

An Efficient Approach For the Nodal Water Demand Estimation in Large-Scale Water Distribution Systems

Shipeng Chu

Zhejiang University

Tuqiao Zhang

Zhejiang University

Xinhong Zhou

Zhejiang University

Tingchao Yu

Zhejiang University

Yu Shao (✉ shaoyu1979@zju.edu.cn)

Zhejiang University <https://orcid.org/0000-0003-2435-5618>

Research Article

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

Posted Date: October 26th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-818509/v1>

License:  This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Version of Record: A version of this preprint was published at Water Resources Management on January 17th, 2022. See the published version at <https://doi.org/10.1007/s11269-021-03024-w>.

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

Shipeng Chu¹, Tuqiao Zhang², Xinhong Zhou³, Tingchao Yu⁴, Yu Shao^{5*}

Abstract: Real-time modeling of the 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 for the modeling of the large-scale network. The Jacobian matrix computation and Hessian matrix inversion are the processes that dominate the main computation time. To address this problem, an approach to shorten the computational time for the real-time demand estimation in the large-scale network is proposed. The approach can efficiently compute the Jacobian matrix based on solving a system of linear equations, and a Hessian matrix inversion method based on matrix partition and Iterative Woodbury-Matrix-Identity Formula is proposed. The developed approach is applied to a large-scale network, of which the number of nodal water demand is 12523, and the number of measurements ranging from 10 to 2000. Results show that the time consumptions of both Jacobian computation and Hessian matrix inversion are significantly shortened compared with the existing approach.

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

¹ Postdoctor, College of Civil Engineering and Architecture, Zhejiang University, Hangzhou, CHINA, 310058. E-mail: chushipeng@zju.edu.cn;

² Professor, College of Civil Engineering and Architecture, Zhejiang University, Hangzhou, CHINA, 310058. E-mail: ztq@zju.edu.cn;

³ PhD student, College of Civil Engineering and Architecture, Zhejiang University, Hangzhou, CHINA, 310058. E-mail: zhouxinhong06@zju.edu.cn;

⁴ Professor, College of Civil Engineering and Architecture, Zhejiang University, Hangzhou, CHINA, 310058. E-mail: yutingchao@zju.edu.cn;

^{5*} Professor, College of Civil Engineering and Architecture, Zhejiang University, Hangzhou, CHINA, 310058. E-mail: shaoyu1979@zju.edu.cn;

1 Introduction

The hydraulic models of the water distribution system (WDS) have been widely used in the analysis and operation of such systems (Chu et al. 2020; Moasheri and Jalili-Ghazizadeh 2019). For the hydraulic models, the nodal water demand is the most uncertain input parameter that can significantly affect the accuracy of the model (Chu et al. 2021a; Diaz et al. 2017; Meirelles et al. 2017; Kapelan et al. 2007). Therefore, real-time estimation of nodal water demand is the major task for the real-time modeling of WDS. With the development of remote metering technologies, the use of a large amount of measured data to estimate the nodal water demand in real-time has gradually received increased attention in recent years.

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

Many techniques have been proposed to estimate the nodal water demand. Examples include evolutionary algorithm (EA) based methods (Dini and Tabesh 2014; Di Nardo et al. 2015), explicit methods (Du et al. 2015), singular value decomposition (SVD) methods (Weber and Hos 2020), and data assimilation methods (Bragalli et al. 2016; Chu et al. 2021a; Kang and Lansey 2009; Qin and Boccelli 2019; Rajakumar et al. 2019; Zhou et al. 2018). For the EA-based methods, the nodal water demand is estimated by minimizing the deviation from the model-simulated values to the measurements. However, these methods require frequent calls to the Epanet hydraulic solver (Rossman 2000), which is time-consuming for the large-scale network. The explicit method is a

computationally efficient method, in which a parameter grouping strategy is adopted to reduce the problem dimension (Goulter and Bouchart 1990; Kumar et al. 2010; Shang, et al.2008). In this method, nodal water demands with similar characteristics are grouped as one parameter to be estimated. However, the nonlinear relationship between the model parameters and outputs is not considered, and improper grouping may result in additional errors (Jung et al. 2016). The SVD method has been used as an efficient method to estimate the nodal water demand. It solves the Moore–Penrose pseudoinverse of the coefficient matrix of the linear equations to get the nodal water demand. However, this method only uses the measured data and the search space is not confined. Therefore, the nodal water demand can be freely adjusted, leading to considerable modeling errors.

The data assimilation methods estimate the nodal water demand by sequentially fusing the prior nodal water demand and the measurements in a probability form and find a value with the largest probability density value as the estimated node water demand (Law et al. 2015). The nodal water demand is estimated in a probability form, typically described by the mean and covariance value. The data assimilation approaches can be classified as sampling-based and numerical-based methods. The sampling-based methods are still time-consuming because it requires frequent calls to the EPANET 2.0 hydraulic solvers to evaluate its probability density. The numerical approach utilizes the first-order gradient (Jacobian matrix) or second order-gradient (Hessian matrix) information to search for the optimal solution. Therefore, it is expected to be more efficient in computation. However, estimation of large-scale networks is still time-consuming for current numerical approaches. Typically, the nodal water demand can be estimated after several iterations, and each iteration involves the update of the Jacobian matrix and Hessian matrix inversion. The update of the two matrices involves the inversion operation of the large matrix, which is time-consuming. Therefore, the computation of the Jacobian matrix and Hessian matrix inversion are the processes that dominant the computation time.

The Jacobian matrix gives the first-order sensitivities of WDS hydraulic model outputs (nodal pressure, pipe flow) to the changes in 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, uncertainty quantification. Piller et al. (2017) provide high-quality research that gives the explicit formulas for the Jacobian matrix of WDS steady-state heads and flows to changes in the modal parameters, including nodal water demands, roughness, resistance factors. For the demand estimation problem, the Jacobian matrix between model outputs to the nodal water demands is critical in minimizing the objective function. The computation of the Jacobian matrix involves an inversion operation of the matrix $W \in \mathbb{R}^{n \times n}$, which typically is a large sparse symmetric matrix (Coulbeck and Orr 1988), where n is the number of nodal water demand. 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) suggest that the Jacobian can be computed by the use of sparse Cholesky factorization, which is expected to be efficient in computation. However, this approach still suffers the disadvantage of being time-consuming for the demand estimation in large-scale WDS. As will be shown in the following sections, direct inversion of matrix W performs a lot of useless operations.

The computational burden of Hessian matrix inversion is also intensive, with the time complexity being $O(n^3)$. Some methods, such as using Hessian-free numerical optimization methods or solving a linear system of equations, can solve the equations more efficiently. However, these methods can hardly be adopted in the data assimilation framework. For the data assimilation approaches, the inverse operation of the Hessian matrix is used to compute the covariance matrix of nodal water demand (Chu et al. 2021b). The covariance matrix is an important variable to quantify 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 to the problem that the number of measurements is

far less than the number of nodal water demands ($m \ll n$), where m is the number of measurements. However, it is not efficient enough when estimating the nodal water demand in the sensor-dense networks ($\frac{m}{n} > 0.54$, e.g.) (Chu et al. 2021b).

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

2 Methodology

2.1 Real-time Nodal Water Demand Estimation

The data assimilation method estimates the nodal water demand by fusion of the prior nodal water demand with the real-time measurements in a probability form,

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

where $X_{t|t} \in \mathbb{R}^{n \times 1}$ is the nodal water demand at the current time step; $Y_t \in \mathbb{R}^{m \times 1}$ is the measurements vector; $P(X_{t|t}|Y_t)$ is the posterior probability distribution of the nodal water demand $X_{t|t}$, 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 from the previous time step ($t - 1$); $X_{t|t-1} \in \mathbb{R}^{n \times 1}$ and $V_{t|t-1} \in \mathbb{R}^{n \times n}$ are the mean and covariance of the nodal water demand that are predicted from previous time step $t - 1$; $N[Y_t|g(X_{t|t}), R]$ is the conditional probability distribution of the measurement Y_t ; $g(X_{t|t}) \in \mathbb{R}^{m \times 1}$ and $R \in \mathbb{R}^{m \times m}$ are the mean and covariance. $g(X_{t|t})$ is the model output corresponding to the measurements, given the model input $X_{t|t}$; m and n are the

numbers of measurements and nodal water demands, respectively. Subscript $t|t-1$ and $t|t$ represent that the state (values) at time step (t) is estimated from time step $t-1$ and t , respectively.

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

$$\min: J = [g(X_{t|t}) - Y_t]^T R^{-1} [g(X_{t|t}) - Y_t] + (X_{t|t} - X_{t|t-1})^T V_{t|t-1}^{-1} (X_{t|t} - X_{t|t-1}) \quad (2)$$

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

$$X_{t|t}^{k+1} = X_{t|t}^k + \varepsilon \Delta X_{t|t}^k \quad (3)$$

$$\Delta X_{t|t}^k = V_{t|t}^k [(G^k)^T R^{-1} (g(X_{t|t}^k) - Y_t) + V_{t|t-1}^{-1} (X_{t|t}^k - X_{t|t-1})] \quad (4)$$

$$G^k = \frac{\partial g(X_{t|t}^k)}{\partial X_{t|t}} \quad (5)$$

$$V_{t|t}^k = [V_{t|t-1}^{-1} + (G^k)^T R^{-1} G^k]^{-1} \quad (6)$$

where ε is the step size and $\varepsilon < 1$. $\Delta X_{t|t}^k \in \mathbb{R}^{n \times 1}$ is the demand corrector at k -th iteration; $V_{t|t} \in \mathbb{R}^{n \times n}$ is the covariance matrix of the estimated nodal water demand; $G^k \in \mathbb{R}^{m \times n}$ is the Jacobian matrix at k -th iteration.

Eq. (5) aims to get the Jacobian matrix of the measurements at each iteration. Besides Eq.(6) involves the inversion of Hessian matrix $H = [V_{t|t-1}^{-1} + (G^k)^T R^{-1} G^k] \in \mathbb{R}^{n \times n}$ to compute the covariance matrix $V_{t|t}$. For the large-scale network, the time consumption of Eqs.(5,6) is intensive. Therefore, Eqs.(5, 6) dominate the main computation time. The bottleneck of computational efficiency lies in the solving of the Jacobian matrix and Hessian matrix inversion (Chu et al. 2021b).

2.2 The solution of the Jacobian Matrix

2.2.1 Jacobian matrix for the pressure measurements

Piller et al. (2017) provide the explicit formulas for the Jacobian matrix between the model outputs (nodal pressure, pipe flow, et.al.) and the model parameters (nodal water demand, pipe roughness,

et.al.). The math formalism for the Jacobian matrix between the nodal pressure and nodal water demand can be written as Eq.(7) (Piller et al. 2017).

$$\bar{G}_p = \frac{\partial p}{\partial x_{t|t}} = -(A^T F A)^{-1} \quad (7)$$

where, $p \in \mathbb{R}^{n \times 1}$ is the pressure at all of the nodes; $A \in \mathbb{R}^{np \times n}$ is the incidence matrix, $F \in \mathbb{R}^{np \times np}$ is a diagonal matrix; $\bar{G}_p \in \mathbb{R}^{n \times n}$ is the Jacobian matrix of the nodal pressures to the nodal water demands at n nodes; np is the number of pipes.

Equation (7) gives the mathematical expression for the Jacobian matrix of n nodal pressures, which involves the inverse operation of matrix $(-A^T F A)$. Matrix $(-A^T F A) \in \mathbb{R}^{n \times n}$ is a square matrix, and the number of non-zero elements in $(-A^T F A)$ is far less than n^2 . Therefore, $(-A^T F A)$ is a large sparse symmetric matrix (Coulbeck and Orr 1988). Treating $(-A^T F A)$ as a dense matrix and direct inversion of $(-A^T F A)$ is time-consuming with the time complexity being $O(n^3)$. Piller et al. (2017) indicates that the inversion of $(-A^T F A)$ can be computed at marginal cost by using sparse Cholesky factorization. As shown in Figure 1, the Jacobian matrix \bar{G}_p consists of n row vectors and each row vector correspond to a nodal pressure. Among the n row vectors, only m_p ($m_p \ll n$) vectors (the right matrix in Figure 1) corresponds to the 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 the large-scale network, as the most vectors ($n - m_p$) in \bar{G}_p are discarded.

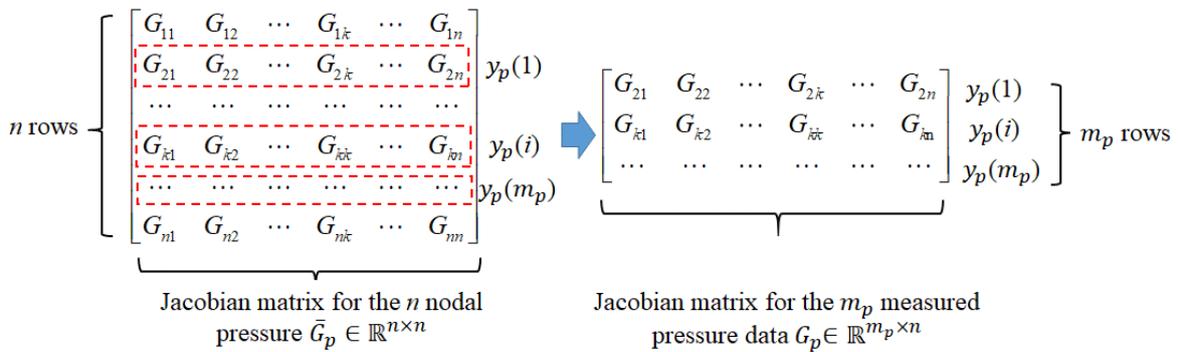


Figure 1 The calculation of the Jacobian matrix for the measured pressure data

To improve the computational efficiency, we developed an approach to avoid the matrix inversion and directly calculate the Jacobian matrix of the pressure measurements (m_p vectors). Denote $(-A^T F A)$ as W , matrix \bar{G}_p can be computed by solving Eq.(8). Eq.(8) can be decomposed into a system of linear equations, as shown in Eq.(9).

$$W\bar{G}_p = I \quad (8)$$

$$W\bar{G}_p(i) = I(i), i = 1, 2, \dots, n \quad (9)$$

where, $I \in \mathbb{R}^{n \times n}$ is identity matrix; $\bar{G}_p(i) \in \mathbb{R}^{n \times 1}$ is the i -th column of \bar{G}_p ; $I(i) \in \mathbb{R}^{n \times 1}$ is the i -th column of I .

By solving Eq.(9), the i -th column vector $\bar{G}_p(i)$ can be directly computed. Consider that \bar{G}_p are symmetric matrices, the i -th row of \bar{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 the m_p column vectors are transposed to get the Jacobian matrix of the pressure measurements.

As mentioned previously, matrix W is a large sparse symmetric matrix. This allows using Cholesky factorization to solve the sparse symmetric system of linear equation (Eq.(9)), which has also been used in the EPANET2.0 (Rossman 2000). The Cholesky factorization of matrix W can be written as,

$$W = L^T L \quad (10)$$

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

$$L^T y(i) = I(i) \quad (11)$$

$$L\bar{G}_p(i) = y(i) \quad (12)$$

To calculate the Jacobian matrix for pressure measurements, a Cholesky factorization of matrix W should be performed firstly (Eq.(10)). By submitting the matrix L to Eqs.(11 and 12), the Jacobian matrix ($\bar{G}_p(i)$) for i -th ($i = 1, 2, \dots, m_p$) can be calculated. Considering that L is an upper

triangular matrix, Eqs.(11 and 12) can be solved efficiently. Considering that Eqs.(11 and 12) of different measurements can be solved independently, parallelization implementation of Eqs.(11 and 12) is adopted to improve the computational efficiency. The pseudo-code for the solution of the Jacobian matrix is shown in Figure S1.

2.2.2 Jacobian matrix for the flow measurements

As shown in Figure S2, N_1 and N_2 are the two nodes of a pipe, of which the flow rate is Q . The head and pressure of the two nodes are h_1, p_1 and h_2, p_2 . The headloss of the pipe can be computed by $h_f = rQ^\alpha = h_1 - h_2$, where r and α are two coefficients. Deriving the head loss equation, we can get Eq.(13). Considering that $\frac{\partial h_1}{\partial X_{t|t}} = \frac{\partial p_1}{\partial X_{t|t}}, \frac{\partial h_2}{\partial X_{t|t}} = \frac{\partial p_2}{\partial X_{t|t}}$, Eq.(13) can be written as Eq.(14).

$$\alpha r Q^{\alpha-1} \frac{\partial Q}{\partial X_{t|t}} = \frac{\partial h_1}{\partial X_{t|t}} - \frac{\partial h_2}{\partial X_{t|t}} \quad (13)$$

$$\frac{\partial Q}{\partial X_{t|t}} = \frac{1}{\alpha r Q^{\alpha-1}} \left(\frac{\partial p_1}{\partial X_{t|t}} - \frac{\partial p_2}{\partial X_{t|t}} \right) \quad (14)$$

To get the Jacobian matrix of measured pipe flow data, we should compute the pressure Jacobian matrix of the two nodal $\left(\frac{\partial p_1}{\partial X_{t|t}}, \frac{\partial p_2}{\partial X_{t|t}} \right)$ of the pipe by the method developed in the previous section. The Jacobian matrix of measured flow data is calculated by substituting $\frac{\partial p_1}{\partial X_{t|t}}$ and $\frac{\partial p_2}{\partial X_{t|t}}$ into Eq.(14).

2.3 The solution of the Hessian matrix inversion

Chu et al. (2021b) proposed an approach to solving the Hessian matrix inversion based on the iterative Sherman-Morrison Formula. This approach is efficient to the problem that the number of measurements is far less than the number of nodal water demands ($m \ll n$). However, this method is not efficient enough when estimating the nodal water demand in the sensor-dense networks ($\frac{m}{n} > 0.54$, e.g.) (Chu et al. 2021b). To fill the research gap, this paper provides a more efficient method to solve the Hessian matrix inversion.

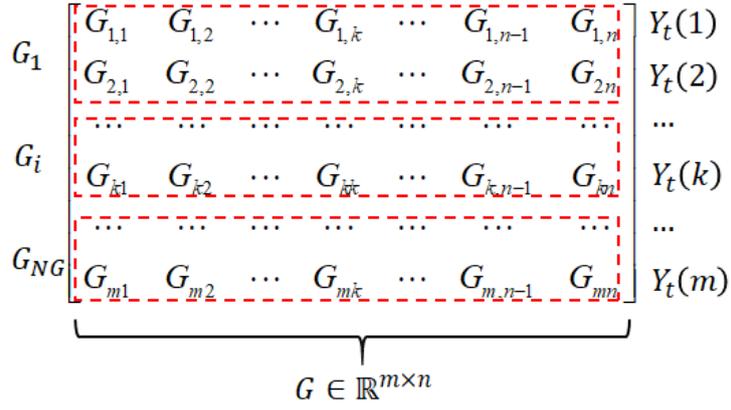


Figure 2 the partitioned Jacobian matrix

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

$$H_{NG} = V_{t|t-1}^{-1} + \sum_i^{NG} (G_i^T R_i^{-1} G_i) \quad (15)$$

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 measurement corresponding to G_i ; m_i is the number of measurements corresponding to block i and $\sum_i^{NG} m_i = m$. From Eq.(15), we can find that the Hessian matrix can be rewritten as Eq.(16).

$$H_i = H_{i-1} + G_i^T R_i^{-1} G_i, H_0 = V_{t|t-1}^{-1} \text{ and } i = 1, 2, \dots, NG \quad (16)$$

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

$$H_i^{-1} = H_{i-1}^{-1} - M^T S^{-1} M, i = 1, 2, \dots, NG \quad (17)$$

$$H_0^{-1} = V_{t|t-1} \quad (18)$$

$$S = R_i + G_i H_{i-1}^{-1} G_i^T \quad (19)$$

As shown in Eq.(17-19), the Hessian matrix inversion is solved after NG iterations. Each iteration involves the inverse operation of matrix $S \in \mathbb{R}^{m_i \times m_i}$, with the time complexity being $O(m_i^3)$. The total time complexity of each iteration can be written as $O(m_i^3) + \vartheta_i$, where ϑ_i is the time complexity of other matrix operations (matrix multiplication and addition, et.al.). Considering

that the algorithm requires NG iterations, the total time complexity can be written as $\sum_i^{NG} [O(m_i^3) + \vartheta_i]$. Assuming that $m_i \approx \frac{m}{NG}$, we can write

$$\sum_i^{NG} [O(m_i^3) + \vartheta_i] \approx O\left(\frac{m^3}{NG^2}\right) + \sum_i^{NG} \vartheta_i \quad (20)$$

Generally, the inverse operation of matrix S is the process that dominates the main computation time in Hessian matrix inversion. In this situation, $O\left(\frac{m^3}{NG^2}\right) \gg \sum_i^{NG} \vartheta_i$ and the time complexity is about $O\left(\frac{m^3}{NG^2}\right)$. At this point, the time consumption reduces exponentially with the increasing number of measurement groups NG . Theoretically, the algorithm takes the shortest time consumption when $NG = m$ (each block corresponds to just one measurement), which is equal to the method developed by Chu et al. (2021b). However, Eq.(20) indicates setting $NG = m$ is not an optimal strategy. The increase of NG will reduce the time complexity of inverse operation of matrix S , however the time consumption of other matrix operations ($\sum_i^{NG} \vartheta_i$) will increase gradually. When $NG = m$, the total time complexity comes to be $O(m) + \sum_i^m \vartheta_i \approx \sum_i^m \vartheta_i$. In this situation, other matrix operations processes dominate the main computation time, leading to the inefficiency of the algorithm developed by Chu et al. (2021b). Comparing with the method provided by Chu et al. (2021b), the developed method provides a more flexible means to select the appropriate NG to balance time consumptions of inverse operation of matrix S and other matrix operations, and thus obtains a significant improvement when dealing with sensor-dense networks.

3 Results

3.1 Case study

The proposed approach is developed in the C language. The algorithm for the Jacobian matrix needs to allocate a large amount of memory to store matrix W , and thus CPU is more suitable to process the parallel algorithm for Jacobian matrix computation. The algorithm for Hessian matrix inversion is executed in parallel on GPU. The simulation was 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.

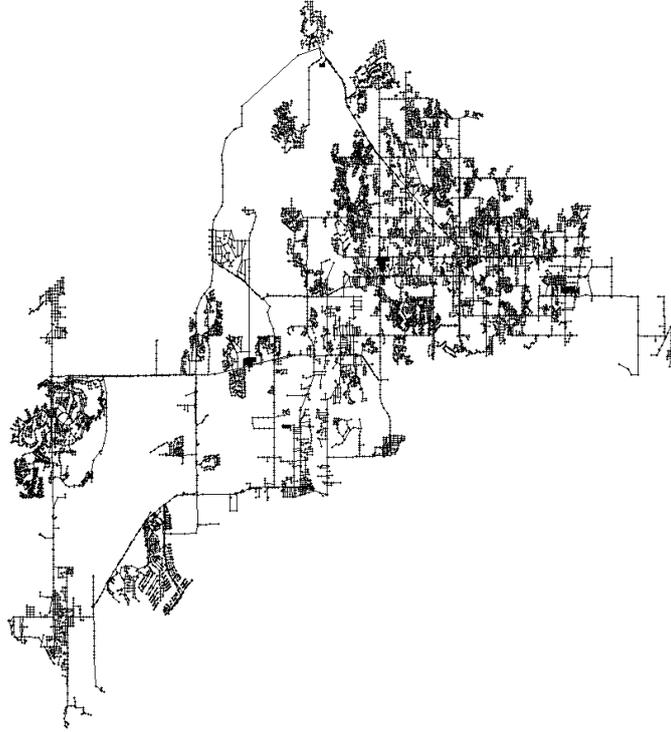


Figure 3 Network layout of BWSN network 2 (Ostfeld et al. 2008)

The developed method is applied to the *BWSN network 2*, a large-scale network with 12523 nodal water demands to be estimated ($n = 12523$), as shown in Figure 3. Detailed information about *BWSN network 2* can be found in Ostfeld et al. (2008). Different simulation scenarios with the numbers of measurements (m) ranging from 10 to 2000 are performed to evaluate the performance of the developed approach. The sensor locations corresponding to these measurements are randomly selected. Both the prior and measurements are synthetic data. The measured pressure data are generated by adding a normally distributed random value with a mean of 0 and a variance of 0.2m. A uniformly distributed random value at $U(-X, X)$ is added to the true nodal water demand (X) to generate the prior nodal water demand $X_{t|t-1}$. The diagonal element of $V_{t|t-1}$ is set to be $(X_{t|t-1})^2$.

3.1.1 Computation Efficiency

The proposed approach, denoting as LE_SCF, computes the Jacobian matrix by solving linear equations (LE) with the utilization of sparse Cholesky factorization (SCF). For comparison,

the direct inversion method, denoting as Direct inversion_MKL, that treat $(-A^TFA)$ as a dense matrix is provided. This method (Direct inversion_MKL) is achieved by the use of the parallel inversion solver based on Intel MKL (Wang et al. 2014), a high-performance parallel matrix computing library provided by the Intel Corporation. Besides, the direction inversion method with the utilization of sparse Cholesky factorization(SCF) is also provided, denoting as Direct inversion_SCF.

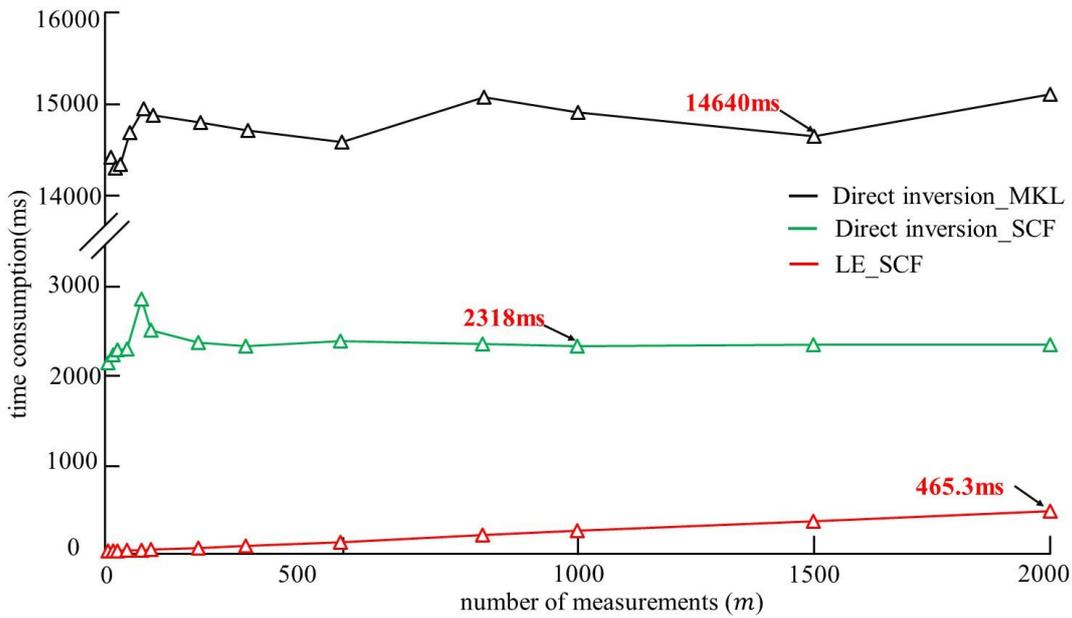


Figure 4 Time consumption of the Jacobian matrix computation for different number of measurements at each iteration

Typically, the algorithm can estimate the nodal water demand after 15 iterations (Figure S3). Figure 4 gives the average time consumption of the Jacobian matrix computed by the three methods at each iteration. From Figure 4 we can find that the time consumption of the LE_SCF method (the red line) increases linearly with the increase of the number of measurements (m), while the time consumption of the direct inversion method (Direct inversion_MKL and Direct inversion_SCF) are relatively stable. This is because that the developed LE_SCF method calculates the Jacobian matrix by solving m linear equations, whereas the direct inversion methods always solve the inversion of the matrix $(-A^TFA) \in \mathbb{R}^{n \times n}$, thus its time consumption does not vary with the change of the number of measurements.

From Figure 4 we can find that the time consumption of the Direct inversion_SCF method (the green line) takes 16% of the Direct inversion_MKL. This highlights the effectiveness of the utilization of sparse Cholesky factorization in improving computation efficiency. In addition, the time consumption of the developed approach (red line) is far less than that of the direct inversion method. Even when the number of measurements reaches 2000, the time consumption of the developed algorithm takes only 20% of the Direct inversion_SCF method. Generally, the developed method can estimate the nodal water demand after 15 iterations (Figure S3). In this situation, the total time consumption of Jacobian matrix computation of the developed method at 15 iterations ranges from 0.18s to 6.98s, with the number of measurements increases from 10 to 2000. For comparison, the total time consumption of the Direct inversion_MKL method and Direct inversion_SCF method are about 219s, 34.77s respectively. The developed method that computes the Jacobian matrix by solving a system of linear equations with the use of sparse Cholesky factorization can significantly shorten the time consumption and improve the usability of the algorithm especially for the real-time modeling of a large-scale network.

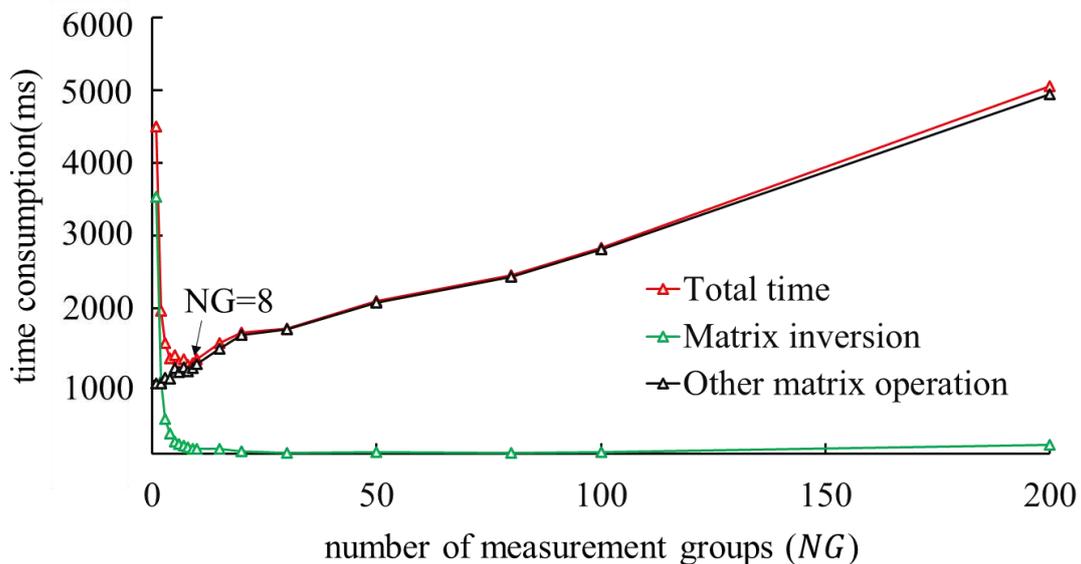


Figure 5 The time consumption of the Hessian matrix inversion for the different number of measurement groups when $m = 2000$

Figure S4 gives the average time consumption of the Hessian matrix inversion at each iteration for the different number 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 Figure 5. It is interesting to note that two different performance-changing stages can be observed from Figure 5 (red line). For the first stage with the number of measurement groups within 8 ($NG < 8$), it shows a rapid decrease of the time consumption; then followed by the second stage with the number of measurement groups exceeding 8 ($NG \geq 8$), the average time consumption increase with the number of measurement groups in a moderate rate. This is because that the total time consumption (red line) is the sum of the time consumption of inversion operation for matrix S (green line) and other matrix operations (black line). As mentioned in the previous section, the time consumption of inversion of matrix S (Eq. (16)) reduces exponentially with the increasing number of measurement groups (NG), as illustrated in Figure 5 (green line). However, the time consumption of other matrix operations ($\sum_i^{NG} \vartheta_i$) increases linearly with the increase of NG (black line). Therefore, the number of measurement groups (NG) should be optimized to balance the time consumption of matrix inversion and other matrix operations. The optimal NG depends on the case networks, the computation environment, and the measurement number. It can be determined by the prearranged numerical experiments. For this case study, NG is optimized by trial and error methods. The optimized NG and time consumptions are shown in Table 1.

Table 1 The optimized NG and 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			

Figure 6 gives 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. (2021 b) is also provided. As shown in Figure 6, the time consumption of the developed approach increases as the number of measurements increases, and it is always within 1000ms. Comparatively, the time consumptions of Sh_Mo based method

increase from 2315ms to 165204 ms. On average, the time consumption of the Sh_Mo method is almost 60 times more time-consuming than the developed approach. The main reason is that the Sh_Mo method calculates the Hessian matrix inversion is calculated after m iterations (Chu et al. 2021 b), which is equal to the developed approach with $NG = m$. Thus Sh_Mo method is not efficient as it consumes intensive time for the other matrix operation. Comparatively, the developed approach can select the appropriate number of groups (NG) to significantly shorten the time consumption of Hessian matrix inversion.

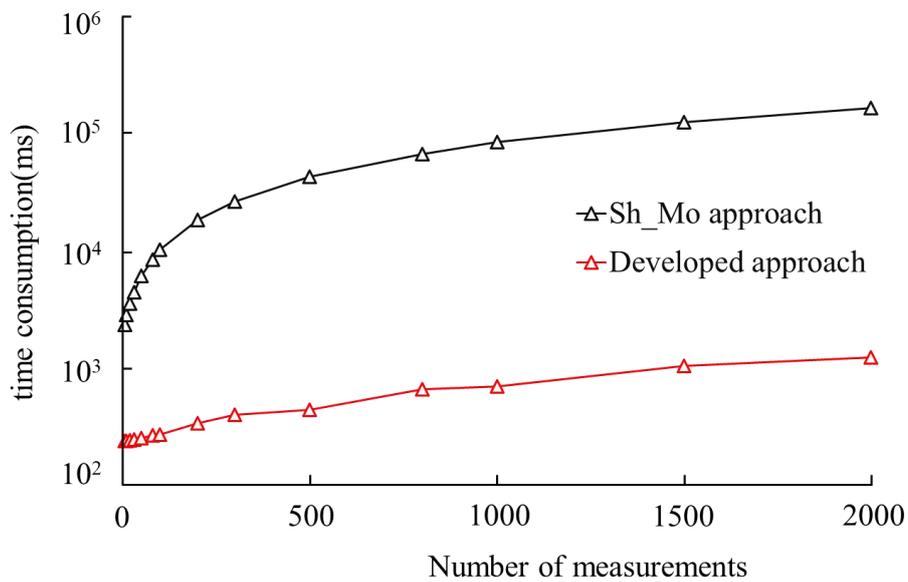


Figure 6 Time consumption of the Hessian matrix inversion for different number of measurements

3.1.2 Modeling Accuracy

Figure S5 gives the cumulative probability of the absolute error for the 12523 nodal water demand. The absolute error is computed by the deviation between the estimated nodal water demand and the true value. The prior error is computed by the absolute deviation from the prior value to the true value. As shown in Figure S5, the estimated nodal water demand error is close to the prior error. This is because that the developed approach tries to minimize the difference between estimated nodal water ($X_{t|t}$) and the prior value ($X_{t|t-1}$). Therefore, the algorithm may have a similar performance as the same prior value is used.

Figure S6 gives the cumulative probability of absolute errors of nodal pressure at the 12523 nodes. The absolute error is computed by the deviation from the estimated nodal pressure to the true value. The prior pressure error is the absolute deviation between the true pressure to the computed pressure that directly using the prior demand as model input. To facilitate the analysis, the estimated results for $m = 30, 100, 1000, 2000$ are shown in Figure 7. As shown in Figure 7, the estimated pressure error is smaller than the prior pressure error. Besides, the nodal pressure errors decrease with the increase of the number of measurements. For example, 50% of the nodes have an error greater than 0.5m when $m = 30$; while 20% of the nodes have an error greater than 0.5m when $m = 100$. This significant improvement highlights the importance of the utilization of measurement to improve modeling accuracy.

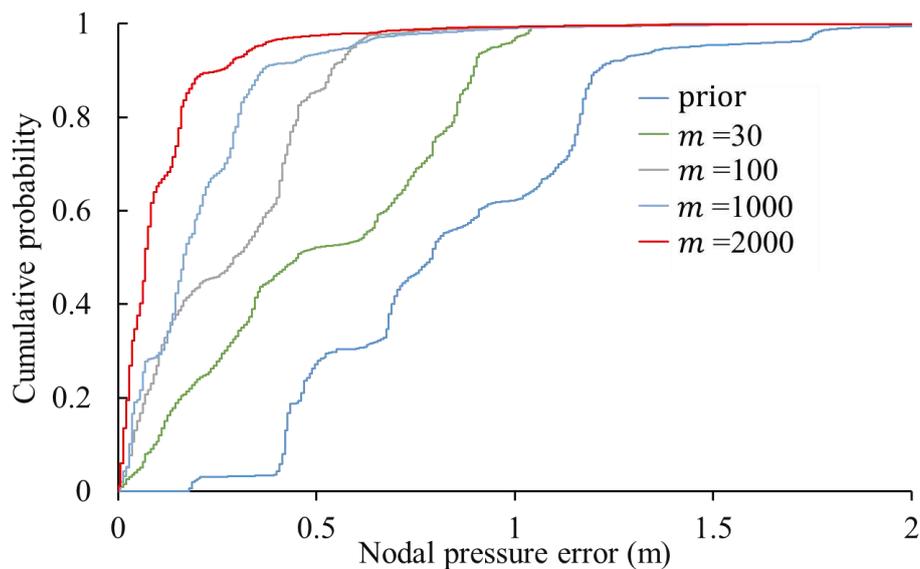


Figure 7 Cumulative probability of absolute errors of nodal pressure

4 Conclusions

The estimation of nodal water demand in a timely manner is critical to the real-time modeling of WDS. The Jacobian matrix computation and Hessian matrix inversion are the processes that dominate the main computation time. This paper developed an efficient approach to improve the

computational efficiency typically for the 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 computational efficiency. An efficient Hessian matrix inversion method based on matrix partition and Iterative Woodbury-Matrix-Identity Formula is developed. The approach provides a more flexible means to balance time consumptions of matrix inverse operation and other matrix operations, and thus obtains a significant improvement when dealing with sensor-dense networks. Results show that the time consumptions of Jacobian matrix computation and Hessian matrix inversion are about 20% and 1.6% of the existing approach, respectively.

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

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

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.

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.

Compliance with Ethical Standards

Ethical Approval Not applicable.

Consent to Publish Not applicable.

Consent to Participate Not applicable.

References

- Bragalli, C., Fortini, M., and Todini, E. (2016). "Enhancing Knowledge in Water Distribution Networks via Data Assimilation." *Water Resources Management*, 30(11), 3689-3706.
- Chu, S., Zhang, T., Xu, C., Yu, T., and Shao, Y. (2021a). "Dealing with Data Missing and Outlier to Calibrate Nodal Water Demands in Water Distribution Systems." *Water Resources Management*.
- Chu, S., Zhang, T., Li, X., Li, K., and Shao, Y. (2021b). "Approach for Water Distribution System Model Calibration Based on Iterative Sherman–Morrison Formula." *Journal of Water Resources Planning and Management*, 147(5), 04021017.
- Chu, S., Zhang, T., Shao, Y., Yu, T., and Yao, H. (2020). "Numerical approach for water distribution system model calibration through incorporation of multiple stochastic prior distributions." *Science of The Total Environment*, 708, 134565.
- Coulbeck, B., and Orr, C.-H. (1988). *Computer applications in water supply*, Research Studies Press.
- Diaz, S., Minguez, R., and Gonzalez, J. (2017). "Calibration via Multi-period State Estimation in Water Distribution Systems." *Water Resources Management*, 31(15), 4801-4819.
- Di Nardo, A., Di Natale, M., Gissoni, C., and Iervolino, M. (2015). "A genetic algorithm for demand pattern and leakage estimation in a water distribution network." *Journal of Water Supply Research and Technology-Aqua*, 64(1), 35-46.
- Dini, M., and Tabesh, M. (2014). "A New Method for Simultaneous Calibration of Demand Pattern and Hazen-Williams Coefficients in Water Distribution Systems." *Water Resources*

- Management, 28(7), 2021-2034.
- Du, K., Long, T.-Y., Wang, J.-H., and Guo, J.-S. (2015). "Inversion Model of Water Distribution Systems for Nodal Demand Calibration." *Journal of Water Resources Planning and Management*, 141(9).
- Goulter, I. C., and Bouchart, F. (1990). "Reliability-Constrained Pipe Network Model." *Journal of Hydraulic Engineering*, 116(2), 211-229.
- Jung, D., Choi, Y., and Kim, J. (2016). "Optimal Node Grouping for Water Distribution System Demand Estimation." *Water*, 8(4), 160.
- Kang, D., and Lansey, K. (2009). "Real-Time Demand Estimation and Confidence Limit Analysis for Water Distribution Systems." *Journal of Hydraulic Engineering*, 135(10), 825-837.
- Kapelan, Z. S., Savic, D. A., and Walters, G. A. (2007). "Calibration of water distribution hydraulic models using a Bayesian-Type procedure." *Journal of Hydraulic Engineering*, 133(8), 927-936.
- Kumar, S. M., Narasimhan, S., and Bhallamudi, S. M. (2010). "Parameter Estimation in Water Distribution Networks." *Water Resources Management*, 24(6), 1251-1272.
- Law, K. J. H., Stuart, A. M., and Zygalakis, K. C. (2015). "Data Assimilation: A Mathematical Introduction." *Revista Brasileira De Meteorologia*, 26(3), 433-442.
- Meirelles, G., Manzi, D., Brentan, B., Goulart, T., and Luvizotto, E., Jr. (2017). "Calibration Model for Water Distribution Network Using Pressures Estimated by Artificial Neural Networks." *Water Resources Management*, 31(13), 4339-4351.
- Moasheri, R., and Jalili-Ghazizadeh, M. (2019). "Locating of Probabilistic Leakage Areas in Water Distribution Networks by a Calibration Method Using the Imperialist Competitive Algorithm." *Water Resources Management*.
- Ostfeld, A., Uber, J. G., Salomons, E., Berry, J. W., Hart, W. E., Phillips, C. A., Watson, J.-P., Dorini, G., Jonkergouw, P., Kapelan, Z., di Pierro, F., Khu, S.-T., Savic, D., Eliades, D., Polycarpou, M., Ghimire, S. R., Barkdoll, B. D., Gueli, R., Huang, J. J., McBean, E. A., James, W.,

- Krause, A., Leskovec, J., Isovitsch, S., Xu, J., Guestrin, C., VanBriesen, J., Small, M., Fischbeck, P., Preis, A., Propato, M., Piller, O., Trachtman, G. B., Wu, Z. Y., and Walski, T. (2008). "The Battle of the Water Sensor Networks (BWSN): A Design Challenge for Engineers and Algorithms." *Journal of Water Resources Planning and Management*, 134(6), 556-568.
- Piller, O., Elhay, S., Deuerlein, J., and Simpson, A. R. (2017). "Local Sensitivity of Pressure-Driven Modeling and Demand-Driven Modeling Steady-State Solutions to Variations in Parameters." *Journal of Water Resources Planning and Management*, 143(2).
- Qin, T., and Boccelli, D. L. (2019). "Estimating Distribution System Water Demands Using Markov Chain Monte Carlo." *Journal of Water Resources Planning and Management*, 145(7).
- Rajakumar, A. G., Kumar, M. S. M., Amrutur, B., and Kapelan, Z. (2019). "Real-Time Water Quality Modeling with Ensemble Kalman Filter for State and Parameter Estimation in Water Distribution Networks." *Journal of Water Resources Planning and Management*, 145(11).
- Rossman, L. A. (2000). "EPANET 2: users manual."
- Shang, F., Uber, J. G., Waanders, B. G. v. B., Boccelli, D., and Janke, R. (2008). "Real Time Water Demand Estimation in Water Distribution System." *Water Distribution Systems Analysis Symposium 2006*, 1-14.
- Wang, E., Q. Zhang, B. Shen, G. Zhang, X. Lu, Q. Wu, and Y. Wang. 2014. "Intel Math Kernel Library." In *High-performance computing on the Intel® Xeon Phi™*, 167–188. Berlin: Springer.
- Weber, R., and Hos, C. (2020). "Efficient Technique for Pipe Roughness Calibration and Sensor Placement for Water Distribution Systems." *Journal of Water Resources Planning and Management*, 146(1).
- Zhou, X., Xu, W. R., Xin, K. L., Yan, H. X., and Tao, T. (2018). "Self-Adaptive Calibration of Real-Time Demand and Roughness of Water Distribution Systems." *Water Resources Research*,

54(8), 5536-5550.

Supplementary Files

This is a list of supplementary files associated with this preprint. Click to download.

- [Supplementary.docx](#)