Networking Lessons: From Computers to Water

Iyswarya Narayanan, Venkatesh Sarangan, Arunchandar Vasan
Innovation Labs, Chennai
Tata Consultancy Services
IIT Madras Research Park, Chennai, India

Aravind Srinivasan
Dept. of CS. and UMIACS
University of Maryland
College Park, USA

Anand Sivasubramaniam
Dept. of Comp. Sci.& Eng.
The Pennsylvania State University
University Park, USA

Abstract—As an instance of using IT to green non-IT domains, we consider the question whether lessons from computer networking can be applied in water distribution networks to improve their energy footprint and/or efficiency. Our contributions in this work are: (i) we identify several areas where principles from computer networking can be used to better water distribution; (ii) we focus on a specific infrastructure enhancement problem caused by increasing demands on a water utility network and present solutions (similar to those used in computer networks) that optimize both operational expenditure and total cost of ownership. We validate our solutions through simulations and compare their efficacy against techniques that are traditionally used in enhancing water networks. Our results show that lessons from computer networks can help in enhancing water networks.

I. Introduction

Widespread deployment of Information and Communication Technology (ICT) hardware coupled with the increasing awareness of energy and carbon footprint of human activities has led to recent research whose goal has been to "green" ICT. A complementary trend has been to use ICT to improve sustainability in other domains. Our work falls in the latter category. Specifically, we study how municipal water delivery networks can be improved by using modeling principles and ideas used in computer networks.

Present day water networks face several challenges such as:

- Age: The estimated refurbishment cost of water infrastructure in the US is nearly $335 billion over 20 years.
- Increased demands: Global demand for water has grown from 770 cubic-kilometers in 1900 to 3840 cubic-kilometers in 2000.
- Accounting: World-wide loss to utilities due to theft or leaks is about $14 billion every year.
- Security: Water delivery systems are potential targets for bio-chemical threats, accidental or deliberate.
- Energy efficiency: Energy constitutes 34% of the operating expenditure of water utilities.

The keen reader will observe that some of the challenges outlined above have their counterparts in large scale data delivery networks such as the Internet. Given this, a natural question arises: *Is it possible to leverage the principles used in designing and operating data delivery networks to solve similar problems in water delivery networks?* This paper attempts to answer this question affirmatively. While there have been efforts to enumerate the similarities and differences between computer networks and electrical grids [9], the connection between computer and water networks is largely unexplored.

To begin with, we enumerate several issues in water delivery networks that have a close parallel with computer networks. We then focus on a specific problem in water network infrastructure planning and use algorithmic techniques similar to those used in computer networks, *viz.* techniques based on tree traversal and minimum weighted dominating set, to solve this problem. Specifically, we consider a water utility which needs to handle increasing demands while meeting the required delivery pressure. While traditional solutions use genetic algorithms (GAs) to enhance the delivery network infrastructure, we find that algorithmic techniques can give solutions that are better than GAs.

This paper is organized as follows. Section II enumerates challenges in water distribution networks that can be addressed through networking principles. Section III describes the planning problem in water distribution networks which we focus on. Section IV presents an analysis for the lower bound to the solution cost. Sections V and VI discuss two solutions to the planning problem that are realizable in practice. Section VII presents results of the two approaches from operational cost and life-cycle cost perspectives. Section VIII discusses related work and Section IX concludes.

II. Networking lessons for water sustainability

Infrastructure planning: In water infrastructure planning, the key questions are often: Where should assets be located, what should be their capacities, how should they be operated to optimize energy consumption, cost of ownership, etc. These optimization problems are typically handled using approaches in operations research (OR) and searches such as genetic algorithms (GA). Planning problems in computer networks use graph theoretic techniques [19] that exploit the structure of the underlying topology. Similar graph theoretic frameworks can be used in the planning of water networks as well.

Threat detection: With the advent of bio-chemical sensors and wireless sensor networks, it has become viable to constantly monitor the quality of water and quickly alert response teams in case of threat detection. Therefore a key problem is the placement of sensors to minimize parameters such as the detection time, contamination spread, exposure, etc. This problem has began to draw the attention of water distribution experts [1] and computer networking researchers [11], [28].

Storage location and management: Increasingly, decentralized waste-water treatment is becoming popular in both developed and developing economies [24]. Taking a leaf out of peer-to-peer (P2P) networking, communities that are geographically closer can co-operate and share their stores of reclaimed water

to average out the variations in production and consumption. Strategies used for P2P server placement [27], [8], [18] can possibly be adapted to identify the optimal location(s) for sewage treatment plants and storage tanks for reclaimed water.

Pricing: There is an inherent trade-off between social and economic angles of water pricing, which are resolved by policy decisions. While one can analyze the costs involved in the supply side rigorously and come up with appropriate price points, it is equally important to analyze the demand response to pricing. Algorithmic game theory, which has been used for network pricing [10], is an ideal tool for modeling the reaction of people to pricing changes.

Network tomography: If metering exists at all nodes in the water network, leaks and thefts can be identified easily. However, the cost will be high, and so one has to improvise. This problem has a parallel in Internet research. Because the Internet is vast and distributed, it is impossible to obtain a global view of the entire system. Hence researchers have evolved tomography to infer the global structure from endpoint measurements. Such techniques could be customized in the context of water delivery to detect and localize leaks even if flow meters are not available at all the junctions.

Striping across delivery channels: Non-piped supply of water is common across the world. In developed countries, (higher quality) water is available containers of various capacities. In developing countries, supply through trucks is common in places lacking piped infrastructure or during times of scarcity. Given that water of potable quality is not needed for all purposes except for drinking and cooking, the delivery of water can be striped across these multiple channels of delivery. Hybrid network interface striping ideas can be adapted to decide when and how to stripe the delivery of water through alternate channels while optimizing cost and energy.

III. INFRASTRUCTURE ENHANCEMENT PROBLEM

Water utilities are required to not only supply treated water, but also do so at a minimum acceptable end-point pressure to customers. This minimum pressure requirement is put in place so that end consumer devices can work satisfactorily. For e.g., in the UK, the utility compensates a consumer if the end-point pressure falls below a minimum of 21.3 psi [21].

Consider a typical water utility network, such as the one at Colorado Springs, USA [23].In this network, we increase the demands of all the nodes uniformly by a scale factor, i.e., each demand is scaled by this factor. Once the original demands have been scaled, we evaluate the network in EPANET – a water network simulator from US EPA [4] and observe the fraction of end points for which the delivery pressure falls below the acceptable value.

Figure 1 shows the results of this experiment. The X-axis shows the scale factor and the Y-axis shows the percentage of nodes for which the supply pressure is less than 21.3 *psi*. As we can see, the network is able to absorb the increased demand quite well until a scale factor of 1.9 due to over-provisioning at the time of installation. However, beyond this scale factor
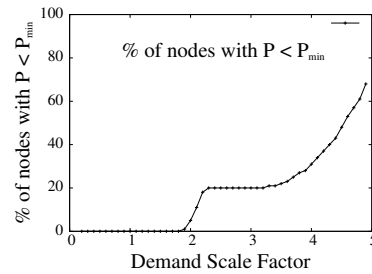

Fig. 1. Pressure violations with increased demands.

of 1.9, significant fraction of nodes become pressure deficient. We have seen a similar behavior with other utility networks too. For the sake of brevity, we do not discuss them. As the demand for water increases in a water delivery network, the flows through pipes increase, and the Physics of piped water flow dictates that as flow increases, the pressure drop across the pipes increases leading to poor end point pressures.

It is clear from this experiment that a water utility should necessarily enhance its network infrastructure to meet the increased demands without violating the minimum pressure requirement. Prior research done in the area of utility delivery infrastructure scaling [26], [5], [15], [17], [2], [16], [6], [20] typically focus on determining cost effective ways to upgrade the delivery pipes and the pumps to meet the new demand. While upgrading pipes along with pumps could lead to cost-effective solutions, pipe upgrades may not always be feasible due to: (i) constraints on capital expenditure, (ii) right-of-way issues on land ownership, and (iii) impact on vehicular traffic flows for pipes that are laid beneath heavy traffic ways. In this work, we focus on enhancing the water distribution network through the placement of additional booster pumps alone. Our problem statement is as follows:

Problem statement: Given the network conditions for increased demand, where should we locate additional booster pumps such that the pressure constraint is satisfied and the cost of these additional pump placement(s) is minimized?

We address the above problem in water delivery networks that have a "tree" topology. While urban water networks typically have looped topologies, tree networks are commonplace in rural and semi-urban areas [15]. Several rural and semi-urban areas have witnessed a tremendous increase in their demands. For example, the population of Faridabad – a city in India, whose water delivery network has a tree topology, has increased by nearly ten-fold in thirty years [13]. Given that water distribution networks have a lifetime of 50+ years, such high population growth rates can create a mismatch between the demand and deployed network's capacity, thereby necessitating research similar to this work.

IV. LOWER BOUNDS FOR PUMP PLACEMENT COSTS

The costs involved in placing a pump are: (i) capital expenditure (capex), the cost of purchasing the pump and associated infrastructure and (ii) operational expenditure (opex) of the pumps in terms of energy costs. As described in section VII-A,

these two costs are combined to give "total cost of ownership (TCO)", which is the cost of the pump over its lifetime. We now obtain a lower bound for this cost. Algorithm Min-Opex determines a pump placement strategy for the lowest opex. Based on this lower bound for opex, we derive the lower bound for capex, which leads to the lower bound for TCO.

A. Algorithm Min-Opex

Algorithm, Min-Opex determines a pump placement strategy that achieves the lowest possible opex. Based on this lower bound for opex, we derive the lower bound for capex, which leads to the lower bound for TCO. Min-Opex employs a "greedy" approach. The 'capacity' or 'power' of a pump is defined as the product of the flow through the pump and the pressure boost it delivers to this flow. Min-Opex assumes that the pump capacities are "continuous" in that the pressure deficiency at any node can be addressed by placing a pump before it of exactly the required capacity. In reality, pumps are manufactured in discrete capacities and hence the solution determined by Min-Opex may not be realizable in practice. Nevertheless, it is a lower-bound to compare the performance of other realizable algorithms. For lack of space, we do not present the proof of correctness of this algorithm.

Algorithm Min-Opex shows our approach. Let P_{min} be the minimum pressure value required to be maintained at each node. The basic idea is that when a node is visited and it is pressure deficient, we place a pump in front of that node just sufficient to handle that node's deficiency. There may be some additional boost due to its parents due to earlier pump placements, which is tracked as *parentBoost*. If this node had a pump placed before it, the extra boost is transferred to its children also, which are visited with an increased parental boost of *parentBoost + currentBoost*.

Min-Opex(N,*parentBoost*)
    ▷ N is a node
1    $N.deficiency \leftarrow N.pressure - P_{min}$
2    $N.deficiency \leftarrow N.deficiency$ - *parentBoost*
3   **if** ($N.deficiency > 0$)
4   **then** Add pump before N
            ▷ Pump's pressure boost is equal to $N.deficiency$
5            $currentBoost \leftarrow N.deficiency$
6   **for** each successor K of N
7   **do** Min-Opex(K,*parentBoost + currentBoost*)
8   **return**

B. Lower bound on TCO

Min-Opex outputs a solution with minimum operational expense in terms of the total pumping power, which is the sum of individual pump powers. Suppose this quantity is Q^*. Then we can calculate a lower bound on the total cost of ownership as follows. Suppose $C_1, C_2, \cdots, C_r$ be the capital expenditure on the prices of varying sizes of pumping stations and $Q_1, Q_2, \cdots, Q_r$ be the corresponding pumping power consumption, such that the list is ordered in terms of increasing

capex cost per watt of pumping power, i.e. C_i/Q_i. Then a lower bound on the total cost of ownership can be calculated by computing the lower bound on the capex. To do this, we "fill" the requirement of Q^* with the maximum number of pumping stations with lowest per-watt capex cost, i.e., C_1/Q_1. That is, we consider $\lfloor \frac{Q^*}{Q_1} \rfloor$ as the number of pumps with power Q_1, where $\lfloor X \rfloor$ denotes the greatest integer lesser than or equal to X. Then for the remaining pumping power, we fill it with pumping stations of the next lowest per-watt capex, i.e., Q_2 and so on. Define $S_0 = Q^*$, $N_i = \left\lfloor \dfrac{S_{i-1}}{Q_i} \right\rfloor$ and $S_i = S_{i-1} - N_iQ_i$. Then the lower bound on the capex is given by $\sum_{i=1}^{i=r} N_iC_i$. Because we are concerned with a lower bound, this can be generalized to include the cheapest real estate cost for the pumping station and other auxiliary costs.

Having obtained the lower bounds for opex and TCO, we now present two approaches that are realizable in practice. The two approaches work with available pump capacities and aim to minimize any one of opex, capex, or TCO involved in pump placement, but do not guarantee any optimality.

V. Algorithm Approx-Pump

Deciding the locations and capacities of assets subject to constraints is a problem that recurs in computer networking (e.g., [3]). Solutions of these problems usually employ variants of the *minimum weighted dominating set* (MWDS) concept. Following these works, we present a solution that appropriately models the nodal pressure deficiencies in a water distribution network and uses the MWDS concept to find the locations and capacities of the pumps.

The MWDS problem can be explained as follows. Consider a directed graph $\mathcal{G} = (V, E)$ with V being the set of vertices and E being the set of directed edges. Each vertex $v, v \in V$ is associated with a a weight (or a cost) given by some function $c(.)$. A subset of vertices $S \subset V$ is called a *dominating set* of $\mathcal{G}$ if every vertex of $\mathcal{G}$ is either in S, or reachable from a vertex in S within one hop. A *minimum weighted dominating set* is a dominating set such that the sum of vertices' weights in the set is minimum among all possible dominating sets. Identifying the minimum weighted dominating set in a graph is an NP-hard problem [14]. However, efficient approximation algorithms that guarantee a solution for MWDS that is within a factor of the true solution (i.e., approximation ratio) have been proposed in the literature [25].
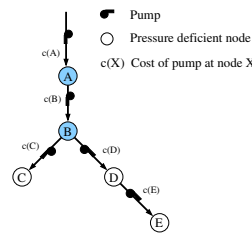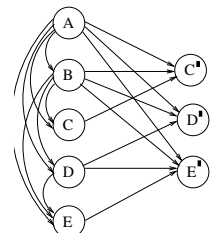

Fig. 2. Input flow graph $\mathcal{P}$ Fig. 3. Transformed graph Q

A. Modeling pump placement as a MWDS problem

Consider Figure 2 which shows a flow network $\mathcal{P}$. Let nodes C, D, and E be pressure deficient. To boost the pressure at these nodes, pumps can be placed at several locations. For instance, a pump of appropriate size can be placed along the edge (D, E) in front of node E to satisfy the pressure requirements of E; as an alternative, a bigger pump could be placed along the edge (B, D) in front of node D to satisfy the pressure requirements of both D and E; getting bolder, one could place an even bigger pump in front of node B along the edge (A, B) to meet the pressure requirements of all the nodes C, D, and E. Thus, placing a pump at a given location in the flow graph has the ability to meet the pressure requirements of zero, one, or more deficient nodes at an appropriate cost. This relation can be modeled as an MWDS problem which can then be solved using algorithms in the literature. Our solution APPROX-PUMP for pump placement in flow graphs does exactly this. Logically, the solution consists of three phases. The first phase is to pre-process the flow graph $\mathcal{P}$ to compute the cost of placing pumps at possible locations. The second phase is to transform the various pump placement options, their costs, and coverage to an instance, Q, of the MWDS problem. The third phase is to solve this MWDS problem instance using solutions in the literature and thereby identify cost effective locations to place the pumps along with the pump capacities. For lack of space, we elaborate only on the TRANSFORM phase.

B. Phase TRANSFORM

The transformation of the pump placement problem into an instance of minimum dominating set is presented in Algorithm TRANSFORM. In lines 2 through 5, the procedure adds all nodes A through E to V_1 as representatives for the pump placement locations. In addition, it adds new nodes C', D' and E' to V_2 as representatives for the pressure deficient nodes. In lines 6 through 9, the procedure adds directed edges from a pump location X to a set of pressure deficient nodes whose requirement can be satisfied by placing a pump at X. For example, placing a pump at B can meet the pressure requirements of nodes C', D', and E'. So, edges $\{(B, C'), (B, D'), (B, E')\}$ are added to Q. Finally, in lines 10 through 13, the procedure adds edges among vertices in V_1 indicating the redundancy existing in pump placement. For example, a pump placed at node A in the original flow graph, will make the pumps placed at nodes B, C, D and E redundant. Therefore, the edges $\{(A, B), (A, C), (A, D), (A, E)\}$ are added to Q. For the flow graph $\mathcal{P}$ shown in Figure 2, the complete transformed graph Q is shown in Figure 3. It can be shown that the worst-case runtime of the algorithm is $O(n^3)$ in the number of nodes n.

VI. ALGORITHM TREE-PUMP

We now discuss a second algorithm for the placement of pumps. In this algorithm, we assume that we have pumps of only one capacity (i.e., operating power) available to boost the pressure in the network. Under this assumption, TREE-PUMP aims to minimize the number of pumps required to

TRANSFORM($\mathcal{P}$)
    ▷ $\mathcal{P}$ is the input flow graph.
    ▷ $Q = (V, E)$ is returned on which a MWDS solution is run.
    ▷ Let $V = V_1 \cup V_2$
    ▷ Let $\mathcal{N}(v)$ be the set of neighbors of vertex v
1    $V_1 \leftarrow \{\}; V_2 \leftarrow \{\}; E \leftarrow \{\}$
2   **for** each node x in $\mathcal{P}$
3   **do** $V_1 \leftarrow V_1 \cup \{x\}$
4   **if** x is pressure deficient
5   **then** $V_2 \leftarrow V_2 \cup \{x'\}$
6   **for** each x in V_1
7   **do for** each y in V_2
8   **if** a pump at x can give required boost at y
9   **then** $E \leftarrow E \cup (x, y)$
10   **for** each x in V_1
11   **do for** each y in $\{V_1 - x\}$
12   **if** $(\mathcal{N}(y) \cap V_2) \subseteq (\mathcal{N}(x) \cap V_2)$
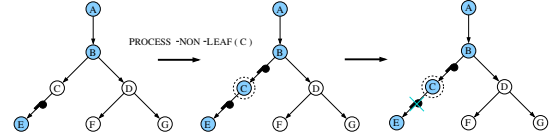13   **then** $E \leftarrow E \cup (x, y)$
14   **return** Q

Fig. 4. Child's over-provisioning removed when processing a non-leaf node.

meet the pressure requirements of the nodes. The algorithm is then repeated for different pump capacities and the least cost solution is chosen. This algorithm has two phases; in the first phase, pumps are placed in-front of pressure deficient nodes and/or their parents; in the second phase, any redundancies in the pumps so placed in the first phase are pruned.

A. Provisioning Phase

In the first phase, a post-order traversal of the nodes in the flow graph is done. If a leaf-node is found to be pressure deficient, a pump is added before it. Figure 4 illustrates this. To begin with, nodes C, D, E, F, and G are pressure deficient.

We start at node E and add a pump before it. It is possible that the pump so added may not be powerful enough to supply the necessary boost individually. If so, the new deficiency (difference between the boost required originally and the capacity of the pump added) is propagated up to its parent and added to the parent's deficiency. When a deficient non-leaf node is encountered in the post order traversal, a pump is added before it like in the case of a leaf node. In addition, if its children have pumps before them, it is possible that they could be rendered redundant due to the pump in front of the node; if so, the redundant pumps in front of the children are removed. Figure 4 illustrates the processing of a non-leaf node, specifically node C. When node C is processed, we add a pump before it, and that causes the pump before its child E to become redundant and so the pump is removed.

B. Pruning Phase

Because we check only one-hop descendants when processing a non-leaf node, it is possible that the flow graph is still

over-provisioned in terms of the pressure boosts. Therefore, we have a pruning phase where the tree is traversed in a pre-order fashion. Whenever any excess pressure is detected, appropriate pumps are pruned as shown in POST-PROCESS. Figure 5 illustrates this idea. The pumps before nodes B and D are sufficient to handle the pressure deficiency of node G. Thus the pump before node G is pruned in this process. The overall runtime of the algorithm for n nodes is $O(rn)$, where r is the number of pump capacities.
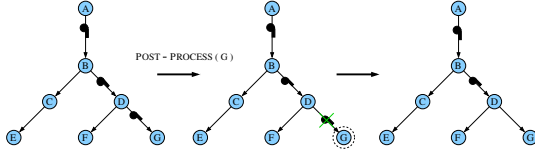

Fig. 5.   Over-provisioning is handled by removing redundant pump.

VII. RESULTS

We studied the performance of the two algorithms on a variety of tree flow graphs. The results presented here are from a flow graph containing one reservoir and 1747 nodes.

A. Cost of the solutions

We first analyze APPROX-PUMP and TREE-PUMP from the perspective of the Net Present Value (NPV) of the pump placement costs defined in terms of the opex and total cost of operation (TCO). Pumps are assumed to be running 24×7 at their full load with a constant efficiency η. Opex includes the cost for the electrical energy consumed by a pump in a year. The total cost of operation (TCO) is the sum of the NPV of the opex and the capex over a pump's life-time which is taken as 15 years. For APPROX-PUMP, we obtain the NPV by rounding-up the pump capacities returned by the algorithm to the nearest (higher) available capacity. For TREE-PUMP, we obtain the NPV by choosing the pump power among the set of available pump capacities, which has the smallest opex or TCO as per our objective.

B. Algorithms for comparison

We compare the cost effectiveness of the two algorithms against the lower bound previously derived. In addition, following most approaches in the literature, we also implemented a Genetic-Algorithm (GA) based solution for the pump placement problem. The GA represents the solution in a chromosome. The initial chromosomes are randomly chosen in a guided manner to ensure that the capacity of the initial set of pumps added is not too much over the deficit. Following typical GA templates, every generation chromosomes are chosen through tournament selection with crossover (of probability 80%) , mutation (of probability 5%), and elitism (i.e., the most fit chromosome is automatically copied over to the next generation). The fitness function used to evaluate the chromosomes has constraints for the pressure and velocity, and tries to minimize the cost. The GA is run for 500 generations with an initial population of 100 chromosomes.

C. Costs

Figure 6(a) shows the results of the proposed algorithms minimized for the operational expenditure (Opex), which is typically a proxy for the energy consumed by the additional pumps. The X-axis shows the factor by which demand is increased and Y-axis shows the NPV of the opex of the solution. As expected, the opex increases with demand. It can be seen from the figure that our algorithms perform better than the GA. We note that, in practice, the APPROX-PUMP solution is quite close to the theoretical lower bound and is roughly within a factor of 2. The behavior of all the three approaches is similar when the Total Cost of Ownership (TCO) is minimized (as opposed to opex alone), as shown in Figure 6(b). These results show that lessons learnt from computer networking can indeed be useful for overcoming challenges in water distribution networks.

D. Distribution of pressure slack at junctions

It is possible that the boost provided is more than the minimum pressure required at a node. We define the pressure slack of a node with new pressure P as follows: $\frac{(P-P_{min})}{P_{min}}$. Figure 6(c) (resp. 6(d)) shows the average pressure slack introduced by the proposed algorithms at the previously pressure deficient nodes when minimized for TCO (resp. Opex). As the demand increases, the over-provisioning introduced by both the algorithms increases. This is due to the discretization of the available pump sizes. The gap between available pump capacities increases as the pump capacity increases. Consequently, the slack introduced due to the rounding up increases with increasing demands. While introducing extra slack is not warranted, it brings in the added advantage of meeting future demands with the same pumping infrastructure.

VIII. RELATED WORK

Related work can be broadly classified into two areas. **Water networks as an application domain for computer networking:** Recently, networking researchers have begun to consider water networks as an application domain. Most work here [7], [12], [11], [22], [28] revolves around the use of wireless sensor networks to monitor the operational parameters of water distribution such as leaks, blocks, pressure drops etc. Our work is different from these in the sense that it does not use computer networks as a *mechanism* in water networks; rather it borrows *policy* (principles) behind the solutions for computer networking problems. The work that comes closest to ours is [9] that brings out the parallels between the Internet and electricity grids and discusses how the lessons learnt in one domain can be used in the other. Our work aims to convey a similar message in the context of water networks. **Water network planning:** The problem of enhancing an existing water utility infrastructure has been looked at so far along two strategies – genetic algorithms [26], [5], [17] and linear programming based operational research techniques [2], [16], [6], [20]. Most existing works focus on discovering cost effective ways to upgrade the delivery pipes along with the pumps to meet the scaled demand. At times, pipe upgrades

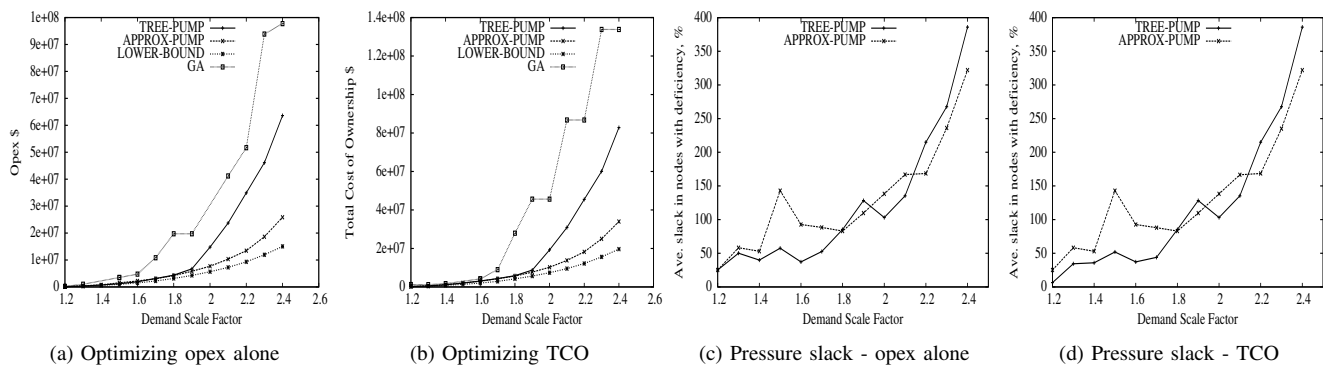|(a) Optimizing opex alone|(b) Optimizing TCO|(c) Pressure slack - opex alone|(d) Pressure slack - TCO|

Fig. 6.   Performance of the proposed solutions.

may not be feasible due to: (i) constraints on capital expenditure, (ii) right-of-way issues on land ownership, and (iii) impact on vehicular traffic flows for pipes that are laid beneath heavy traffic ways. Existing works may not be helpful in such scenarios since they recommend both pump and pipe replacement. Further, we have provided a lower bound for the total operational cost of any improvement.

IX. Conclusions and Future Work

With increasing focus on climate change and sustainability, water is being considered as the next "gold". We identified several challenges in water networks that can be tackled by principles used in computer networking. We focused on a specific municipal water network enhancement problem and adopted techniques used in computer networking to solve it. Our results show that that solutions used in computer networking perform better or at par with traditional solutions. We also acknowledge that it may be possible to leverage principles from other related domains such as transportation and oil/gas pipeline networks to overcome challenges in water networks. At present, we are working on extending the ideas presented in this paper to do integrated pipe and pump additions/enhancements on water networks with looped topologies.

References

[1] A. Ostfeld et al. The Battle of the Water Sensor Networks (BWSN): A Design Challenge for Engineers and Algorithms. *Journal of Water Resources Planning and Management*, 134(6), November/December 2008.

[2] E. Alperovits and U. Shamir. Design of Optimal Water Distribution Systems. *Water Resources*, 13(6):885 – 900, 1977.

[3] B. Aoun, R. Boutaba, Y. Iraqi, and G. Kenward. Gateway placement optimization in wireless mesh networks with qos constraints. *IEEE Journal on Selected Areas in Communications*, 24(11):2127–2136, 2006.

[4] EPANET - Drinking Water Research Software. http://www.epa.gov/nrmrl/wswrd/dw/epanet.html.

[5] W. G. Farmani R. and S. D.A. Evolutionary multi-objective optimization of the design and operation of water distribution network:total cost vs. reliability vs. water quality. *Journal of Hydroinformatics*, 8(3):165–179, 2006.

[6] Fujiwara, O., Jenchaimahakoon, B. and Edirisinghe, N.C. P. A modified linear programming gradient method for optimal design of looped water distribution systems. *Water Resources Research*, 23(6):977–982, 1987.

[7] M. Iqbal and H. B. Lim. A cyber-physical middleware framework for continuous monitoring of water distribution systems. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, SenSys '09, pages 401–402, 2009.

[8] K. R. Jun Wu. Optimization algorithms for proxy server placement in content distribution networks. In *IFIP/IEEE International Symposium on Integrated Network Management-Workshops*, pages 193–198, 2009.

[9] S. Keshav and C. Rosenberg. How internet concepts and technologies can help green and smarten the electrical grid. *SIGCOMM Comput. Commun. Rev.*, 41:109–114.

[10] P. B. Key and D. R. McAuley. Differential QoS and pricing in networks: where flow-control meets game theory. In *IEE Proceedings Software*, 1999.

[11] Y. Kim, T. Schmid, Z. M. Charbiwala, J. Friedman, and M. B. Srivastava. Nawms: Nonintrusive autonomous water monitoring system. In *Proceedings of the 6th ACM Conference on Embedded Networked Sensor Systems*, SenSys '08, 2008.

[12] T.-t. T. Lai, Y.-h. T. Chen, H.-h. Chu, and P. Huang. Pipeprobe: mapping hidden water pipelines. In *Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems*, SenSys '09, pages 375–376, 2009.

[13] Municipal Corporation of Faridabad. http://www.mcfbd.org/profile.htm.

[14] Papadimitriou and Stieglitz. *Combinatorial optimization: algorithms and complexity*. Prentice Hall, 1982.

[15] Prabhata Swamee and Ashok Sharma. *Design of Water Supply Pipe Networks*. Wiley-Interscience, 2008.

[16] G. Quindry, E. Brill, and J.C.Liebman. Optimization of looped water distribution systems. *ACSE Journal of Environmental Engineering Division*, 107(EE4):665–679, 1981.

[17] J. Reca and J. Martinez. Genetic algorithms for the design of looped irrigation water distribution networks. *Water Resour. Res.*, 42, 2006.

[18] F. B. S. Robbert van Renesse. Chain replication for supporting high throughput and availability. In *6th Usenix conference on Symposium on Operating Systems Design and Implementation (ODSI)*, 2004.

[19] Robert Cahn. *Wide Area Network Design: Concepts and Tools for Optimization*. Morgan Kaufmann, 1998.

[20] H. D. Sherali and E. P. Smith. A Global Optimization Approach to a Water Distribution Network Design Problem. *Journal of Global Optimization*, 11(2):107 – 132, 1997.

[21] Southwest Water. Customer Code of Practice, 2007-2008.

[22] I. Stoianov, L. Nachman, S. Madden, and T. Tokmouline. Pipenet- a wireless sensor network for pipeline monitoring. In *Proceedings of the 6th international conference on Information processing in sensor networks*, IPSN '07, 2007.

[23] University of Exeter, Colorado Springs Water Utility. http://centres.exeter.ac.uk/cws/benchmarks/expansion.

[24] US EPA. Decentralized/Cluster Wastewater Treatment, a Green Development Tool. In *Worksop on Decentralized Wastewater*, 2007.

[25] Vasek Chvatal. A greedy heuristic for the set-covering problem. *Mathematics of Operatations Research*, 4(3), 1979.

[26] G. A. Walters, D. Halhal, D. Savic, and D. Ouazar. Improved design of anytown distribution network using structured messy genetic algorithms. *Urban Water*, 1(1):23 – 38, 1999.

[27] J. K. Yan Chen, Randy H. Katz. Dynamic Replica Placement for Scalable Content Delivery. In *First International Workshop on Peer-to-Peer Systems*, 2001.

[28] S. Yoon, W. Ye, J. Heidemann, B. Littlefield, and C. Shahabi. Swats: Wireless sensor networks for steamflood and waterflood pipeline monitoring. *IEEE Network*, 25(1):50–56, 2011.