Available online at www.sciencedirect.com

ScienceDirect

Procedia Engineering 119 (2015) 554 – 563

Procedia Engineering

www.elsevier.com/locate/procedia

13th Computer Control for Water Industry Conference, CCWI 2015

Modelling and simulation of water distribution systems with quantised state system methods

Daniel Paluszczyszyn[a,*], Piotr Skworcow[a], Bogumil Ulanicki[a]

[a] *Water Software Systems, De Montfort University, The Gateway, Leicester, LE1 9BH, UK*

Abstract

The work in this paper describes a study of quantised state systems in order to formulate a new framework within which water distribution systems can be modelled and simulated. In contrast to the classic time-slicing simulators, depending on the numerical integration algorithms, the quantisation of system states would allow accounting for the discontinuities exhibited by control elements in a more efficient manner, and thereby, offer a significant increase in speed of the simulation of water network models. The proposed approach is evaluated on a case study and compared against the results obtained from the Epanet2 simulator and OpenModelica.

Peer-review under responsibility of the Scientific Committee of CCWI 2015

Keywords: modelling; quantised state systems; QSS; simulation; water distribution systems;

1. Introduction

This paper tries to open a new paradigm for modelling and simulation of water distribution system (WDS). It is proposed within this article to model and simulate WDSs with use of the quantised state system (QSS) methods. Majority of water network simulators use a time-slicing approach, typical for simulation of continuous systems, here quantisation of the states is proposed leading to an asynchronous discrete-event simulation model. Such an approach in which water network are simulated using the quantisation-based integration methods has not been applied to WDSs. Section 2 provides an epitome of numerical difficulties may be encountered in WDSs simulation. Next, an explanatory introduction is given to the QSS methods in Section 3 illustrating their properties on a simple water system. Section 4 describes tools used and evaluates the QSS methods on a case study. Finally, Section 5 concludes this paper.

* Corresponding author. Tel.: +44-116-207-8939.
E-mail address: paluszcol@dmu.ac.uk

Peer-review under responsibility of the Scientific Committee of CCWI 2015
doi:10.1016/j.proeng.2015.08.908

Nomenclature

ΔQ quantum
$h(t)$ head of tanks
q fixed flow
$\mathbf{q}(t)$ quantised state vector
t time
$\mathbf{x}(t)$ state vector

2. Time-slicing simulation of water networks

Simulation of water distribution networks aims to provide solution of differential algebraic equations used to formulate the mathematical representation of a WDS. Simulation is an invaluable tool in the assessment of WDS response to different operational actions (e.g. valves opening and closing) or control strategies prior to applying the actions to a real water network.

The WDS simulation techniques can be classified into: (i) steady-state simulation, (ii) extended-period simulation and (iii) transient simulation. Steady-state simulations represent a snapshot of the WDS operation; i.e. demands and pressures at all nodes and flows in all pipes do not vary in time. In real systems, however, the loading conditions and states vary in time. Thus, to evaluate performance of a WDS over a period of time, an extended-period simulation (EPS) is used. This type of analysis considers fluctuations of water level in tanks and demands in discrete time intervals, but with an assumption that in each time interval, the system is in a steady state. The transient simulation provides the most accurate simulation of WDS as it considers naturally unsteady flow conditions incorporating transient analysis. Due to the complexity of this approach it is not yet adopted by many practitioners; mainly limited to specialised applications such as surge studies due to switching control elements.

Simulation of a WDS is not an easy task; solution cannot be obtained analytically. For tree-shaped networks a solution can be obtained by applying the flow continuity equation at all the nodes, but in practice, WDSs are almost never pure tree-shaped networks. The analysis of a water network in a loop configuration presents more challenges. Although, over the last decades a number of methods were proposed. The most noticeable methods include: (i) Hardy Cross method [7], (ii) linear theory method [24], (iii) Netwon-Raphson method [22], (iv) linear graph theory method [12], (v) an approach involving optimisation methods [6], (vi) global gradient method [23]. Implementations of the methods and their hybrids resulted in dozens hydraulic simulators created over the years; e.g. Epanet2, Finesse, H2Onet, WaterCAD. More extensive list of simulators and their reviews can be found in [21] or [2].

Majority of water networks analysis methods and simulators are based on a time slicing approach; i.e. numerical methods, used in computer simulation of a system characterised by differential equations (e.g. tank dynamics), require the system to be approximated by discrete quantities. The solution of difference equation is calculated at fixed points in time. This feature of mapping a discrete-time set to a continuous-state set made the discrete-time approach to simulation applicable in many fields, including water networks analysis.

Additionally, it is assumed in EPS, that the system is in a steady state between the successive time stamps. But in fact, a real WDS continually adjusts itself in response to changing requirements of the users. This rises an important issue about the model's hydraulic fidelity; especially in WDS models with pumps operated on the water level in tanks, as if the time interval is not appropriate the events that actually happened in the real water network might be overlooked. Figure 1 illustrates effect of different time intervals on the hydraulic simulation results obtained from Epanet2.

Furthermore, some elements of WDS (see Table 1) may cause numerical difficulties (convergence problems) in simulation, due to their inherent non-smooth and discontinuous characteristics [1,10,16]. For example, serious convergence problems may be encountered when simulating in Epanet2 a complex and large-scale WDS consisting of hundreds of elements such as those listed in Table 1. This is mainly due to the fact that switching events may not happen at the pre-selected time steps and then additional intermediate time steps need to be introduced. Such an ap-

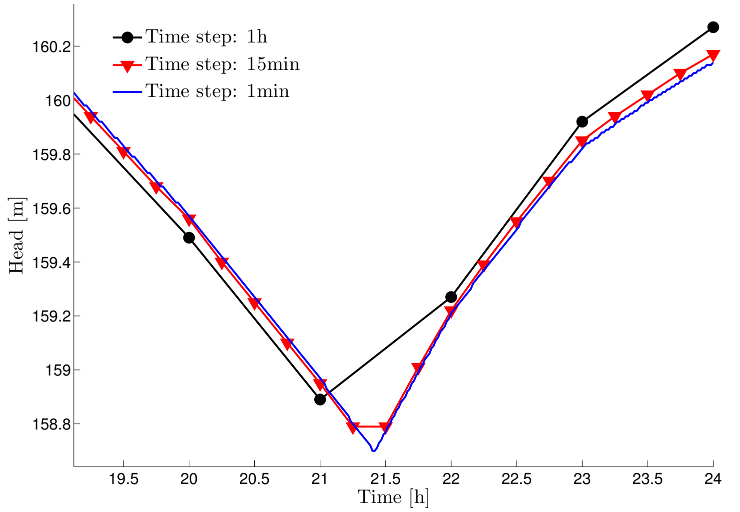
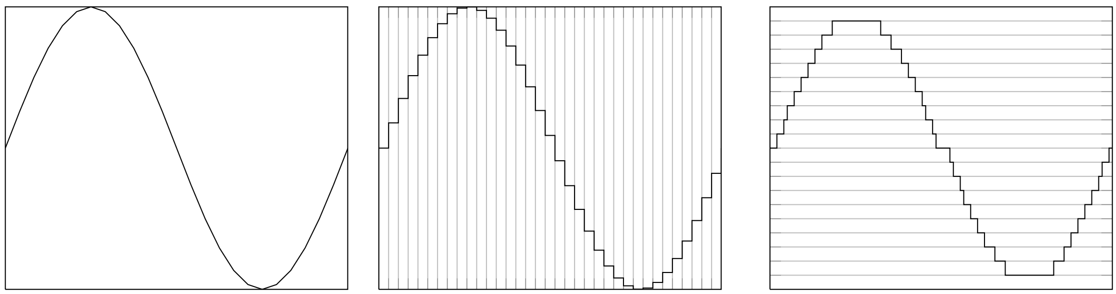


Fig. 1. Illustrating the impact of a different sampling time on the hydraulic analysis results obtained in Epanet2.



(a) Continuous (b) Discrete time (c) Discrete state

Fig. 2. Time and state discretisation of a continuous system [18].

proach is used in Epanet2, which introduces the intermediate steps when simulating water network models containing control elements.

Table 1. Non-smooth and discontinuous elements in water distribution systems.

Water distribution system components
control valves (e.g. pressure reducing valve (PRV), non-return valve (NRV), altitude control valve (ACV), pressure sustaining valve (PSV))
pump controlled by tank level
pump controlled by many tanks' levels and time
control expressed as a computer program

In order to perform a more accurate state calculation an approach based on a discrete event solution can be used. Two aspects of the system can be made discrete, time and state. These two types of discretisations are illustrated in Figure 2. In the discrete-event approach due to the event-dependent time advance, only important simulation points regarding the dynamics of the system are simulated, while idle periods of the system (intervals where no changes in states occur) are simply skipped whereas in the fixed-increment approach time advance also simulates inactive periods [5].

3. Event-based simulation of water networks

Discrete-event systems may be modelled using Petri nets, finite-state machines, Markov chains, state charts or the discrete-event specification (DEVS) formalism. However, WDSs are hybrid systems; i.e. can exhibit both discrete and continuous dynamics simultaneously e.g. a discrete pump control based on the water level in the tank. While discrete part of WDS can be naturally and straightforward represented in discrete-event framework, to account for the continuous part within a discrete-event framework, a quantisation-based integration technique can be used to transform the considered continuous system into a system described by a sequence of events. The idea to approximate a continuous system by a discrete-event simulation, is based on a quantisation function, that enables transformation the continuous state variables into the quantised discrete valued variables. The idea initially introduced in [25] was later reformulated in [15] and defined as the QSS methods.

3.1. Quantised state systems

The purpose of employing the QSS methods is replace the time discretisation of classic numerical integration algorithms by the quantisation of the state variables [9], thereby provide means to simulate the continuous part of the WDS model formulated within the discrete-event framework. The QSS methods can be defined as follows.

Consider a time-invariant state equation system

$$\dot{\mathbf{x}}(t) = \mathbf{f}(\mathbf{x}(t), t) \tag{1}$$

where $\mathbf{x}(t)$ is the state vector. The QSS method integrates an approximate system, which is called the quantised state system.

$$\dot{\mathbf{x}}(t) \approx \mathbf{f}(\mathbf{q}(t), t) \tag{2}$$

where $\mathbf{q}(t)$ is the quantised version of the state vector $\mathbf{x}(t)$. The $\mathbf{q}(t)$ and $\mathbf{x}(t)$ are related components by hysteretic quantisation function (i.e each component of $q_i(t)$ is related to the corresponding state variable $x_i(t)$ by a hysteretic quantisation function). The hysteretic quantisation function could be defined as follows:

$$q_i(t) = \begin{cases} x_i(t) & \text{if } |x_i(t) - q_i(t^-)| \geq \Delta Q_i \\ q_i(t^-) & \text{otherwise} \end{cases} \tag{3}$$

where ΔQ_i is defined as the quantum, which is usually constant.

Figure 3 illustrates a typical quantisation function $q(t)$ with uniform quantisation intervals, obtained with a hysteresis window ε. Depending on quantisation method the quantisation function can be piecewise constant (QSS1) [15], linear (QSS2) [13] or parabolic (QSS3) [14]. The family of QSS methods include also methods for stiff systems: Backward QSS and Linearly Implicit QSS [17].

The QSS-based algorithms are of particular interest for the simulation of systems exhibiting discontinuities, as state events can be handled much more efficiently by state-quantisation algorithms compared to time-slicing algorithms. Hence, the QSS methods are well suited for the simulation of hybrid systems. Moreover, the QSS-based solvers are very promising for the simulation of large-scale models, as they exploit the sparsity inherent in these models naturally and directly. Additionally, QSS provides features that ensure convergence and stability even for nonlinear systems [5].

3.2. Illustrative example

In order to illustrate properties of the QSS methods, consider a simple water system, namely WDS1, illustrated in Figure 4a. For simplicity reasons, the tanks in WDS1 are assumed to be independent. Notations h_1, h_2 and h_3 represent heads of those cylindrical tanks with different diameters of 5, 20 and 40 m respectively.

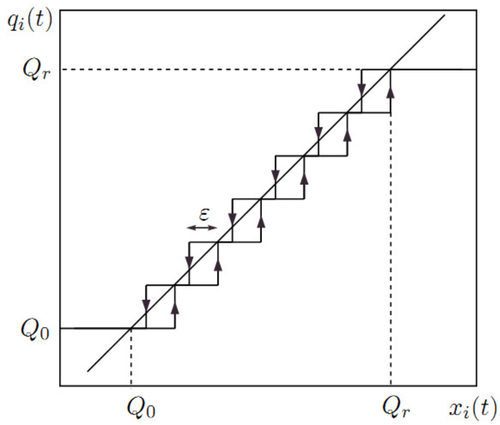


Fig. 3. A quantisation function with the uniform quantisation levels [15].

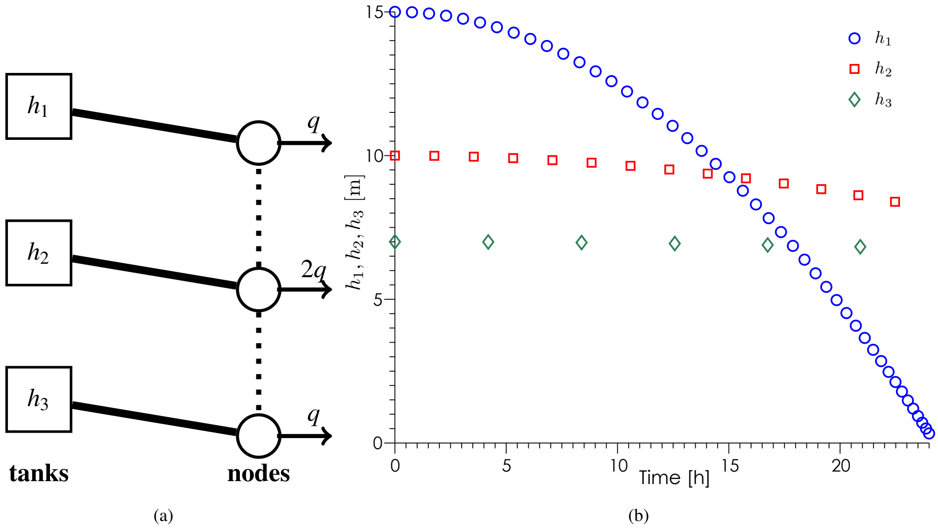


Fig. 4. (a) A simple water distribution system (WDS1); (b) Tanks' trajectories simulated with use of the QSS2 method.

The differential equations describing dynamics of these tanks are as follows:

$$\begin{aligned}\frac{dh_1(t)}{dt} &= \frac{1}{6.25\pi}(-q)\\ \frac{dh_2(t)}{dt} &= \frac{1}{100\pi}(-2q)\\ \frac{dh_3(t)}{dt} &= \frac{1}{400\pi}(-q)\end{aligned} \tag{4}$$

where q represents fixed outflows from the tanks.

For elucidation purposes only the tanks' dynamics are considered here and other hydraulic elements are neglected. The system was simulated using the QSS2 method and the quantum of 0.001 for all the states. The heads' trajectories obtained from simulation, with initial heads of $h_1(0) = 15$, $h_2(0)=10$ and $h_3(0) = 7$, are depicted in Figure 4b.

It can be seen that the QSS2 method adapts to the different tank's dynamics by scheduling more integrator events when the trajectory has a high curvature, while the number of integrations decrease as the trajectory resembles a straight line. This example also illustrates the asynchronous property of the QSS methods as the variable h_1 changed its state 45 times while the variables h_2 and h_3 changed 14 and 6 times, respectively. In simulation of water distribution systems with a large number of tanks with different dynamics, the asynchronous property might be especially useful; tanks' states will evolve individually as there is no need to update them simultaneously.

4. Modelling and simulating water networks with QSS methods

The goal of this section is to compare the simulation run-times and the hydraulic accuracy of the QSS approach against the conventional time-slicing methods. More specifically, the different QSS methods in QSS Solver against the differential algebraic system solver (DASSL) solver of OpenModelica and Epanet2 simulation engine. A representative theoretical WDS was used to perform investigation of the accuracy and performance of simulators.

4.1. Modelling and simulation environments

A number of software environments and tools were used in this work. While some of the investigated tools were fully developed and supported other were still in the development phase. Such tools offer only limited functionalities and often were released without manuals what significantly hindered work with them. The final list of modelling and simulation environments includes: (i) Epanet2, (ii) QSS Solver and (iii) OpenModelica.

Epanet2 [20] is one of the most recognised water network solver in the water distribution research area. The Epanet2 version used was 2.00.12.

In addition to Epanet2, it was decided to establish the computational implementation of the benchmark network with use of the Modelica object-oriented modelling language. Modelica is a free modelling language that supports the equation-based, object-oriented modelling methodology, and facilitates the description of large and complex systems. An open-source OpenModelica [11] was used to investigate the performance of DASSL [19] numerical engine in simulation of WDSs. DASSL uses the backward differentiation formulas of orders one through five and it solves the nonlinear system at each time-step by Newton's method. The OpenModelica version used was 1.9.1Beta2.

For simplicity reasons, the most of QSS implementations were incorporated into the discrete-event simulation engines such as DEVS, e.g. the QSS methods in PowerDEVS [3]. Such an approach, however, is not fully efficient due to increased computational load within discrete event simulation mechanism. To overcome this problem [9] created a stand-alone QSS Solver. Modelling in QSS Solver is done with the μModelica language, a subset of the standard Modelica language. Although, the used QSS Solver version enabled to model a simple systems, the tool is still under development, and a number functionalities such as linking to external libraries with nonlinear equations solvers were not available at the time this research was conducted.

Simulations were performed on a PC workstation powered by Intel i5-2500K processor. The measured CPU time should not be considered as an absolute ground-truth since it will vary from one computer system to another, but the relative ordering of the methods is expected to remain the same.

4.2. Case study: Network with pump controlled by tank's level

The next water network, namely WDS2, considered for a preliminary investigation is illustrated in Figure 5. The aim of this theoretical WDS is to provide water to an industrial user. The fixed-speed pump, initially closed, is controlled by the water level in the cylindrical tank (the initial level assumed to be 1 m). the pump was set to operate when the water level in tank reaches 15 m and when the water level drops below 10 m the pump was switched off. The extended period simulations were performed for a period of 24 hours.

Table 2 presents the simulations' details for all the simulators. The comparison includes the measured simulation run-time and the number of result points describing the tank's dynamics for different settings of their respective tolerance/quantum parameter.

In the time-slicing environments, Epanet2 and OpenModelica, the shorter step interval or increased tolerance precision improved the simulation accuracy (see Figure 6 and Figure 7), but resulted in the longer simulation run-times. When the run-times of the time-slicing simulators were compared against the QSS1-based simulation, it was noticed, that for the QSS quantum $\leq$1e-4, the QSS1 simulations lasted significantly longer due to high number of detected state

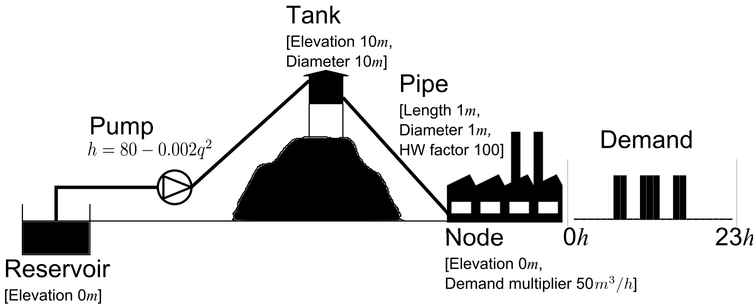


Fig. 5. A simple water distribution system (WDS2).

Table 2. Comparison of simulations for WDS2. (⋆) - 24 result points were requested in OpenModelica simulations, but the output file contained 46 points. This is due to switching events, around which OpenModelica engine introduced additional points to determine the accurate time of the event.

Simulator	Interval time [s]	Tolerance/ Quantum	Number of output points describing tank dynamics	Run-time [s]
Epanet2	3600	1e-3	24	0.0011
	60	1e-3	1440	0.0069
OpenModelica	3600	1e-3	46(⋆)	0.0049
	3600	1e-6	46(⋆)	0.0072
QSS Solver (QSS1)	-	1e-1	19	<1e-3
	-	1e-2	180	<1e-3
	-	1e-3	1850	<1e-3
	-	1e-4	18534	0.0156
	-	1e-5	185388	0.0468
QSS Solver (QSS2)	-	1e-1	6	<1e-3
	-	1e-2	9	<1e-3
	-	1e-3	14	<1e-3
	-	1e-4	30	<1e-3
	-	1e-5	80	<1e-3
QSS Solver (QSS3)	-	1e-1	6	<1e-3
	-	1e-2	9	<1e-3
	-	1e-3	10	<1e-3
	-	1e-4	17	<1e-3
	-	1e-5	22	<1e-3

events. However, this was expected, as in the QSS methods, the choice of the quantum is both reverse proportional to the simulation time and directly proportional to the quantisation error [4].

Since the obtained head trajectories differ in the number of result points, the accuracy of the particular simulation can only be approximated. Therefore, the obtained results were plotted in Figure 6 for a visual inspection. Results obtained from OpenModelica were used as a benchmark, instead the results obtained from Epanet2, as nowadays, DASSL represents state-of-art multi-purpose differential algebraic equation (DAE) solver used in many commercial simulation environments e.g. in Dymola [8] whereas Epanet2, despite its established position, has not been improved for years.

It can be clearly seen in Figure 6 that QSS1-based simulations with the quantisation step of 1e-1 and 1e-2 are imprecise. This is because the QSS method used in the simulation was a first-order accurate. In this method, to achieve a small simulation error the tolerance needs to be decreased but then the number of output points will be larger. Also, the Epanet2 simulation with the sampling time of 1 hour cannot be treated as accurate when compared against the OpenModelica results

For a more detailed inspection the plot was zoomed around area of the first switching event occurrence (around 1h 34min). The zoomed section is pictured in Figure 7. Amongst the plotted trajectories only the QSS1-based simulation, with the quantum of 1e-5, detected the event at the same time as the reference time obtained from the OpenModelica

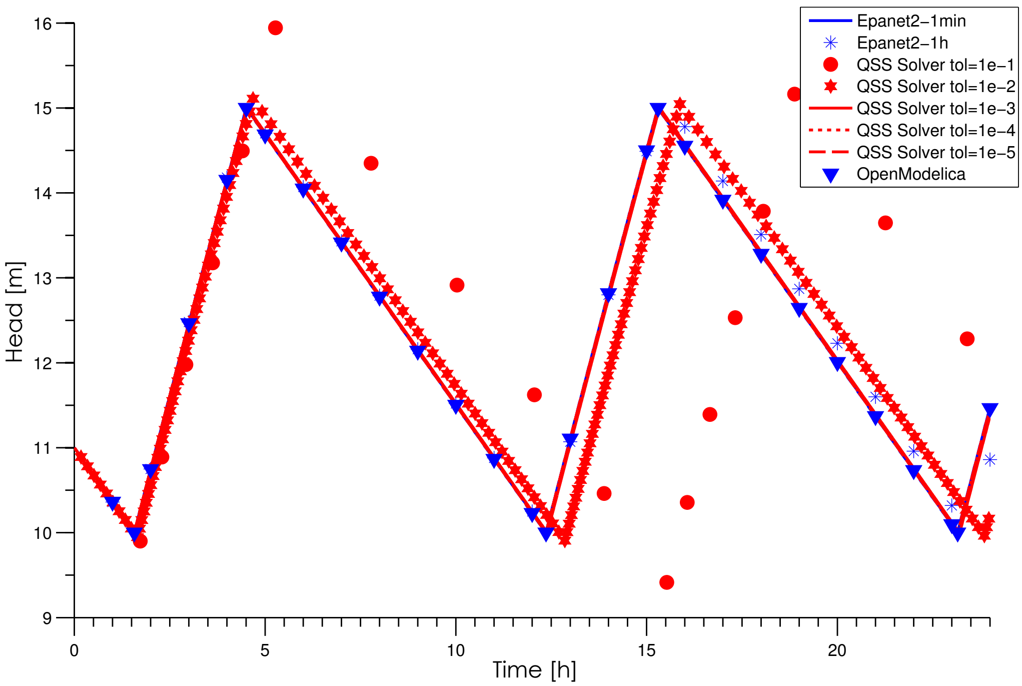


Fig. 6. Illustrating the tank's head trajectories obtained from all simulators.

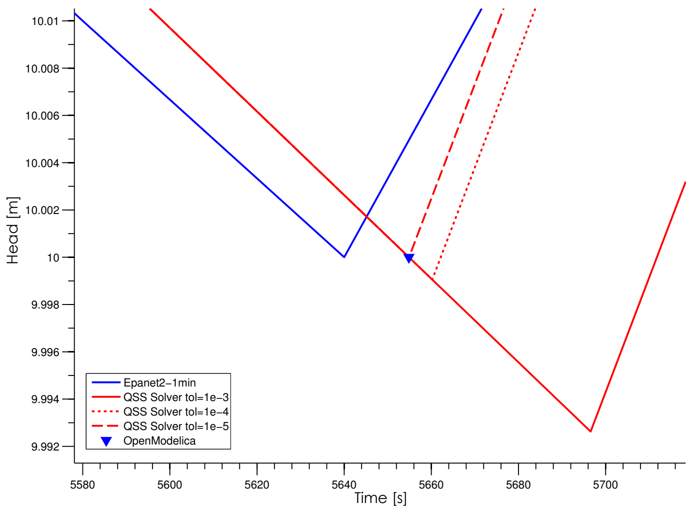


Fig. 7. Depicting a zoomed section of the tank's head trajectories.

simulation. Although, time of event occurrence in QSS 1e-5 was identical with the benchmark time this resulted in the longer simulation run-time and significantly higher number of output points than other solutions.

The QSS1 methods were clearly not outperforming the classical time-domain approaches when simulation of WDS2 was considered. While the first-order QSS method was capable to simulate accurately (for quantum ≤1e-4) the considered system it required a longer simulation run-time and produced a large number of output points. But choosing the quantum is not a straightforward task. One way to determine the quantum is to know the signal magnitude and to

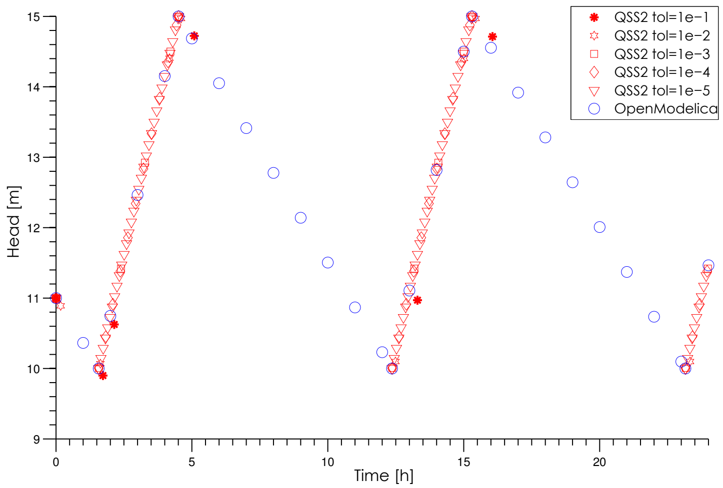


Fig. 8. Comparison of QSS2 against OpenModelica.

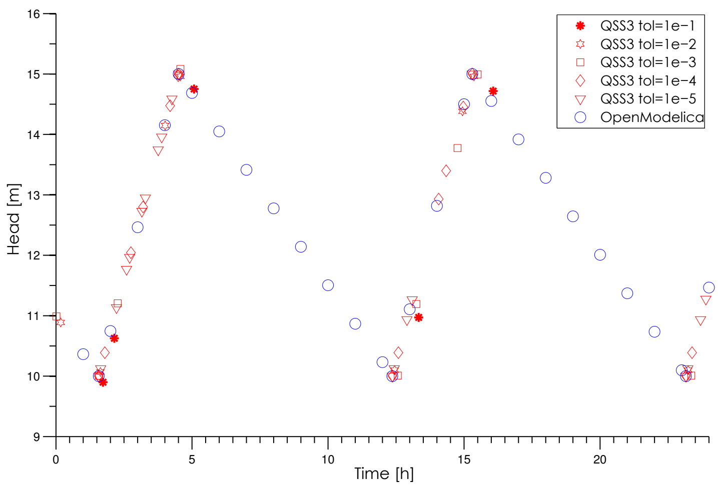


Fig. 9. Comparison of QSS3 against OpenModelica.

divide it by a reasonable number to obtain acceptable results. Other solution, is to use a method defined in [5], which allows to calculate the quantum based on the error rate defined by the user. However, the recommenced approach is to use the higher order QSS methods, especially the third order method, since it provides the smaller simulation time and it gives more freedom in the choice of the quantum without any impact on the quantisation error [14].

In QSS2 [13], the quantised state variables evolve in a piecewise linear way with the state variables following piecewise parabolic trajectories. In the third-order extension, QSS3 [14], the quantised states follow piecewise parabolic trajectories, while the states themselves exhibit piecewise cubic trajectories.

Results from the simulations in QSS Solver with the higher order QSS methods are presented in Table 2, Figure 8 and Figure 9. QSS2 and QSS3 methods not only significantly decreased the simulation run-times, but also, they were much more accurate than the QSS1 method and aligned with results obtained from OpenModelica. Additionally, even for quantum values ≤1e-4, the number of the output points was significantly smaller than in the discrete-time simulators. While the first order state integration algorithm, QSS1, did not bring significant improvement over the classical time-discrete methods, the higher order QSS methods outperformed the time-domain-based simulators; QSS2 and QSS3 algorithms can achieve a good accuracy without excessive increment in the number of steps.

5. Conclusions

In this paper a novel QSS-based approach has been applied to model and simulate WDSs. The preliminary evaluations on a water network with the pump operated on the water level margin, have demonstrated that the QSS methods compete well against conventional simulators in terms of accuracy. From a functional level the use of the QSS methods is analogous to ordinary time-discrete methods; if a reasonably small quantisation step is chosen then the algorithm provides a simulated trajectory that is sufficiently consistent with the analytical solution. However, the smaller quantum have resulted in increasing the simulation time. This has been especially visible in case of QSS1 simulations. Nevertheless, when the higher order QSS methods, QSS2 and QSS3, were employed the simulation run-time have decreased drastically while obtaining the same accuracy as the benchmark simulation performed in OpenModelica. It has been clearly seen that the higher order QSS methods are more efficient than conventional discrete-time solvers. However, the current state-of-art of the associated tools do not yet allow to exploit fully the potential of the QSS methods. To fully appreciate the efficiency of the QSS methods a simulation tool must be developed dedicated explicitly to model and simulate water distribution networks.

References

[1] Afshar, M. and Rohani, M. (2009). Embedded modified Euler method: an efficient and accurate model. *Water Management*, 162(3):199.
[2] Balut, A. and Urbaniak, A. (2011). Management of water pipeline networks supported by hydraulic models and information systems. In *Proceedings of the 12th International Carpathian Control Conference (ICCC)*, pages 16–21.
[3] Bergero, F. and Kofman, E. (2011). PowerDEVS: a tool for hybrid system modeling and real-time simulation. *Simulation*, 87(1-2):113–132.
[4] Capocchi, L., Federici, D., Yazidi, A., Henao, H., and Capolino, G.-A. (2009). New trends and solutions for the simulation of electrical circuits using a discrete event approach. In *Industrial Electronics, 2009. IECON '09. 35th Annual Conference of IEEE*, pages 4365–4370.
[5] Cellier, F. E. and Kofman, E. (2006). *Continuous System Simulation*. Springer, 1 edition.
[6] Collins, M., Cooper, L., Helgason, R., Kennington, J., and LeBlanc, L. (1978). Solving the pipe network analysis problem using optimization techniques. *Management Science*, 24(7):747–760.
[7] Cross, H. (1936). *Analysis of flow in networks of conduits or conductors*. Number Bulletin 286. University of Illinois.
[8] Dassault Systems (2014). Dymola. http://www.3ds.com/products-services/catia/capabilities/systems-engineering/modelica-systems-simulation/dymola/. Accessed on 20-8-2014.
[9] Fernández, J. and Kofman, E. (2014). A stand-alone quantized state system solver for continuous system simulation. *Simulation: Transactions of the Society for Modeling and Simulation International*, 90(7):782–799.
[10] Filion, Y. and Karney, B. (2003). Sources of error in network modelling - a question of prespective. *Journal of the American Water Works Association*, 95(2):119–130.
[11] Fritzson, P. (2010). *Principles of object-oriented modeling and simulation with Modelica 2.1*. John Wiley & Sons.
[12] Kesavan, H. and Chandrashekar, M. (1972). Graph-theoretical models for pipe network analysis. *Journal of the Hydraulics Division*, 98(2):345–364.
[13] Kofman, E. (2002). A second-order approximation for DEVS simulation of continuous systems. *Simulation*, 78(2):76–89.
[14] Kofman, E. (2006). A third order discrete event method for continuous system simulation. *Latin American applied research*, 36(2):101–108.
[15] Kofman, E. and Junco, S. (2001). Quantized state systems. A DEVS approach for continuous system simulation. *Transaction of The Society for Modeling and Simulation International*, 18(3):123–132.
[16] Kovalenko, Y., Gorev, N., Kodzhespirova, I., lvarez, R., Prokhorov, E., and Ramos, A. (2010). Experimental analysis of hydraulic solver convergence with genetic algorithms. *Journal of Hydraulic Engineering*, 136(5):331–335.
[17] Migoni, G., Bortolotto, M., Kofman, E., and Cellier, F. E. (2013). Linearly implicit quantization-based integration methods for stiff ordinary differential equations. *Simulation Modelling Practice and Theory*, 35:118–136.
[18] Nutaro, J. (2005). Discrete event simulation of continuous systems. In *Handbook of Dynamic Systems Modeling*.
[19] Petzold, L. R. (1982). Description of DASSL: A differential/algebraic system solver. Technical report, Sandia National Labs., Livermore, CA (USA).
[20] Rossman, L. (2000). *EPANET Users manual*. US: Risk Reduction Engineering Laboratory, Office of Research & Development, US Enviromental Protection Agency, Cincinnati, Ohio, USA.
[21] Schmid, R. (2002). Review of modelling software for piped distribution networks. In *Working Papers on Water Supply and Environmental Sanitation*. SKAT.
[22] Shamir, U. and Howard, C. (1968). Water distribution systems analysis. *Journal of the Hydraulics Division*, 94(HY1):219–234.
[23] Todini, E. and Pilati, S. (1987). A gradient algorithm for the analysis of pipe networks. In Coulbeck, B. and Orr, C.-H., editors, *Computer Applications In Water Supply: Vol. 1— Systems Analysis And Simulation*, pages 1–20. Research Studies Press Ltd., Letchworth, Hertfordshire, UK.
[24] Wood, D. J. and Charles, C. O. (1972). Hydraulic network analysis using linear theory. *Journal of the Hydraulics division*, 98(7):1157–1170.
[25] Zeigler, B. and Lee, J. (1998). Theory of quantized systems: Formal basis for DEVS/HLA distributed simulation environment. In *SPIE Proceedings*, volume 3369, pages 49–58.