estimation stage) greater than predefined accuracy. C2 changes to zero when the state estimation process is completed and we go to the confidence limits analysis stage.

C3 is a logical negation of C2. When C3=1 (equivalent of C2=0) the C4 =LEC. At this stage signal y is directed to the "Confidence Limits Integrator" block.

STAGE 1

At the stage 1 (C2=1) after every adjusting of the state vector **x**, the change of the Jacobian and the right hand side of the linearised system (RHSLS) of equations, synchronised by C1, is carried out. These newly calculated values are than set in the "Neural Estimator" block (Jacobian - Matrix A, Jacobian transposed - Matrix AT, RHSLS - Vector b) and the new adjustments of the state vector are computed. It has to be stressed that there must be enough time available between two subsequent impulses of LEC for calculating Jacobian, RHSLS and setting these parameters in the "Neural Estimator" block. If the time of carrying out those operations is not known or can not be determined the easy solution could be an introduction of a new control signal. The function of

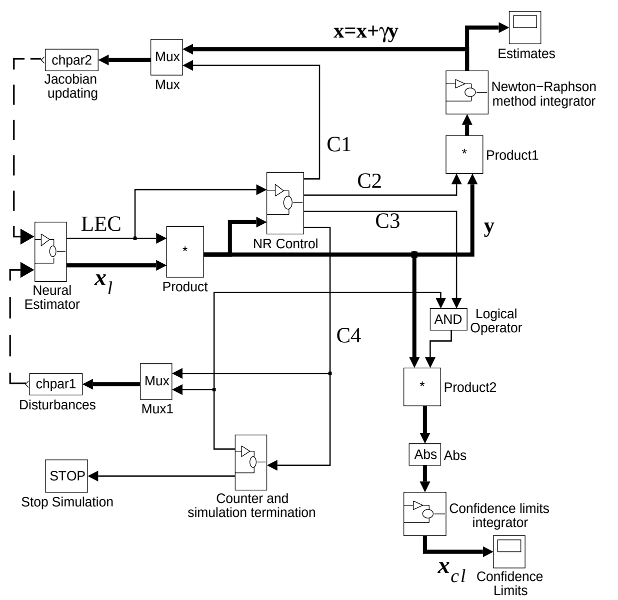

Figure 1: The system for estimation (based on Newton-Raphson method) and confidence limit analysis.

this signal would be to make sure that after generating the first LEC impulse there would not be generated the next LEC impulse before the newly calculated Jacobian and RHSLS were not set up in the "Neural Estimator" subsystem.

STAGE 2

Since the adapted method of the confidence limits finding is based on sequential solving of a system of linear equations, we can use the "Neural Estimator" to achieve it. As it has been described previously the matrix $\boldsymbol{A} = \boldsymbol{J}^{opt}$ of the system of equations Ax=b (at this stage) does not change. It is the Jacobian calculated for the optimal state estimate vector $\hat{\boldsymbol{x}}$. Making vector b successively $\begin{bmatrix}\Delta b_1 & 0 & \dots & 0\end{bmatrix}^T$, $\begin{bmatrix}0 & \Delta b_2 & \dots & 0\end{bmatrix}^T$,..., $\begin{bmatrix}0 & 0 & \dots & \Delta b_m\end{bmatrix}^T$ (synchronised by the signal C4) and summing the influences of each disturbance on state vector ("Confidence limit integrator" block), we finally get the confidence limits for given vector of disturbances $\Delta\mathbf{b}$ and calculated vector of state estimates $\hat{\boldsymbol{x}}$.

The simulation terminates when the influence of the last disturbance on the state vector has been adjusted to the vector of confidence limits.

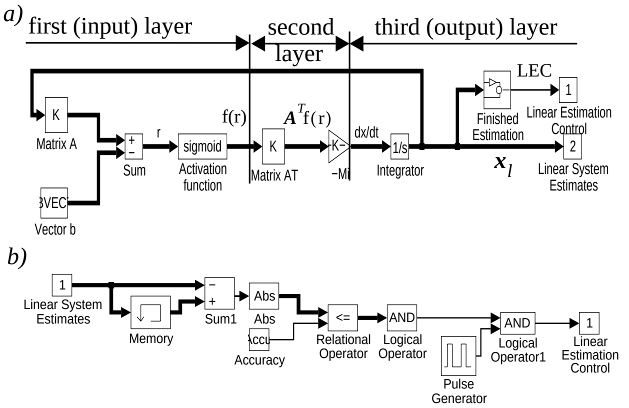

Figure 2: a) Neural network for solving systems of linear equations - Subsystem of the system from Figure 1, b) The "Finished Estimation" subsystem of a) - generating impulse when the optimal state estimates are found.

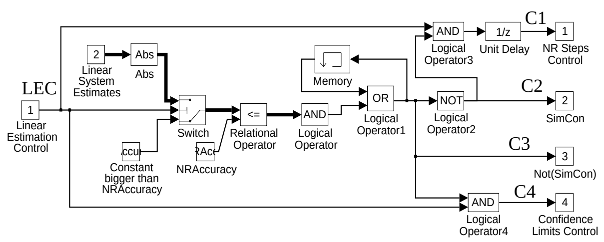

Figure 3: The "NR Control" subsystem of the system from Figure 1

SIMULATION RESULTS

The performance of the proposed methods for water-system state estimation and CLA was tested on a realistic 34-node network (42 state variables). A complete definition of the network parameters is contained in (Sterling and Bargiela 1984).

The results are presented in two forms:

a) table containing the state estimates and corresponding confidence limits for three different cases

b) time diagrams - in order to show the relations between control signals and iteratively calculated values of state estimates and confidence limits.

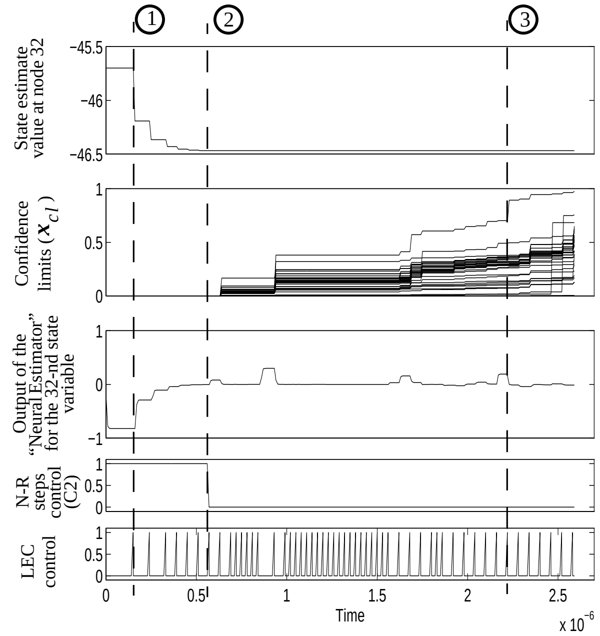

Figure 4: The simulation time diagram for 34-node water network - case 3. Dashed lines mark three phases of simulation: 1) First stage - state estimates adjusting; 2) State estimation finished, beginning of the stage two; 3) Second stage - confidence limits finding

Three different cases are considered below.

First we consider the case of minimal set of measurements when we have available only one reference head measurement at node 30. The state estimates and corresponding confidence limits for this case are shown in columns 3 and 4 of Table 1 respectively. The purpose of calculating the confidence limits is to obtain an information about how far from the real state the estimated values could be in the worst case. The requirement to have the state estimates as close as possible to the real state is equivalent to the requirement of having the confidence limits to be as tight as possible. The means of achieving that is the introduction of additional accurate measurements into the system.

In the highlighted row (columns 3 to 6) of Table 1 we can see the worst (the biggest) confidence limit for the estimated state vector. In order to improve it, in the second case apart from the reference head (node 30) we have also measured the head at node 28. With the accuracy of meter equal 2%, the measured value can vary within the range of ± 0.7024.

The state estimates and corresponding confidence limits for the second case are shown in columns 5 and 6 of Table 1 respectively. Comparing the confidence limits for the 28-th state variable it can be seen that the considerable improvement has been achieved.

Repeating the procedure of finding the biggest confidence limits for the second case two new measurements have been added to the system in the third case. These were the heads in nodes 33 and 34. The state estimates and corresponding confidence limits for the third case are shown in columns 7 and 8 of Table 1 respectively. The two highlighted rows (columns 5 to 8) show the variables of interest. In this case apart from the improvement regarding state variables for nodes 33 and 34 we can observe the big improvement in nodes in their direct vicinity (nodes 1, 26, 29) as well as in nodes laying a bit further (nodes 13, 14,15,16,17,18, and 19). The effect wears off with the increasing distance from the meter.

The simulation time diagram for this case is shown in Figure 4.

In all cases the simulated time of the calculations (the time that would be required by the actual neural network) was in order of microseconds. It needs to be pointed out, however, that the simulation of the neural network was performed on a serial computer (SparcStation IPC) and the corresponding lapsed time for the simulation was of order of hundreds of seconds.

It is commonly known that with the bigger number of measurements the reliability of estimation increases. It is due to averaging property of systems with high redundancy ratio. Redundancy ratio is defined as the ratio of the number of equations to the number of unknowns. In other words the influence of single measurement or pseudomeasurement or strictly speaking its inaccuracy is smaller (averaged) for the system with high redundancy ratio.

On the other hand introducing a new measurement we introduce a new source of inconsistency which is the finite accuracy of a meter. The conclusion that can be drawn from these considerations and results is as follows. Addition of the new measurement for *i*-th state variable can have the tightening effect on the confidence limit of this variable only if the error resulted from the inaccuracy of the meter is smaller than confidence limit calculated for existing set of meters.

CONCLUSIONS

Present day state estimation techniques are very efficient 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. 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.

Presented algorithm shows that using relatively simple neural network solving systems of linear equations could significantly improve the time of obtaining results of state estimation and confidence limit analysis in such complex and nonlinear systems as water distribution networks.

REFERENCES

Bargiela, A. and G.Hainsworth 1988 “Pressure and flow uncertainty in water systems”, *ASCE Journal of Water Resources, Planning and Management*, vol.115, pp.212-29

Cichocki, A. and A.Bargiela 1996 “Neural networks for solving linear inequality systems” to appear in *Int. J. of Parallel Computing*

Cichocki, A. and R.Unbehauen. 1992 “Neural networks for solving systems of linear equations and related problems”, *IEEE Trans. Circuits Syst.*, vol.39 (Feb.), pp.124-138.

Cichocki, A. and R.Unbehauen. 1992. “Neural networks for solving systems of linear equations - Part II: Minimax and absolute value problem”, *IEEE Trans. Circuits Syst.*, vol. 39 (September), pp.619-633.

Gabrys, B. and A.Bargiela. 1995 “Neural simulation of water systems for efficient state estimation”, *Proceedings of Modelling and Simulation Conference ESM’95*, Prague, ISBN1-56555-080-3, pp.775-779

Sterling, M.J.H and A.Bargiela. 1984. “Minimum norm state estimation for computer control of water distribution systems’, *IEE Proceedings*, vol.131 (March), no.2.

State variable	Exact state	State estimates (case 1)	Confidence limits (case 1)	State estimates (case 2)	Confidence limits (case 2)	State estimates (case 3)	Confidence limits (case 3)
1	32.6566	33.2501	1.9336	33.2500	1.9251	32.6906	0.6503
2	43.7173	43.7581	0.3647	43.7843	0.3728	43.7254	0.4234
3	46.0554	46.1320	0.4666	46.1814	0.4754	46.0530	0.4621
4	46.6476	46.7040	0.4468	46.7568	0.4535	46.6187	0.4199
5	43.2138	43.1581	0.2449	43.1734	0.2560	43.1439	0.3001
6	42.9352	42.8410	0.1701	42.8553	0.1768	42.8168	0.2339
7	42.3122	42.1258	0.3563	42.1339	0.3574	42.1239	0.3619
8	42.0405	41.8311	0.3776	41.8418	0.3804	41.8194	0.4069
9	43.7703	43.7536	0.0415	43.7686	0.0469	43.7054	0.1279
10	47.9759	48.0797	0.8288	48.0949	0.8263	47.8126	0.5228
11	44.6783	44.6814	0.1452	44.7043	0.1574	44.6290	0.1930
12	44.0043	44.0056	0.0565	44.0271	0.0686	43.9596	0.1301
13	49.3371	49.5421	1.0835	49.5603	1.0814	49.2088	0.5546
14	49.1419	49.3029	1.0651	49.3169	1.0616	48.9583	0.6168
15	49.0994	49.2563	1.0576	49.2698	1.0541	48.9131	0.6175
16	49.3563	49.5702	1.0824	49.5900	1.0805	49.2422	0.5369
17	47.9938	48.1156	0.8156	48.1294	0.8141	47.8566	0.4793
18	49.3736	49.5980	1.0817	49.6149	1.0804	49.2728	0.5213
19	49.0700	49.2514	0.8981	49.2762	0.9069	48.9854	0.5082
20	46.6476	46.7040	0.4468	46.7566	0.4530	46.6181	0.4182
21	45.6441	45.6825	0.2950	45.7180	0.3069	45.5983	0.2866
22	46.6183	46.6792	0.4385	46.7255	0.4480	46.5794	0.3936
23	48.4099	48.5566	0.7486	48.5869	0.7649	48.3352	0.4772
24	43.1961	43.1332	0.1117	43.1468	0.1178	43.1066	0.1753
25	42.4415	42.2816	0.2864	42.2931	0.2902	42.2661	0.3257
26	32.1117	32.6021	1.7594	32.6035	1.7515	32.0839	0.6300
27	-15.1991	-14.9994	0.9501	-14.9739	0.9568	-15.2686	0.5679
28	-33.4978	-35.1219	3.0669	-33.4718	0.6839	-33.4716	0.6840
29	31.7221	32.1481	1.6298	32.1521	1.6229	31.6796	0.6221
30	43.5819	43.5820	0.0000	43.6007	0.0229	43.5372	0.1659
31	44.1703	44.3208	0.5354	44.3604	0.5494	44.2578	0.5855
32	-46.3812	-46.4228	0.9293	-46.3792	0.9382	-46.4679	0.9750
33	-36.5478	-36.1842	2.2204	-36.1941	2.2091	-36.6956	0.7564
34	-12.1963	-11.6190	2.1379	-11.6198	2.1279	-12.2956	0.3880
35	0.0723	0.0729	0.0022	0.0729	0.0022	0.0728	0.0023
36	0.0927	0.0908	0.0025	0.0926	0.0015	0.0928	0.0015
37	-0.0229	-0.0236	0.0012	-0.0236	0.0012	-0.0240	0.0014
38	-0.0518	-0.0516	0.0018	-0.0517	0.0018	-0.0513	0.0021
39	-0.0391	-0.0396	0.0015	-0.0396	0.0015	-0.0394	0.0016
40	0.0254	0.0251	0.0013	0.0251	0.0013	0.0253	0.0013
41	0.0614	0.0612	0.0020	0.0612	0.0020	0.0611	0.0022
42	0.1063	0.1061	0.0028	0.1061	0.0028	0.1048	0.0038

Table 1: 34-node water network state estimates and confidence limits

1-34: nodal heads (m Aq) at nodes 1-34;

35-42: fixed-head nodes in/out flows (m^3/s) at nodes 27-34

All the results have been obtained for the following parameters: variability of consumptions - ±10 %, accuracy of head measurements - ±2 %, coefficient γ of Newton-Raphson method - 0.6, integration time constant - 10e-8 [s]