Unified Framework for Deriving Simultaneous Equation Algorithms for Water Distribution Networks

Ezio Todini[1] and Lewis A. Rossman, M.ASCE[2]

Abstract: The known formulations for steady-state hydraulics within looped water distribution networks are rederived in terms of linear and nonlinear transformations of the original set of partly linear and partly nonlinear equations that express conservation of mass and energy. All of these formulations lead to a system of nonlinear equations that can be linearized as a function of the chosen unknowns using either the Newton-Raphson (NR) or the linear theory (LT) approaches. This produces a number of different algorithms, some of which are already known in the literature, whereas others have been originally developed within this work. For the sake of clarity, all the different algorithms were rederived using the same analytical approach and a unified notation. They were all applied to the same test case network with randomly perturbed demands to compare their convergence characteristics. The results show that all of the linearly transformed formulations have exactly the same convergence rate, whose value depends on whether a NR or LT algorithm was used, and that they converge faster than the nonlinearly transformed formulations do. A number of computational factors suggest that the global algorithm, in either its NR or LT form, is the most attractive of the various formulations to implement. **DOI: 10.1061/(ASCE)HY.1943-7900.0000703.** © *2013 American Society of Civil Engineers.*

CE Database subject headings: Water distribution systems; Algorithms; Hydraulics.

Author keywords: Water distribution networks; Simultaneous equations algorithms; Newton-Raphson; Linear theory.

Introduction

Numerous algorithms have been developed over the years for solving the mixed set of linear and nonlinear equations that govern the steady-state hydraulics of looped water distribution networks (WDNs). As can be seen from Fig. 1, the different approaches can be divided into the early local approaches (Cross 1936), which dealt with one equation at a time, and the more recent simultaneous equation approaches, which use either the Newton-Raphson (NR) linearization approach or the linear theory (LT) successive approximation approach to accommodate the system's nonlinear equations. Examples of the NR approach include the head method of Martin and Peters (1963), the flow and loop methods of Wood and Rayes (1981), and the global gradient algorithm of Todini and Pilati (1988). Only two algorithms (Wood and Charles 1972; Isaacs and Mills 1980) are known to use the LT approach, while a third one by Gupta and Prasad (2000) is using a nonlinear approach that could be classified as LT in the log space.

Todini (1999, 2006) showed that several of the NR-based algorithms could be derived through simple linear transformations of the original global algorithm (GA) formulation of Todini and Pilati (1988). Nothing similar has been done to extend the LT approach to the different formulations. The aim of this paper is to extend Todini's results to the domain of LT-based formulations and to show that all the algorithms that can be obtained via linear transformation of the GA produce exactly the same convergence results whose rate only depends on whether a NR or LT approach is used. This unifying effort is applied to the classical definition of the WDN problem (where demands are fixed quantities not dependent on pressure and no additional hydraulic devices other than pipes are considered) for the sake of clarity. The solution algorithms derived from this basic formulation serve as the starting point from which more realistic representations of water distribution systems, which include pumps, valves, and both time-varying and pressure-dependent demands, can be modeled.

Following Fig. 2, one can see all the possible alternatives that can be obtained by simple linear transformation as well as by nonlinear ones. Note that the extension to LT of the nodal head (NH) formulation, drawn in Fig. 2 with dashed line, showed total lack of convergence. This is why it will not be considered in the rest of the paper. Three new LT algorithms, namely LT-GA, LT-loop flow (LF), and LT-loop node (LN) were originally developed within the frame of this work, in analogy with the NR-GA, NR-LF, and NR-LN algorithms.

Basic Equations for a WDN

Conservation principles lead to two sets of equations that describe the distribution of flows **Q** and hydraulic heads **H** throughout a looped network of pipes subjected to a given set of water withdrawal rates. The first set of equations represents conservation of energy as described by Bernoulli's principle applied across the length of each pipe k (Bhave 1991)

$$H_i - H_j - \Phi(Q_k) = 0 \tag{1}$$

where i and j represent the nodes at the extremes of pipe k and $\Phi(Q_k)$ describes the head loss due to friction as a function of flow

[1]Professor of Alma Mater, Dept. of Earth and Geo-Environmental Sciences, Univ. of Bologna, Piazza di Porta S. Donato 1, I-40127 Bologna, Italy (corresponding author). E-mail: ezio.todini@unibo.it

[2]Environmental Scientist, National Risk Management Research Laboratory, U.S. Environmental Protection Agency, Cincinnati, OH 45268. E-mail: Rossman.Lewis@epamail.epa.gov

Note. This manuscript was submitted on January 5, 2012; approved on November 8, 2012; published online on November 10, 2012. Discussion period open until October 1, 2013; separate discussions must be submitted for individual papers. This paper is part of the ***Journal of Hydraulic Engineering***, Vol. 139, No. 5, May 1, 2013. © ASCE, ISSN 0733-9429/2013/5-511-526/$25.00.

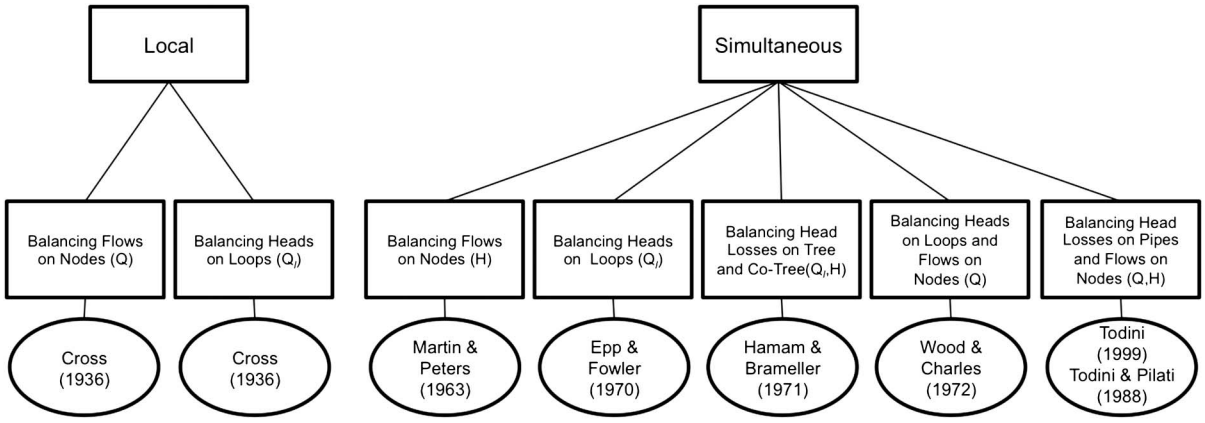

Fig. 1. Different approaches to the solution of the pipe network analysis problem; the variable(s) on which linearization is based are shown in parentheses

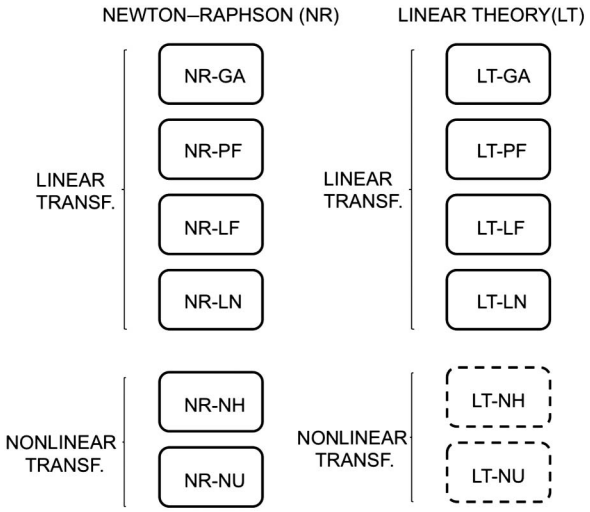

Fig. 2. Categorization of the different WDN algorithms; the first two letters in the acronym relate to the nonlinear solution approach: NR = Newton-Raphson, while LT = linear theory; the last two letters relate to the different bases of representation: GA = global algorithm; PF = pipe flows algorithm; LF = loop flows algorithm; LN = loop node algorithm; NH = nodal head algorithm; NU = nodal head algorithm with under-relaxation

rate. The second set of equations represent conservation of mass, which requires that total inflow equal total outflow at each network node to which pipes are connected. This linear set of equations can be expressed for each node i as follows:

$$\sum_{k=1}^{n_i} Q_{k_{i,j}} + q_i = 0 \tag{2}$$

where $Q_{k_{i,j}}$ = flow in the generic pipe $k_{i,j}$ connected to node i; n_i = number of pipes connected to node i; and q_i = known demand (or withdrawal rate) at node i. The sign convention used here is that flow into a node is positive while flow out of a node is negative.

The frictional head loss function appearing in Eq. (1) is typically expressed as a power function of Q_k

$$\Phi = r_k |Q_k|^{\alpha-1} Q_k \tag{3}$$

where r_k = coefficient that depends on the dimensions used as well as on the pipe diameter, length, and possibly the flow rate while α is an exponent, whose value is 2 when using the Darcy-Weisbach equation or 1.852 when using the Hazen-Williams equation. Note that when written in this manner, Φ maintains the same sign as Q_k. Substituting Eq. (3) into Eq. (1) produces the following restatement of the conservation of energy equation:

$$H_i - H_j - r_k |Q_k|^{\alpha-1} Q_k = 0 \tag{4}$$

Eqs. (2) and (4) are the elementary equations describing a water distribution system. When dealing with real WDNs, these equations must be modified in order to account for localized losses (also known as minor losses), pumps, different types of valves, etc. This can be done without changing, unless they affect the topology, the overall properties of the network. This is why in this paper, for the sake of clarity and without loss of generality, these equations will be kept as simple as possible in the sequel.

In order to compare all the algorithms and to show how they can be mutually derived via linear or nonlinear transformation, a matrix formulation was essentially required and used in this paper. Nonetheless, all the resulting algorithms may be transformed in their simpler scalar analogue version, similar to what can be found in the *EPANET 2 Users Manual* (Rossman 2000, Appendix D).

Topological Representation and Properties of Looped Networks

In order to derive all the algorithms in Fig. 2, it is necessary to introduce a number of topological relations that are present in looped networks. In very broad terms, a WDN is represented by pipes that convey water between nodes which connect pipes to one another and where inflows and outflows of water occur. A WDN can have the shape of an open tree, commonly used in irrigation schemes, or include a number of closed loops to increase its service reliability, as in urban WDNs. Following graph theory, a loop will be defined as any sequence of pipes within the network that begin and end at the same node. This includes the simple case of two parallel pipes connecting the same pair of nodes. Several authors (Kesavan and Chandrashekar 1972; Rahal 1995) have used graph theory definitions, where a looped WDN can be represented as a connected graph formed by vertices (nodes) and edges (pipes), which can be decomposed into a tree. A tree is defined as the connected graph, which will include the minimum number of edges

that allow all vertices to remain connected. A co-tree can then be defined, which is composed by all the edges that must be added to the tree to complete the graph, by closing the loops. For any given graph many alternative trees, as well as the corresponding co-trees, can be identified. For a connected graph with n_S subnetworks and n_0 sources, if p is the number of edges (pipes), N the number of vertices (nodes), n_0 the number of source nodes (nodes with known head) and l the number of chords (loops), the following topological relation holds: $p = (N - n_0) + l + n_S - 1 = n + l + n_S - 1$, where $n = N - n_0$ is the number of unknown head nodes. Please note that in WDN analysis it is common practice to add fictitious links among the n_0 fixed head nodes in order to obtain a fully looped network (e.g., Gupta and Prasad 2000), namely $n_S = 1$, which leads to the following relation: $p = l + N - n_0$.

Therefore, in a WDN characterized by $N = n + n_0$ nodes, where $n_0 \geq 1$ is the number of known head nodes, the number of unknown head nodes n equals t, the number of pipes in the tree, while l the number of loops equals the number of chords c. Given the identities $n = t$ and $l = c$, in this paper for the sake of clarity only n and l will be used instead of t and c as in Osiadacz (1988) and Osiadacz and Pienkosz (1988).

The topology of the WDN can then be described by means of two topological incidence matrices, the first one relating the pipes to the nodes, and the second one relating the pipes to the loops. Following the convention that inflows to a node are assumed positive and outflows negative, the first incidence matrix $\bar{\mathbf{A}}_{12}$ relating the pipes to the nodes can be defined as follows:

$$\bar{\mathbf{A}}_{12}(i,j) = \begin{cases} -1 & \text{if pipe } i \text{ leaves node } j \\ 0 & \text{if pipe } i \text{ is not connected to node } j \\ +1 & \text{if pipe } i \text{ enters node } j \end{cases}$$

This matrix of size $[p; N]$ relates all the pipes and all the nodes in the WDN, including the fixed head nodes. For the purpose of writing the equations needed for the solution of the unknown head nodes, it is convenient to partition this matrix into two submatrices

$$\bar{\mathbf{A}}_{12} = \begin{bmatrix} \mathbf{A}_{12} & \vdots & \mathbf{A}_{10} \end{bmatrix} \tag{5}$$

where $\mathbf{A}_{12}$ is a $[p; n]$ matrix relating the pipes to the unknown head nodes and $\mathbf{A}_{10}$ is a $[p; n_0]$ matrix relating the pipes to the fixed head nodes. For the sake of clarity the following notation is used in the rest of the paper $\mathbf{A}_{21} = \mathbf{A}_{12}^T$ and $\mathbf{A}_{01} = \mathbf{A}_{10}^T$.

The second topological matrix of interest is $\mathbf{M}_{13}$ with dimension $[p; l]$, relating the loops to the pipes. It can be defined as follows:

$$\mathbf{M}_{13}(i,k) = \begin{cases} -1 & \text{if pipe } i \text{ is in loop } k \text{ and their directions are opposed} \\ 0 & \text{if pipe } i \text{ is notin loop } k \\ +1 & \text{if pipe } i \text{ is in loop } k \text{ and their directions} \end{cases}$$

Here again the following notation $\mathbf{M}_{31} = \mathbf{M}_{13}^T$ will be used in the rest of the paper.

While matrices $\bar{\mathbf{A}}_{12}$, $\mathbf{A}_{12}$ and $\mathbf{A}_{10}$ are uniquely defined when the topology of the WDN is known, matrix $\mathbf{M}_{13}$ is not unique. Under the condition that $n + l = p$, the matrix $\mathbf{A}_{21}$ can additionally be partitioned into two submatrices as follows:

$$\mathbf{A}_{12} = \begin{bmatrix} \mathbf{A}_{n2} \\ \cdots \\ \mathbf{A}_{l2} \end{bmatrix} \tag{6}$$

$$\mathbf{A}_{21} = \begin{bmatrix} \mathbf{A}_{2n} & \vdots & \mathbf{A}_{2l} \end{bmatrix} \tag{7}$$

where $\mathbf{A}_{n2} = \mathbf{A}_{2n}^T$ are $[n, n]$ (0, 1) matrices that must be invertible, which corresponds to the fact that all the n unknown head nodes are connected by the set of the n selected pipes. The reason for matrix $\mathbf{M}_{13}$ not being unique is that there are $\binom{p}{n}$ possible combinations of pipes leading to different matrices $\mathbf{A}_{n2}$, not all of them invertible. Only those matrices containing pipes that connect all the n unknown head nodes will be invertible and could then be used to derive matrix $\mathbf{M}_{13}$. The n pipes that form a tree for the network will have this property and thus form the $\mathbf{A}_{n2}$ partition. The remaining co-tree pipes belong to the $\mathbf{A}_{2l}$ partiton. As previously stated and as clearly shown by Osiadacz (1988), the co-tree subdivision corresponds to the loops $c = l$ and in practice the different choices of the co-tree incidence sub-matrix $\mathbf{A}_{2l}$ correspond to all the alternative ways of selecting the loops in the network.

Matrices $\mathbf{M}_{13}$ and $\mathbf{M}_{31}$ can then be represented as follows:

$$\mathbf{M}_{13} = \begin{bmatrix} -\mathbf{A}_{2n}^{-1}\mathbf{A}_{2l} \\ \cdots \\ \mathbf{I}_{ll} \end{bmatrix} \tag{8}$$

$$\mathbf{M}_{31} = \begin{bmatrix} -\mathbf{A}_{l2}\mathbf{A}_{n2}^{-1} & \vdots & \mathbf{I}_{ll} \end{bmatrix} \tag{9}$$

with $\mathbf{I}_{ll}$ an $[l, l]$ identity matrix.

It is easy to verify that

$$\mathbf{M}_{31}\mathbf{A}_{12} = \begin{bmatrix} -\mathbf{A}_{l2}\mathbf{A}_{n2}^{-1} & \vdots & \mathbf{I}_{ll} \end{bmatrix} \begin{bmatrix} \mathbf{A}_{n2} \\ \cdots \\ \mathbf{A}_{l2} \end{bmatrix} = [-\mathbf{A}_{l2} + \mathbf{A}_{l2}] = 0 \tag{10}$$

Given these definitions of the incidence matrices and owing to the topological properties of nodes, pipes, and loops in a network, it is not difficult to verify that the following properties hold: $\mathbf{M}_{31}\mathbf{A}_{12} = 0$ and $\mathbf{A}_{21}\mathbf{M}_{13} = 0$, while the property $\mathbf{M}_{31}\mathbf{A}_{10} = 0$ and $\mathbf{A}_{01}\mathbf{M}_{13} = 0$ will only hold if there is just a single fixed head node. When the number of fixed head nodes n_0 is larger than one, $\mathbf{M}_{31}\mathbf{A}_{10} \neq 0$ and $\mathbf{A}_{01}\mathbf{M}_{13} \neq 0$, with no significant consequences regarding the development of the different algorithms, as it will be demonstrated in the rest of the paper. It is from these properties that it is possible to project the system of equations [Eq. (1)] onto vector bases smaller in size than the original base of size $p + n$.

J. Hydraul. Eng., 2013, 139(5): 511-526

Another condition imposed by a network's topology is the possibility that not all the pipe flows (PFs) are independent, but n of them can be expressed as a linear combination of the remaining l flows in the pipes (chords) that were added to close the loops. The flow vector $\mathbf{Q}$ can in fact be partitioned into two subvectors, the first one of size n and the second one of size l by using the continuity of mass equation

$$\mathbf{A}_{21}\mathbf{Q} = \begin{bmatrix} \mathbf{A}_{2n} & \vdots & \mathbf{A}_{2l} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_n \\ \cdots \\ \mathbf{Q}_l \end{bmatrix} = -\mathbf{q} \tag{11}$$

By inverting the matrix $\mathbf{A}_{2n}$ (which is invertible because it was appropriately chosen) one obtains

$$\mathbf{Q}_n = -\mathbf{A}_{2n}^{-1}\mathbf{q} - \mathbf{A}_{2n}^{-1}\mathbf{A}_{2l}\mathbf{Q}_l \tag{12}$$

and, setting $\bar{\mathbf{Q}}_n = -\mathbf{A}_{2n}^{-1}\mathbf{q}$ as a known flow vector, this leads to

$$\mathbf{Q}_n = \bar{\mathbf{Q}}_n - \mathbf{A}_{2n}^{-1}\mathbf{A}_{2l}\mathbf{Q}_l \tag{13}$$

which shows that $\mathbf{Q}_n$ can be represented as a linear function of $\mathbf{Q}_l$, and $\mathbf{Q}$ the full set of flows in all the pipes can be written as follows:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_n \\ \cdots \\ \mathbf{Q}_l \end{bmatrix} = \begin{bmatrix} \bar{\mathbf{Q}}_n \\ \cdots \\ 0 \end{bmatrix} + \begin{bmatrix} -\mathbf{A}_{2n}^{-1}\mathbf{A}_{2l} \\ \cdots \\ \mathbf{I}_{ll} \end{bmatrix} \mathbf{Q}_l = \bar{\mathbf{Q}} + \mathbf{M}_{13}\mathbf{Q}_l \tag{14}$$

where

$$\bar{\mathbf{Q}} = \begin{bmatrix} \bar{\mathbf{Q}}_n \\ \cdots \\ 0 \end{bmatrix} = \begin{bmatrix} -\mathbf{A}_{2n}^{-1}\mathbf{q} \\ \cdots \\ 0 \end{bmatrix} \tag{15}$$

Thus, $\bar{\mathbf{Q}}$ represents a flow solution that satisfies nodal mass continuity with nonzero flow in the tree pipes and zero flow in the co-tree pipes.

Derivation of the Different Reduced Representations

Global Algorithm Representation

The GA representation, as derived by Todini and Pilati (1988) through the constrained minimization of the Collins et al. (1978) content problem, provides the starting point for the other transformed formulations discussed in this paper. It is simply a restatement of the governing Eqs. (2) and (4) in the following matrix notation:

$$\begin{bmatrix} \mathbf{A}_{11} & \vdots & \mathbf{A}_{12} \\ \cdots & \cdots & \cdots \\ \mathbf{A}_{21} & \vdots & 0 \end{bmatrix} \begin{bmatrix} \mathbf{Q} \\ \cdots \\ \mathbf{H} \end{bmatrix} = \begin{bmatrix} -\mathbf{A}_{10}\mathbf{H}_0 \\ \cdots \\ -\mathbf{q} \end{bmatrix} \tag{16}$$

where

$\mathbf{Q}^T = [Q_1, Q_2, \ \ldots, \ Q_p]$, the $[1, p]$ unknown pipe discharges

$\mathbf{H}^T = [H_1, H_2, \ \ldots \ H_n]$, the $[1, n]$ unknown nodal heads

$\mathbf{H}_0^T = [H_{n+1}, H_{n+2}, \ \ldots \ H_N]$, the $[1, N-n]$ known nodal heads

$\mathbf{q}^T = [q_1, q_2, \ \ldots \ q_n]$, the $[1, n]$ known nodal demands

with p as the number of pipes, n as the number of unknown head nodes, N as the total number of nodes in the network leaving $N-n$ as the number of nodes with known head.

In Eq. (16) $\mathbf{A}_{11}$ is a diagonal matrix whose elements are defined as

$$\mathbf{A}_{11}(k,k) = r_k|Q_k|^{\alpha-1} \tag{17}$$

for $k \in 1, p; i \in 1, N; j \in 1, N$, where k = index of the pipe connecting two generic nodes i, j.

As previously shown, this formulation leads to a set of $p+n$ unknowns, namely p unknown discharges in the pipes and n unknown heads at the nodes. Although alternative starting points could be used, it is the intent of this paper to show how several existing representations can be derived from Eq. (16) by linear transformations to produce algorithms, which differ in terms of the set of unknowns chosen and consequently in terms of matrix sizes and matrix characteristics.

Pipe Flows (PF) Representation

The most immediate reduced representation to be derived is the Pipe Flow formulation used in the original LT introduced by Wood and Charles (1972). If one premultiplies both sides of Eq. (16) by the following matrix

$$\begin{bmatrix} \mathbf{M}_{31} & \vdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \vdots & \mathbf{I}_{nn} \end{bmatrix} \tag{18}$$

where $\mathbf{I}_{nn}$ is an $[n; n]$ identity matrix, one obtains

$$\begin{bmatrix} \mathbf{M}_{31} & \vdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \vdots & \mathbf{I}_{nn} \end{bmatrix} \begin{bmatrix} \mathbf{A}_{11} & \vdots & \mathbf{A}_{12} \\ \cdots & \cdots & \cdots \\ \mathbf{A}_{21} & \vdots & 0 \end{bmatrix} \begin{bmatrix} \mathbf{Q} \\ \cdots \\ \mathbf{H} \end{bmatrix} = \begin{bmatrix} \mathbf{M}_{31} & \vdots & 0 \\ \cdots & \cdots & \cdots \\ 0 & \vdots & \mathbf{I}_{nn} \end{bmatrix} \begin{bmatrix} -\mathbf{A}_{10}\mathbf{H}_0 \\ \cdots \\ -\mathbf{q} \end{bmatrix} \tag{19}$$

Taking into account that $\mathbf{M}_{31}\mathbf{A}_{12} = 0$, as discussed in the previous section, the following system of equations is obtained:

$$\begin{bmatrix} \mathbf{M}_{31}\mathbf{A}_{11} \\ \cdots \\ \mathbf{A}_{21} \end{bmatrix} \mathbf{Q} = \begin{bmatrix} -\mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0 \\ \cdots \\ -\mathbf{q} \end{bmatrix} \tag{20}$$

As one can see, the original problem of Eq. (16) in $p+n$ unknowns, namely the pipe flows $\mathbf{Q}$ together with the nodal heads $\mathbf{H}$ is now transformed into the solution of a problem with p unknown pipe flows.

Loop Flows Representation

A second reduced representation, the so-called Loop Flows formulation (Epp and Fowler 1970), solves the problem in terms of a subset of flows. This is possible by substituting for $\mathbf{Q}$ from Eq. (14) into Eq. (20)

$$\begin{bmatrix} \mathbf{M}_{31}\mathbf{A}_{11} \\ \cdots \\ \mathbf{A}_{21} \end{bmatrix} (\bar{\mathbf{Q}} + \mathbf{M}_{13}\mathbf{Q}_l) = \begin{bmatrix} -\mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0 \\ \cdots \\ -\mathbf{q} \end{bmatrix} \tag{21}$$

Given that $\mathbf{A}_{21}\bar{\mathbf{Q}} = -\mathbf{q}$ and $\mathbf{A}_{21}\mathbf{M}_{13}\mathbf{Q}_l = 0$ (as $\mathbf{A}_{21}\mathbf{M}_{13} = 0$), one can drop the lower part of the system that does not contain any

unknowns. Note that the information content of the demand is not lost but is now incorporated into the term $\mathbf{M}_{31}\mathbf{A}_{11}\bar{\mathbf{Q}}$.

The Loop Flows representation can thus be stated as follows:

$$\mathbf{M}_{31}\mathbf{A}_{11}(\bar{\mathbf{Q}} + \mathbf{M}_{13}\mathbf{Q}_l) = -\mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0 \tag{22}$$

which can be more conveniently rewritten as

$$\mathbf{M}_{31}\mathbf{A}_{11}\mathbf{M}_{13}\mathbf{Q}_l = -\mathbf{M}_{31}(\mathbf{A}_{10}\mathbf{H}_0 + \mathbf{A}_{11}\bar{\mathbf{Q}}) \tag{23}$$

where the unknowns are now the l unknown loop flows $\mathbf{Q}_l$. Note that this representation requires an initial set of flows $\bar{\mathbf{Q}}$ satisfying the nodal continuity equation to be assigned to all the pipes. In this work, given the liberty of choosing an initial mass balanced solution, the following initial solution was assumed $\bar{\mathbf{Q}} = -\mathbf{A}_{2n}^{-1}\mathbf{q}$ in order to preserve consistency with the LN representation derived next.

Loop-Node Representation

A third reduced representation is the so-called LN formulation, which is the basis of the hybrid algorithm (Hamam and Brameller 1971; Osiadacz 1988; Osiadacz and Pienkosz 1988). In order to derive this formulation, the original problem of Eq. (16) can be rewritten using the following partitioned matrices:

$$\mathbf{A}_{11} = \begin{bmatrix} \mathbf{A}_{nn} & 0 \\ 0 & \mathbf{A}_{ll} \end{bmatrix} \tag{24}$$

$$\mathbf{A}_{12} = \begin{bmatrix} \mathbf{A}_{n2} \\ \mathbf{A}_{l2} \end{bmatrix} = \mathbf{A}_{21}^T \tag{25}$$

$$\mathbf{A}_{10} = [\,\mathbf{A}_{n0} \quad \mathbf{A}_{l0}\,] = \mathbf{A}_{01}^T \tag{26}$$

After substituting these into Eq. (16) one has

$$\begin{bmatrix} \mathbf{A}_{nn} & 0 & \mathbf{A}_{n2} \\ 0 & \mathbf{A}_{ll} & \mathbf{A}_{l2} \\ \mathbf{A}_{2n} & \mathbf{A}_{2l} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{Q}_n \\ \mathbf{Q}_l \\ \mathbf{H} \end{bmatrix} = \begin{bmatrix} -\mathbf{A}_{n0}\mathbf{H}_0 \\ -\mathbf{A}_{l0}\mathbf{H}_0 \\ -\mathbf{q} \end{bmatrix} \tag{27}$$

Substituting for $\mathbf{Q}_n$ from Eq. (13), the vector of unknowns can be expressed as a function of $\mathbf{Q}_l$ and $\mathbf{H}$

$$\begin{bmatrix} \mathbf{Q}_n \\ \mathbf{Q}_l \\ \mathbf{H} \end{bmatrix} = \begin{bmatrix} -\mathbf{A}_{2n}^{-1}\mathbf{q} \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} -\mathbf{A}_{2n}^{-1}\mathbf{A}_{2l} & 0 \\ \mathbf{I}_{ll} & 0 \\ 0 & \mathbf{I}_{nn} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_l \\ \mathbf{H} \end{bmatrix} \tag{28}$$

Eq. (28) can be substituted into Eq. (27) to give

$$\begin{bmatrix} \mathbf{A}_{nn} & 0 & \mathbf{A}_{n2} \\ 0 & \mathbf{A}_{ll} & \mathbf{A}_{l2} \\ \mathbf{A}_{2n} & \mathbf{A}_{2l} & 0 \end{bmatrix} \begin{bmatrix} -\mathbf{A}_{2n}^{-1}\mathbf{A}_{2l} & 0 \\ \mathbf{I}_{ll} & 0 \\ 0 & \mathbf{I}_{nn} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_l \\ \mathbf{H} \end{bmatrix} = \begin{bmatrix} -\mathbf{A}_{n0}\mathbf{H}_0 \\ -\mathbf{A}_{l0}\mathbf{H}_0 \\ -\mathbf{q} \end{bmatrix} - \begin{bmatrix} \mathbf{A}_{nn} & 0 & \mathbf{A}_{n2} \\ 0 & \mathbf{A}_{ll} & \mathbf{A}_{l2} \\ \mathbf{A}_{2n} & \mathbf{A}_{2l} & 0 \end{bmatrix} \begin{bmatrix} -\mathbf{A}_{2n}^{-1}\mathbf{q} \\ 0 \\ 0 \end{bmatrix} \tag{29}$$

which after algebraic manipulation leads to

$$\begin{bmatrix} -\mathbf{A}_{nn}\mathbf{A}_{2n}^{-1}\mathbf{A}_{2l} & \mathbf{A}_{n2} \\ \mathbf{A}_{ll} & \mathbf{A}_{l2} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{Q}_l \\ \mathbf{H} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{nn}\mathbf{A}_{2n}^{-1}\mathbf{q} - \mathbf{A}_{n0}\mathbf{H}_0 \\ -\mathbf{A}_{l0}\mathbf{H}_0 \\ 0 \end{bmatrix} \tag{30}$$

and by dropping the null set of equations, one finally obtains the following:

$$\begin{bmatrix} -\mathbf{A}_{nn}\mathbf{A}_{2n}^{-1}\mathbf{A}_{2l} & \vdots & \mathbf{A}_{n2} \\ \cdots & \cdots & \cdots \\ \mathbf{A}_{ll} & \vdots & \mathbf{A}_{l2} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_l \\ \cdots \\ \mathbf{H} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{nn}\bar{\mathbf{Q}}_n - \mathbf{A}_{n0}\mathbf{H}_0 \\ \cdots \\ -\mathbf{A}_{l0}\mathbf{H}_0 \end{bmatrix} \tag{31}$$

The original problem has now been transformed into one with p unknowns, similar to the PF-based problem used in LT, but with a different set of unknowns. In the LN representation, the unknowns are the l flows $\mathbf{Q}_l$ plus the n nodal heads $\mathbf{H}$ for a total of $p = l + n$ unknowns, while the remaining $\mathbf{Q}_n$ unknowns can be derived as a function of $\mathbf{Q}_l$ using Eq. (13). Thus as with the loop representation, an initial set of flows for the nonchord pipes that satisfy nodal continuity ($\bar{\mathbf{Q}}_n = -\mathbf{A}_{2n}^{-1}\mathbf{q}$) is required.

Nodal Head Representation

This reduced representation differs from the previous ones because it applies a nonlinear transformation of the original GA representation given by Eq. (16). In this case, the head loss relation described in Eq. (3) is reversed by initially taking the absolute value of both sides. This allows one to express $|Q_k|$ as

$$|Q_k| = \frac{|\Delta H_k|^{1/\alpha}}{r_k^{1/\alpha}}$$

where $\Delta H_k = H_i - H_j$. Substituting this back into the energy Eq. (4) gives

$$Q_k = \Psi(H_k) = r_k^{-1/\alpha}|\Delta H_k|^{\frac{1-\alpha}{\alpha}}\Delta H_k \tag{32}$$

which can be used to represent the flows that appear in the nodal continuity Eq. (2). This allows one to represent the problem with a set of n equations in the n unknown heads $\mathbf{H}$, which can be written in matrix form as

$$\mathbf{A}_{21}\hat{\mathbf{A}}_{11}\mathbf{A}_{12}\mathbf{H} = \mathbf{q} - \mathbf{A}_{10}\mathbf{H}_0 \tag{33}$$

where

$$\hat{\mathbf{A}}_{11}(k,k) = \text{diag}\left\{\frac{\Psi(H_k)}{\Delta H_k}\right\} = \text{diag}\left\{r_k^{-1/\alpha}|\Delta H_k|^{\frac{1-\alpha}{\alpha}}\right\}$$

Summary of the Different Representations

Table 1 summarizes the five different representations of the WDN problem. For ease of reference, Table 2 restates the definitions of the various matrices used in these formulations. The GA representation is the starting point for the remaining four. The three loop-based formulations (PF, LF, and LN) are all derived by applying a linear set of transformations to the GA that implement the following topological conditions:

1. Kirchoff's second law—the total head loss around a closed loop of pipes must be zero;
2. Flows in the tree pipes that satisfy nodal flow continuity are uniquely determined by the flows assigned to the co-tree pipes.

The fourth representation, the NH, results in a nonlinear transformation of GA as it substitutes the inverse of the nonlinear head loss equation used in the GA into the nodal continuity equations.

Derivation of the Different NR Algorithms

The Newton-Raphson (NR) iterative method for solving a general set of n nonlinear equations $\mathbf{f}(\mathbf{x}) = 0$ for n unknowns $\mathbf{x}$ is usually expressed in the following form (Carnahan et al. 1969):

J. Hydraul. Eng., 2013, 139(5): 511-526

Table 1. Summary of the Different Reduced Representations of the WDN Problem

Representation	Governing equations	Unknowns
Global algorithm	$\mathbf{A}_{11}\mathbf{Q} + \mathbf{A}_{12}\mathbf{H} = -\mathbf{A}_{10}\mathbf{H}_0$	Nodal heads
	$\mathbf{A}_{21}\mathbf{Q} = -\mathbf{q}$	Pipe flows
Pipe flows	$\mathbf{M}_{31}\mathbf{A}_{11}\mathbf{Q} = -\mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0$	Pipe flows
	$\mathbf{A}_{21}\mathbf{Q} = -\mathbf{q}$	
Loop flows	$\mathbf{M}_{31}\mathbf{A}_{11}\mathbf{M}_{13}\mathbf{Q}_l = -\mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0 - \mathbf{M}_{31}\mathbf{A}_{11}\bar{\mathbf{Q}}$	Loop flows
Loop node	$\mathbf{A}_{ll}\mathbf{Q}_l + \mathbf{A}_{l2}\mathbf{H} = -\mathbf{A}_{l0}\mathbf{H}_0$	
	$-\mathbf{A}_{nn}\mathbf{A}_{2n}^{-1}\mathbf{A}_{2l}\mathbf{Q}_l + \mathbf{A}_{n2}\mathbf{H} = -\mathbf{A}_{n0}\mathbf{H}_0 + \mathbf{A}_{nn}\bar{\mathbf{Q}}$	Nodal heads Loop flows
Nodal head	$\mathbf{A}_{21}\hat{\mathbf{A}}_{11}\mathbf{A}_{12}\mathbf{H} = -\mathbf{A}_{10}\mathbf{H}_0 + \mathbf{q}$	Nodal heads

Table 2. Definitions of the Matrices Used in the Reduced Representations

$\mathbf{A}_{11}$	$[p;p]$ matrix whose diagonal entries are $r\|Q\|^{\alpha-1}$
$\hat{\mathbf{A}}_{11}$	$[n;n]$ matrix whose diagonal entries are $r^{-(1/\alpha)}\|\Delta H\|^{\frac{1-\alpha}{\alpha}}$
$\mathbf{A}_{12}$	$[p;n]$ matrix relating pipes to nodes with unknown heads
$\mathbf{A}_{10}$	$[p;n_0]$ matrix relating pipes to fixed head nodes
$\mathbf{M}_{31}$	$[p;l]$ matrix relating pipes to loops
$\mathbf{A}_{nn}$, $\mathbf{A}_{ll}$	partition of $\mathbf{A}_{11}$ containing the tree and co-tree pipes, respectively
$\mathbf{A}_{n0}$, $\mathbf{A}_{l0}$	partition of $\mathbf{A}_{10}$ containing the tree and co-tree pipes, respectively
$\mathbf{A}_{n2}$, $\mathbf{A}_{l2}$	partition of $\mathbf{A}_{12}$ containing the tree and co-tree pipes, respectively

$$\mathbf{J}(\mathbf{x}^\tau)(\mathbf{x}^{\tau+1} - \mathbf{x}^\tau) = -\mathbf{f}(\mathbf{x}^\tau)$$

or as simply

$$\mathbf{J}(\mathbf{x}^\tau)\mathbf{dx} = \mathbf{f}(\mathbf{x}^\tau)$$

where $\mathbf{J}$ is the $[n;n]$ Jacobian matrix of first derivatives of $\mathbf{f}(\mathbf{x})$ with respect to $\mathbf{x}$ with elements $\mathbf{J}(i,j) = \partial f_i/\partial x_j$, τ denotes the current iteration, and $\mathbf{dx} = \mathbf{x}^\tau - \mathbf{x}^{\tau+1}$. In one dimension, the method can be thought of approximating the function at its current solution point by its tangent line and using the point where this line intersects 0 as the next solution estimate.

Newton-Raphson Global Algorithm

The GA representation of the WDN problem in Eq. (16) can be rearranged as follows:

$$\begin{bmatrix} \mathbf{A}_{11} & \vdots & \mathbf{A}_{12} \\ \cdots & \cdots & \cdots \\ \mathbf{A}_{21} & \vdots & 0 \end{bmatrix} \begin{bmatrix} \mathbf{Q} \\ \cdots \\ \mathbf{H} \end{bmatrix} + \begin{bmatrix} \mathbf{A}_{10}\mathbf{H}_0 \\ \cdots \\ \mathbf{q} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \cdots \\ \mathbf{f}_2 \end{bmatrix} \tag{34}$$

where $\mathbf{f}_1 = \mathbf{f}_1(\mathbf{Q},\mathbf{H})$ and $\mathbf{f}_2 = \mathbf{f}_2(\mathbf{Q})$ indicate how far from zero the relevant equations are, for any given approximated solution $\mathbf{Q}$, $\mathbf{H}$. Applying the NR method directly to this system results in the following set of equations for the flow and head corrections at iteration τ of the process:

$$\begin{bmatrix} \mathbf{D}_{11}^\tau & \mathbf{A}_{12} \\ \mathbf{A}_{21} & 0 \end{bmatrix} \begin{bmatrix} \mathbf{dQ} \\ \mathbf{dH} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1^\tau \\ \mathbf{f}_2^\tau \end{bmatrix} \tag{35}$$

where

$$\mathbf{dQ} = \mathbf{Q}^\tau - \mathbf{Q}^{\tau+1} \tag{36}$$

$$\mathbf{dH} = \mathbf{H}^\tau - \mathbf{H}^{\tau+1} \tag{37}$$

$$\mathbf{f}_1^\tau = \mathbf{A}_{11}^\tau\mathbf{Q}^\tau + \mathbf{A}_{12}\mathbf{H}^\tau + \mathbf{A}_{10}\mathbf{H}_0 \tag{38}$$

$$\mathbf{f}_2^\tau = \mathbf{A}_{21}\mathbf{Q}^\tau + \mathbf{q} \tag{39}$$

and $\mathbf{D}_{11}$ is a $[p;p]$ diagonal matrix that represents the Jacobian of $\mathbf{A}_{11}\mathbf{Q}$. Its kth diagonal element is

$$\mathbf{D}_{11}(k,k) = \alpha_k r_k |Q_k|^{\alpha-1} \tag{40}$$

Following the derivation in the Appendix, the iterative steps of the NR-GA are

1. Solve the following $[n;n]$ system of equations for $\mathbf{H}^{\tau+1}$:

$$[\mathbf{A}_{21}(\mathbf{D}_{11}^\tau)^{-1}\mathbf{A}_{12}]\mathbf{H}^{\tau+1} = \mathbf{A}_{21}(\mathbf{D}_{11}^\tau)^{-1}[(\mathbf{D}_{11}^\tau - \mathbf{A}_{11}^\tau)\mathbf{Q}^\tau - \mathbf{A}_{10}\mathbf{H}_0] + \mathbf{q} \tag{41}$$

   where $\mathbf{D}_{11}^\tau$ is $\mathbf{D}_{11}$ evaluated for the current flows $\mathbf{Q}^\tau$.
2. Update flows by evaluating the scalar set of equations as follows:

$$\mathbf{Q}^{\tau+1} = \mathbf{Q}^\tau - (\mathbf{D}_{11}^\tau)^{-1}(\mathbf{A}_{11}^\tau\mathbf{Q}^\tau + \mathbf{A}_{12}\mathbf{H}^{\tau+1} + \mathbf{A}_{10}\mathbf{H}_0) \tag{42}$$

Thus each step of the NR-GA involves the solution of a $[n;n]$ sparse system of linear equation [Eq. (41)] to find nodal heads followed by the evaluation of p scalar equation [Eq. (42)] to find pipe flows. Note the inversion of matrix $\mathbf{D}_{11}$ is trivial, given that it is a diagonal matrix.

Newton-Raphson Pipe Flows Algorithm

The NR method will next be applied to the PF formulation of Eq. (20) where the unknowns are the pipe flows. Eq. (20) can be rearranged as follows:

$$\begin{bmatrix} \mathbf{M}_{31}\mathbf{A}_{11} \\ \cdots \\ \mathbf{A}_{21} \end{bmatrix} \mathbf{Q} + \begin{bmatrix} \mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0 \\ \cdots \\ \mathbf{q} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \cdots \\ \mathbf{f}_2 \end{bmatrix} \tag{43}$$

where $\mathbf{f}_1 = \mathbf{f}_1(\mathbf{Q})$ and $\mathbf{f}_2 = \mathbf{f}_2(\mathbf{Q})$ indicate how far from zero the relevant equations are, for any given approximated solution $\mathbf{Q}$.

Applying the NR formula to this set of equations results in

$$\begin{bmatrix} \mathbf{M}_{31}\mathbf{D}_{11}^\tau \\ \cdots \\ \mathbf{A}_{21} \end{bmatrix} \mathbf{dQ} = \begin{bmatrix} \mathbf{f}_1^\tau \\ \cdots \\ \mathbf{f}_2^\tau \end{bmatrix} \tag{44}$$

where

$$\mathbf{dQ} = \mathbf{Q}^\tau - \mathbf{Q}^{\tau+1}$$

$$\mathbf{f}_1^\tau = \mathbf{M}_{31}\mathbf{A}_{11}^\tau\mathbf{Q}^\tau + \mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0$$

$$\mathbf{f}_2^\tau = \mathbf{A}_{21}\mathbf{Q}^\tau + \mathbf{q}$$

and once again $\mathbf{D}_{11}$ is the Jacobian of $\mathbf{A}_{11}\mathbf{Q}$.

Replacing $\mathbf{f}_1^\tau$ and $\mathbf{f}_2^\tau$ in Eq. (44) with their respective definitions leads directly to the following NR-PF algorithm:

1. Solve the $[p;p]$ system of equations for $\mathbf{dQ}$

$$\begin{bmatrix} \mathbf{M}_{31}\mathbf{D}_{11}^{\tau} \\ \cdots \\ \mathbf{A}_{21} \end{bmatrix} \mathbf{dQ} = \begin{bmatrix} \mathbf{M}_{31}\mathbf{A}_{11}^{\tau} \\ \cdots \\ \mathbf{A}_{21} \end{bmatrix} \mathbf{Q}^{\tau} + \begin{bmatrix} \mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0 \\ \cdots \\ \mathbf{q} \end{bmatrix} \tag{45}$$

where $\mathbf{D}_{11}^{\tau}$ is $\mathbf{D}_{11}$ evaluated for the current flows $\mathbf{Q}^{\tau}$.

2. Update flows by evaluating

$$\mathbf{Q}^{\tau+1} = \mathbf{Q}^{\tau} - \mathbf{dQ} \tag{46}$$

Thus the original global problem in $p+n$ unknowns, namely the pipe flows $\mathbf{Q}$ together with the nodal heads $\mathbf{H}$, is now transformed into the solution of a problem with p unknown flows. Once the final flows have been found, the final heads $\mathbf{H}$ can be computed from

$$\mathbf{H} = (\mathbf{A}_{21}\mathbf{A}_{12})^{-1}\mathbf{A}_{21}(\mathbf{A}_{10}\mathbf{H}_0 + \mathbf{A}_{11}\mathbf{Q}) \tag{47}$$

However, to avoid having to take the inverse of $(\mathbf{A}_{21}\mathbf{A}_{12})$, one can evaluate the heads by traversing the network's spanning tree starting from each fixed head node and applying Eq. (1).

Newton-Raphson Loop Flows Algorithm

The NR method can also be applied to the so-called loop representation of the WDN problem as formulated in Eq. (23). First, one rearranges Eq. (23) as follows:

$$\mathbf{M}_{31}\mathbf{A}_{11}\mathbf{M}_{13}\mathbf{Q}_l + \mathbf{M}_{31}(\mathbf{A}_{10}\mathbf{H}_0 + \mathbf{A}_{11}\bar{\mathbf{Q}}) = \mathbf{f}_1 \tag{48}$$

where $\mathbf{f}_1 = \mathbf{f}_1(\mathbf{Q}_l)$ represents how far from zero the relevant equations are.

Applying the NR formula to this set of equations results in

$$\mathbf{M}_{31}\mathbf{D}_{11}^{\tau}\mathbf{M}_{13}\mathbf{dQ}_l = \mathbf{f}_1^{\tau} \tag{49}$$

where

$$\mathbf{dQ}_l = \mathbf{Q}_l^{\tau} - \mathbf{Q}_l^{\tau+1}$$

$$\mathbf{f}_1^{\tau} = \mathbf{M}_{31}\mathbf{A}_{11}^{\tau}\mathbf{M}_{13}\mathbf{Q}_l^{\tau} + \mathbf{M}_{31}(\mathbf{A}_{10}\mathbf{H}_0 + \mathbf{A}_{11}^{\tau}\bar{\mathbf{Q}}) \tag{50}$$

and $\mathbf{D}_{11}$ is the Jacobian of $\mathbf{A}_{11}\mathbf{M}_{13}\mathbf{Q}_l$. Solving Eq. (49) for $\mathbf{dQ}_l$ gives

$$\mathbf{dQ}_l = (\mathbf{M}_{31}\mathbf{D}_{11}^{\tau}\mathbf{M}_{13})^{-1}\mathbf{f}_1^{\tau} \tag{51}$$

Substituting for $\mathbf{f}_1^{\tau}$ from Eq. (50) leads directly to the following iterative NR-LF algorithm:

1. Solve the $[l,l]$ system of equations

$$\mathbf{M}_{31}\mathbf{D}_{11}^{\tau}\mathbf{M}_{13}\mathbf{dQ}_l = \mathbf{M}_{31}\mathbf{A}_{11}\mathbf{M}_{13}\mathbf{Q}_l + \mathbf{M}_{31}(\mathbf{A}_{10}\mathbf{H}_0 + \mathbf{A}_{11}\bar{\mathbf{Q}}) \tag{52}$$

for $\mathbf{dQ}_l$ where $\mathbf{D}_{11}^{\tau}$ is $\mathbf{D}_{11}$ evaluated for the current flows $\mathbf{Q}^{\tau}$.

2. Update the loop chord flows by evaluating

$$\mathbf{Q}_l^{\tau+1} = \mathbf{Q}_l^{\tau} - \mathbf{dQ}_l \tag{53}$$

3. Use Eq. (14) to update the full set of PFs

$$\mathbf{Q}^{\tau+1} = \bar{\mathbf{Q}} + \mathbf{M}_{13}\mathbf{Q}_l^{\tau+1} \tag{54}$$

The problem is thus reduced to the iterative solution of a sparse and symmetric system of linear equations of size $[l,l]$. Once the final flows are found, the final heads can be found using the same procedure as for the NR-PF algorithm.

Newton-Raphson Loop-Node Algorithm

The NR method will now be applied to the LN representation of Eq. (31), whose resulting algorithm is known as the hybrid method. As before, one begins by rewriting the representation in the following format:

$$\begin{bmatrix} -\mathbf{A}_{nn}\mathbf{A}_{2n}^{-1}\mathbf{A}_{2l} & \vdots & \mathbf{A}_{n2} \\ \cdots & \cdots & \cdots \\ \mathbf{A}_{ll} & \vdots & \mathbf{A}_{l2} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_l \\ \cdots \\ \mathbf{H} \end{bmatrix} - \begin{bmatrix} \mathbf{A}_{nn}\mathbf{A}_{2n}^{-1}\mathbf{q} - \mathbf{A}_{n0}\mathbf{H}_0 \\ \cdots \\ -\mathbf{A}_{l0}\mathbf{H}_0 \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1 \\ \cdots \\ \mathbf{f}_2 \end{bmatrix} \tag{55}$$

where $\mathbf{f}_1 = \mathbf{f}_1(\mathbf{Q},\mathbf{H})$ and $\mathbf{f}_2 = \mathbf{f}_2(\mathbf{Q},\mathbf{H})$ indicate how far from zero the relevant equations are, for any given approximated solution $\mathbf{Q}$, $\mathbf{H}$.

Applying the NR formula to this set of equations results in

$$\begin{bmatrix} -\mathbf{D}_{nn}^{\tau}\mathbf{A}_{2n}^{-1}\mathbf{A}_{2l} & \vdots & \mathbf{A}_{n2} \\ \cdots & \cdots & \cdots \\ \mathbf{D}_{ll}^{\tau} & \vdots & \mathbf{A}_{l2} \end{bmatrix} \begin{bmatrix} \mathbf{dQ}_l \\ \cdots \\ \mathbf{dH} \end{bmatrix} = \begin{bmatrix} \mathbf{f}_1^{\tau} \\ \cdots \\ \mathbf{f}_2^{\tau} \end{bmatrix} \tag{56}$$

where $\mathbf{D}_{nn}$ is the Jacobian of that portion of $\mathbf{A}_{11}\mathbf{Q}$ that corresponds to the pipes included in $\mathbf{A}_{nn}$ while $\mathbf{D}_{ll}$ is the Jacobian for the pipes included in $\mathbf{A}_{ll}$ and

$$\mathbf{dQ}_l = \mathbf{Q}_l^{\tau} - \mathbf{Q}_l^{\tau+1}$$

$$\mathbf{dH} = \mathbf{H}^{\tau} - \mathbf{H}^{\tau+1}$$

$$\mathbf{f}_1^{\tau} = -\mathbf{A}_{nn}^{\tau}\mathbf{A}_{2n}^{-1}(\mathbf{A}_{2l}\mathbf{Q}_l^{\tau} + \mathbf{q}) + \mathbf{A}_{n2}\mathbf{H}^{\tau} + \mathbf{A}_{n0}\mathbf{H}_0 \tag{57}$$

$$\mathbf{f}_2^{\tau} = \mathbf{A}_{ll}^{\tau}\mathbf{Q}_l^{\tau} + \mathbf{A}_{l2}\mathbf{H}^{\tau} + \mathbf{A}_{l0}\mathbf{H}_0 \tag{58}$$

Following the derivation in Appendix, the resulting iterative steps of the NR-LN algorithm are

1. Solve the following $[n;n]$ system of equations for new heads $\mathbf{H}^{\tau+1}$:

$$[\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{12}]\mathbf{H}^{\tau+1} = -\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}[\mathbf{A}_{11}^{\tau}\mathbf{Q}^{\tau} + \mathbf{A}_{10}\mathbf{H}_0] \tag{59}$$

where $\mathbf{D}_{11}^{\tau}$ is $\mathbf{D}_{11}$ evaluated for the current flows $\mathbf{Q}^{\tau}$.

2. Find new loop chord flows by evaluating the following set of scalar equations:

$$\mathbf{Q}_l^{\tau+1} = \mathbf{Q}_l^{\tau} - (\mathbf{D}_{ll}^{\tau})^{-1}[\mathbf{A}_{ll}^{\tau}\mathbf{Q}_l^{\tau} + (\mathbf{A}_{l2}\mathbf{H}^{\tau+1} + \mathbf{A}_{l0}\mathbf{H}_0)] \tag{60}$$

3. Use Eq. (14) to update the full set of pipe flows

$$\mathbf{Q}^{\tau+1} = \bar{\mathbf{Q}} + \mathbf{M}_{13}\mathbf{Q}_l^{\tau+1} \tag{61}$$

In this algorithm the independent unknowns are represented by the l unknown flows $\mathbf{Q}_l$ plus the n nodal heads $\mathbf{H}$ for a total of $l+n$ unknowns, but with the problem reduced to the iterative solution of a sparse and symmetric system of linear equations of size $[n;n]$.

Newton-Raphson Nodal Head Algorithms

The NH representation of the WDN problem, Eq. (32), can be rewritten as

J. Hydraul. Eng., 2013, 139(5): 511-526

$$\mathbf{A}_{21}\hat{\mathbf{A}}_{11}\mathbf{A}_{12}\mathbf{H} - \mathbf{q} + \mathbf{A}_{21}\hat{\mathbf{A}}_{11}\mathbf{A}_{10}\mathbf{H}_0 = \mathbf{f}_1 \quad (62)$$

where $\mathbf{f}_1 = \mathbf{f}_1(\mathbf{H})$ indicates how far from zero the relevant equations are, for any given approximated solution $\mathbf{H}$. Applying the NR formula to this set of equations yields

$$\mathbf{A}_{21}\hat{\mathbf{D}}_{11}^{\tau}\mathbf{A}_{12}\mathbf{dH} = \mathbf{f}_1 \quad (63)$$

where

$$\mathbf{dH} = \mathbf{H}^{\tau} - \mathbf{H}^{\tau+1}$$

$$\mathbf{f}_1 = \mathbf{A}_{21}\hat{\mathbf{A}}_{11}^{\tau}(\mathbf{A}_{12}\mathbf{H}^{\tau} + \mathbf{A}_{10}\mathbf{H}_0) - \mathbf{q}$$

and $\hat{\mathbf{D}}_{11}$, the Jacobian of $\hat{\mathbf{A}}_{11}\mathbf{H}$, is a $[p;p]$ diagonal matrix the elements of which are defined for $k \in 1, p$ as

$$\hat{\mathbf{D}}_{11}(k,k) = \text{diag}\left\{\frac{\partial\Psi(H_k)}{\partial H_k}\right\} = \text{diag}\left\{\frac{r_k^{-1/\alpha}}{\alpha}|\Delta H_k|^{\frac{1-\alpha}{\alpha}}\right\}$$

By direct substitution into Eq. (62), the following iterative scheme can be used to solve the NR-NH formulation:

1. Solve the following $[n;n]$ system of equations for head changes $\mathbf{dH}$:

$$\mathbf{A}_{21}\hat{\mathbf{D}}_{11}^{\tau}\mathbf{A}_{12}\mathbf{dH} = \mathbf{A}_{21}\hat{\mathbf{A}}_{11}^{\tau}(\mathbf{A}_{12}\mathbf{H}^{\tau} + \mathbf{A}_{10}\mathbf{H}_0) - \mathbf{q} \quad (64)$$

   where $\hat{\mathbf{D}}_{11}^{\tau}$ is $\hat{\mathbf{D}}_{11}$ evaluated at the current heads $\mathbf{H}^{\tau}$.
2. Update the nodal heads

$$\mathbf{H}^{\tau+1} = \mathbf{H}^{\tau} - \mathbf{dH} \quad (65)$$

Once convergence is reached, the flow vector $\mathbf{Q}$ can be obtained by means of Eq. (32).

Several authors (Shamir and Howard 1968; Lam and Wolla 1972a, b; Lemieux 1972; Donachie 1974; Mignosa 1987) have discussed the convergence problems exhibited by the NH-NR algorithm. The problems are due to the type of nonlinearity of the equations to be solved, which are of the square root type when the equations are expressed in terms of nodal heads [Eq. (32)], as opposed to almost quadratic when expressed in terms of pipe flows [Eq. (3)]. While the application of the NR algorithm to functions characterized by exponents $\alpha \geq 1$, as in the previous approaches, guarantees a rapid one-sided convergence (Fig. 3), the same thing is not guaranteed when the exponent is $1/\alpha < 1$. This type of nonlinearity may induce a slow alternating convergence when using the NR algorithm (see Fig. 3).

This is why the previously mentioned authors proposed the inclusion of an under-relaxation coefficient $0 \leq \vartheta \leq 1$, which tends to stabilize the convergence of the NR-NH approach

$$\mathbf{H}^{\tau+1} \leftarrow (1-\vartheta)\mathbf{H}^{\tau} + \vartheta\mathbf{H}^{\tau+1} \quad (66)$$

This has given rise to an additional approach, which will be called NR-NH with under-relaxation (NR-NU). Both algorithms have been tested and compared in this work.

Derivation of the Different LT Algorithms

The linear theory (LT) algorithms are predicated on being able to represent a system of nonlinear equations $\mathbf{f}(\mathbf{x}) = 0$ in the form $\mathbf{x} - \mathbf{g}(\mathbf{x}) = 0$, which gives rise to the recursive formula

$$\mathbf{x}^{\tau+1} = (1-\vartheta)\mathbf{x}^{\tau} + \vartheta\mathbf{g}(\mathbf{x}^{\tau}) \quad (67)$$

where $\mathbf{x}^{\tau}$ is the value of $\mathbf{x}$ at iteration τ and ϑ is an under-relaxation parameter chosen to improve convergence. In a one-dimensional model, this approach is known as the method of successive approximations (Carnahan et al. 1969).

Alternatively, for systems $\mathbf{f}(\mathbf{x},\mathbf{y}) = 0$ containing two sets of variables $\mathbf{x}$ and $\mathbf{y}$ that can be arranged as

$$\mathbf{x} - \mathbf{g}(\mathbf{y}) = 0 \qquad \mathbf{y} - \mathbf{h}(\mathbf{x}) = 0$$

an iterative solution can be found by evaluating

$$\mathbf{x}^{\tau+1} = \mathbf{g}(\mathbf{y}^{\tau}) \qquad \mathbf{y}^{\tau+1} = (1-\vartheta)\mathbf{y}^{\tau} + \vartheta\mathbf{h}(\mathbf{x}^{\tau+1}) \quad (68)$$

Linear Theory Global Algorithm

The LT technique can be applied to the GA representation of the WDN problem as stated in Eq. (16) by recasting it into a form comparable to Eq. (68). Beginning with the system

$$\mathbf{A}_{11}\mathbf{Q} + \mathbf{A}_{12}\mathbf{H} = -\mathbf{A}_{10}\mathbf{H}_0 \quad (69)$$

$$\mathbf{A}_{21}\mathbf{Q} = -\mathbf{q} \quad (70)$$

one can solve Eq. (69) for $\mathbf{Q}$

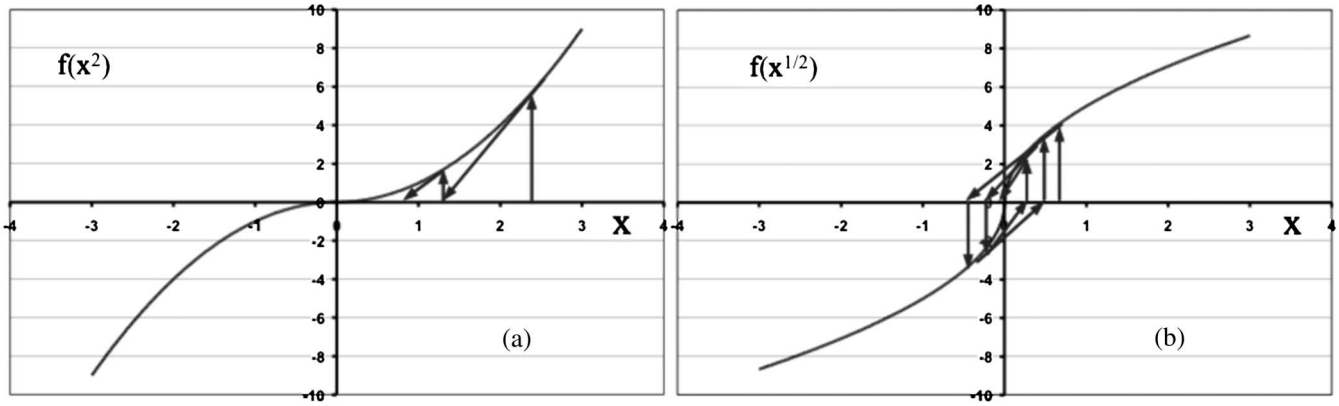

Fig. 3. Convergence behavior of the Newton-Raphson algorithm: (a) One-sided convergence: when the function's exponent is larger than one, the convexity enhances the performance of the NR algorithm resulting in a high rate of convergence; (b) alternating convergence: when the exponent is smaller than one, the convexity reduces the performance of the NR algorithm resulting in a low rate of convergence

$$\mathbf{Q} = -(\mathbf{A}_{11})^{-1}[\mathbf{A}_{12}\mathbf{H} + \mathbf{A}_{10}\mathbf{H}_0] \tag{71}$$

Substituting for $\mathbf{Q}$ in Eq. (70) gives

$$-\mathbf{A}_{21}(\mathbf{A}_{11})^{-1}\mathbf{A}_{12}\mathbf{H} - \mathbf{A}_{21}(\mathbf{A}_{11})^{-1}\mathbf{A}_{10}\mathbf{H}_0 = -\mathbf{q} \tag{72}$$

This leads to the following iterative algorithm for solving LT-GA:

1. Solve the $[n;n]$ system of equations for $\mathbf{H}^{\tau+1}$

$$[\mathbf{A}_{21}(\mathbf{A}_{11}^{\tau})^{-1}\mathbf{A}_{12}]\mathbf{H}^{\tau+1} = [\mathbf{q} - \mathbf{A}_{21}(\mathbf{A}_{11}^{\tau})^{-1}\mathbf{A}_{10}\mathbf{H}_0] \tag{73}$$

   using the flows at iteration τ to evaluate $\mathbf{A}_{11}$.
2. Update flows by evaluating Eq. (71) and averaging it with the previous estimates

$$\mathbf{Q}^{\tau+1} = \frac{1}{2}[\mathbf{Q}^{\tau} - (\mathbf{A}_{11}^{\tau})^{-1}(\mathbf{A}_{12}\mathbf{H}^{\tau+1} + \mathbf{A}_{10}\mathbf{H}_0)] \tag{74}$$

Each step only involves the solution of an $[n;n]$ sparse system of linear equations, because the inversion of matrix $\mathbf{A}_{11}$ is trivial, given that it is a diagonal matrix. This algorithm, although differently derived, corresponds in practice to the nodal LT introduced by Isaacs and Mills (1980).

Linear Theory Pipe Flows Algorithm

The PF formulation given by Eq. (20)

$$\begin{bmatrix} \mathbf{M}_{31}\mathbf{A}_{11} \\ \cdots \\ \mathbf{A}_{21} \end{bmatrix}\mathbf{Q} = \begin{bmatrix} -\mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0 \\ \cdots \\ -\mathbf{q} \end{bmatrix} \tag{75}$$

can be put in the form of Eq. (67) by solving for $\mathbf{Q}$ as follows:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{M}_{31}\mathbf{A}_{11} \\ \cdots \\ \mathbf{A}_{21} \end{bmatrix}^{-1}\begin{bmatrix} -\mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0 \\ \cdots \\ -\mathbf{q} \end{bmatrix} \tag{76}$$

This leads directly to the following iterative algorithm for solving LT-PF:

1. Solve the $[p;p]$ system of equations for $\mathbf{Q}$

$$\begin{bmatrix} \mathbf{M}_{31}\mathbf{A}_{11} \\ \cdots \\ \mathbf{A}_{21} \end{bmatrix}\mathbf{Q} = \begin{bmatrix} -\mathbf{M}_{31}\mathbf{A}_{10}\mathbf{H}_0 \\ \cdots \\ -\mathbf{q} \end{bmatrix} \tag{77}$$

   using the flows at iteration τ to evaluate $\mathbf{A}_{11}$.
2. Average these flows with the previous ones to arrive at a new set of flows

$$\mathbf{Q}^{\tau+1} = \frac{1}{2}(\mathbf{Q}^{\tau} + \mathbf{Q}) \tag{78}$$

After final convergence for flow has been reached, the same procedure as used with the NR version of the algorithm can be used to compute nodal heads $\mathbf{H}$.

As one can see, the original problem of $p+n$ unknowns, namely the pipe flows $\mathbf{Q}$ together with the nodal heads $\mathbf{H}$, is now transformed into the solution of a problem with p unknown flows, which requires the iterative solution of a sparse but nonsymmetric system of linear equations of size $[p;p]$. This algorithm corresponds to the original LT solution introduced by Wood and Charles (1972).

Linear Theory Loop Flow Algorithm

Similar to how the LT-PF algorithm was derived, the Loop Flow representation of the WDN problem given by Eq. (21) can be rearranged to place the unknown loop flow variables $\mathbf{Q}_l$ alone on the left-hand side of the equation thus making it compatible with the format of Eq. (67)

$$\mathbf{Q}_l = -(\mathbf{M}_{31}\mathbf{A}_{11}\mathbf{M}_{13})^{-1}\mathbf{M}_{31}(\mathbf{A}_{10}\mathbf{H}_0 + \mathbf{A}_{11}\bar{\mathbf{Q}}) \tag{79}$$

This leads to the following iterative algorithm for solving LT-LF:

1. Solve the $[l,l]$ system of equations for $\mathbf{Q}_l$

$$(\mathbf{M}_{31}\mathbf{A}_{11}^{\tau}\mathbf{M}_{13})\mathbf{Q}_l = -\mathbf{M}_{31}(\mathbf{A}_{10}\mathbf{H}_0 + \mathbf{A}_{11}\bar{\mathbf{Q}}) \tag{80}$$

   using the flows at iteration τ to evaluate $\mathbf{A}_{11}$.
2. Average these flows with the previous ones to arrive at a new set of LFs

$$\mathbf{Q}_l^{\tau+1} = \frac{1}{2}(\mathbf{Q}_l^{\tau} + \mathbf{Q}_l) \tag{81}$$

3. Use Eq. (14) to update the full set of PFs

$$\mathbf{Q}^{\tau+1} = \bar{\mathbf{Q}} + \mathbf{M}_{13}\mathbf{Q}_l^{\tau+1} \tag{82}$$

The problem is again reduced to the iterative solution of a sparse and symmetric system of linear equations of size $[l,l]$. The same procedure used with the loop NR algorithm to compute the nodal heads can also be used here after a final set of flows has been found.

Linear Theory Loop-Node Algorithm

The LN formulation of Eq. (31) can be put into a form suitable for solution via LT by applying the steps described in Appendix. The resulting iterative algorithm proceeds as follows:

1. Using the current set of pipe flows $\mathbf{Q}^{\tau}$ to evaluate the entries of $\mathbf{A}_{11}^{\tau}$, solve the following set of $[n;n]$ equations for new heads $\mathbf{H}^{\tau+1}$:

$$[\mathbf{A}_{21}(\mathbf{A}_{11}^{\tau})^{-1}\mathbf{A}_{12}]\mathbf{H}^{\tau+1} = [\mathbf{q} - \mathbf{A}_{21}(\mathbf{A}_{11}^{\tau})^{-1}\mathbf{A}_{10}\mathbf{H}_0] \tag{83}$$

2. Use the new heads and the previous LFs $\mathbf{Q}_l^{\tau}$ to update the latter by evaluating the set of scalar equations

$$\mathbf{Q}_l^{\tau+1} = \frac{1}{2}[\mathbf{Q}_l^{\tau} - (\mathbf{A}_{ll}^{\tau})^{-1}(\mathbf{A}_{l2}\mathbf{H}^{\tau+1} + \mathbf{A}_{l0}\mathbf{H}_0)] \tag{84}$$

3. Use Eq. (14) to update the full set of pipe flows

$$\mathbf{Q}^{\tau+1} = \bar{\mathbf{Q}} + \mathbf{M}_{13}\mathbf{Q}_l^{\tau+1} \tag{85}$$

In this case, the unknowns are the l unknown flows $\mathbf{Q}_l$ plus the n NHs $\mathbf{H}$ for a total of $l+n$ unknowns, but the problem is finally reduced to the iterative solution of a sparse and symmetric system of linear equations of size $[n;n]$. A somewhat similar algorithm has been described by Gupta and Prasad (2000), where the weighted flow updating of Eq. (84) is done in log space.

Comparison of the Solution Matrices for NR and LT Algorithms

Each of the NR and LT solution algorithms repeatedly solves a linear system of equations whose coefficient matrix is derived from the current estimate of pipe flows (or heads for NR-NH). Table 3 compares the structure of the solution matrix used with each algorithm. Notice the similarities between these matrices. In particular

- The global and LN matrices are the same for both the NR and LT algorithms.
- The coefficients of the global and LN matrices contain $|Q|^{-(\alpha-1)}$ terms in them while the coefficients of the PF and LF formulations contain $|Q|^{\alpha-1}$ terms.

J. Hydraul. Eng., 2013, 139(5): 511-526

Table 3. Solution Matrices Used in the Various WDN Algorithms

Representation	Newton-Raphson	Linear theory
Global algorithm	$\mathbf{A}_{21}\mathbf{D}_{11}^{-1}\mathbf{A}_{12}$	$\mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}$
Pipe flows	$\begin{bmatrix}\mathbf{M}_{31}\mathbf{D}_{11}\\ \cdots \\ \mathbf{A}_{21}\end{bmatrix}$	$\begin{bmatrix}\mathbf{M}_{31}\mathbf{A}_{11}\\ \cdots \\ \mathbf{A}_{21}\end{bmatrix}$
Loop flows	$\mathbf{M}_{31}\mathbf{D}_{11}\mathbf{M}_{13}$	$\mathbf{M}_{31}\mathbf{A}_{11}\mathbf{M}_{13}$
Loop-node	$\mathbf{A}_{21}\mathbf{D}_{11}^{-1}\mathbf{A}_{12}$	$\mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12}$
Nodal head	$\mathbf{A}_{21}\hat{\mathbf{D}}_{11}\mathbf{A}_{12}$	—

- The diagonal matrix $\mathbf{D}_{11}$ appearing in the NR algorithms differs from the diagonal matrix $\mathbf{A}_{11}$ used in the LT methods only by a factor of α (the exponent in the head loss equation).

Convergence Properties of the Different Algorithms

Description of the Test Network

The convergence properties of the different algorithms were tested using the simple example network sketched in Fig. 4. The hydraulic characteristics of the network are summarized in Tables 4 and 5,

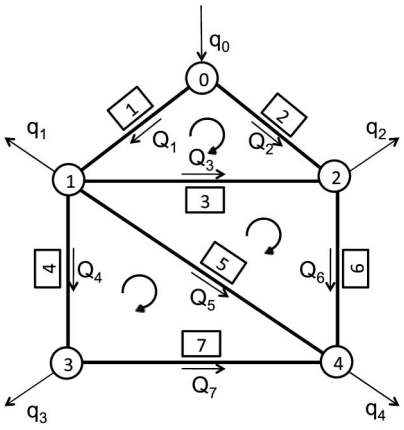

Fig. 4. Schematic representation of the example network with Node 0 being known head

Table 4. Pipe Characteristics

Pipe	True flow ($\mathrm{L\,s^{-1}}$)	r (Hazen-Williams)
1	80	0.00029886657
2	20	0.00778973070
3	10	0.01406047524
4	40	0.00215781840
5	20	0.01168459605
6	10	0.02812095048
7	10	0.01406047524

Table 5. Node Characteristics

Node	q ($\mathrm{L\,s^{-1}}$)	H (m)
0	100	100
1	−10	99
2	−20	98
3	−30	97
4	−40	96

where the sign of demands is consistent to the chosen convention: the flow is positive if it enters and negative if it leaves a node. The Hazen-Williams formula was used to estimate the head losses, with the resistance constant of each pipe given by

$$r = \frac{10.67L}{C^{1.852}D^{4.871}} \tag{86}$$

where L = pipe length (m), D = pipe diameter (m), and C = roughness factor. In order to minimize information, only the Hazen-Williams coefficient r resulting from Eq. is given in Table 4.

Two tests were carried out using this example network. The first one was designed to assess the convergence rate of the different algorithms based on the number of iterations needed to solve the network when the required accuracy was set to 10^{-12}. Accuracy was defined as the maximum absolute deviation between the computed and true flows in $1\,\mathrm{s^{-1}}$. The second test was intended to measure the sensitivity of the convergence rate to the boundary conditions (the demands) imposed. It was run by solving the example network 100 times using perturbations on nodal demands with a uniform random variation of $\pm 50[1\,\mathrm{s^{-1}}]$. This statistical analysis was performed twice by setting the required accuracy to 10^{-2} and to 10^{-6} in order to compare both low and high accuracy requirements.

Each algorithm was begun with the same initial set of conditions that were selected as follows:

1. $\mathbf{A}_{11}^0$ was computed using a flow of 1 ($1\,\mathrm{s^{-1}}$) in each pipe

$$\mathbf{A}_{11}^0 = \mathrm{diag}\left\{\frac{\Phi(Q_k)}{Q_k}\bigg|_{Q_{k=1}}\right\} \tag{87}$$

2. Initial heads $\mathbf{H}^0$ were computed by using this value of $\mathbf{A}_{11}^0$ in

$$\mathbf{H}^0 = [\mathbf{A}_{21}(\mathbf{A}_{11}^0)^{-1}\mathbf{A}_{12}]^{-1}[\mathbf{q} - \mathbf{A}_{21}(\mathbf{A}_{11}^0)^{-1}\mathbf{A}_{10}\mathbf{H}_0] \tag{88}$$

3. Initial flows $\mathbf{Q}^0$ were computed by using $\mathbf{A}_{11}^0$ and $\mathbf{H}^0$ in

$$\mathbf{Q}^0 = \mathbf{Q} = -(\mathbf{A}_{11}^0)^{-1}(\mathbf{A}_{12}\mathbf{H}^0 + \mathbf{A}_{10}\mathbf{H}_0) \tag{89}$$

Discussion of Results

Results from the first test of the different algorithms are shown in Fig. 5, which plots the logarithm of the maximum error in terms of discharges (in $1\,\mathrm{s^{-1}}$) as a function of the number of iterations.

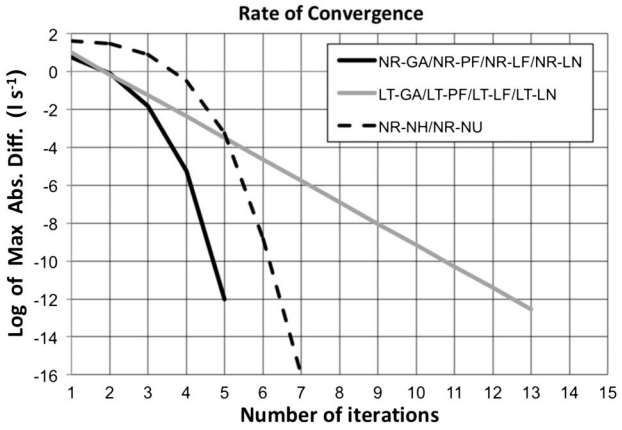

Fig. 5. Convergence rate for the various algorithms; note the identical convergence behavior of the linearly related algorithms (GA, PF, LF, and LN), which is due to the fact that the convergence properties of both NR and LT are not altered by a linear transformation of the original system

The results appear in groups that show exactly the same behavior. One group includes all the NR formulations derived by linear transformation from the GA; a second group encompasses all the LT formulations derived by linear transformation from the GA; and a third group contains the NR formulations derived by nonlinear transformation of the GA. Please note that both NR and LT formulations are totally transparent to linear transformations, which results in the fact that all the linearly related approaches (GA, PF, LF, LN) perform exactly the same showing an identical rate of convergence.

If one just compares the number of iterations required to meet the final accuracy target of 10^{-12} (which was considered sufficiently close to floating-point double precision accuracy 10^{-16}) it appears that the NR linearly transformed group requires the fewest iterations [Eq. (5)], followed by the NR nonlinearly transformed group [Eq. (7)], and then the LT group [Eq. (13)]. However, a closer look at these results reveals that if the required accuracy is not too high (10^{-2} to 10^{-3} $1s^{-1}$ is a more than adequate accuracy when dealing with practical problems), the LT group shows convergence rates which are similar to the ones of the linearly transformed NR approaches. Nevertheless, it is possible that this result on the convergence rate of the LT group could be affected by the inclusion of pumps, valves, etc. or by the extension of the algorithms to head driven problems.

While it might be somewhat surprising that the convergence rates for the different linearly transformed formulations differ only due to the method of solution and not by problem formulation, one can presume that this results from them all being derived by applying a sequence of linearly transformation steps to the same initial formulation.

The results of the second set of experiments that compare the convergence rates of the different algorithms on the test network for 100 different sets of randomly generated demand conditions are synthesized in a series of histograms shown in Fig. 6. The histograms on the top row of Fig. 6 are for an accuracy 10^{-2} while those on the bottom row are for 10^{-6}. The histograms in the left column of Figs. 6(a and c) only compare the linearly transformed formulations while those in the right column of Figs. 6(b and d) include

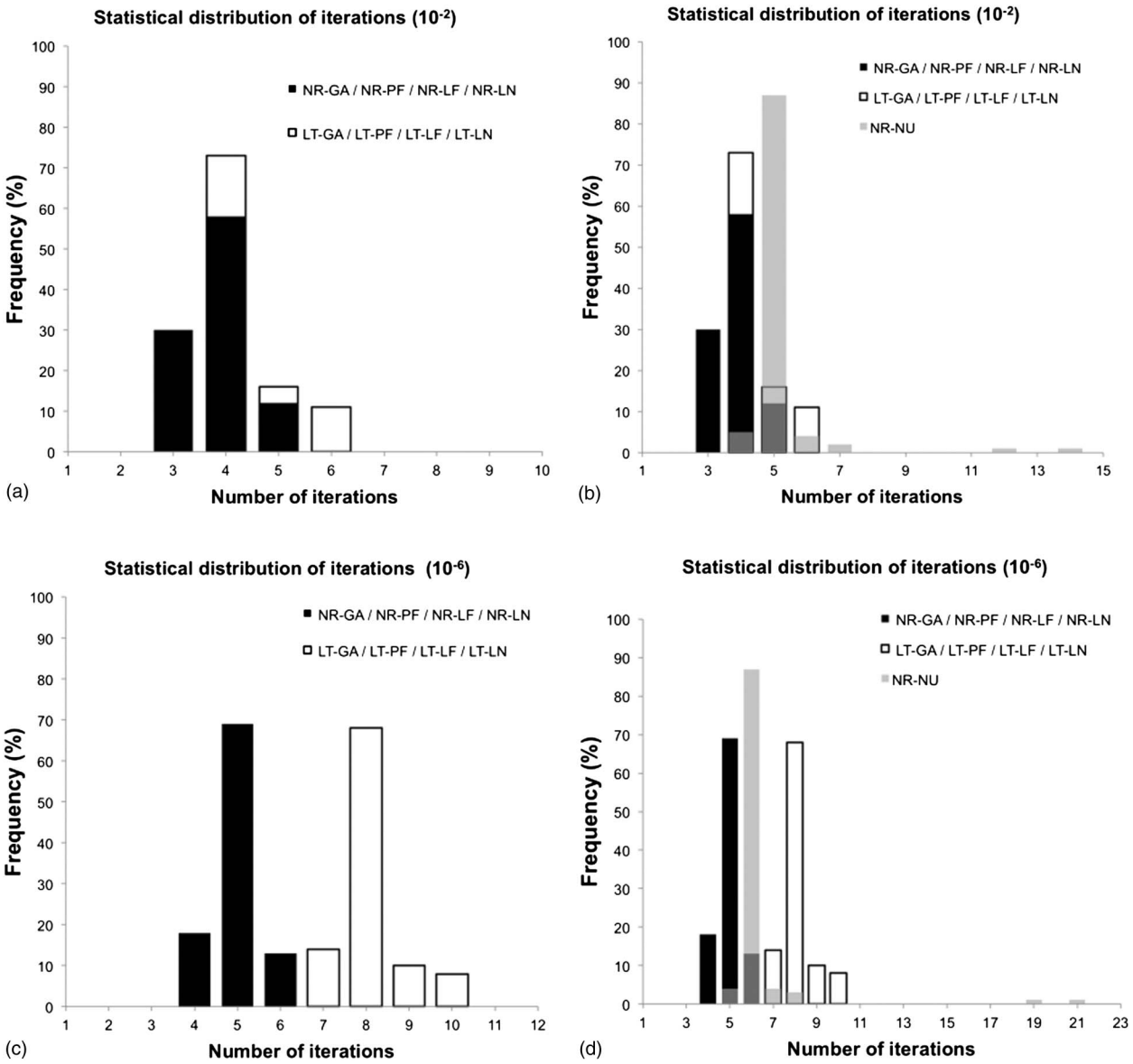

Fig. 6. Histograms of the convergence rates of the different algorithms: (a) NR and LT linearly transformed methods with 10^{-2} accuracy; (b) all methods with 10^{-2} accuracy; (c) NR and LT linearly transformed methods with 10^{-6} accuracy; (d) all methods with 10^{-6} accuracy

J. Hydraul. Eng., 2013, 139(5): 511-526

the nonlinearly transformed NR methods as well (requiring an expanded scale to accommodate the larger spread in iterations for this method).

While Fig. 6(c) shows that on average more iterations are needed when using the LT instead of the NR approaches, from Fig. 6(a) it is clear that the behavior of LT at low accuracy requirements is almost indistinguishable from that of the NR group. From Figs. 6(b and d), where the results for the NR-NO approach are also shown, one notices that LT at low accuracy requirements performs better than NR applied to the NHs, which not only on the average requires a higher number of iterations, but also suffers a wider spread.

Fig. 7 shows the beneficial effect of the under-relaxation factor in the case of the NR method applied to the NHs formulation. One can see that the factor helps reduce the spread of the required number of iterations independent from the accuracy requirements.

Finally Fig. 8 shows the mean and the range (min–max) of the iterations needed by the different approaches at the two accuracy levels. What appears evident from the figure is how risky the use of the NR approach based on the NH can be given that conditions may arise that require a very large number of iterations to converge. This happens when the flow in a pipe, although not necessarily close to zero, becomes substantially small so to produce the alternating convergence shown in Fig. 3(b). At the same time one can notice the small range shown by the other NR as well as the LT approaches. It is also apparent that the LT approaches can converge quickly for low accuracy requirements, while they tend to be slower at higher accuracy levels.

Suitability of the Different Algorithms for Operational Use

The main result found from the previous section is that, apart from the NH-NR approaches (NR-NH, NR-NU), all the linearly related NR algorithms (NR-GA, NR-PF, NR-LF, NR-LN) have exactly the same convergence rate. The same holds true for the linearly related LT algorithms as well (LT-GA, LT-PF, LT-LF, LT-LN). Thus, while convergence rate is certainly one aspect to consider, additional factors that affect computational efficiency must also be taken into account when evaluating the alternative algorithms. The most relevant of these factors include the following:

1. The size of the matrix that must be inverted (or factorized) to solve the system;
2. Whether the solution matrix is symmetric or not;
3. The density (number of nonzero elements) in the matrix;
4. The convergence rate of the algorithm;
5. Whether or not the algorithm requires a fundamental set of loops be identified; and
6. Whether or not the algorithm requires an initial set of balanced flows.

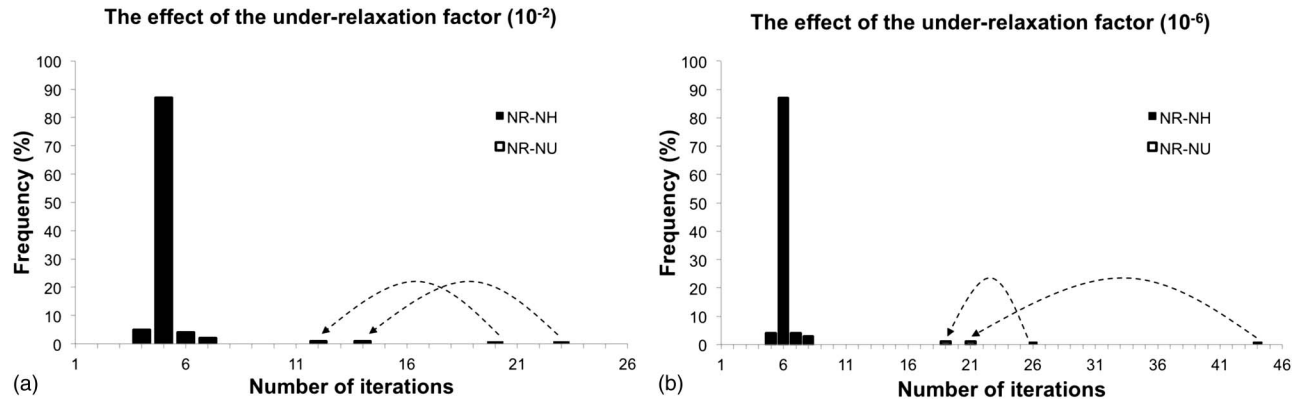

Fig. 7. Effect of the underrelaxation factor on the nonlinearly transformed nodal head formulation: (a) 10^{-2} accuracy; (b) 10^{-6} accuracy

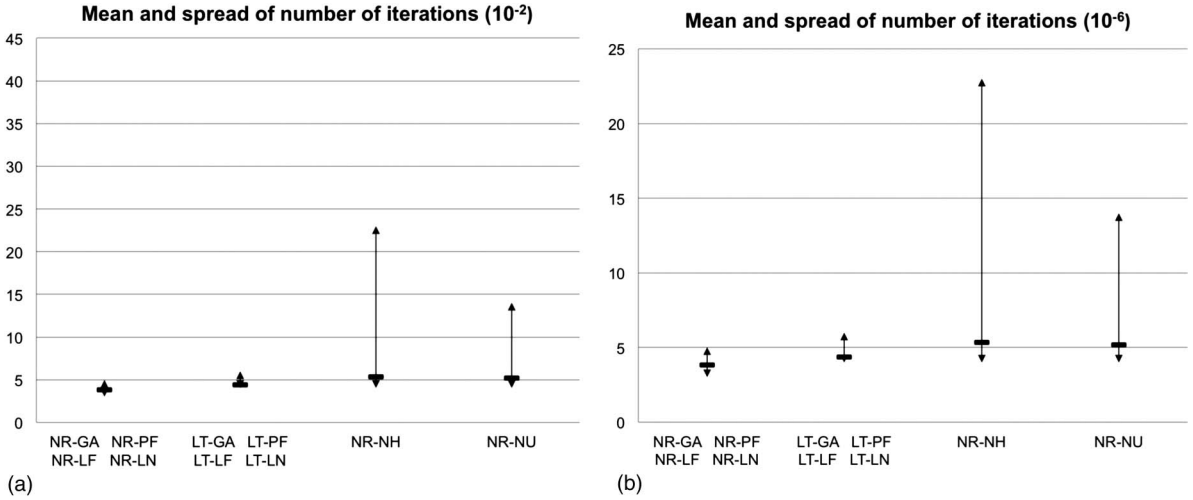

Fig. 8. Mean and spread of iterations required by the different solution methods: (a) 10^{-2} accuracy; (b) 10^{-6} accuracy

Table 6. Qualitative Comparison of Computational Factors for the Different WDN Algorithms

Algorithm	Unknowns	Matrix size	Matrix symmetric	Matrix density	Solution linear system	Nonlinear convergence rate	Loops required	Initial flows balanced
NR-GA	Q, H	$[n;n]$	Yes	B	A	A	No	No
NR-PF	Q	$[p;p]$	No	A	C	A	Yes	No
NR-LF	Q_l	$[l,l]$	Yes	C	A–B	A	Yes	Yes
NR-LN	Q_l, H	$[n;n]$	Yes	B	A	A	Yes	Yes
LT-GA	Q, H	$[n;n]$	Yes	B	A	B	No	No
LT-PF	Q	$[p;p]$	No	A	C	B	Yes	No
LT-LF	Q_l	$[l,l]$	Yes	C	A–B	B	Yes	Yes
LT-LN	Q_l, H	$[n;n]$	Yes	B	A	B	Yes	Yes
NR-NH	H	$[n;n]$	Yes	B	A	D	No	No
NR-NU	H	$[n;n]$	Yes	B	A	C	No	No

Note: A = best; D = worst.

Table 6 compares specific features of the different solution algorithms with respect to the impact they have on computational effort. A smaller size, sparse (low density) symmetric matrix can be factorized much faster than a nonsymmetric higher density matrix. This fact is reflected in the solution of linear system ranking, which refers to the speed (A = best, D = worst) to perform a single NR or LT iteration that requires the solution of a linear system. The nonlinear convergence rate refers to the number of iterations required by the algorithm to achieve a desired level of accuracy (as shown in Fig. 5). Additional comparisons are also provided in the columns named loops required (if one needs to first identify loops) and initial flows balanced (if one needs to provide a feasible set of initial flows). These requirements are particularly costly for extended period simulations where network topology can change due to the way that closed pipes and pressure reducing valves are handled, requiring a redefinition of loops and new estimates of initial flows.

From the standpoint of convergence, the NR algorithms, and in particular the linearly based ones, converge faster than the LT-based algorithms. However, the latter do perform better than the NH-NR approaches when the required accuracy is not too high. This suggests that the NH-NR approaches be avoided because under particular sets of loading conditions they may converge slowly, while none of the other approaches suffers this problem.

Among the four NR algorithms, the NR-GA stands out for the following reasons:

1. It is the only method that does not require the definition of loops, a subjective operation that may prove to be delicate and time consuming. This is why NR-GA is more suitable than NR-LN, which although solves the same $[n;n]$ system with a symmetric and sparse matrix, requires the definition of the loops to derive the reduced set of unknowns.
2. NR-GA solves an $[n;n]$ system with a symmetric and sparse matrix, as opposed to NR-PF, which results in a nonsymmetric $[p;p]$ system that has a much higher computational and time requirement.
3. Although the $[n;n]$ symmetric and sparse matrix used by NR-GA is larger than the $[l,l]$ symmetric and sparse matrix associated with NR-LF, the density of the NR-GA matrix is lower than the NR-LF matrix, as discussed by Osiadacz (1988) and Osiadacz and Pienkosz (1988), and is not dependent on the choice of loops. This leads to reduced computer load and computation time when using sparse matrix solvers that perform row reordering to reduce the amount of fill-in created when the matrix is factorized. The density of NR-LF is in fact heavily affected by the choice of loops, because the minimum density corresponds to a selection of loops that minimizes the number of loops adjacent to one another.

For exactly the same reasons, given that the resulting matrices are of the same structure, the LT-GA method would be the most preferred among the linearly related LT approaches.

Conclusions

This work sought to provide a unified development and comparison of the most widely available simultaneous equation algorithms for the analysis of looped WDNs. In order to be consistent with the original derivation of these algorithms, only the classic demand-driven WDN problem without special components such as pumps, valves was analyzed. As part of this process, several new LT algorithms were formulated and used to complement the existing ones. Four LT algorithms were identified as were four NR-based ones, all derived from different linear transformations applied to the GA formulation of the governing set of mass and energy conservation equations. All these algorithms produce exactly the same convergence rate, which depends only on the chosen solution approach (NR or LT). In addition to these algorithms, another approach was derived as a nonlinear transformation of the GA, which shows poorer convergence properties. Although additional tests will be required to extend the comparisons to account for pumps, valves, distributed withdrawals along pipes (Giustolisi 2010), and pressure-dependent demands, the authors believe that the results obtained are fairly general because they reflect the basic structure of the underlying equations. By comparing the different algorithms with respect to a number of computational factors, the NR-GA method (also known as the global gradient algorithm) and the LT-GA (for lower tolerance requirements) emerge as the most preferred.

Appendix. Derivations of Selected WDN Algorithms

NR-GA

Solving the upper part of Eq. (35) with respect to $\mathbf{dQ}$ gives

$$\mathbf{dQ} = (\mathbf{D}_{11}^{\tau})^{-1}(\mathbf{f}_1^{\tau} - \mathbf{A}_{12}\mathbf{dH}) \tag{90}$$

and then substituting it into the lower part, one obtains

$$\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}(\mathbf{f}_1^{\tau} - \mathbf{A}_{12}\mathbf{dH}) = \mathbf{f}_2^{\tau} \tag{91}$$

After some algebraic manipulation, the following result is reached:

$$\mathbf{dH} = [\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{12}]^{-1}[\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{f}_1^{\tau} - \mathbf{f}_2^{\tau}] \quad (92)$$

Substituting for $\mathbf{f}_1^{\tau}$ and $\mathbf{f}_2^{\tau}$ from Eqs. (38) and (39), respectively, into Eqs. (91) and (92), the iterative formulation of the NR-GA can be found

$$\mathbf{H}^{\tau+1} = [\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{12}]^{-1}\{\mathbf{A}_{21}[\mathbf{I}_{pp} - (\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{11}^{\tau}]\mathbf{Q}^{\tau} - \mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{10}\mathbf{H}_0 + \mathbf{q}\} \quad (93)$$

$$\mathbf{Q}^{\tau+1} = [\mathbf{I}_{pp} - (\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{11}^{\tau}]\mathbf{Q}^{\tau} - (\mathbf{D}_{11}^{\tau})^{-1}(\mathbf{A}_{12}\mathbf{H}^{\tau+1} + \mathbf{A}_{10}\mathbf{H}_0) \quad (94)$$

where $\mathbf{I}_{pp}$ = identity matrix of size $[p; p]$. This can be written more compactly by factoring out the $(\mathbf{D}_{11}^{\tau})^{-1}$ term to produce the final result

$$\mathbf{H}^{\tau+1} = [\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{12}]^{-1}\{\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}[(\mathbf{D}_{11}^{\tau} - \mathbf{A}_{11}^{\tau})\mathbf{Q}^{\tau} - \mathbf{A}_{10}\mathbf{H}_0] + \mathbf{q}\} \quad (95)$$

$$\mathbf{Q}^{\tau+1} = \mathbf{Q}^{\tau} - (\mathbf{D}_{11}^{\tau})^{-1}(\mathbf{A}_{11}^{\tau}\mathbf{Q}^{\boldsymbol{\tau}} + \mathbf{A}_{12}\mathbf{H}^{\tau+1} + \mathbf{A}_{10}\mathbf{H}_0) \quad (96)$$

These two sets of equations form the NR-GA.

NR-LN Algorithm

Premultiplying the upper part of Eq. (56) by $\mathbf{A}_{2n}\mathbf{D}_{nn}^{-1}$ gives

$$\begin{bmatrix} -\mathbf{A}_{2l} & \vdots & \mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{n2} \\ \cdots & \cdots & \cdots \\ \mathbf{D}_{ll}^{\tau} & \vdots & \mathbf{A}_{l2} \end{bmatrix} \begin{bmatrix} \mathbf{dQ}_l \\ \cdots \\ \mathbf{dH} \end{bmatrix} = \begin{bmatrix} \mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{f}_1^{\tau} \\ \cdots \\ \mathbf{f}_2^{\tau} \end{bmatrix} \quad (97)$$

Solving the lower part of this system of equations with respect to $\mathbf{dQ}_l$ gives

$$\mathbf{dQ}_l = -\mathbf{D}_{ll}^{-1}(\mathbf{A}_{l2}\mathbf{dH} - \mathbf{df}_2) \quad (98)$$

and substituting it into the upper part the following result is obtained:

$$\mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}(\mathbf{A}_{l2}\mathbf{dH} - \mathbf{f}_2^{\tau}) + \mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{n2}\mathbf{dH} = \mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{f}_1^{\tau} \quad (99)$$

Eq. (99) can also be rearranged as follows:

$$[\mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{n2} + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}\mathbf{A}_{l2}]\mathbf{dH} = \mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{f}_1^{\tau} + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}\mathbf{f}_2^{\tau} \quad (100)$$

Substituting into Eq. (100) for $\mathbf{f}_1^{\tau}$ and $\mathbf{f}_2^{\tau}$ from Eqs. (57) and (58), respectively, the following equation is obtained:

$$[\mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{n2} + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}\mathbf{A}_{l2}]\mathbf{dH} = \mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}[-\mathbf{A}_{nn}^{\tau}\mathbf{A}_{2n}^{-1}(\mathbf{A}_{2l}\mathbf{Q}_l^{\tau} + \mathbf{q}) + \mathbf{A}_{n2}\mathbf{H}^{\tau} + \mathbf{A}_{n0}\mathbf{H}_0] + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}[\mathbf{A}_{ll}^{\tau}\mathbf{Q}_l^{\tau} + \mathbf{A}_{l2}\mathbf{H}^{\tau} + \mathbf{A}_{l0}\mathbf{H}_0] \quad (101)$$

Bearing in mind the definition of $\mathbf{Q}_n^{\tau} = -\mathbf{A}_{2n}^{-1}(\mathbf{A}_{2l}\mathbf{Q}_l^{\tau} + \mathbf{q})$ given in Eq. (12), Eq. (101) can be rewritten as

$$[\mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{n2} + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}\mathbf{A}_{l2}]\mathbf{dH} = \mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{nn}^{\tau}\mathbf{Q}_n^{\tau} + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}\mathbf{A}_{ll}^{\tau}\mathbf{Q}_l^{\tau} + [\mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{n2} + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}\mathbf{A}_{l2}]\mathbf{H}^{\tau} + [\mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{n0} + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}\mathbf{A}_{l0}]\mathbf{H}_0 \quad (102)$$

Recognizing that

$$[\mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{n2} + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}\mathbf{A}_{l2}] = [\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{12}] \quad (103)$$

$$[\mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{n0} + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}\mathbf{A}_{l0}] = [\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{10}] \quad (104)$$

$$\mathbf{A}_{2n}(\mathbf{D}_{nn}^{\tau})^{-1}\mathbf{A}_{nn}^{\tau}\mathbf{Q}_n^{\tau} + \mathbf{A}_{2l}(\mathbf{D}_{ll}^{\tau})^{-1}\mathbf{A}_{ll}^{\tau}\mathbf{Q}_l^{\tau} = \mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{11}^{\tau}\mathbf{Q}^{\tau} \quad (105)$$

allows one to write

$$[\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{12}]\mathbf{dH} = [\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{11}^{\tau}]\mathbf{Q}^{\tau} + [\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{12}]\mathbf{H}^{\tau} + [\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{10}]\mathbf{H}_0 \quad (106)$$

Substituting $\mathbf{H}^{\tau} - \mathbf{H}^{\tau+1}$ for $\mathbf{dH}$ gives

$$\mathbf{H}^{\tau} - \mathbf{H}^{\tau+1} = \mathbf{H}^{\tau} + [\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{12}]^{-1}\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1} \times [\mathbf{A}_{11}^{\tau}\mathbf{Q}^{\tau} + \mathbf{A}_{10}\mathbf{H}_0] \quad (107)$$

Simplifying Eq. (107) leads to the final recursive algorithm for heads $\mathbf{H}$

$$\mathbf{H}^{\tau+1} = -[\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}\mathbf{A}_{12}]^{-1}\mathbf{A}_{21}(\mathbf{D}_{11}^{\tau})^{-1}[\mathbf{A}_{11}^{\tau}\mathbf{Q}^{\tau} + \mathbf{A}_{10}\mathbf{H}_0] \quad (108)$$

The updated flows can then be estimated by substituting for $\mathbf{dQ}_l$, $\mathbf{dH}$, and $\mathbf{f}_2$ into Eq. (108)

$$\mathbf{Q}_l^{\tau} - \mathbf{Q}_l^{\tau+1} = -(\mathbf{D}_{ll}^{\tau})^{-1}[\mathbf{A}_{l2}(\mathbf{H}^{\tau} - \mathbf{H}^{\tau+1}) - (\mathbf{A}_{ll}^{\tau}\mathbf{Q}_l^{\tau} + \mathbf{A}_{l2}\mathbf{H}^{\tau} + \mathbf{A}_{l0}\mathbf{H}_0)] \quad (109)$$

which can be simplified and rearranged to give

$$\mathbf{Q}_l^{\tau+1} = \mathbf{Q}_l^{\tau} - (\mathbf{D}_{ll}^{\tau})^{-1}[\mathbf{A}_{ll}^{\tau}\mathbf{Q}_l^{\tau} + (\mathbf{A}_{l2}\mathbf{H}^{\tau+1} + \mathbf{A}_{l0}\mathbf{H}_0)] \quad (110)$$

Eqs. (108) and (110) form the basis of the NR-LN algorithm.

LT-LN Algorithm

The LN formulation of Eq. (31) can be put into a form suitable for solution via LT by applying the following steps:

1. Premultiply the upper part of Eq. (31) by $\mathbf{A}_{2n}\mathbf{A}_{nn}^{-1}$ to obtain the following:

$$\begin{bmatrix} -\mathbf{A}_{2l} & \mathbf{A}_{2n}\mathbf{A}_{nn}^{-1}\mathbf{A}_{n2} \\ \mathbf{A}_{ll} & \mathbf{A}_{l2} \end{bmatrix} \begin{bmatrix} \mathbf{Q}_l \\ \mathbf{H} \end{bmatrix} = \begin{bmatrix} \mathbf{q} - \mathbf{A}_{2n}\mathbf{A}_{nn}^{-1}\mathbf{A}_{n0}\mathbf{H}_0 \\ -\mathbf{A}_{l0}\mathbf{H}_0 \end{bmatrix} \quad (111)$$

2. Solve the lower part with respect to $\mathbf{Q}_l$

$$\mathbf{Q}_l = -\mathbf{A}_{ll}^{-1}(\mathbf{A}_{l2}\mathbf{H} + \mathbf{A}_{l0}\mathbf{H}_0) \quad (112)$$

3. Substitute this into the upper part of Eq. (111)

$$\mathbf{A}_{2l}\mathbf{A}_{ll}^{-1}(\mathbf{A}_{l2}\mathbf{H} + \mathbf{A}_{l0}\mathbf{H}_0) + \mathbf{A}_{2n}\mathbf{A}_{nn}^{-1}\mathbf{A}_{n2}\mathbf{H} = \mathbf{q} - \mathbf{A}_{2n}\mathbf{A}_{nn}^{-1}\mathbf{A}_{n0}\mathbf{H}_0 \quad (113)$$

which can be rearranged as

$$(\mathbf{A}_{2l}\mathbf{A}_{ll}^{-1}\mathbf{A}_{l2} + \mathbf{A}_{2n}\mathbf{A}_{nn}^{-1}\mathbf{A}_{n2})\mathbf{H} = \mathbf{q} - (\mathbf{A}_{2l}\mathbf{A}_{ll}^{-1}\mathbf{A}_{l0} + \mathbf{A}_{2n}\mathbf{A}_{nn}^{-1}\mathbf{A}_{n0})\mathbf{H}_0 \quad (114)$$

and is identical to

$$(\mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12})\mathbf{H} = \mathbf{q} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{10}\mathbf{H}_0 \quad (115)$$

4. Isolate $\mathbf{H}$ on the left-hand side

$$\mathbf{H} = [(\mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{12})]^{-1}[\mathbf{q} - \mathbf{A}_{21}\mathbf{A}_{11}^{-1}\mathbf{A}_{10}\mathbf{H}_0] \quad (116)$$

5. Solve the lower part of Eq. (111) for $\mathbf{Q}_l$

$$\mathbf{Q}_l = (\mathbf{A}_{ll})^{-1}(\mathbf{A}_{l2}\mathbf{H} + \mathbf{A}_{l0}\mathbf{H}_0) \quad (117)$$

Eqs. (116) and (117) form the basis of the LT-LN algorithm.

Acknowledgments

A large portion of the programming and simulation work was carried out by Professor Todini during his Fellowship (October–December 2009) at the Advanced Study Institute of the University of Durham (U.K.). The U.S. Environmental Protection Agency, through its Office of Research and Development, collaborated in the research described herein. It has been subjected to the Agency's peer and administrative review and has been approved for external publication. Any opinions expressed are those of the authors and do not necessarily reflect the views of the Agency, therefore, no official endorsement should be inferred. Any mention of trade names or commercial products does not constitute endorsement or recommendation for use.

Notation

The following symbols are used in this paper:

$\mathbf{A}_{10} = \mathbf{A}_{01}^T = [p, n_0]$ incidence matrix relating pipes to known head nodes;
$\mathbf{A}_{11} = [p; p]$ diagonal matrix, whose generic term is $\mathbf{A}_{11}(k,k) = r_k|Q_k|^{\alpha_k - 1}$;
$\hat{\mathbf{A}}_{11} = [p; p]$ diagonal matrix, whose generic term is $\hat{\mathbf{A}}_{11}(k,k) = r_k^{-1/\alpha_k}|\Delta H_k|^{1-\alpha_k/\alpha_k}$;
$\mathbf{A}_{12} = \mathbf{A}_{21}^T = [p, n]$ incidence matrix relating pipes to unknown head nodes;
$\bar{\mathbf{A}}_{12} = \bar{\mathbf{A}}_{21}^T = [p, N]$ incidence matrix relating pipes to nodes;
$\mathbf{A}_{11}^0 = [p; p]$ diagonal matrix of initial guess for $\mathbf{A}_{11}$;
$\mathbf{A}_{l0} = \mathbf{A}_{0l}^T = [l, n_0]$ partition of $\mathbf{A}_{10} = \mathbf{A}_{01}^T$ incidence matrix;
$\mathbf{A}_{l2} = \mathbf{A}_{2l}^T = [l, n]$ remaining partition of $\mathbf{A}_{12} = \mathbf{A}_{21}^T$ incidence matrix;
$\mathbf{A}_{ll} = [l, l]$ partition of $\mathbf{A}_{11}$, diagonal matrix;
$\mathbf{A}_{n0} = \mathbf{A}_{0n}^T = [n, n_0]$ partition of $\mathbf{A}_{10} = \mathbf{A}_{01}^T$ incidence matrix;
$\mathbf{A}_{n2} = \mathbf{A}_{2n}^T = [n; n]$ invertible partition of $\mathbf{A}_{12} = \mathbf{A}_{21}^T$ incidence matrix;
$\mathbf{A}_{nn} = [n; n]$ partition of $\mathbf{A}_{11}$, diagonal matrix;
$\mathbf{D}_{11} = [p; p]$ diagonal matrix of derivatives of $\mathbf{A}_{11}$ with respect to $\mathbf{Q}$;
$\hat{\mathbf{D}}_{11} = [p; p]$ diagonal matrix of derivatives of $\hat{\mathbf{A}}_{11}$ with respect to $\mathbf{H}$;
$\mathbf{D}_{ll} = [l, l]$ partition of $\mathbf{D}_{11}$, diagonal matrix (co-tree);
$\mathbf{D}_{nn} = [n; n]$ partition of $\mathbf{D}_{11}$, diagonal matrix (tree);
$\mathbf{f}, \mathbf{g}, \mathbf{h}$ = generic function expressions;
$\mathbf{f}_1, \mathbf{f}_2$ = residual errors in satisfying an equation;
$\mathbf{G} = [n; n]$ linear system matrix $\mathbf{A}_{21}(\mathbf{D}_{11})^{-1}\mathbf{A}_{12}$;
$\mathbf{H} = [n]$ length vector of unknown nodal heads;
$\mathbf{H}^0 = [n]$ length vector of initial heads solution;
$\mathbf{H}_0 = [n_0]$ length vector of known fixed head nodes;
$\mathbf{I}_{nn}, \mathbf{I}_{ll}, \mathbf{I}_{pp}$ = identity matrices of size $[n; n]$, $[l, l]$, $[p; p]$, respectively;
i, j, k = generic indexes;
$\mathbf{J}$ = Jacobian matrix;
L, D, C = pipe length (L), diameter (D), and roughness factor (C);
l = number of independent loops;
$\mathbf{M}_{13} = \mathbf{M}_{31}^T = [p, l]$ incidence matrix relating pipes to loops;
N = total number of nodes ($N = n + n_0$);
n = number of nodes with unknown head;
n_0 = number of nodes with known head;
p = number of pipes;
$\mathbf{Q} = [p]$ length vector of pipe discharges;
$\bar{\mathbf{Q}} = [p]$ length vector defined as $\bar{\mathbf{Q}}^T = [\bar{\mathbf{Q}}_n^T \vdots 0]$;
$\mathbf{Q}^0 = [p]$ length vector of initial pipe flows solution;
$\mathbf{Q}_l = [l]$ length partition vector of loop discharges;
$\mathbf{Q}_n = [n]$ length partition vector of discharges defined as $\mathbf{Q}_n = \bar{\mathbf{Q}}_n - \mathbf{A}_{2n}^{-1}\mathbf{A}_{2l}\mathbf{Q}_l$;
$\bar{\mathbf{Q}}_n = [n]$ length known flow vector defined as $\bar{\mathbf{Q}}_n = -\mathbf{A}_{2n}^{-1}\mathbf{q}$;
$\mathbf{q} = [n]$ length vector of nodal demands;
r = loss constant in the head loss equation;
$\mathbf{x}, \mathbf{y}$ = vectors of unknown variables;
α = exponent in the head loss equation;
ΔH = head loss in a generic pipe;
τ = iteration counter;
ϑ = under-relaxation weight $(0 \le \vartheta \le 1)$;
$\Phi(\boldsymbol{Q})$ = head loss equation as a function of friction and flow rate; and
$\Psi(\mathbf{H})$ = inverse of head loss equation giving flow as a function of friction and head loss.

References

Bhave, P. R. (1991). *Analysis of flow in water distribution networks*, Technomic Publishing, Lancaster, PA.
Carnahan, B., Luther, H. A., and Wilkes, J. O. (1969). *Applied numerical methods*, Wiley, New York.
Collins, M., Cooper, L., Helgason, R., Kennington, J., and LeBlanc, L. (1978). "Solving the pipe network analysis problem using optimization techniques." *Manage. Sci.*, 24(7), 747–760.
Cross, H. (1936). "Analysis of flow in networks of conduits or conductors." *Bulletin No. 286*, Engineering Experiment Station, Univ. of Illinois, Urbana, IL.
Donachie, R. P. (1974). "Digital program of water analysis." *J. Hydraul. Div.*, 100(HY3), 393–403.
Epp, R., and Fowler, A. G. (1970). "Efficient code for steady-state flows in networks." *J. Hydraul. Div.*, 96(HY1), 43–56.
Giustolisi, O. (2010). "Considering actual pipe connections in WDN analysis." *J. Hydraul. Eng.*, 136, 889–900.
Gupta, R., and Prasad, T. D. (2000). "Extended use of linear graph theory for analysis of pipe networks." *J. Hydraul. Eng.*, 126(1), 56–62.
Hamam, Y., and Brameller, A. (1971). "Hybrid method for the solution of piping network." *Proc., IEE*, 118(11), 1607–1612.
Isaacs, L. T., and Mills, K. G. (1980). "Linear theory for pipe network analysis." *J. Hydraul. Div.*, 106(HY7), 1191–1201.
Kesavan, H. K., and Chandrashekar, M. (1972). "Graph-theoretic models for pipe network analysis." *J. Hydraul. Div.*, 98(HY2), 345–363.

Lam, C. F., and Wolla, M. L. (1972a). "Computer analysis of water distribution systems: Part I—Formulation of equations." *J. Hydraul. Div.*, 98(HY2), 335–344.

Lam, C. F., and Wolla, M. L. (1972b). "Computer analysis of water distribution systems: Part II—Numerical solution." *J. Hydraul. Div.*, 98(HY3), 447–460.

Lemieux, P. F. (1972). "Efficient algorithms for distribution networks." *J. Hydraul. Div.*, 98(HY11), 1911–1920.

Martin, D. W., and Peters, G. (1963). "The application of Newton's method to network analysis by digital computer." *J. Inst. Water Eng.*, 17, 115–129.

Mignosa, P. (1987). "Sui problemi di verifica delle reti di distribuzione idrica complesse." *Idrotecnica*, 6, 257–273 (in Italian).

Osiadacz, A. J. (1988). "Comparison of numerical methods for steady-state simulation of gas networks." *Civ. Eng. Syst.*, 5, 25–30.

Osiadacz, A. J., and Pienkosz, K. (1988). "Methods of steady-state simulation for gas networks." *Int. J. Syst. Sci.*, 19(7), 1311–1321.

Rahal, H. (1995). "A co-tree flows formulation for steady state in water distribution networks." *Adv. Eng. Softw.*, 22, 169–178.

Rossman, L. A. (2000). "EPANET 2 users manual." *Rep. No. EPA/600/R-00/057*, Water Supply and Water Resources Division, National Risk Management Research Laboratory, Cincinnati.

Shamir, U., and Howard, C. D. D. (1968). "Water distribution systems analysis." *J. Hydraul. Div.*, 94(HY1), 219–234.

Todini, E. (1999). "A unifying view on the different looped pipe network analysis algorithms." *Computing and control for the water industry*, R. Powell and K. S. Hindi, eds., Research Studies, Baldock, Hertfordshire, UK, 63–80.

Todini, E. (2006). "On the convergence properties of the different pipe network algorithms." *Proc., 8th Annual Water Distribution Systems Analysis Symposium* (CD-ROM), ASCE, Cincinnati, 1–16.

Todini, E., and Pilati, S. (1988). "A gradient method for the solution of looped pipe networks." *Computer applications in water supply*, B. Coulbeck and C. H. Orr, eds., Vol. 1 (System analysis and simulation), Wiley, London, 1–20.

Wood, D. J., and Charles, C. O. A. (1972). "Hydraulic network analysis using linear theory." *J. Hydraul. Div.*, 98(HY7), 1157–1170.

Wood, D. J., and Rayes, A. G. (1981). "Reliability algorithms for pipe network analysis." *J. Hydraul. Div.*, 107(HY10), 1145–1161.