An Efficient Approach for Nodal Water Demand Estimation in Large-scale Water Distribution Systems

Shipeng Chu1 · Tuqiao Zhang1 · Xinhong Zhou1 · Tingchao Yu1 · Yu Shao1

Received: 16 August 2021 / Accepted: 3 November 2021 / Published online: 17 January 2022
© The Author(s), under exclusive licence to Springer Nature B.V. 2021

Abstract
Real-time modeling of a water distribution system (WDS) is a critical step for the control and operation of such systems. The nodal water demand, as the most important time-varying parameter, must be estimated in real time. The computational burden of nodal water demand estimation is intensive, leading to inefficiency in the modeling of large-scale networks. The Jacobian matrix computation and Hessian matrix inversion are the main processes that dominate the computation time. To address this problem, an approach for shortening the computation time for real-time demand estimation in large-scale network is proposed. This approach allows the Jacobian matrix to be efficiently computed based on solving a system of linear equations, and a Hessian matrix inversion method based on matrix partitioning and the iterative Woodbury-Matrix-Identity Formula is proposed. The developed approach is applied to a large-scale network, in which the number of nodal water demands is 12523, and the number of measurements ranges from 10 to 2000. The results show that the time consumptions for the Jacobian computation and Hessian matrix inversion are within 465.3 ms and 1219.0 ms, respectively. The time consumption is significantly shortened compared with the existing approach, especially for nodal water demand estimation in large-scale WDSs.

Keywords Nodal water demand estimation · Jacobian matrix · Hessian matrix inversion · Water distribution · Real-time modeling

Yu Shao
shaoyu1979@zju.edu.cn
Shipeng Chu
chushipeng@zju.edu.cn
Tuqiao Zhang
ztq@zju.edu.cn
Xinhong Zhou
zhouxinhong06@zju.edu.cn
Tingchao Yu
yutingchao@zju.edu.cn

1 College of Civil Engineering and Architecture, Zhejiang University, Hangzhou 310058, China
1 Introduction

Hydraulic models of water distribution system have been applied extensively in the analysis and operation of such systems (Abu-Mahfouz et al. 2019; Chu et al. 2020; Moasheri and Jalili-Ghazizadeh 2020; Khatavkar and Mays 2019). For these hydraulic models, the nodal water demands, which are the most uncertain input parameters, can greatly influence the model accuracy (Chu et al. 2021a; Meirelles et al. 2017; Kapelan et al. 2007). Therefore, real-time estimation of nodal water demand has become a major task for the real-time modeling of WDSs. Thanks to the development of remote metering technology (Sankaranarayanan et al. 2020), the use of a large amount of measured data to estimate the nodal water demands in real-time has gradually received increasing attention in recent years (Vrachimis et al. 2018).

The sensors distributed in a WDS network usually upload measurements at regular time intervals, typically 5-15 minutes in China, and then these data are used for nodal water demand estimation. Therefore, the nodal water demands must be estimated within the time interval of data uploading. However, the computational burden of demand estimation is intensive, and the time consumption grows exponentially as the number of network nodes increases (Chu et al. 2021b). This leads to inefficiency of the existing approach when estimating the nodal water demands for a large-scale network. There is an urgent need to address such inefficiency problems for real-time nodal water demand estimation in large-scale networks.

Many techniques have been proposed for estimating nodal water demands (Wang et al. 2019; Zhang et al. 2018). Examples include evolutionary algorithm (EA) based methods (Di Nardo et al. 2015; Zanfei et al. 2020), explicit methods (Berardi and Giustolisi 2021), singular value decomposition (SVD) methods (Weber and Hos 2020), and data assimilation methods (Chu et al. 2021a; Kang and Lansey 2009; Zhou et al. 2018). EA-based methods estimate nodal water demands by minimizing the deviation from the measured to the model-simulated values. However, these methods require frequent calls to the EPANET hydraulic solver (Rossman 2000), which is time-consuming for a large-scale network. Explicit methods are computationally efficient methods, in which a parameter grouping strategy is adopted to reduce the dimensionality of the problem (Goulter and Bouchart 1990; Shang et al. 2008; Freitas et al. 2018). This method groups the nodal water demands with similar characteristics as one parameter to be estimated (Rathi et al. 2020). However, the nonlinear relationship between the model outputs and model parameters is not considered, and improper grouping may result in additional errors (Jung et al. 2016). SVD methods have been used as an efficient means of estimating nodal water demands by solving the Moore–Penrose pseudoinverse of the coefficient matrix. However, such methods use only measured data, and the search space is not confined. Therefore, the nodal water demands can be arbitrarily adjusted, leading to considerable modeling errors.

Data assimilation methods estimate water demand by sequentially fusing the prior nodal water demand and measurements in a probabilistic form and finding a value with the maximum probability density value as the estimated nodal water demand (Law et al. 2015). Thus, the nodal water demand is estimated in a probabilistic form, typically described by the mean and covariance values. Data assimilation approaches can be further classified into sampling-based and numerical-based methods. Sampling-based methods are still time-consuming because they require frequent calls to the EPANET hydraulic solver to evaluate the probability density. In the numerical approach, the first-order gradient (Jacobian matrix) or second order-gradient (Hessian matrix) information
is utilized to search for the optimal solution. Therefore, such a method is expected to be more computationally efficient. However, estimation for large-scale networks is still time consuming when using current numerical approaches. Typically, the nodal water demands can be estimated only after several iterations, where each iteration involves the updating of the Jacobian matrix and Hessian matrix inversion. The updating of these two matrices involves an inversion operation for large matrix, which is time consuming. Therefore, the computation of the Jacobian matrix and Hessian matrix inversion are the processes that dominate the computation time.

The Jacobian matrix gives the first-order sensitivities of the outputs of a WDS hydraulic model (nodal pressure, pipe flow) to changes in the model parameters (nodal water demands, roughness) (Piller et al. 2017). The Jacobian matrix is widely used in many areas of water distribution system analysis, such as model parameter estimation, sensor placement, and uncertainty quantification. Piller et al. (2017) have presented high-quality research that provides explicit formulas for the Jacobian matrix of WDS steady-state heads and flows with respect to changes in the model parameters, including nodal water demands, roughness, and resistance factors. For the demand estimation problem, the Jacobian matrix between the model outputs and the nodal water demands is critical for minimizing the objective function. The computation of the Jacobian matrix involves an inversion operation on the matrix $W \in \mathbb{R}^{n \times n}$, which is typically a large, sparse, symmetric matrix (Coulbeck and Orr 1988), where $n$ is the number of nodal water demands. Computing the Jacobian matrix using the conventional dense matrix inversion method is time-consuming, as its time complexity is $O(n^3)$. Piller et al. (2017) suggested that the Jacobian can be computed through the use of sparse Cholesky factorization, which is expected to be computationally efficient. However, this approach still suffers the disadvantage of being time consuming for demand estimation in a large-scale WDS, as the direct inversion of the matrix $W$ involves many useless operations.

The computational burden of Hessian matrix inversion is also intensive, with a time complexity of $O(n^3)$. Some methods, such as using Hessian-free numerical optimization methods or solving a linear system of equations, allow the equations to be solved more efficiently. However, these methods are difficult to adopt in the data assimilation framework. For data assimilation approaches, the inversion of the Hessian matrix is used to compute the covariance matrix of the nodal water demands (Chu et al. 2021b). The covariance matrix is an important variable for quantifying the parameter uncertainty (Law et al. 2015). This highlights the importance of calculating the covariance matrix (Hessian matrix inversion) in a timely manner. Chu et al. (2021b) developed an efficient Hessian matrix inversion algorithm to shorten the time consumption of the demand estimation problem. This approach is efficient for problems in which the number of measurements $m$ is far smaller than the number of nodal water demands ($m \ll n$). However, it is not efficient enough when estimating the nodal water demands in sensor-dense networks (e.g., $\frac{m}{n} > 0.54$) (Chu et al. 2021b).

This paper develops a computationally efficient demand estimation approach, especially for large-scale WDS networks. The contributions are as follows: 1) an efficient method is developed to calculate the Jacobian matrix based on solving a system of linear equations; and 2) an efficient Hessian matrix inversion method is developed based on matrix partitioning and the iterative Woodbury-Matrix-Identity Formula. The developed approach is applied to a large-scale network, in which the number of nodal water demands is 12523 and the number of measurements ranges from 10 to 2000. The results demonstrate that the time consumptions for both the Jacobian computation and Hessian matrix inversion are significantly shortened compared with the existing approach.
2 Methodology

2.1 Real-time Nodal Water Demand Estimation

A data assimilation method estimates nodal water demands through the fusion of real-time measurements with the prior nodal water demands in a probabilistic form,

\[ P(X_{t|t}|Y_t) = N[Y_t|g(X_{t|t}), R] \times N(X_{t|t-1}|X_{t|t-1}, V_{t|t-1}) \]  

where \( X_{t|t} \in \mathbb{R}^{nx1} \) represents the nodal water demands at the current time step; \( Y_t \in \mathbb{R}^{mx1} \) is the measurement vector; \( P(X_{t|t}|Y_t) \) is the posterior probability distribution of \( X_{t|t} \), given the measurements \( Y_t \); \( N(X_{t|t}|X_{t|t-1}, V_{t|t-1}) \) is the prior probability distribution, describing the demand uncertainty predicted based on the previous time step \((t-1)\); \( X_{t|t-1} \in \mathbb{R}^{nx1} \) and \( V_{t|t-1} \in \mathbb{R}^{nxn} \) are the mean and covariance, respectively, of the nodal water demands predicted based on the previous time step \((t-1)\); \( N[Y_t|g(X_{t|t}), R] \) is the conditional probability distribution of the measurements \( Y_t \); \( g(X_{t|t}) \in \mathbb{R}^{mx1} \) and \( R \in \mathbb{R}^{mxm} \) are the mean and covariance. \( g(X_{t|t}) \) is the model outputs corresponding to the measurements, given the model input \( X_{t|t} \); and \( m \) and \( n \) are the numbers of measurements and nodal water demands, respectively. The subscripts \( t|t-1 \) and \( t|t \) represent that the states (values) at time step \( (t) \) are estimated from time steps \( t-1 \) and \( t \), respectively.

By maximizing the logarithm of \( P(X_{t|t}|Y_t) \), the nodal water demands \( X_{t|t} \) can be obtained. This is equivalent to minimizing the objective function shown in Eq. (2). The details of Eqs. (1, 2) can be found in (Chu et al. 2021b).

\[ \min : J = \left[ g(X_{t|t}) - Y_t \right]^T R^{-1} \left[ g(X_{t|t}) - Y_t \right] + (X_{t|t} - X_{t|t-1})^T V_{t|t-1}^{-1} (X_{t|t} - X_{t|t-1}) \]  

The objective function Eq. (2) is a nonlinear function and can be solved iteratively. The solution is achieved by solving Eqs. (3-6) iteratively.

\[ X_{t|t}^{k+1} = X_{t|t}^k + \varepsilon \Delta X_{t|t}^k \]  

\[ \Delta X_{t|t}^k = V_{t|t}^k G^T R^{-1} \left( g(X_{t|t}^k) - Y_t \right) + V_{t|t-1}^{-1} (X_{t|t}^k - X_{t|t-1}) \]  

\[ G^k = \frac{\partial g(X_{t|t}^k)}{\partial X_{t|t}} \]  

\[ V_{t|t}^k = [V_{t|t-1}^{-1} + \left( G^k \right)^T R^{-1} G^k]^{-1} \]

where \( \varepsilon \) is the step size, satisfying \( \varepsilon < 1 \); \( \Delta X_{t|t}^k \in \mathbb{R}^{nx1} \) is the demand corrector in the \( k \)-th iteration; \( V_{t|t}^k \in \mathbb{R}^{nxn} \) is the covariance matrix of the estimated nodal water demands; and \( G^k \in \mathbb{R}^{nxm} \) is the Jacobian matrix in the \( k \)-th iteration.

Equation (5) aims to produce the Jacobian matrix of the measurements in each iteration. In addition, Eq. (6) involves the inversion of the Hessian matrix \( H = [V_{t|t-1}^{-1} + \left( G^k \right)^T R^{-1} G^k] \in \mathbb{R}^{nxn} \) for the computation of the covariance matrix \( V_{t|t}^k \). For a large-scale network, the time consumption of Eqs. (5, 6) is intensive. Therefore, Eqs.
(5, 6) dominate the computation time. The bottleneck of computational efficiency lies in the solving of the Jacobian matrix and the Hessian matrix inversion (Chu et al. 2021b).

### 2.2 Solution of the Jacobian Matrix

#### 2.2.1 Jacobian Matrix of the Pressure Measurements

Piller et al. (2017) provided explicit formulas for the Jacobian matrix between the model outputs (nodal pressure, pipe flow, etc.) and the model parameters (nodal water demand, pipe roughness, etc.). The mathematical formalism for the Jacobian matrix between the nodal pressures and nodal water demands can be written as shown in Eq. (7) (Piller et al. 2017).

\[
\overline{G}_p = \frac{\partial p}{\partial \bar{X}_{fr}} = -(A^T FA)^{-1}
\]

where, \( p \in \mathbb{R}^{n \times 1} \) represents the pressures at all nodes; \( A \in \mathbb{R}^{np \times np} \) is the incidence matrix, \( F \in \mathbb{R}^{np \times np} \) is a diagonal matrix; \( \overline{G}_p \in \mathbb{R}^{n \times np} \) is the Jacobian matrix between the nodal pressures and the nodal water demands at the \( n \) nodes; and \( np \) is the number of pipes.

Equation (7) gives the mathematical expression for the Jacobian matrix of \( n \) nodal pressures, which involves the inversion of the matrix \((-A^T FA)\). The matrix \((-A^T FA)\) is a square matrix, and the number of nonzero elements in \((-A^T FA)\) is far smaller than \( n^2 \). Therefore, \((-A^T FA)\) is a large, sparse symmetric matrix (Coulbeck and Orr 1988). Treating \((-A^T FA)\) as a dense matrix and performing the direct inversion of \((-A^T FA)\) is time consuming with a time complexity of \(O(n^3)\). Piller et al. (2017) reported that the inversion of \((-A^T FA)\) can be computed at a marginal cost by using sparse Cholesky factorization.

As shown in Fig. 1, the Jacobian matrix \( \overline{G}_p \) consists of \( n \) row vectors, where each row vector corresponds to a nodal pressure. Among the \( n \) row vectors, only \( m_p \) (\( m_p \ll n \)) vectors (the right matrix in Fig. 1) correspond to measured pressure data and will be used for nodal water demand estimation, where \( m_p \) is the number of pressure measurements. Even if sparse Cholesky factorization is used, the direct inversion method is not the most efficient approach especially for large-scale network, as most vectors (\( n - m_p \)) in \( \overline{G}_p \) are discarded.

To improve the computational efficiency, we have developed an approach in which matrix inversion is avoided and the Jacobian matrix of the pressure measurements (\( m_p \) vectors) is directly calculated. Denoting \((-A^T FA)\) by \( W \), the matrix \( \overline{G}_p \) can be computed by solving Eq. (8). Equation (8) can be decomposed into a system of linear equations, as shown in Eq. (9).

![Fig. 1 Calculation of the Jacobian matrix for the measured pressure data](image-url)
where, \( I \in \mathbb{R}^{n \times n} \) is the identity matrix; \( \bar{G}_p(i) \in \mathbb{R}^{n \times 1} \) is the \( i \)th column of \( \bar{G}_p \); and \( I(i) \in \mathbb{R}^{n \times 1} \) is the \( i \)th column of \( I \).

By solving Eq. (9), the \( i \)th column vector \( \bar{G}_p(i) \) can be directly computed. Considering that \( G_p \) is a symmetric matrix, the \( i \)th row of \( G_p \) is the transposition of the column vector \( \bar{G}_p(i) \). Therefore, \( m_p \) column vectors corresponding to the pressure measurements can be computed by solving \( m_p \) linear equations. Then, these \( m_p \) column vectors are transposed to obtain the Jacobian matrix of the pressure measurements.

As mentioned previously, the matrix \( W \) is a large, sparse symmetric matrix. This allows the use of Cholesky factorization to solve the sparse symmetric system of linear equations in Eq. (9), an approach that has also been used in EPANET (Rossman 2000). The Cholesky factorization of matrix \( W \) can be written as

\[
W = L^T L
\]

where \( L \) is an upper triangular matrix with positive diagonal elements. Then, Eq. (9) can be written in its equivalent form,

\[
L^T y(i) = I(i)
\]

\[
L \bar{G}_p(i) = y(i)
\]

To calculate the Jacobian matrix for the pressure measurements, the Cholesky factorization of the matrix \( W \) should be performed first (Eq. 10). By submitting the matrix \( L \) to Eqs. (11 and 12), the Jacobian matrix \( \bar{G}_p(i) \) for \( i \)-th \((i = 1, 2, \ldots, m_p)\) pressure measurement can be calculated. Considering that \( L \) is an upper triangular matrix, Eqs. (11 and 12) can be efficiently solved. Considering that Eqs. (11 and 12) for different measurements can be solved independently, parallelization implementation of Eqs. (11 and 12) is adopted to improve the computational efficiency. The pseudocode for the solution of the Jacobian matrix is shown in Fig. S1.

### 2.2.2 Jacobian Matrix for the Flow Measurements

As shown in Fig. S2, \( N_1 \) and \( N_2 \) denote the two nodes of a pipe, in which the flow rate is \( Q \). The heads and pressures of the two nodes are \( h_1 \) and \( p_1 \) for \( N_1 \) and \( h_2 \) and \( p_2 \) for \( N_2 \). The head loss of the pipe can be computed as \( h_f = rQ^\alpha = h_1 - h_2 \), where \( r \) and \( \alpha \) are two coefficients.

By taking the derivative of the head loss equation, we can obtain Eq. (13). Considering that

\[
\frac{\partial h_1}{\partial X_{it}} = \frac{\partial p_1}{\partial X_{it}} \quad \text{and} \quad \frac{\partial h_2}{\partial X_{it}} = \frac{\partial p_2}{\partial X_{it}},
\]

Eq. (13) can be written as Eq. (14).

\[
arQ^{\alpha-1} \frac{\partial Q}{\partial X_{it}} = \frac{\partial h_1}{\partial X_{it}} - \frac{\partial h_2}{\partial X_{it}}
\]

\[
\frac{\partial Q}{\partial X_{it}} = \frac{1}{arQ^{\alpha-1}} \left( \frac{\partial p_1}{\partial X_{it}} - \frac{\partial p_2}{\partial X_{it}} \right)
\]
To obtain the Jacobian matrix of the measured pipe flow data, we should first compute the pressure Jacobian matrix of the two nodes \( \frac{\partial p_l}{\partial X_{it}} \) of the pipe using the method developed in the previous section. The Jacobian matrix of the measured flow data is then calculated by substituting \( \frac{\partial p_l}{\partial X_{it}} \) into Eq. (14).

### 2.3 Inversion of the Hessian Matrix

Chu et al. (2021b) proposed an approach to solve the Hessian matrix inversion by adopting the iterative Sherman-Morrison Formula. This approach is efficient for problems in which the number of measurements is far smaller than the number of nodal water demands \( (m \ll n) \). However, this method is not efficient enough when estimating the nodal water demands in sensor-dense networks \( (m \approx n) \) (Chu et al. 2021b). To fill this research gap, this paper provides an efficient method of computing the inversion of Hessian matrix.

As shown in Fig. 2, the Jacobian matrix can be partitioned into several blocks, each of which corresponds to a group of measurements. Denoting the Hessian matrix by \( H_{NG} \), the Hessian matrix can be written as Eq. (15).

\[
H_{NG} = V^{-1}_{lit} + \sum_{i=1}^{NG} (G_i^T R_i^{-1} G_i)
\]  

(15)

where \( NG \) is the number of blocks (measurement groups); \( G_i \in \mathbb{R}^{m_i \times n} \) is the \( i \)-th block of \( G \in \mathbb{R}^{m \times n} \); \( R_i \in \mathbb{R}^{m_i \times m_i} \) is the noise covariance of the measurements corresponding to \( G_i \); and \( m_i \) is the number of measurements corresponding to block \( i \) and satisfies \( \sum_{i=1}^{NG} m_i = m \). Based on Eq. (15), the Hessian matrix has the following form.

\[
H_i = H_{i-1} + G_i^T R_i^{-1} G_i, H_0 = V^{-1}_{lit} \quad \text{and} \quad i = 1, 2, \ldots, NG
\]  

(16)

Denoting \( G_i H_{i-1}^{-1} \) by \( M \in \mathbb{R}^{m_i \times n} \), based on Woodbury-Matrix-Identity Formula, the Hessian matrix inversion can be calculated as,

\[
H_i^{-1} = H_{i-1}^{-1} - M^T S^{-1} M, i = 1, 2, \ldots, NG
\]  

(17)

\[
H_0^{-1} = V^{-1}_{lit}
\]  

(18)

\[
S = R_i + G_i H_{i-1}^{-1} G_i^T
\]  

(19)
As shown in Eqs. (17-19), the Hessian matrix inversion problem is solved after \( NG \) iterations. Each iteration involves the inversion of the matrix \( S \in \mathbb{R}^{m_i \times m_i} \), with a time complexity of \( O(m_i^3) \). The total time complexity of each iteration can be written as \( O(m_i^3) + \theta_i \), where \( \theta_i \) is the time complexity of other matrix operations (matrix multiplication and addition, etc.). Considering that the algorithm requires \( NG \) iterations, the total time complexity can be written as \( \sum_{i=1}^{NG} [O(m_i^3) + \theta_i] \). Assuming that \( m_i \approx \frac{m}{NG} \), we can write

\[
\sum_{i=1}^{NG} [O(m_i^3) + \theta_i] \approx O \left( \frac{m^3}{NG^2} \right) + \sum_{i=1}^{NG} \theta_i \tag{20}
\]

Generally, the inversion of the matrix \( S \) is the process that dominates the computation time in Hessian matrix inversion. In this situation, \( O \left( \frac{m^3}{NG^2} \right) \gg \sum_{i=1}^{NG} \theta_i \), and the time complexity is approximately \( O \left( \frac{m^3}{NG^2} \right) \). At this point, the time consumption decreases exponentially with an increasing number of measurement groups \( NG \). Theoretically, the algorithm consumes the least time when \( NG = m \) (each block corresponds to just one measurement), in which case it is equivalent to the method developed by Chu et al. (2021b). However, Eq. (20) indicates that setting \( NG = m \) is not an optimal strategy. An increase in \( NG \) will reduce the time complexity of the inversion of the matrix \( S \); however, the time consumption of the other matrix operations (\( \sum_{i=1}^{NG} \theta_i \)) will gradually increase. When \( NG = m \), the total time complexity is \( O(m) + \sum_{i=1}^{m} \theta_i \approx \sum_{i=1}^{m} \theta_i \). In this situation, other matrix operations dominate the computation time, leading to inefficiency of the algorithm developed by Chu et al. (2021b). Compared with the method presented by Chu et al. (2021b), the developed method provides a more flexible means to select the appropriate \( NG \) to balance the time consumptions of the inversion of the matrix \( S \) and other matrix operations, and thus obtains a significant improvement when dealing with sensor-dense networks. The resulting nodal water demand estimation algorithm is shown in Fig. S3.

### 3 Results

#### 3.1 Case Study

The proposed approach has been developed in the C language. The algorithm for solving the Jacobian matrix needs to allocate a large amount of memory to store the matrix \( W \); thus, the CPU is most suitable for processing the parallel algorithm for Jacobian matrix computation. The algorithm for Hessian matrix inversion is executed in parallel on the GPU. Simulations were performed on an ASUS computer equipped with an Intel (R) Core i7-8750H CPU (6 cores and 12 threads) and an NVIDIA GeForce GTX 1050 Ti.

The developed method was applied to BWSN network 2, a large-scale network with 12523 nodal water demands to be estimated (\( n = 12523 \)), as shown in Fig. 3. Detailed information about BWSN network 2 can be found in Ostfeld et al. (2008). Various scenarios with numbers of measurements (\( m \)) ranging from 10 to 2000 were simulated to assess the properties of the developed approach. The sensor locations corresponding to these measurements were randomly selected. Both the prior demands and measurements were generated as synthetic data. The measured pressure data were generated by adding a normally distributed random value with a mean of 0 and a variance of 0.2 m. A uniformly distributed random value in \( U(-X, X) \) was added to the true nodal water demand (\( X \)) to generate the prior nodal water demand \( X_{t|t-1} \). The diagonal elements of \( V_{t|t-1} \) were set to \( (X_{t|t-1})^2 \).
3.1.1 Computational Efficiency

The proposed approach, denoted by LE_SCF, computes the Jacobian matrix by solving linear equations (LEs) through the utilization of sparse Cholesky factorization (SCF). For comparison, the direct inversion method, denoted by Direct inversion_MKL, in which \((-A^TFA)\) is treated as a dense matrix is provided. This method (Direct inversion_MKL) is implemented through the use of a parallel inversion solver based on Intel MKL (Wang et al. 2014), a high-performance parallel matrix computing library provided by the Intel Corporation. In addition, a direct inversion method utilizing sparse Cholesky factorization (SCF) is also provided, denoted by Direct inversion_SCF.

Typically, algorithms can estimate the nodal water demands after 15 iterations (Fig. S4). Figure 4 gives the average time consumption of the Jacobian matrix computation using each of the three methods in each iteration. From Fig. 4, we can see that the time...
consumption of the LE_SCF method (the red line) increases linearly with an increasing number of measurements \((m)\), while the time consumptions of the direct inversion methods (Direct inversion_MKL and Direct inversion_SCF) are relatively stable. This is because the developed LE_SCF method calculates the Jacobian matrix by solving \(m\) linear equations, whereas the direct inversion methods always directly invert the matrix \((-A^TFA) \in \mathbb{R}^{n \times n}\); thus, their time consumptions do not vary with the change in the number of measurements.

From Fig. 4 we can see that the time consumption of the Direct inversion_SCF method (the green line) is 16% of that of the Direct inversion_MKL method. This highlights the effectiveness of sparse Cholesky factorization in improving computational efficiency. In addition, the time consumption of the developed approach (red line) is far less than those of the direct inversion methods. Even when the number of measurements reaches 2000, the time consumption of the developed algorithm is only 20% of that of the Direct inversion_SCF method. Generally, the developed method can estimate the nodal water demands after 15 iterations (Fig. S4). In this situation, the total time consumption of the Jacobian matrix computation in the developed method over 15 iterations ranges from 0.18 s to 6.98 s as the number of measurements increases from 10 to 2000. For comparison, the total time consumptions of the Direct inversion_MKL method and the Direct inversion_SCF method are approximately 219 s and 34.77 s, respectively. The developed method, in which the Jacobian matrix is computed by solving a system of linear equations with the use of sparse Cholesky factorization, offers significantly shortened time consumption and improved usability, especially for the real-time modeling of a large-scale network.

Figure S5 shows the average time consumption of Hessian matrix inversion in each iteration for different numbers of measurement groups \((NG)\), with the number of measurements ranging from 10 to 2000. To facilitate the analysis, the time consumption for \(m = 2000\) is shown in Fig. 5. It is interesting to note that two different performance-changing stages can be observed from Fig. 5 (red line). For the first stage, in which the number of measurement groups is fewer than 8 \((NG < 8)\), a rapid decrease in time consumption is observed; then follows the second
stage, in which the number of measurement groups is 8 or greater ($NG \geq 8$) and the average time consumption increases with the number of measurement groups at a moderate rate. This is because the total time consumption (red line) is the sum of the time consumptions of the inversion operation for the matrix $S$ (green line) and the other matrix operations (black line). As mentioned in the previous section, the time consumption of the inversion of the matrix $S$ (Eq. 16) decreases exponentially with an increasing number of measurement groups ($NG$), as illustrated in Fig. 5 (green line). However, the time consumption of the other matrix operations ($\sum_{i}^{NG} \theta_i$) increases linearly with increasing $NG$ (black line). Therefore, the number of measurement groups ($NG$) should be optimized to balance the time consumptions of matrix inversion and other matrix operations. The optimal $NG$ depends on the case networks, the computation environment, and the number of measurements. It can be determined on the basis of prearranged numerical experiments. For this case study, $NG$ was optimized by trial and error. The optimized $NG$ values and time consumptions are shown in Table 1.

Figure 6 shows the time consumption of Hessian matrix inversion after the number of measurement groups has been optimized (Table 1). For comparison, the time consumption of the Sh_Mo based method developed by Chu et al. (2021b) is also provided. As shown in Fig. 6, the time consumption of the developed approach increases as the number of measurements increases, and it is always within 1000 ms. In contrast, the time consumption of the Sh_Mo based method increases from 2315 ms to 165204 ms. On average, the time consumption of the Sh_Mo method is almost 60 times greater than that of the developed approach. The main reason is that the Sh_Mo method calculates the Hessian matrix inversion after $m$ iterations (Chu et al. 2021b), making it equivalent to the developed approach with $NG = m$. Thus, the Sh_Mo method is not efficient, as it consumes considerable time for the other matrix operations. In contrast, in the developed approach, the appropriate number of groups ($NG$) can be selected to significantly shorten the time consumption of Hessian matrix inversion.

| $m$ | $NG$ | time consumption (ms) | $m$ | $NG$ | time consumption (ms) |
|-----|-----|-----------------------|-----|-----|-----------------------|
| 10  | 2   | 235.27                | 300 | 2   | 395.60                |
| 20  | 2   | 240.47                | 500 | 2   | 432.20                |
| 30  | 2   | 243.07                | 800 | 4   | 650.00                |
| 50  | 2   | 249.53                | 1000| 4   | 690.13                |
| 80  | 2   | 265.20                | 1500| 4   | 1029.20               |
| 100 | 2   | 267.53                | 2000| 8   | 1219.00               |
| 200 | 2   | 333.93                |     |     |                       |
3.1.2 Modeling Accuracy

The cumulative probability of absolute error for the 12523 nodal water demands is shown in Fig. S6. The absolute error is computed as the deviation from the estimated water demands to the true values. The prior error is computed as the absolute deviation from the prior
values to the true values. As shown in Fig. S6, the estimated water demand error approaches the prior error. This is because the developed approach attempts to minimize the difference between the estimated nodal water demands ($X_{t+1}$) and the prior values ($X_{t+1-1}$). Therefore, the algorithm may show similar performance when the same prior values are used.

Figure S7 presents the cumulative probability of absolute error of the nodal pressures at the 12523 nodes. The absolute error is computed as the deviation from the estimated nodal pressures to the true values. The prior pressure error is the absolute deviation between the true pressures to the pressures computed by directly using the prior demands as model input. To facilitate the analysis, the estimated results for $m = 30, 100, 1000, 2000$ are shown in Fig. 7. As shown in Fig. 7, the estimated pressure error is smaller than the prior pressure error. In addition, the nodal pressure error decreases with an increasing number of measurements. For example, 50% of the nodes have an error greater than 0.5 m when $m = 30$, while 20% of the nodes have an error greater than 0.5 m when $m = 100$. This significant improvement highlights the importance of the utilization of measurements to improve the modeling accuracy.

4 Conclusions

The timely estimation of nodal water demands is critical for the real-time modeling of WDS. The Jacobian matrix computation and Hessian matrix inversion are the processes that dominate the computation time. This paper has developed an efficient approach for improving the typical computational efficiency of nodal water demand estimation in the large-scale network.

The developed approach calculates the Jacobian matrix based on sparse Cholesky factorization and solving a system of linear equations. Each equation can be solved independently, allowing the use of parallel programming to improve the computational efficiency. An efficient Hessian matrix inversion method based on matrix partitioning and Iterative Woodbury-Matrix-Identity Formula has been developed. This approach provides a more flexible means to balance the time consumptions of the matrix inversion and other matrix operations, and thus obtains a significant improvement when dealing with sensor-dense networks. The results show that the time consumptions of the Jacobian matrix computation and Hessian matrix inversion are approximately 20% and 1.6%, respectively, of those in the existing approach.

Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s11269-021-03024-w.

Acknowledgments This work was supported by the National Natural Science Foundation of China (No. 52070165).

Authors Contributions Shipeng Chu: Conceptualization, Methodology, Software, Validation, Writing-Original Draft. Tuqiao Zhang: Writing-Review & Editing, Resources, Project administration. Xinhong Zhou: Visualization, Investigation, Formal analysis. Tingchao Yu: Writing - Review & Editing. Yu Shao: Writing - Review & Editing, Supervision, Funding acquisition.

Data/Code Availability All data, models, and codes that support the findings of this study are available from the corresponding author upon reasonable request.

Declarations

Ethical Approval Not applicable.
Consent to Publish Not applicable.

Consent to Participate Not applicable.

Conflicts of Interest/Competing Interests The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

References

Abu-Mahfouz AM, Hamam Y, Page PR, Adedeji KB, Anele AO, Todini E (2019) Real-time dynamic hydraulic model of water distribution networks. Water 11(3)

Berardi L, Giustolisi O (2021) Calibration of design models for leakage management of water distribution networks. Water Resour Manag 35(8):2537–2551

Chu S, Zhang T, Xu C, Yu T, Shao Y (2021a) Dealing with data missing and outlier to calibrate nodal water demands in water distribution systems. Water Resour Manag 35(9):2863–2878

Chu S, Zhang T, Li X, Li K, Shao Y (2021b) Approach for water distribution system model calibration based on iterative Sherman-Morrison formula. J Water Resour Plan Manag 147(5):04021017

Chu S, Zhang T, Shao Y, Yu T, Yao H (2020) Numerical approach for water distribution system model calibration through incorporation of multiple stochastic prior distributions. Sci Total Environ 708:134565

Coulbeck B, Orr CH (1988) Computer applications in water supply. Res Stud Press

Di Nardo A, Di Natale M, Gisonni C, Iervolino M (2015) A genetic algorithm for demand pattern and leakage estimation in a water distribution network. J Water Supply Re Technol Aqu 64(1):35–46

Freitas R, Brentan B, Meirelles G, Luvizotto E (2018) Dynamic clustering and a hybrid optimization for roughness calibration in water distribution model. 1st International WDSA/CCWI 2018 Joint Conference

Goulter IC, Bouchart F (1990) Reliability-constrained pipe network model. J Hydraul Eng 116(2):211–229

Jung D, Choi Y, Kim J (2016) Optimal node grouping for water distribution system demand estimation. Water 8(4):160

Kang D, Lansey K (2009) Real-time demand estimation and confidence limit analysis for water distribution systems. J Hydraul Eng 135(10):825–837

Kapelan ZS, Savic DA, Walters GA (2007) Calibration of water distribution hydraulic models using a Bayesian-type procedure. J Hydraul Eng 133(8):927–936

Khatavkar P, Moasheri R, Jalili-Ghazizadeh M (2020) Locating of probabilistic leakage areas in water distribution networks by a calibration method using the imperialist competitive algorithm. Water Resour Manag 34(13):4339–4351

Meirelles G, Manzi D, Brentan B, Goulart T, Luvizotto E Jr (2017) Calibration model for water distribution network using pressures estimated by artificial neural networks. Water Resour Manag 31(13):4339–4351

Rossman, L. A. (2000). "Epanet 2 users manual, US Environmental Protection Agency." Water Supply and Water Resources Division, National Risk Management Research Laboratory, Cincinnati, OH, 45268.
Sankaranarayanan S, Sivakumaran N, Radhakrishnan TK, Swaminathan G (2020) Dynamic soft sensor based parameters and demand curve estimation for water distribution system: theoretical and experimental cross validation. Control Eng Pract 102
Shang F, Uber JG, van Bloemen Waanders BG, Boccelli D, Janke R (2008) Real time water demand estimation in water distribution system. Water Distrib Syst Anal Symp 2006:1–14
Vrachimis SG, Eliades DG, Polycarpou MM (2018) Leak detection in water distribution systems using hydraulic interval state estimation. 2018 IEEE Conference on Control Technology and Applications, CCTA 565–570
Wang E, Zhang Q, Shen B, Zhang G, Lu X, Wu Q, Wang Y (2014) Intel math kernel library. In High-performance computing on the Intel® Xeon Phi™ 167–188. Berlin: Springer
Wang S, Taha AF, Sela L, Gatsis N, Giacomoni MH (2019) State estimation in water distribution networks through a new successive linear approximation. Proceedings of the IEEE Conference on Decision and Control 5474-5479
Weber R, Hos C (2020) Efficient technique for pipe roughness calibration and sensor placement for water distribution systems. J Water Resour Plan Manag 146(1):04019070
Zanfei A, Menapace A, Santopietro S, Righetti M (2020) Calibration procedure for water distribution systems: comparison among hydraulic models. Water 12(5):1421
Zhang H, Wang K, Zhou X, Wang W (2018) Using DFP algorithm for nodal demand estimation of water distribution networks. KSCE J Civil Eng 22(8):2747–2754
Zhou X, Xu WR, Xin KL, Yan HX, Tao T (2018) Self-adaptive calibration of real-time demand and roughness of water distribution systems. Water Resour Res 54(8):5536–5550

Publisher’s Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.