Chapter 1

Contaminant Transport in Municipal Water Systems

Presented at the *3rd PIMS Industrial Problem Solving Workshop*
held at the University of Victoria, Canada, May 31–June 4, 1999.

Seema Ali[1], Jeremy Bell[2], Chris Budd[3], Adriana Davidova[2], Nathan Krislock[4], Margaret Liang[5], Yves Lucet[6], Scott MacLachlan[5], Marni Mishna[6], Shelly Pinder[5], Peilin Shi[2], Cristina Stoica[2], JF Williams[6], Zili Wu[2]

Report written by John Stockie[6]

1.1 Introduction.

The quality of municipal water supplies has become an issue of great importance for large, urban centres. Widespread industrialisation, pollution, and development of areas close to traditional water sources have all contributed to the degradation of our drinking water, and made the topic a subject of heated debate. Officials and planners at all levels of government have recognised the need to manage water distribution networks so as to maintain acceptable levels of water quality. Because of the complexity of even the simplest of flow networks, numerical computations have become an essential tool in the water quality management process, and simulations of networks with hundreds of junctions, pumping stations, tanks and reservoirs are commonplace [4].

The forward or dynamic simulation of water networks with known characteristics and contaminant inputs is fairly well-understood, and software packages such as EPANET [5], which is freely-available over the Internet, are widely used in the industry. However, a more difficult aspect of water networks is the "inverse" or "forensic" problem where, for example, one wants to know how much of a contaminant has been released given measurements of contaminant levels at certain locations[7]. This is the problem that Charles Howard & Associates Ltd. (CHAL), a local Victoria engineering consulting firm, brought to the PIMS problem-solving workshop.

[1] University of Calgary
[2] University of Victoria
[3] University of Bath
[4] University of Regina
[5] University of British Columbia
[6] Simon Fraser University

[7] The importance of forensic water quality analysis has received widespread attention with the recent release of the Hollywood movie "Civil Action" starring John Travolta.

Our first task from CHAL was to examine the dynamic (forward) algorithm used in EPANET, and ensure that it is computing physically reasonable solutions for networks that experience rapid “flow reversals,” in which the flow in a pipe section changes direction over a short time intervals. This is a particular concern for EPANET, because the calculation of water flow in the pipe network is totally decoupled from the contaminant transport step. The discrete time step used for the flow is typically much larger than for the contaminant, and so EPANET may have difficulties handling flow reversals. CHAL is also concerned about whether or not the inverse problem is well-posed, and if it is possible to use existing forward solvers to construct an algorithm for solving the inverse problem.

We begin in the following sections with an overview of the physics of flow in water distribution networks, and the implementation used in the EPANET flow solver. The bulk of our results from the workshop are concerned with analysing simple flow networks and using the results to draw conclusions about the well-posedness of both the forward and inverse problems.

1.2 Flow in Water Distribution Networks.

A typical water distribution network, pictured in Figure 1.1, can be considered as a collection of *links* and *nodes*. The links come in two varieties[8]: *pipes* and *pumps*. The pipes convey water passively from one

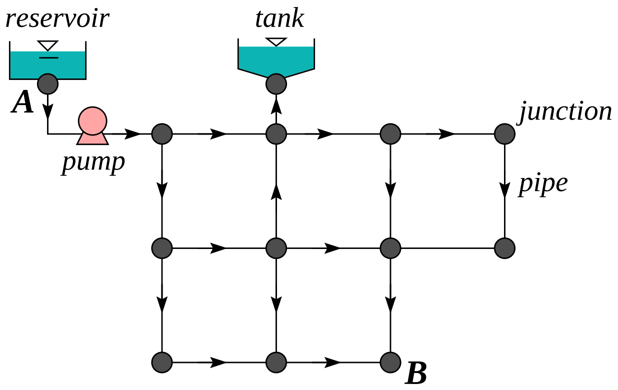


Figure 1.1: A typical water distribution network.

network junction to another in the direction of decreasing pressure, except where forced by pumps which apply some external forcing to raise the hydraulic head of the water in the network. The nodes are either *junctions*, *tanks* or *reservoirs*. Junctions connect two pipe sections together and may also serve as points of water consumption (e.g., a residential or industrial customer) or external water sources (e.g., a water treatment plant). Tanks and reservoirs are external nodes that serve as water storage areas.

1.2.1 Network components.

Before moving on to a mathematical description of the physical processes in each of the network components, we need to define the basic variables that will be used to describe water transport in the network. Rather than the variables velocity and pressure that are typically used in fluid mechanics to describe flow, we will instead formulate the problem in terms of

[8] There are also several types of *valves* [5] which we will not consider here for the sake of simplicity.

- *hydraulic grade line* or *head* (H) which for 1D, steady, incompressible flow is given by Bernoulli's equation:

$$H = \frac{P}{\rho g} + \frac{V^2}{2g} + H^o,$$

where P is the pressure, g is the acceleration due to gravity, V the fluid velocity, and H^o is the elevation;

- *volumetric flow rate* (Q) through a section of pipe, which is related to the velocity V by $Q = V \cdot A$, where A is the cross-sectional area of the pipe.

We now proceed to describe briefly the physics governing each of the network components implemented in EPANET, with complete details given in [5]:

Pipes: The flow in each pipe is assumed to be steady, uniform and characterised by fully-developed turbulence. As water flows along the pipe section joining node i to node j, there will be a loss in head, $\Delta H_{ij} = H_i - H_j$, due to friction between the water and the walls of the pipe. There are several semi-empirical equations used in hydraulics that relate this head loss to the flow rate, one of which is the *Hazen–Williams equation* [2, 8]

$$Q_{ij} = R_{ij} \frac{\Delta H_{ij}}{|\Delta H_{ij}|^{0.46}}, \tag{1.1}$$

where the constant R_{ij} (called the *conductance*) depends on the dimensions of the pipe and its frictional resistance. Notice from the form of (1.1) that Q_{ij} is always positive when the flow in pipe $i - j$ is directed from node i to node j (refer to Figure 1.2).

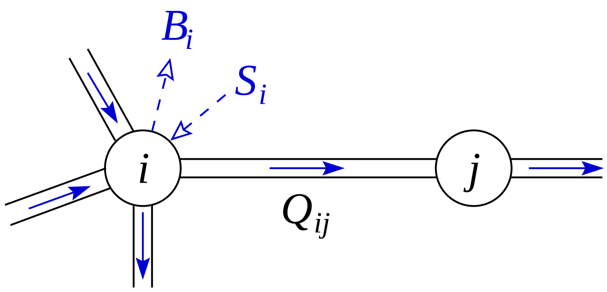


Figure 1.2: A typical pipe link in a network.

Pumps: Pumps are devices used to raise the hydraulic head of the water in the network. The flow rate in pump-driven links is related to the head gain by the following relationship:

$$Q_{ji} = \begin{cases} \left(\dfrac{H_{ij}^{off} - \Delta H_{ij}}{\alpha} \right)^{1/\beta} & \text{if } 0 \leq \Delta H_{ij} \leq H_{ij}^{off}, \\ 0 & \text{otherwise,} \end{cases} \tag{1.2}$$

where H_{ij}^{off} is the *shut-off head*, above which the pump is unable to operate. Notice that the pump only operates when the head at node i is greater than that at node j and hence always pumps water from node j to node i; i.e. pumps always work against the pressure gradient. The coefficients α and β are parameters specific to each pump that fit the flow rates to an operational *pump curve*.

Junctions: Pipe junctions may also be consumers or suppliers of water, with a given supply $S(t)$ or demand $B(t)$ flow rate, as pictured in Figure 1.2.

Tanks: A storage tank is a special type of node that has a free water surface where the head is simply given by the height above sea level. The time variation of the head H_s at a storage node connected by a single pipe to node s is described by the equation

$$\frac{\partial H_s}{\partial t} = \frac{Q_{is}}{A_s}, \tag{1.3}$$

where A_s is the cross-sectional area of the storage vessel.

Reservoirs: These are another type of storage node in which the water surface remains at the same level no matter what the flow rate is, and so the head is a constant. Reservoir nodes typically represent external sources of water such as lakes or rivers.

1.2.2 Hydraulic simulations.

The dynamic equations describing a flow network can be derived based on conservation principles. Suppose that a network contains N_j junctions, N_t tanks and N_r reservoirs. Within each node j (see Figure 1.2), the total of all flows into the node must equal that directed out of the node. Mathematically, this can be written as

$$S_j + \sum_{i=1}^{N} Q_{ij} = B_j, \tag{1.4}$$

where $N = N_j + N_t$ are the number of nodes with heads that vary (reservoirs are excluded here because they have a constant head). Since the flow rates satisfy $Q_{ij} = -Q_{ji}$, there is one additional conservation constraint that must be satisfied by the supply/demand input values; namely, that

$$\sum_{j=1}^{N} (S_j - B_j) = 0. \tag{1.5}$$

Based on the assumption that consumptions B_j, supplies S_j, and pump operations change slowly in time, we treat the system as one that can be modeled over time as a series of quasi–steady state simulations. By substituting into (1.4) the expressions for Q_{ij} from (1.1) and (1.2), and the discrete form $Q_{ij} = (H_j^{new} - H_j^{old}) \cdot (A_j/\Delta t)$ of the tank equation (1.3), we obtain a system of N nonlinear equations for the N unknown heads H_j at any given time.

1.2.3 Contaminant transport.

Imagine that a dissolved substance is introduced at a certain node (e.g., the reservoir node **A** in Figure 1.1), which is then transported through the network. The substance has no effect on the water flow rates we obtained in the hydraulic model and so its concentration can be determined independently.

Let $C_{ij}(x,t)$ represent the concentration of the substance in link $i - j$. While our assumption that the flow rate is constant along a given pipe link is reasonable, we cannot say the same for the contaminant, which must be taken as a function of time, t, and distance along the pipe, x. The equation of conservation for contaminant mass within each link then reads

$$\frac{\partial C_{ij}}{\partial t} + \frac{Q_{ij}}{A_{ij}} \frac{\partial C_{ij}}{\partial x} = -k_{ij}\, C_{ij}. \tag{1.6}$$

A_{ij} is the pipe's cross-sectional radius and k_{ij} is a rate constant that describes the decay of contaminant due to reaction with the pipe walls, and bulk reactions with the water[9].

The quantity (Q_{ij}/A_{ij}) represents a known velocity with which the substance is convected in the pipe and therefore, (1.6) is simply a linear advection equation. In order to solve this equation we need only determine the value $C_i^o(t)$ of the concentration at $x_{ij} = 0$, where we have used the convention that the coordinate x along each pipe is chosen so that the origin is at the upstream end (i.e., at node i if $Q_{ij} > 0$ and node j if $Q_{ij} < 0$). This situation is pictured in Figure 1.3.

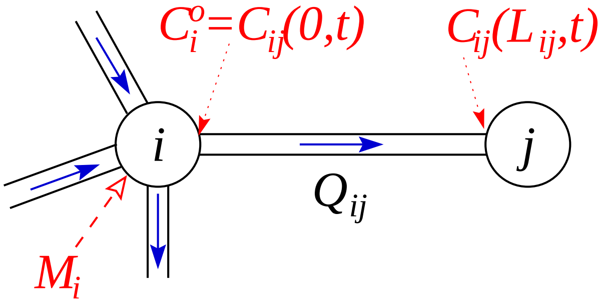


Figure 1.3: The boundary condition on the concentration in a link, $C_i^o(t)$, is obtained by combining the mass inflow of contaminant, M_i, and the mixing of inflows into node i.

The boundary values $C_i^o(t)$ can be found for each node i by assuming that the turbulent flow into the node is well-mixed and then applying the following conservation argument. We require that the total mass of contaminant transported into junction i

$$M_i + \sum_{k,Q_{ki}>0} C_{ki}(L_{ki}, t)\, Q_{ki}$$

(where M_i is the mass inflow rate of contaminant and L_{ki} is the length of the pipe k–i) must be balanced by the total mass out

$$C_i^o \cdot \left[B_i + \sum_{k,Q_{ki}<0} (-Q_{ki}) \right].$$

Equating these two expressions and using (1.4), we can solve for the boundary value

$$C_i^o(t) = \frac{M_i + \displaystyle\sum_{k,Q_{ki}>0} C_{ki}(L_{ki}, t)\, Q_{ki}}{S_i + \displaystyle\sum_{k,Q_{ki}>0} Q_{ki}} \tag{1.7}$$

Equation (1.6) can now be solved in each link using (1.7) as a boundary condition, and with the flow rate Q held constant over the time interval.

1.3 The EPANET solution algorithm.

The solution procedure within the EPANET water quality simulation package proceeds as follows in each hydraulic time step:

[9] The overall rate constant takes the form $k = k_b + 2k_w\, k_f / [R(k_w + k_f)]$ where k_b is the first-order bulk reaction rate constant, k_f is the mass transfer coefficient between wall and pipe, k_w is the reaction rate at the wall, and R is the pipe radius.

1. Calculate the supply and demand at each node, and the status of the pump links using user-specified time "patterns."

2. Solve the non-linear system of equations (1.1)–(1.4) for the head values H_i using a full Newton solver. The Jacobian matrix is constructed explicitly and the resulting linear system is solved using a Cholesky factorisation which takes advantage of the sparse structure of the matrix.

3. Perform the water quality calculations using the *Discrete Volume Element Method* (or DVEM), which proceeds as follows:

   (a) Select the water quality time step $\tau = \min_{ij}(L_{ij}/V_{ij})$, where $V_{ij} = Q_{ij}/A_{ij}$ is the flow velocity in pipe $i - j$. Consequently, τ is equal to the shortest travel time through all pipes in the system, which ensures that water is not transported beyond the downstream node in any link within a single time step.

   (b) Interpolate the concentrations from the discrete volume elements at the beginning of the hydraulic time step onto their current locations (since the position of the elements changes in each hydraulic time step).

   (c) Each pipe $i-j$ is then re-partitioned into η_{ij} discrete elements of equal volume, with the number of elements chosen so that

   $$\eta_{ij} = \left\lfloor \frac{L_{ij}}{\tau\, V_{ij}} \right\rfloor .$$

   This ensures that the transport in all links is resolved adequately over the time step τ.

   (d) Propagate the concentrations through the network using equation (1.6) and boundary condition (1.7). The water quality time step τ is typically much smaller than the hydraulic time step, and so this will require several integration steps.

Details of the Newton solver for the network flow are given in [8] and the DVEM contaminant transport algorithm is detailed in [6].

1.3.1 EPANET's limitations: reverse flow.

One of the major limitations of EPANET is the total decoupling of the hydraulic and water quality computations, so that the time steps used in each component need not coincide. The default hydraulic time step in EPANET is $\Delta t_h = 1$ *hour*, whereas the water quality computations are typically done on a much finer time scale of $\Delta t_q \lessapprox 0.1\ h$. Solving the network flow problem can be expensive because EPANET uses a full Newton solver, and so the choice of a larger Δt_h is justified in terms of efficiency. However, holding the flow rates constant over a large hydraulic time step can lead to inaccurate or even non-physical results. For example, if the actual flow in a pipe link reverses in the middle of a hydraulic time step, then the contaminant in that pipe may be transported in the wrong direction!

To illustrate the non-physical results that can occur when choosing a Δt_h that is too large, consider a pipe section of length 1 unit wherein the flow velocity varies periodically as shown in Figure 1.4(a). Suppose that a pulse of contaminant is released at time $t = \frac{1}{4}\ h$ in the upstream node (i) of the pipe section $i - j$. The pulse will travel only $x = \frac{9}{16}$ of the distance down the pipe (which is simply the area under the velocity curve between $t = \frac{1}{4}$ and 1), at which time the pulse will reverse direction. Therefore, the pulse returns eventually to node i and never gets transported into the rest of the network through node j.

If this same problem were to be computed using EPANET with a hydraulic time step of 1 h, then the velocity variation would be represented as a step function, as pictured in Figure 1.4(d). A pulse of contaminant released at time $t = \frac{1}{4}$ would reach the downstream node at time $t = \frac{3}{4}$ and be transported into the rest of the network before the pipe flow experiences a reversal. Consequently, taking a hydraulic time step that is too large to resolve variations in the water flow or contaminant transport can lead to incorrect water quality solutions.

(a) True flow velocity in pipe $i-j$. (b) Chlorine at node i. (c) Chlorine at node j.

(d) EPANET velocity ($\Delta t_h = 1\ h$). (e) Chlorine at node i. (f) Chlorine at node j.

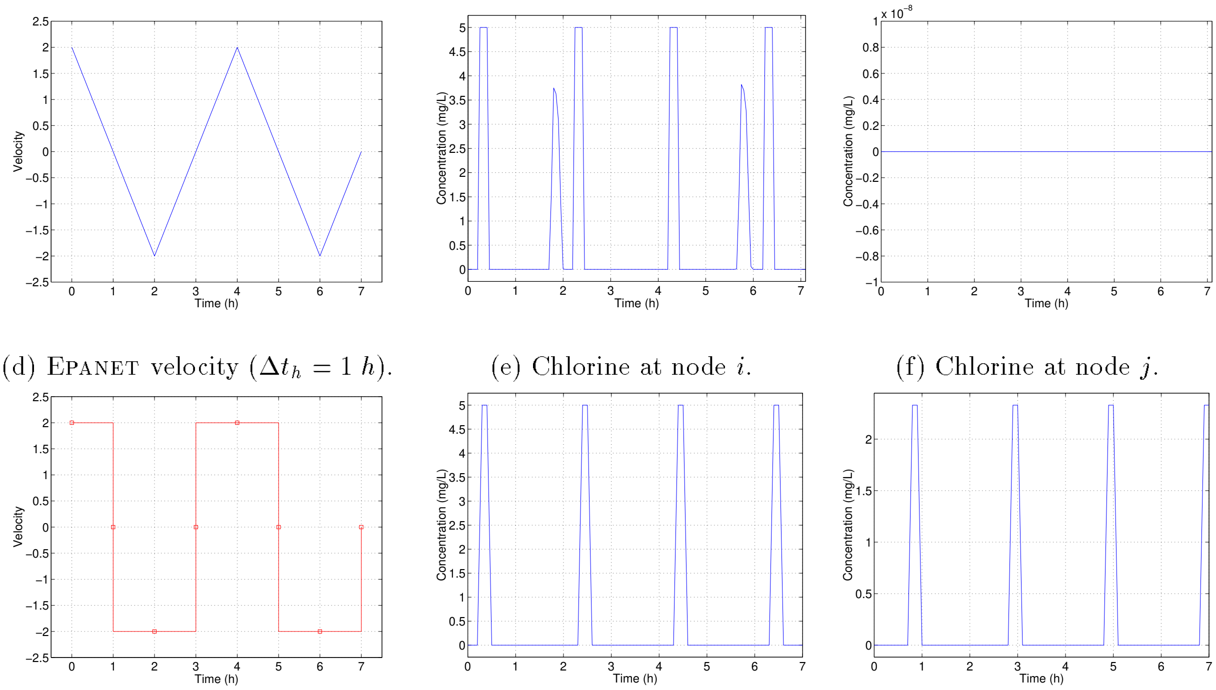


Figure 1.4: Computed chlorine concentrations at the upstream node for periodically-reversed flow: (a,b,c): "true" triangular velocity profile, computed with $\Delta t_h = 0.025\ h$; (d,e,f): typical EPANET calculation, with $\Delta t_h = 1\ h$.

To see that this is in fact what happens in EPANET, Figures 1.4(a,b,c) depict a simulation of the "true flow" conditions with a triangular velocity profile and a small hydraulic time step. The upstream node (i) is given periodic pulses of contaminant, and the return pulse is clearly superimposed on the input pulse. The downstream node (j) remains at a contaminant level of zero throughout the simulation. Figures 1.4(d,e,f) on the other hand, show the results for a hydraulic time step of 1 hour corresponding to the step function velocity profile. Contaminant is released at the same frequency, and the pulses are clearly seen to exit the loop via the downstream node before the flow reverses.

There is no reason in principle that Δt_h cannot be chosen much smaller so that it is equal to Δt_q in order to minimise this phase error due to flow reversal. However, the calculations become much more expensive. Not only this, the format of the input files requires that the user provide a "pattern" for the time variation of nodal supply and demand and pump operations, which must be given at each hydraulic time step (i.e., EPANET does no interpolation). In reality, demand patterns are known only at 1– or 2–hour intervals, and so manual interpolation is required if a smaller hydraulic time step is to be taken.

Nonetheless there is a practical limit to how small the time step can be taken. Within the EPANET algorithm is embedded a Newton solver that inverts a full Jacobian matrix in each iteration. This is not much of a concern when the hydraulic time step is on the order of hours, but could be prohibitively expensive if the time step is taken small enough to accurately resolve frequent flow reversals. An obvious alternative is the class of *quasi–Newton* methods, which never explicitly construct the Jacobian matrix, and solve the system using iterative techniques rather than direct methods. Many public–domain codes are available that could be incorporated in the EPANET software, one of which is DNSQE[10]. We did not have the time during the Workshop to implement a non-linear solver in EPANET, but we do believe that the modifications required to the code for such an addition are straightforward.

1.4 Analysis of Simple Networks.

We now proceed to analyse some very simple distribution networks, in order to make conclusions about the solvability of the forward and inverse problems for contaminant transport. We assume for the remainder that we have a given set of flow rates Q_{ij} for the links in the network.

1.4.1 Exact solution for contaminant flow

We consider the simplest possible network consisting of two junctions connected by a single pipe. Letting $V(t) = Q(t)/A$ denote the flow velocity, we are reduced to solving the equation

$$\begin{aligned} C_t + V(t)\, C_x &= -k\, C, \\ C(0,t) &= C^o(t) \qquad \text{(given)}, \end{aligned}$$

where the boundary condition is given at the upstream node. If we define

$$\xi(t) = \int_0^t V(\tau)\, d\tau$$

to be the distance traveled by fluid particles from time 0 to t, then the solution to this problem can be written explicitly as

$$C(x,t) = C^o(\xi^{-1}(\xi - x))\, e^{-k(t-\xi^{-1}(\xi-x))}. \tag{1.8}$$

A very important special case is the situation of constant velocity, $V(t) \equiv V_o$, for which

$$C(x,t) = C^o(\underbrace{t - x/V_o}_{delay})\ \underbrace{e^{-kx/V_o}}_{attenuation}\ . \tag{1.9}$$

[10]DNSQE is part of the SLATEC library, which is freely available on the Internet through Netlib (`http://www.netlib.org/bib/gams.html`).

From this form of the solution, it is clear that flow in the pipe is a combination of *delay* (due to the finite propagation velocity of contaminant in the system) and *attenuation* (from the decay of contaminant due to reactions at the pipe wall and in the bulk flow).

An important consequence of the form of (1.9) is that even though the transport of contaminant is a linear process (i.e., $C(x,t) \propto C^o$), it is *not* a convolution (because of the exponential decay term). Therefore, we expect that there may be some difficulties with the solution of the inverse problem.

Nevertheless, let us continue on and investigate the solution of the inverse problem for a simple "loop" network, pictured in Figure 1.5. Two nodes, A and B, are connected by a pair of pipes with different lengths

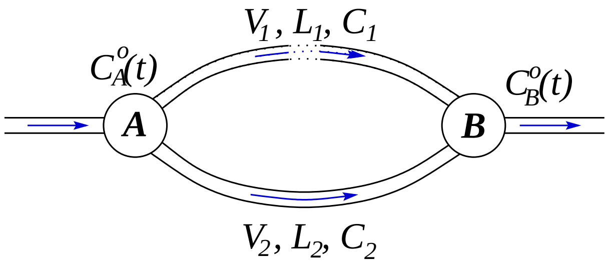


Figure 1.5: A simple reduced network consisting of two nodes connected in a loop, with both flows in the same direction. The dotted lines in pipe 1 represent the fact that there cannot be two pipes joining a single pair of junctions in EPANET; rather there must be some collection of other nodes and links in between for which we take "effective" variables V_1, L_1 and C_1.

(L_1 and L_2) and flow velocities (V_1 and V_2). Supposing that contaminant enters node A with a variation given by $C^{in} = C_A^o(t)$, we begin by determining how the outflow concentration C_B^o measured at node B depends on C_A^o. Assuming the velocities are positive and constant, equation (1.9) gives us concentrations of the contaminant at the end of the pipe section:

$$C_1(L_1,t) = C_A^o(t - L_1/V_1)\, e^{-kL_1/V_1},$$
$$C_2(L_2,t) = C_A^o(t - L_2/V_2)\, e^{-kL_2/V_2}.$$

Once the inflows from the two pipes meet in node B and mix, the outflow from the node is given by

$$\begin{aligned} C_B^o &= \frac{V_1 C_1 + V_2 C_2}{V_1 + V_2} \\ &= \frac{V_1 C_A^o(t - L_1/V_1)\, e^{-kL_1/V_1} + V_2 C_A^o(t - L_2/V_2)\, e^{-kL_2/V_2}}{V_1 + V_2}. \end{aligned} \tag{1.10}$$

1.4.2 Ill-posedness of inverse problem for simple loops.

We will now use the exact solution (1.10) to answer the following questions:

> *Is the inverse problem well-posed? Or, in other words, is there a unique inflow of contaminant $C_A^o(t)$ corresponding to each measured outflow $C_B^o(t)$?*

In fact, we will show that a constant outflow $C_B^o = 0$ can be generated by more than one input for a given set of network parameter and flow rates.

Consider the single-loop network in Figure 1.5, and assume that $V_2 > V_1 > 0$. To simplify the situation, we look for input concentrations of the form $C_A^o(t) = \overline{C}\, e^{\theta t}$, where θ is a complex number. Substituting this expression into (1.10) and setting $C_B^o = 0$, we obtain

$$e^{\delta\theta} = -\frac{V_2}{V_1}\, e^{-\delta k},$$

where $\delta = (\frac{L_2}{V_2} - \frac{L_1}{V_1})$ is the difference in time it takes for the concentration pulse to travel through pipe 2 versus pipe 1. Taking the logarithm of this expression, we can solve for θ:

$$\theta = -k + \frac{1}{\delta} \log\left(\frac{V_2}{V_1}\right) + \frac{(2n+1)\pi}{\delta} \sqrt{-1}, \tag{1.11}$$

where n is any integer. Therefore, the inflows we are interested in take the form

$$C_A^o(t) = P(t)\, e^{[-k + \log(V_2/V_1)/\delta]\, t}, \tag{1.12}$$

where $P(t)$ is *any* periodic function with period $2\pi/\delta$.

Consequently, there are an *infinite* number of possible inflows for which the measured concentration at node B will be zero (or constant). A second such inflow is pictured in Figure 1.6, where it is clear that after an initial transient, the measured concentration is identically constant, even though the contaminant continues to vary at node A.

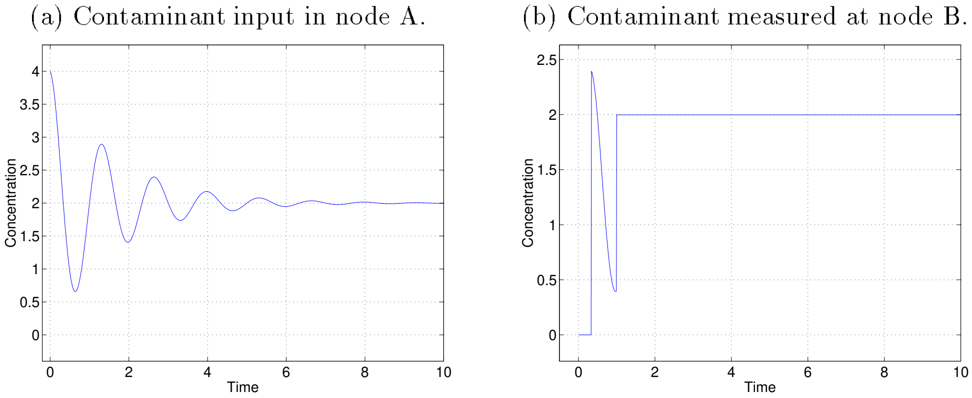


Figure 1.6: An exact solution of (1.10) with a decaying sinusoidal input concentration, and $L_1 = 10$, $V_1 = 10$, $L_2 = 5$, $V_2 = 15$, $k = 0.002$ (so that $\delta = -2/3$ and $\theta \approx -0.610198$).

In real water distribution networks, the sources do not decay in time in an exponential fashion. Instead, it is much more typical for sources to vary periodically *without* decay, corresponding, for example, to 24–hour cycles of demand and supply. In order for the solution in (1.12) to be undamped, the term in the exponent must be zero, so that

$$\delta = \frac{1}{k} \log\left(\frac{V_2}{V_1}\right),$$

which can also be written as a restriction on network parameters

$$L_2 = V_2 \left[\frac{L_1}{V_1} + \frac{1}{k} \log\left(\frac{V_2}{V_1}\right)\right].$$

Figure 1.7 presents an undamped solution for another network which again clearly exhibits a constant concentration at node B after the initial transients have died out.

All of our efforts in this section have so far been theoretical and so it is worthwhile for us to verify that the same type of behaviour is observed in actual EPANET computations. We constructed a simple 3-node network consisting of a loop, as pictured in Figure 1.8(a) (the extra junction is needed for the lower pipe since EPANET has a restriction that only one pipe can connect any two junctions). We applied a periodic, square-wave input of contaminant to node A, as shown in Figure 1.8(b), and then measured the concentration at node B. The measured chlorine levels at node B in Figure 1.8(c) again show that the contaminant reaches a constant level after the initial transients have died out.

(a) Contaminant input at node A. (b) Contaminant measured at node B.

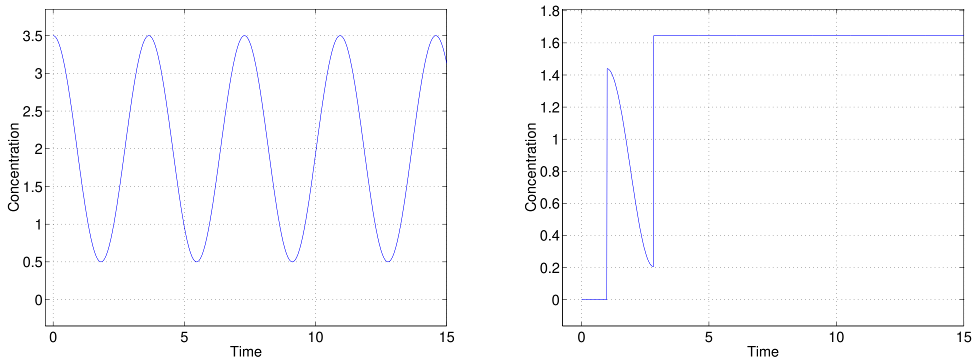
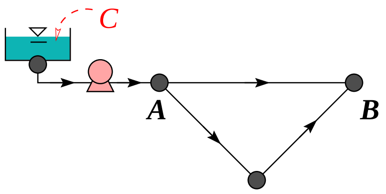


Figure 1.7: An exact solution of (1.10) for an undamped periodic input, with $L_1 = 10$, $V_1 = 10$, $L_2 \approx 33.8786$, $V_2 = 12$, $k = 0.1$ ($\delta \approx 1.823216$, $\theta = 0.0$).

(a) Loop network in EPANET.



(b) Chlorine source at node A. (c) Chlorine measured at node B.

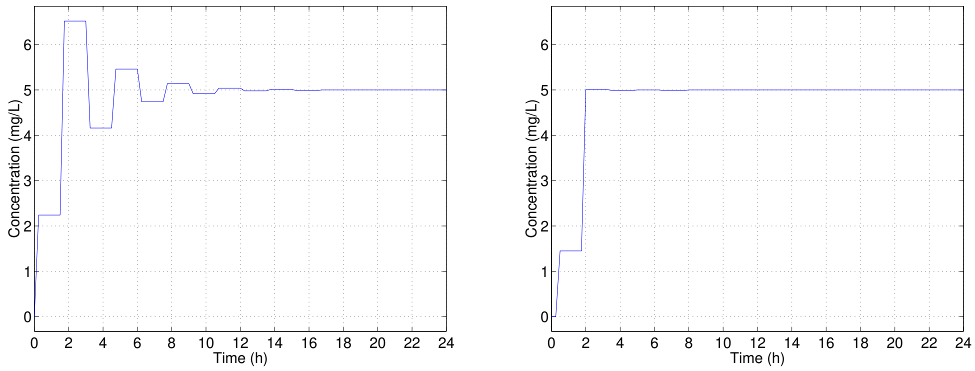


Figure 1.8: An EPANET computation with a decaying, periodic, square wave input that produces a constant output at node B.

1.4.3 Ill-posedness of forward problem.

We will now show that even the forward contaminant problem can be ill-posed. For this purpose, we will consider a similar loop network to that just considered, except that the velocity $V_1 < 0$ so that there is a *return loop*. The situation is pictured in Figure 1.9. The important distinction here is that we are not

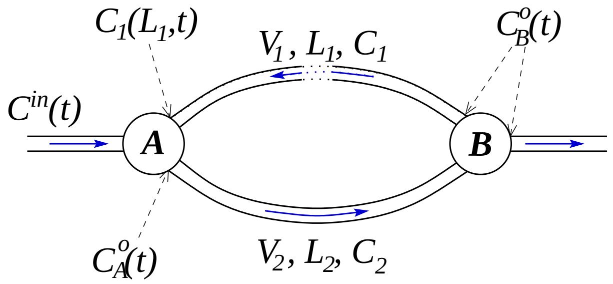


Figure 1.9: A second network with a "feedback" loop.

attempting to find C^{in}, the concentration flowing into node A. Instead, C^{in} is a given function.

Using the same mixing arguments from the previous section, the concentration of the boundary value at the head of pipe 2, C_A^o, is a linear combination of C^{in} and the inflow from pipe 1:

$$C_A^o(t) = aC^{in}(t) + bC_1(L_1, t),$$

(where a and b are constants depending on the flow velocities). However, if we apply mixing to the inflows at node B, then $C_1(L_1, t)$ is an attenuated and time-shifted version of $C_A^o(t)$:

$$C_1(L_1, t) = C_A^o(t - \delta)\, e^{-k\delta},$$

with $\delta = \frac{L_2}{V_2} + \frac{L_1}{V_1}$ (notice the change in sign from the expression for δ in Section1.4.3!). If we then assume $C_A^o = \overline{C} e^{\theta t}$ (and take $C^{in} = 0$ for simplicity), then we can solve these equations as before to obtain

$$C_A^o(t) = P(t)\, e^{(-k + \log b/\delta)t},$$

where $P(t)$ is any periodic function with period $\frac{2\pi}{\delta}$. Physically, this behaviour can be interpreted in terms of a given contaminant pulse generating *echoes* or *ringing* in networks containing a return loop.

Consequently, for a *given* contaminant inflow $C^{in}(t)$, there can be more than one outflow C_A^o into pipe 2. This suggests that the forward problem is also ill-posed. Whether or not such a situation is actually encountered in practice is not clear. However, at the very least it does suggest that we proceed with caution when attempting to compute solutions to the network flow problem.

1.4.4 Solving the Inverse Problem.

There are many possible approaches to solving the inverse water quality problem, one of which is described in [1]. During the workshop, we were not able to implement any inverse solution algorithms or to investigate in detail the merits of the possible approaches. However, with an efficient forward solver in hand, there are many public domain software packages available for non-linear optimisation that use the forward solution as a "black box" in solving the inverse problem. One package which would be worth considering is DASOPT [3], or its successor COOPT [7], which are based on the well-known nonlinear differential-algebraic system solver DASSL. Provided that a forward time step of the water quality calculation could be incorporated into this package, then the contaminant concentrations or positions could be treated as free parameters and optimised using these algorithms.

1.5 Summary and Future Directions.

Our main conclusion is that the EPANET code is calculating the correct solution, provided that the hydraulic time step is taken small enough to resolve flow reversals. As a result, we make the following suggestions:

- First, the hydraulic time step and the water quality time step should be explicitly chosen equal to each other, instead of letting EPANET default to a hydraulic step of one hour.
- Second, patterns for the time variation of consumption, supply and pump/valve operations must be interpolated onto the hydraulic time steps from known (measured) data. Two options for dealing with this are:
  - preprocessing the input data files to interpolate the known patterns onto a denser series of time points. This requires no modifications to EPANET and can easily be made transparent to the user.
  - incorporating an interpolation algorithm into EPANET, which is much more problematic since one must deal with the internal data structures used to store the flow information.
- Because the flow equations are now being solved at a much larger number of time steps, faster algorithms are required for solving the nonlinear system. Our suggestion is to use a quasi-Newton algorithm, such as DNSQE.
- We have found concrete examples where the inverse problem for contaminant transport is ill-posed when there are "loops" in the network, and so it may be difficult or impossible to solve in practice. Nevertheless, we have no evidence that this is true in general and so it may still be worthwhile to pursue the implementation of an inverse solver. Our advice: *Proceed with caution!*
- Once an efficient forward solution algorithm is working, there are many free software packages available on the Internet for solving the inverse problem, in which the forward solver is used as a simple "black box."

Bibliography

[1] Timothy R. Ginn and Ellyn M. Murphy. A transient flux model for convective infiltration: Forward and inverse solutions for chloride mass balance studies. *Water Resources Research*, 33(9):2065–2079, 1997.

[2] Horace W. King, Chester O. Wisler, and James G. Woodburn. *Hydraulics*. John Wiley & Sons, New York, 5th edition, 1948.

[3] Linda Petzold, J. Ben Rosen, Philip E. Gill, Laurent O. Jay and Kihong Park. Numerical optimal control of parabolic PDEs using DASOPT, in *Proceedings of the IMA Workshop on Large-Scale Optimization*, 1996. Available at `http://www.engineering.ucsb.edu/~cse/dasopt.ps`.

[4] L. S. Reddy, Lindell E. Ormsbee, and Don J. Wood. Time-averaging water quality assessment. *Journal of the American Water Works Association*, pages 64–73, July 1995.

[5] Lewis A. Rossman. *EPANET Users Manual (Version 1.1)*. Risk Reduction Engineering Laboratory, U.S. Environmental Protection Agency, Cincinnati, Ohio, January 1994. Code and documentation are available from `http://www.epa.gov/docs/RREL/`.

[6] Lewis A. Rossman, Paul F. Boulos, and Tom Altman. Discrete volume-element method for network water-quality models. *Journal of Water Resources Planning and Management*, 119(5):505–517, 1993.

[7] Radu Serban. COOPT – Control and optimization of dynamic systems: User's guide. Technical Report UCSB-ME-99-1, University of California at Santa Barbara, 1999. Available at `http://www.engineering.ucsb.edu/~cse/coopt.ps`.

[8] Uri Shamir and Charles D. D. Howard. Water distribution systems analysis. *Journal of the Hydraulics Division, Proceedings of the American Society of Civil Engineers*, 94(HY1):219–234, January 1968.