HIGH PERFORMANCE NEURAL OPTIMISATION FOR REAL TIME PRESSURE CONTROL

A. Bargiela

Real Time Telemetry Systems, Department of Computing
The Nottingham Trent University, Burton St., Nottingham NG1 4BU
tel: +44 (0) 115 9418418, fax: +44 (0) 115 9486518, e-mail: andre@doc.ntu.ac.uk

Abstract

Real time control of industrial processes requires that the computations are performed in a time scale that is compatible with the dynamics of the physical system. This in turn implies that such applications need sufficiently powerful computing hardware. This paper presents a high performance neural optimisation network and discusses its application in the context of real time minimisation of overpressures in large-scale water distribution systems. The nonlinear optimisation problem has been formulated as a state estimation process and then expressed as a set of nonlinear ordinary differential equations. Based on these equations, an analog neural network has been proposed and evaluated by means of extensive simulations. The neural optimisation system has been shown to deliver high computational performance which is independent of the problem size. This is due to the massively parallel nature of the neural network. The neural optimisation method presented in this paper readily generalises to a wide spectrum of optimisation problems.

1. Introduction

Many industrial optimisation problems involve solving large sets of nonlinear equations. However, frequently, direct solution of such equations using standard sequential computers, requires computational power that is either not available or it cannot be economically justified. This paper presents an economically viable high performance neural optimisation system and applies it to a real-life problem of pressure control in large-scale water distribution systems.

The primary aim of water distribution control is to maintain sufficient pressure to ensure that all demands for water are met. The idealised requirement is to keep pressure in every node constant relative to its ground level. However, owing to the pressure/flow relationship in the network this requirement can be satisfied only in few nodes while in others the pressure has to be higher than the required minimum. The control of pressures in the network is accomplished by throttling control valves according to the changing demand pattern. However, computation of the optimum valve settings is a difficult task because of the high dimensionality of the optimisation problem and the nonlinearity of the network model. In this paper, the optimisation task is expressed as a state estimation problem formulated within an augmented state space that include both the nodal pressures and the control valve settings as the state variables. Such state estimation problem has been translated onto a set of nonlinear ordinary differential equations which are implemented as an analog neural network. The neural network has been simulated using an analog circuit simulation software.

2. Problem Formulation

The task of minimisation of overpressures in water distribution network can be formulated as follows:

$$\underset{v_k}{\text{minimise}} \quad \sum_j \left| h_j - h_j^o \right| \qquad (1)$$

$$\text{s.t.} \quad 0 < v_k < v_k^{max} \qquad (2)$$

where $j = 1, \ldots, R$ - is the number of pressure controlled nodes (a subset of all nodes N),

$k = 1, \ldots, K$ - is the number of control valves,

v_k^{max} - is an operational limit on valve v_k

.

To be able to perform the optimisation (1) it is necessary to specify the functional relationship between the nodal heads h_j and valve controls v_k. This relationship is defined through element flow functions $f_{ij}(\mathbf{h,v})$ that depend on both nodal pressures $\mathbf{h}$ and valve throttling $\mathbf{v}$. Using the functions f_{ij} the mass balance equations are formulated for each node in the network.

$$\sum_{j \in M_i} f_{ij}(\mathbf{h,v}) = b_i \qquad i = 1, \dots, N \qquad (3)$$

where $\mathbf{h} = [h_1, \dots, h_N]^T$ is a vector of nodal pressures,

$\mathbf{v} = [v_1, \dots, v_K]^T$ is a vector of valve controls,

M_i is a set of nodes incident to node i,

b_i is a node balance.

However, the valve controls v_k are not known in advance. The information about the value of v_k can only be approximated by the following

$$v_k + r_k = v_k^o \qquad k = 1, \dots, K \qquad (4)$$

$$v_k + \nu_k = v_k^{max} \qquad k = 1, \dots, K \qquad (5)$$

where equation (4) represents uncertainty about the current approximation of valve control v_k^o, and equation (5) represents an operational limit imposed on valve control $\left(\dot{v}_k > 0\right)$.

Similarly, equations representing head in reference nodes can be expressed as:

$$h_j + r_j^h = h_j^o \qquad j = 1, \dots, R \qquad (6)$$

where r_j^h - is a discrepancy between the current and the optimal head.

Using the notation introduced above the optimisation problem (1) can now be written as follows

$$\underset{x}{\text{minimise}} \quad \mathbf{w}^T|\mathbf{r}| \qquad (7)$$

$$\text{s.t.} \quad g(\mathbf{x}) + \mathbf{r} = \mathbf{z} \qquad (8)$$

where $g(.)$ is a non-linear function of $\mathbf{x}$,

$$\mathbf{x} = [h_1, \dots, h_N, v_1, \dots, v_K, \nu_1, \dots, \nu_K]^T,$$

$$\mathbf{w} = \left[w_1^h, \dots, w_N^h, w_1^v, \dots, w_K^v, w_1^\nu, \dots, w_K^\nu, w_1^h, \dots, w_R^h\right]^T,$$

$$\mathbf{r} = \left[r_1^h, \dots, r_N^h, r_1^v, \dots, r_K^v, r_1^\nu, \dots, r_K^\nu, r_1^h, \dots, r_R^h\right]^T,$$

$$\mathbf{z} = \left[b_1, \dots, b_N, v_1^o, \dots, v_K^o, v_1^{max}, \dots, v_K^{max}, h_1^o, \dots, h_R^o\right]^T.$$

The values of the elements of the weighting vector $\mathbf{w}$ are chosen in such a way so as to reflect the requirement of the residuals vector $\mathbf{r}$. Since the mass balance equations (3) and the equations representing the operational limits of valves (5) must be satisfied exactly, the corresponding weights are high, which has an effect of zeroing r_i^h, r_i^ν. Conversely, the weights corresponding to the equations (4), representing the valve controls, are set to zero since the cost of valve control is neglected in the cost function (1). The equations (6) representing a discrepancy between the current and the optimal pressure in selected nodes are biased with small positive weights and effectively are the only ones which contribute in a non-trivial way to the value of the performance index.

3. Solution Technique

To cope with the non-linearity of the expressions in (7) a method of iterative linearisation has been used. This can be summarised as follows:

1. Expand $g(\mathbf{x})$ to first order using Taylor series about an initial guess of the state vector $\mathbf{x}^k, k = 0$

$$\mathbf{z} - g\left(\mathbf{r}^k\right) = J\Delta\mathbf{r} + \mathbf{r}(\Delta\mathbf{x}) \qquad (9)$$

where $J = \frac{d}{dx}g(\mathbf{x})$ and

$J \in R^{\overline{m} \times \overline{n}}, \quad \overline{m} = N + K + K + R, \quad \overline{n} = N + K + K$

.

2. Solve the optimisation problem for the linearised constraints

$$\underset{\Delta\mathbf{x}}{\text{minimise}} \quad \mathbf{w}^T|\mathbf{r}(\Delta\mathbf{x})| \qquad (10)$$

$$\text{s.t.} \quad \Delta\mathbf{z} = J\Delta\mathbf{x} + \mathbf{r}(\Delta\mathbf{x}) \qquad (11)$$

where $\Delta\mathbf{z} = g(\mathbf{x}) - g\left(\mathbf{x}^{o}\right)$

3. Update the state vector

$$\mathbf{x}^{k+1} = \mathbf{x}^{k} + \Delta\mathbf{x} \qquad (12)$$

4. If $\Delta\mathbf{x}$ satisfies the convergence test then stop, otherwise repeat iteration from step 1.

The optimisation problem in step 2 represents the major computational effort. In order to achieve a computational performance compatible with the real time control requirements, the problem is now formulated in terms of analog neural networks (NN). As a pre-requisite for such a formulation, the non-differentiable absolute values function $|.|$ is approximated by the function $\beta\mathbf{lncosh}(\mathbf{r}(\Delta\mathbf{x})/\beta)$. By selecting small positive values for the parameter β, $(0 < \beta < 1)$ this function can provide an arbitrarily close fit to the $|.|$ while being differentiable in the whole range of its argument (in particular for $r_i(\Delta\mathbf{x}) = 0$. The optimisation problem is therefore expressed as follows

$$\underset{\Delta\mathbf{x}}{\text{minimise}} \quad \mathbf{w}^{T}\beta\mathbf{lncosh}(\mathbf{r}(\Delta\mathbf{x})/\beta) \qquad (13)$$

$$\text{s.t.} \quad \Delta\mathbf{z} = J\Delta\mathbf{x} + \mathbf{r}(\Delta\mathbf{x}) \qquad (14)$$

Minimisation of the cost function (13) according to the gradient descent method leads to a system of differential equations

$$\frac{d}{dt}\Delta\mathbf{x} = \mu\left[J^{T}W\mathbf{tanh}(\mathbf{r}(\Delta\mathbf{x})/\beta)\right] \qquad (15)$$

where $r(\Delta\mathbf{x}) = \Delta\mathbf{z} - J\Delta\mathbf{x}$, $\Delta\mathbf{z} = \mathbf{z} - g\left(\mathbf{x}^{k}\right)$ and

$$W = \operatorname{diag}[w_1, \ldots, w_{\bar{m}}] \quad .$$

The neural network implementation of the system of differential equations (15) is given in Figure 1. The network can be seen as consisting of three layers: the input layer that adaptively calculates residuals; the second layer that calculates the derivatives of the increments to the state vector; and the third layer that calculates the increments to the state vector themselves.

Elaborating the differential equation (15) for each of the state variables one can write

$$\frac{d}{dt}\Delta x_j = \mu\sum_{i=1}^{\bar{m}} a_{ij}w_i\Psi_i(r_i) \qquad j = 1, \ldots, \bar{n} \qquad (16)$$

where Δx_j - is the j-th state variable,

$\Psi_i(r_i) = \tanh(r_i(\Delta x)/\beta)$ and

a_{ij} - is the i,j-th element of the Jacobian matrix

$J = [a_{i,j}]$, $i = 1, \ldots, \bar{m}$, $j = 1, \ldots, \bar{n}$.

The detailed neural network, implementing equation (16), is presented on Figure 2 .

It should be noted that simple bounds constraints on state variables: $x_{jmin} < x_j < x_{jmax}$ are conveniently implemented using integrators with signal limiters at their out-

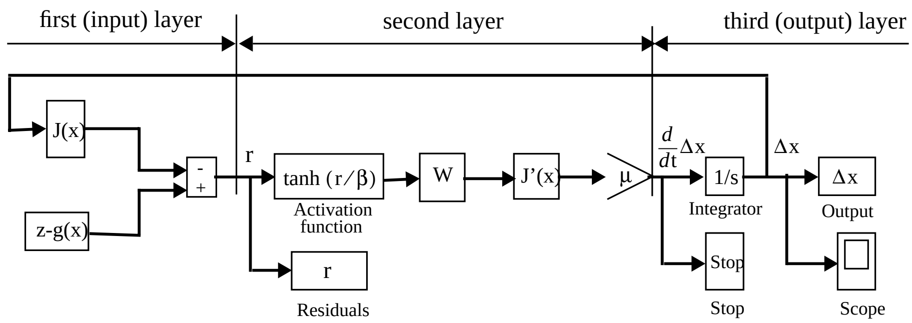


Figure 1: ANN for solving the optimisation problem (10)-(11)

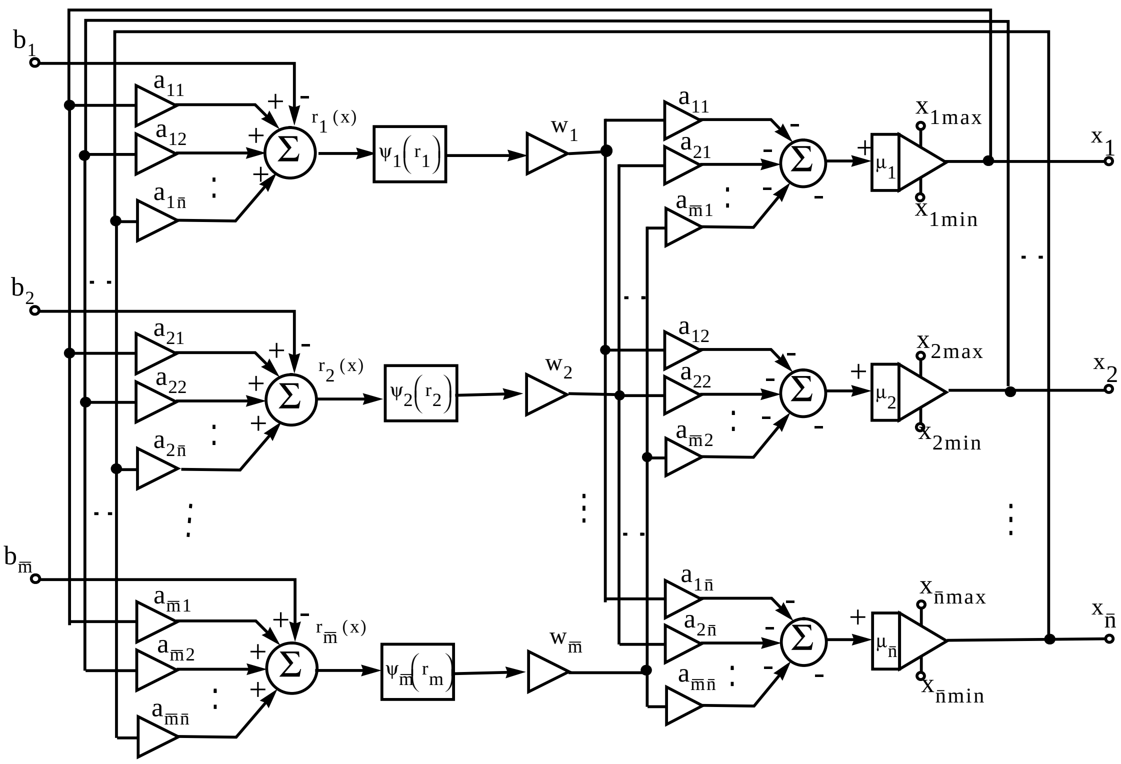


Figure 2. Detailed neural network (implementing Equation (16)

put. The input signals are integrated but cannot drive the output beyond the specified limits $[x_{jmin}, x_{jmax}]$. In such an approach, all simple bounds constraints are "hard", i.e., the constraints must not be violated either at the final solution or during the optimisation process. Equivalently one can employ a nonlinear transformation which maps an unlimited output signal u_j into an output-limited signal x_j of the j-th integrator.

$$x_j = q_j(u_j) \qquad (17)$$

e.g.

$$x_j = x_{j_{min}} + \frac{x_{j_{max}} - x_{j_{min}}}{1 + e^{-\gamma u_j}} \qquad (18)$$

$\gamma > 0$.

4. Computer Simulation Results

The performance of the neural network based pressure optimisation system was tested on several networks of various sizes. An example network is presented in Figure 3. The network is supplied from nodes 23, 24 and 25 and it contains three control valves in links 12-13, 12-15 and 21-22. The service pressure is monitored in nodes 6,13, 18 and 22 which have locally the highest ground level. The optimal pressure profile is defined as 30mAq above the ground level in the four pressure monitoring nodes. In order to test the neural optimisation network a full range of supply conditions has been considered, as documented in [1]. The results obtained with the neural network matched exactly the results calculated analytically. Also, it has been found through extensive simulation study that the calculated results were relatively insensitive to the value of the neural network parameters. Accurate state vectors were obtained for the parameter β in the range 10^{-3} to 10^{-1} and the network learning parameter μ up to 10^9. The time constant of the integrators has been assumed to be $0.01 \mu s$.

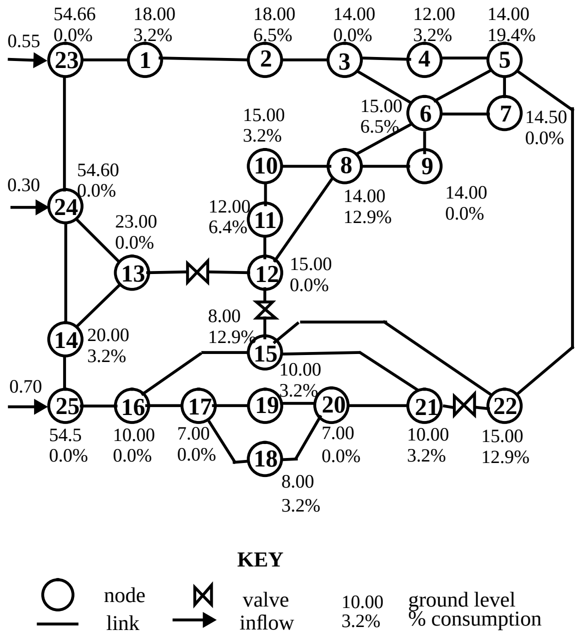
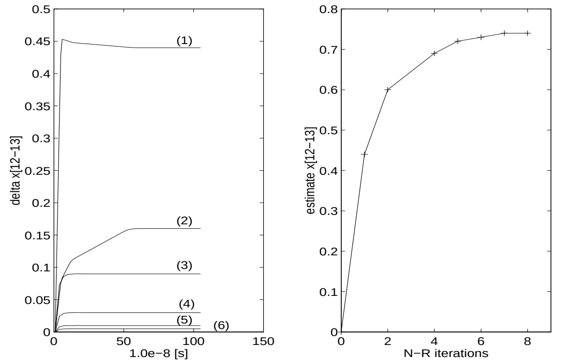


Figure 3: Test water distribution network

Figure 4 presents a representative example of the convergence of the state variable representing throttling of the valve 12-13 in consecutive Newton-Raphson iterations, calculated by the neural network. The parameters of the neural network have been set as follows: $\beta = 10^{-2}$ and $\mu = 10^{8}$. The corresponding supply conditions in the water network were: 0.65 m^3/s in node 23, 0.38 m^3/s in node 24 and 0.65 m^3/s in node 25. The calculated value of 74% throttling for the valve 12-13 (see Figure 4(b)), 15% for valve 13-15 and 92% for valve 21-22 matched exactly the analytical computations [1].
As can be seen from Figure 4 the linearised state variables Δx converged to their final values in a time of order of $1\mu s$ so the six Newton-Raphson iterations require less than $10\mu s$.

5. Conclusions

This paper discussed the high performance neural optimisation technique and its application to the real time pressure control problem. The control problem has been formulated as a state estimation in an augmented state space and then represented by a set of ordinary nonlinear differential equations. The implementation of these



(a) *(b)*

Figure 4:State variable - throttling of valve 12-13

(a) Convergence of the state increments,
(b) Convergence of the Newton-Raphson process

equations, as an analog neural network has been found to be computationally very efficient and eminently suitable for real time applications.

The proposed neural network can be constructed from simple analog elements such as adders, multipliers, function generators and limiting integrators.

It must be noted however that the above simulation results are intended to quantify the performance of the analog neural network of Figure 1 and DO NOT suggest that the optimisation problem (10) -(11) can be solved efficiently by means of simulation of this network using serial computers. Indeed the CPU times on SUN Sparc 10 workstation running MATLAB/SIMULINK software ranged from 180 to 220 [s] depending on the initial state.

References

1.M.Sterling, A.Bargiela, Leakage reduction by optimised control of valves in water networks, Trans. Inst. MC, Vol. 6, No 6, 1984, pp. 293-298

2.A.Cichocki, A.Bargiela, Neural networks for solving linear inequality systems, (preprints available from authors)

3.B.Gabrys, A. Bargiela, Neural simulation of water systems for efficient state estimation, Proc. of ESM'95, Soc. for Computer Simulation, ISBN1-56555-0803

4.A. Dax, The l_1 solution of linear equations subject to linear constraints, SIAM J. Science Stat. Comput., Vol. 10, No. 2, March 1989, pp. 328-340.

5.J.J. Hopfield and D.W. Tank, Neural computation of decision in optimising problems, Biological Cybernetics, Vol. 52, 1985, pp. 141-152.