Water Age Computation for Distribution Networks

Auteur : Stoquart, Colin
Promoteur(s) : Dewals, Benjamin; Archambeau, Pierre
Faculté : Faculté des Sciences appliquées
Diplôme : Master en ingénieur civil physicien, à finalité approfondie
Année académique : 2020-2021
URI/URL : http://hdl.handle.net/2268.2/11447

Water Age Computation for Distribution Networks

By

Colin Stoquart

Prof. Benjamin Dewals, Thesis Advisor

Prof. Pierre Archambeau, Thesis Advisor

Master thesis presented in partial fulfillment of the requirements
for the master of science degree in engineering physics.

University of Liège - Faculty of Applied Sciences

Academic year 2020-2021

Acknowledgments

First and foremost, I would like to express my deepest thanks to Prof. B. Dewals, who introduces me to the subject. His guidance, help and advice were invaluable throughout this work. This work also owes a lot to my co-supervisor Prof. P. Archambeau.

The quality of this work has been greatly improved thanks to meetings in which Prof. E. Deleersnijder, I. Draoui, T. Pirard, and my two supervisors took part. I am grateful to all of them for their precious feedbacks.

I would like to thank, again, Prof. E. Deleersnijder and Prof. E. Delhez for being the jury members of this thesis.

I would also like to thank Mr. Ralf Friedmann and Markus Justen from Zweckverband für Wasserversorgung Germersheimer Südgruppe for providing the data of the Jockgrim network. My gratitude also goes to T. Pirard who helps me to understand and use this network.

I have the chance to be very well supported. For their support, I am grateful to my parents, my brother, my girlfriend, and all these persons that I do not explicitly mention here but who I am thinking of. Thank you for being my adventure companion.

Colin.

"The world cannot be understood without numbers.
But the world cannot be understood with numbers alone."

Hans Rosling, *Factfulness*.

Abstract

As you start reading these lines, you may have already taken a shower, drank tap water, started a washing machine, or any of these small daily miracles that a water distribution system offers us. The omnipresence of water distribution networks (WDNs) in our daily life may lead us to reduce WDNs to simple and well-understood systems, which they are not. Among many concerns for WDNs, a major and essential one is to ensure to every customer good water quality. A well-known technique to easily assess water quality is to evaluate the water age ([9], [16], [22]).

In this work, two existing methods to evaluate the water age in WDNs have been reviewed and implemented: the flow weighted method from the software EPANET [27], and the bins method introduced by Machel et al. [20]. First, assuming steady-state, a method with a new representation of the age distribution has been implemented. This last method is shown to be more accurate and more computationally efficient. It has been successfully applied on a full-scale network. Some models to extends this new method to unsteady-state have been reviewed and implemented. The new method is shown to not include any further complication compared to the already existing methods.

All the previously cited methods make two common assumptions: the complete mixing at the junctions and the plug flow assumption. These two hypotheses have been shown to be inaccurate by several authors ([24], [29], [10]). In the second part of this work, numerical techniques are developed to model more complex phenomena on academic networks. Some non-homogeneous mixing models are reviewed and implemented with the newly developed method. Dispersion effects are modelled thanks to an approximate analytical solution and a finite volume scheme to go beyond the plug flow assumption. The optimal model to consider the dispersion effects is shown to depend on the Peclet number in each pipe of the network.

Contents

List of Figures

List of Tables

— 1 —

Introduction

Motivations

As you start reading these lines, you may have already taken a shower, drank tap water, started a washing machine, or any of these small daily miracles that a water distribution system offers us. The omnipresence of water distribution networks (which will be abbreviated as WDNs for the rest of this thesis) in our daily life may lead us to reduce WDNs to simple and well-understood systems, which they are not. Among many concerns for WDNs (energy efficiency, leaks, hydraulic integrity,...), a major and essential one is to ensure to every customer good water quality.

However, water quality management is challenging due to the ubiquitous and complex physical, chemical, and biological interactions occurring inside WDNs [4]. Water quality can be managed by sample analyses. Unfortunately, this technique is inefficient, costly, and does not allow to systematically detect quality issues everywhere and at any time in the network.

Another technique to evaluate water quality is to compute the water ag, such as in Grayman et al. [9], Kourbasis et al. [16]), or Rhoads et al. [22] where the water age is taken as an indicator to evaluate water quality inside WDNs. As shown by the American Water Works Association [2], water age is a major factor in water quality deterioration within the distribution system. Several studies have linked the water age and water quality. Blokker et al. [4] have correlated water age and various microbial parameters. Master et al. [21] relate higher water age in WDNs to change in chemical and biological properties that impact corrosivity and regulatory compliance with lead and copper action levels. Imran et al. [12] emphasise the importance of water age and temperature for the development of indicators of corrosion-related problems, like red water release. Water quality is also linked to nitrification, which is a biochemical reaction that causes a decrease in chloramine residual, a large increase in heterotrophic bacteria population, a change in PH, and other adverse effects [15]. Ensuring minimal residence time of water inside WDNs may be the 'key operational tool' to prevent nitrification [15]. Other water quality problems associated with an increase in water age include disinfectant decay, disinfectant by-product formation, taste, and odour,... [2]. A summary of water quality problems associated with age, adapted from [2], is shown in Table 1.1.

Current practices are often based on flow weighted mean age at the junctions. Indeed, this value can be easily accessed thanks to the widespread public domain software EPANET,

Chemical issues	Biological issues	Physical issues
Disinfectant decay	*Nitrification	Sediment deposition
*Desinfectant by-product formation	*Desinfectant by-product biodegradation	Temperature increases
Corrosion	*Microbial	Colour
Taste and odour	Taste and odour	

Table 1.1: Summary of water quality problems associated with water age. * Emphasises potential health impact. Adapted from [2].

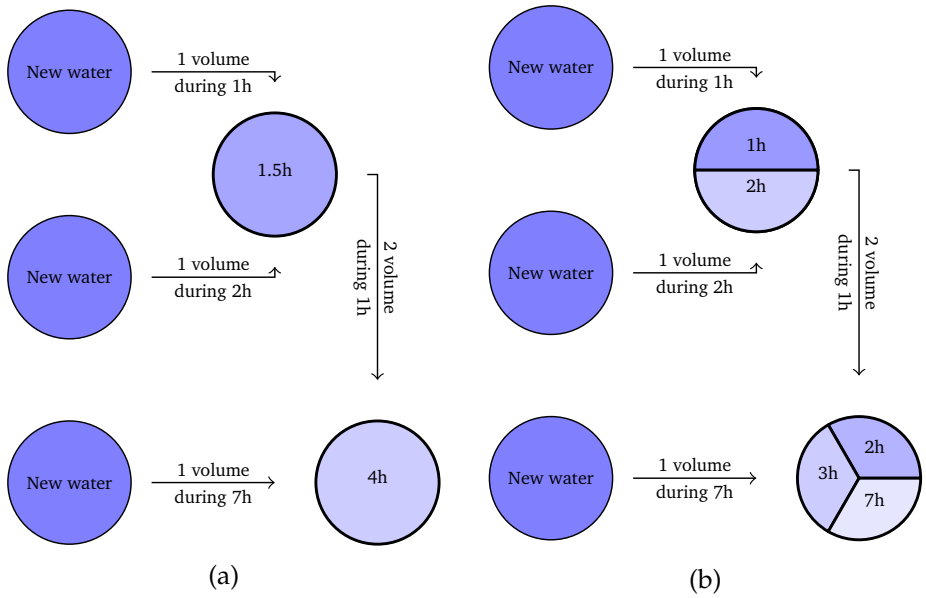


Figure 1.1: Going beyond mean age computation. Adapted from [20].

developed by the United States Environmental Protection Agency [27]. The flow weighted mean age technique is briefly explained in Fig. 1.1a. In this simple example, two nodes supply a third node that in turn supply a fourth node with the help of a last fifth node. At each junction, water age is computed by averaging incoming water age, weighted with their respective incoming flows. However, mean age computation has serious limitations [20]. To illustrate these limitations, let consider the same network without averaging water. This situation is shown in Fig. 1.1b. The key information here is not that the mean age of the water is 4 hours at the last node, but that a third of the water has an age equal to 7 hours [20]. These two pieces of information, correct but different, will lead to different water quality evaluations and eventually wrong decisions for water quality management.

Previous works from Delhez et al. [7] show the water age to be more complex than a single mean value: the water age should actually be described by a water age *distribution*. Without mentioning it, Machell et al. have shown that the key information is not the water mean age, but the water age distribution. This is what motivated this master thesis, which aims at improving water age representation in WDNs by computing water age distributions.

Study Framework

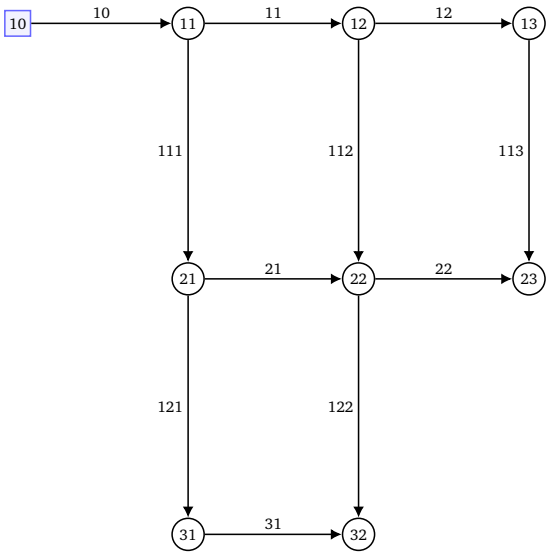
The purpose of this master thesis is not the hydraulic computation but the water age computation thanks to hydraulic data. The hydraulic computations have been done thanks to the software EPANET. The results of the hydraulic simulations are taken as input for the present work.

Four WDNs have been investigated in this master thesis. They are of different levels of complexity. The simplest one, which is named 'Net1' for the rest of this work, originates from examples delivered with the software EPANET. It has been slightly modified compared to the original one for the need of this work. It is composed of 8 junctions, 1 reservoir, and 11 pipes. More details are available in Fig. 1.2a. The second network is an experimental one developed by a Brasilian team. It is referred as 'Test rig' in the next (Fig. 1.2b). The last 'academic' network that has been considered is the one described in Jowitt et al. [14]. It has 3 reservoirs, 23 junctions, and 36 pipes. Eventually, a real network was available for this master thesis. It is a network from the municipality of Jockgrim in Germany composed of 4551 junctions, 5114 pipes, and 6 reservoirs. These four networks may have been slightly modified for the need of the master thesis. More details are available in appendix A.

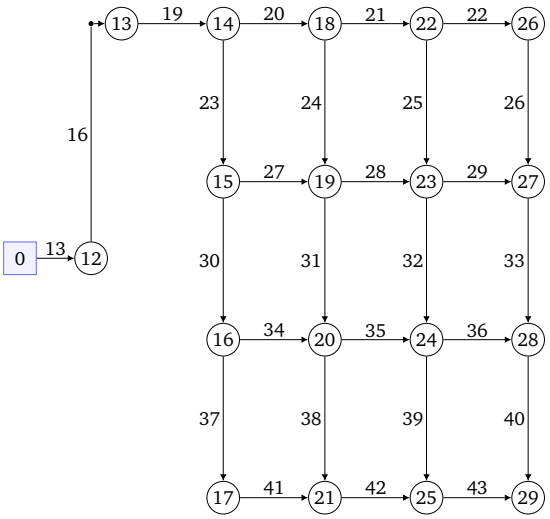
The present work is divided into two parts. The first part tries to improve the water age representation using common modelling assumptions. The developed methods are tested on all networks, up to the full-scale Jockgrim network. The second part of this master thesis investigates underlying assumptions in current practices of modelling water age in WDNs. Current practices neglect dispersion effects and assume perfect mixing at junctions. These two assumptions have been shown to be inaccurate by several authors ([29], [24], [10]). The complete mixing assumption will be mainly studied on network 'Test rig', whereas dispersion effects will be studied on the simplest network 'Net1'. The philosophy of this work has been to developed a simple model applicable to a full-scale network, and to study more complex phenomena on simpler networks, as shown in Table 1.2. A last third chapter briefly explained how the different numerical methods have been implemented.

	Modelling water age with state-of-the-art assumptions Chapter 2		Relaxing assumptions Chapter 3	
	Steady-state	Unsteady-state	Dispersion	Non-homogeneous mixing
Net1	✓	✓	✓	
Test rig				✓
Jowitt et al.	✓	✓		
Jockgrim	✓			✓

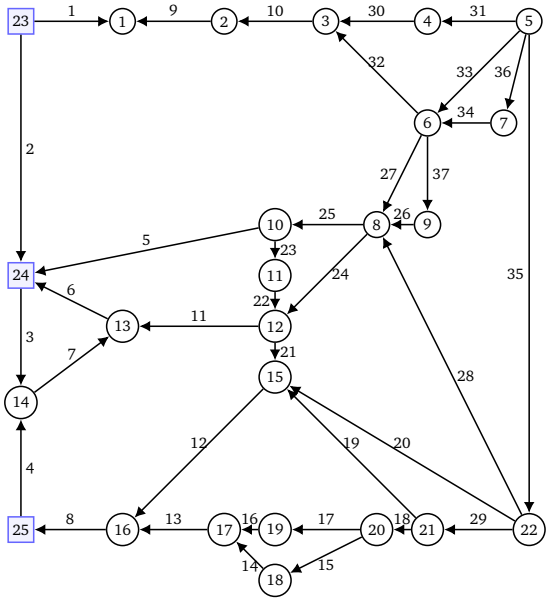
Table 1.2: Tabular plan.



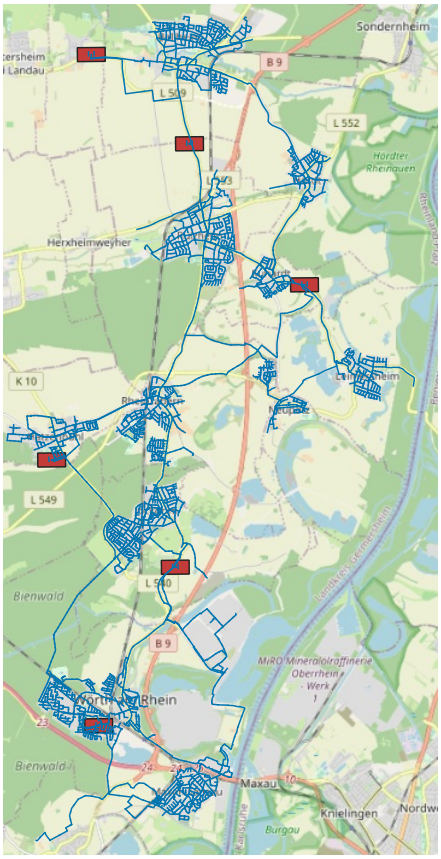
(a) Net1 [27]



(b) Test rig



(c) Jowitt et al. 1990 [14]



(d) Jockgrim

Figure 1.2: Studied WDNs. Round nodes are junctions, square nodes reservoirs, and arrows indicated the conventional positive direction of the flows. All parameters can be found in appendix A.

— 2 —

Modelling water age with state-of-the-art assumptions

2.1 Introduction

This section investigates water age computation with common assumptions in the field, the main one being the plug-flow assumption that neglects any dispersion effects. These assumptions and the mathematical framework in which the numerical methods will be described are first introduced. This mathematical framework will be used to describe two existing methods: the one from the software EPANET and the one developed by Machel et al. [20], called the bins method. These two methods have been reviewed and reimplemented.

Then, a new method that has been implemented will be introduced with the aim of improving water age representation in WDNs. This method will be introduced for steady-state first. The results obtained with this method will be compared to results that can be obtained with the already existing methods. The goal of all the techniques described in this chapter is to be used on real full-scale networks. The extension for such networks is thus discussed.

In the optic to extend the developed methods for unsteady-state, how to model such states is described afterwards. The different methods for unsteady-state have been implemented and discussed.

2.2 Mathematical Framework

This section introduces the general mathematical framework that is used for the rest of this thesis. According to Delhez et al. [7], the age distribution function, $c(t, \boldsymbol{x}, \tau)$, can be defined as follows: 'at time t, the mass of water contained in the volume

$$\delta\Omega = \{(x,y,z)|x(x,y,z) \in \mathbb{R}^3, \\ \text{and } (x-\Delta x/2, y-\Delta y/2, z-\Delta z/2) \leqslant (x,y,z) \leqslant (x+\Delta x/2, y+\Delta y/2, z+\Delta z/2)\},$$

with an age lying in the interval

$$\delta\tau = \left\{\tau|\tau \in \mathbb{R}^+, \text{and } \tau - \Delta\tau/2 \leqslant \tau \leqslant \tau + \Delta\tau/2\right\},$$

tends to $\rho c(t, \boldsymbol{x}, \tau)\delta\Omega\delta\tau$ where ρ is the water density'. A mass conservation budget allows Delhez et al. to derive the governing equation for water age at a time t and a location $\boldsymbol{x}$

$$\frac{\partial c}{\partial t} + \frac{\partial c}{\partial \tau} = -\nabla \cdot (c\mathbf{u} - \boldsymbol{k} \cdot \nabla c)\,, \tag{2.1}$$

where τ is the water age, and $c(t, \boldsymbol{x}, \tau)$, $u(t, \boldsymbol{x})$, $\boldsymbol{k}(t, x)$ denote the the water age distribution, the fluid velocity field, and the diffusivity tensor, respectively.

The governing equation will now be specified for confined flow inside pipes of WDNs. To help the reading, Fig. 2.1 illustrates the mathematical notations.

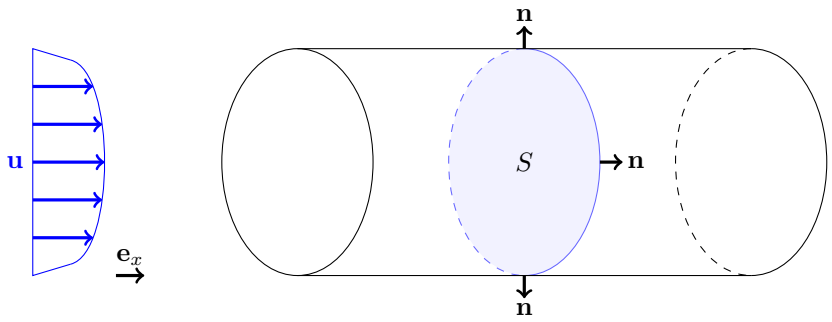


Figure 2.1: Graphical representation of a single pipe.

Integrating Eq. 2.1 over the cross-section of a given pipe gives

$$\int_S \frac{\partial c}{\partial t} dS + \int_S \frac{\partial c}{\partial \tau} dS = -\int_S \nabla \cdot (\mathbf{u}c - \mathbf{k} \cdot \nabla c) dS. \tag{2.2}$$

The cross-section does not depend on the time and the water age. Furthermore, the cross-section is assumed constant in the streamwise direction, x. Eq. 2.2 can thus be rewritten as

$$\frac{\partial}{\partial t}\int_S c dS + \frac{\partial}{\partial \tau}\int_S c dS = -\frac{\partial}{\partial x}\int_S \left(uc - k\frac{\partial c}{\partial x}\right) dS - \int_S \nabla_{2D} \cdot (\mathbf{u}_{2D}c - \mathbf{k}_{2D} \cdot \nabla_{2D} c) dS,$$

where the streamwise component and the cross-sectional components (indicate with the 2D indices) of the last term have been separated. The diffusion tensor has been assumed isotropic $k\mathbf{I} = \mathbf{k}$, and $u = \mathbf{u} \cdot \mathbf{e}_x$. Applying 2D Gauss theorem for the last term leads to

$$\frac{\partial}{\partial t}\int_S c dS + \frac{\partial}{\partial \tau}\int_S c dS = -\frac{\partial}{\partial x}\int_S \left(uc - k\frac{\partial c}{\partial x}\right) dS - \oint_{\partial S} (\mathbf{u}c - \mathbf{k} \cdot \nabla c) \cdot \mathbf{n} dS, \tag{2.3}$$

where ∂S and $\mathbf{n}$ are the boundary and the normal to the cross-section S of the pipe. This term is equal to zero because of the impermeability condition for pipes. Let introduce the mean value over the cross-section for some quantities

$$C = \frac{1}{S}\int_S c dS, \quad \overline{uc} = \frac{1}{S}\int_S uc dS, \text{ and } \quad \overline{k\frac{\partial c}{\partial x}} = \frac{1}{S}\int_S k\frac{\partial c}{\partial x} dS. \tag{2.4}$$

With these notations, Eq. 2.3 becomes

$$S\frac{\partial C}{\partial t} + S\frac{\partial C}{\partial \tau} + S\frac{\partial \overline{uc}}{\partial x} = S\frac{\partial}{\partial x}\left(\overline{k\frac{\partial c}{\partial x}}\right). \tag{2.5}$$

Let rewrite the term $\overline{uc}$

$$\overline{uc} = \frac{1}{S}\int_S [U + (u-U)][C + (c-C)] dS, \tag{2.6}$$

if U is chosen to be the mean value of u over the cross-section, $U = \frac{1}{S}\int_S u dS$, then

$$\overline{uc} = UC + \frac{1}{S}\int_S (u-U)(c-C) dS,$$

since

$$\frac{1}{S}\int_S U(c-C) dS = \frac{U}{S}\int_S (c-C) dS = 0, \text{and} \quad \frac{1}{S}\int_S C(u-U) dS = \frac{C}{S}\int_S (u-U) dS = 0,$$

by definition of the mean. With these results, Eq. 2.5 becomes

$$\frac{\partial C}{\partial t} + \frac{\partial C}{\partial \tau} + \frac{\partial}{\partial x}(UC) = \underbrace{-\frac{1}{S}\int_S (u-U)(c-C) dS}_{\star} + \underbrace{\frac{\partial}{\partial x}\left(\overline{k\frac{\partial c}{\partial x}}\right)}_{\dagger}.$$

The term † represents the effect of diffusion in the streamwise direction, whereas the term $\star$ is due to the dispersion effects. In the following, these two effects will be modelled with a single diffusion term

$$-\frac{1}{S}\int_S (u-U)(c-C) dS + \frac{\partial}{\partial x}\left(\overline{k\frac{\partial c}{\partial x}}\right) \approx \frac{\partial}{\partial x}\left(K\frac{\partial C}{\partial x}\right),$$

where K is a new parameter, with the unit of a diffusion coefficient. It may also be used to model the influence of turbulence. Assuming that the discharge is uniform throughout the pipe, our modelling equation eventually writes

$$\frac{\partial C}{\partial t} + \frac{\partial C}{\partial \tau} = -U\frac{\partial C}{\partial x} + \frac{\partial}{\partial x}\left(K\frac{\partial C}{\partial x}\right). \tag{2.7}$$

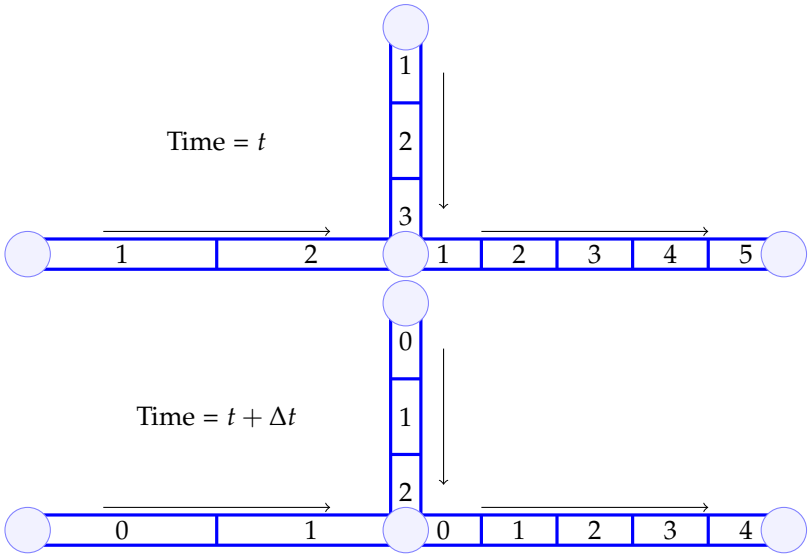
The dispersion effects will be investigated in chapter 3. For now, they are neglected, and the governing equation for this chapter is

$$\frac{\partial C}{\partial t} + \frac{\partial C}{\partial \tau} = -U\frac{\partial C}{\partial x}. \tag{2.8}$$

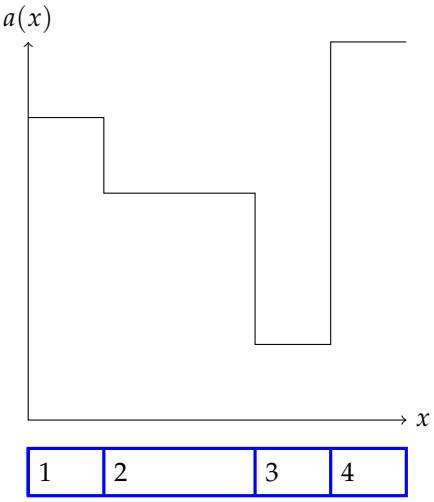
2.3 Two already available methods

2.3.1 Epanet [27]

Nowadays, if an end-user wants to compute the water age in a network, it will probably use the software EPANET [27]. EPANET assumes perfect mixing and no dispersion. It is able to compute water age for a time-dependent simulation. Its major drawback is that it does not compute the water age distribution but only mean ages. The spatial discretisation in the EPANET software is based on a Lagrangian method as illustrated in Fig. 2.2a. It relies on a sequence of water particles that represent water parcels inside the WDN. At each time step, the location of the parcel is shifted by the distance traveled during this given time step, and the mean age inside the water segment is incremented by Δt. This is the intuitive explanation that can be deduced from the EPANET docmentation. Hereafter is described a formal description of how a mean age computation method, similar to the one of EPANET, can be implemented.



(a) Lagrangian scheme of EPANET software illustrated on a simple network. Illustration adapted from EPANET documentation [27]. (Here the water particles are represented equidistant for simplicity.)



(b) Piecewise constant reconstruction of the mean age profile through a fictitious pipe.

Figure 2.2: Illustration of the numerical scheme of EPANET.

Starting from Eq. 2.8

$$\frac{\partial C}{\partial t} + \frac{\partial C}{\partial \tau} = -U\frac{\partial C}{\partial x},$$

a formal mathematical framework can be established for the age computation performed by EPANET. Since C is the water age distribution across the cross-section of the pipe, the mean water age for the cross-section can be found thanks to its first moment

$$a(x,t) = \int_0^{+\infty} \tau C(x,t,\tau) d\tau, \tag{2.9}$$

as it was first established by Delhez et al. [7]. The governing equation for mean age can be determined by multiplying Eq. 2.8 with the age τ and then integrating over the age (see [7] or [6])

$$\int_0^\infty \tau \frac{\partial C}{\partial t} d\tau + \int_0^\infty \tau \frac{\partial C}{\partial \tau} d\tau = -\int_0^\infty \tau U \frac{\partial C}{\partial x} d\tau. \tag{2.10}$$

Injecting Eq. 2.9, and considering

$$\frac{\partial (\tau C)}{\partial \tau} - C = \tau \frac{\partial C}{\partial \tau},$$

Eq. 2.10 can be rewritten as

$$\frac{\partial a}{\partial t} + [\tau C]_0^\infty = 1 - U \frac{\partial a}{\partial x},$$

where the normalisation of the age distribution property has been used $\int_0^\infty C d\tau = 1$. Following common assumptions ([7])

$$\lim_{\tau \to 0} \tau C = 0 = \lim_{\tau \to \infty} \tau C, \tag{2.11}$$

one has

$$\frac{\partial a}{\partial t} = 1 - U \frac{\partial a}{\partial x}. \tag{2.12}$$

From the right-hand side of this last equation, it is clear that the first term is responsible for aging and the second one for the advection. Let us introduce a temporal discretisation $a(x, n\Delta t) = a^{(n)}(x)$ with a time-stepping Δt and an Euler explicit scheme

$$a^{(n)} \approx a^{(n-1)} + \Delta t - \Delta t U \frac{\partial a^{(n-1)}}{\partial x}. \tag{2.13}$$

The spatial derivative in Eq. 2.13 is approximated with an upwind finite difference on an adaptive mesh of length $\Delta x^{(n)}$, where $\Delta x^{(n)}$ is the traveled distance at time step n. Assuming the mean age at time step $n-1$ is known at location x, the mean age at the next time step and at location $x + \Delta x^{(n)}$ can be found

$$\begin{cases} \Delta x^{(n)} \equiv U(n\Delta t)\Delta t, \\ a^{(n)}(x + \Delta x^{(n)}) \approx a^{(n-1)}(x + \Delta x^{(n)}) + \Delta t - \Delta t U \dfrac{a^{(n-1)}(x + \Delta x^{(n)}) - a^{(n-1)}(x)}{\Delta x^{(n)}} = a^{(n-1)}(x) + \Delta t. \end{cases}$$

Since Epanet numerical scheme is based on segments, the age between computational nodes is retrieved by constant upwind reconstruction as depicted in Fig. 2.2b.

When several water 'segments' merge at the junction j, complete mixing is assumed to compute the age at the junction

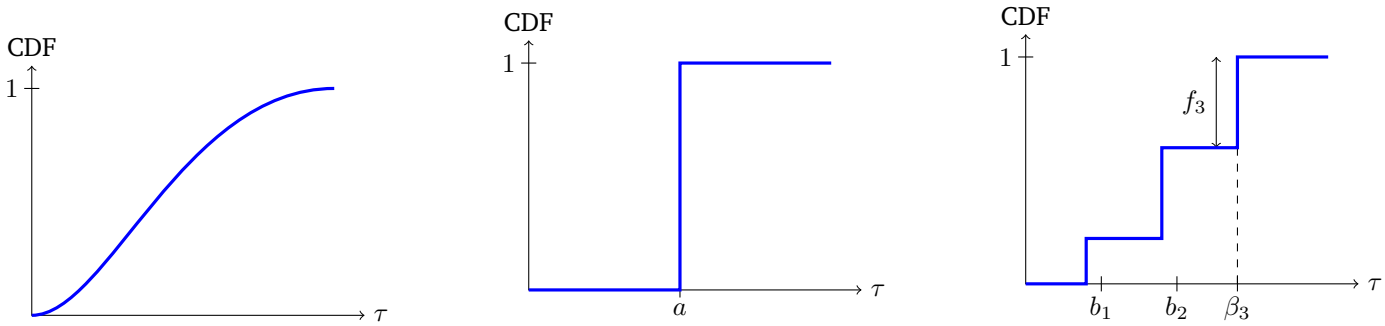
$$A_j^{(n)} = \frac{\sum_{i \in P^+} \left| Q_i^{(n)} \right| a_i^{(n)}}{\sum_{i \in P^+} \left| Q_i^{(n)} \right|}, \tag{2.14}$$

where $a_i^{(n)}$ is the mean water age entering the junction with a discharge Q_i from pipe i at time step n, and P^+ is the set of pipes with a flow entering the junction. This method has been implemented and compared to other methods.

2.3.2 Machell et al. 2009 [20]

A second method has been described by Machell et al. (2009) [20]. The presentation of the method has been modified compared to the original paper to make the reading easier and consistent with the rest of this thesis. Although the presentation is different, the resulting method is identical in practice.

As stated in the introduction, mean age computation may be unsatisfactory for water quality monitoring. The mean age computation result may not reflect the actual complexity of the age distribution. To illustrate this point, the cumulative distribution function (CDF) of a fictitious age distribution is depicted in Fig. 2.3a. The CDF that would be obtained with mean age computation, thanks to the software EPANET, is a Heaviside function centred at the mean age, as shown in Fig. 2.3b. Machell et al. thus proposed a method to access more information on the age distribution at the junctions than simply its mean value. For this purpose, they introduced so-called 'age bins' in the model, and they tracked the fractions of water within each bin. A bin is characterised thanks to a lower and an upper age limit. Components of the age distribution that lie in the same bin at a given node at any time steps are grouped to form a single value. An example of results that could be obtained with this method is shown in Fig. 2.3c.



(a) Fictitious CDF.

(b) CDF from mean age computation (Section 2.3.1).

(c) Potential CDF from the bins method with three bins.

Figure 2.3: CDFs that can be computed thanks to already available methods.

First, it introduces N_b age bins, noted B_i,

$$B_1 = [0,\ b_1[\,, \quad B_2 = [b_1,\ b_2[\,, \quad \dots \quad , \quad B_{N_b} = [b_{N_b-1},\ +\infty[\,.$$

Formally, the method is equivalent to approximate the age distribution as a sum over the N_b bins, where each term is the product of a Dirac impulse located somewhere in the bin (noted $\beta_b(x,t)$) and the fraction ($f_b(x,t)$) of the water in a given bin B_b

$$C(x, n\Delta t, \tau) \approx \sum_{b=1}^{N_b} f_b^{(n)}(x)\delta\left(\tau - \beta_b^{(n)}(x)\right), \quad \text{where } \beta_b^{(n)} \in B_b^{(n)}\ \forall b, \text{ and } \sum_{b=1}^{N_b} f_b^{(n)}(x) = 1. \tag{2.15}$$

The unknowns in this method are the fractions, f_b, and the location of the Dirac impulse, β_b, for each bin B_b at all computation nodes, and at all time steps. In the original paper,

the numerical method is developed on a fix-grid. For the sake of simplicity, the numerical method is here described on an adaptive-grid such as described in the previous Section for the software EPANET. The effects of a fix-grid scheme will be discussed later on. The 'bins method' is described step-by-step hereafter.

1. First, the fractions and the ages are initialized such that all water has an age zero for all junctions of the network, *i.e* $C_j(t = 0, \tau) = \delta(\tau = 0)$ for all junctions j. The f and β's should then be initialized as

$$\left[f_1^0 = 1, \quad f_b^0 = 0 \ \forall b \neq 1, \text{ and } \quad \beta_b^0 = 0 \ \forall b\right] \quad \forall \text{ junctions.}$$

2. Then, each age component, β_b, is updated for the next time step similarly as the implementation described for the software EPANET.

$$\beta_b^{(n)}(x + \Delta x^{(n)}) = \beta_b^{(n-1)}(x) + \Delta t \quad \text{if } f_b^{(n-1)} > 0.$$

3. If, because of the previous step, one has an age component out of its bin (*i.e* $\beta_b \notin B_b$), water that was in this bin is transferred in the relevant bin.

$$\text{If } \beta_b^{(n)} \in B_c \text{ with } b \neq c, \quad \Rightarrow \frac{f_b^{(n)}\beta_b^{(n)} + f_c^{(n)}\beta_c^{(n)}}{f_b + f_c} \rightarrow \beta_c^{(n)}, \quad f_b + f_c \rightarrow f_c,$$

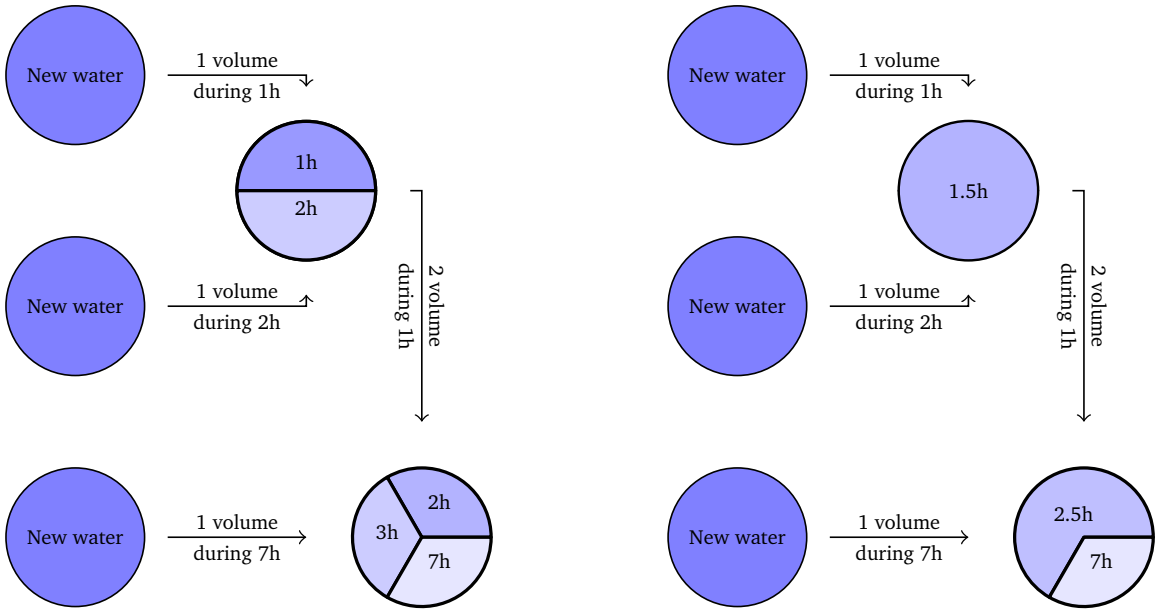
   and the β_b and f_b are then reset to zero.

4. Eventually, the age at each junction j is updated according to complete mixing

$$C_j(n\Delta t, \tau) \approx \sum_{b=1}^{N_b} f_b^{(n)} \delta\left(\tau - \beta_b^{(n)}\right), \quad \text{with } \beta_b^{(n)} = \frac{\sum_{i \in P^+} \left|Q_i^{(n)}\right| f_{i,b}^{(n)} \beta_{i,b}^{(n)}}{\sum_{i \in P^+} \left|Q_i^{(n)}\right| f_{i,b}^{(n)}},$$
$$\text{and } f_b^{(n)} = \frac{\sum_{i \in P^+} \left|Q_i^{(n)}\right| f_{i,b}^{(n)}}{\sum_{i \in P^+} \left|Q_i^{(n)}\right|}.$$

5. The steps 2, 3, and 4 are iterated for all time steps.

Note that, in the original paper, up to nine bins were investigated, $N_b = 9$. The limitation on the bins number arises from the computational power limit. This method has been reimplemented for comparison with the new approach described in the following. A simple example of computations for a five junctions network is shown in Fig. 2.4 to illustrate this Section. In these simple examples, the computation has been done for steady-state and two different sets of bins.



(a) Age bins are: $B_1 = [0,\ 1.5[$, $B_2 = [1.5,\ 2.5[$, $B_3 = [2.5,\ 3.5[$, and $B_4 = [3.5,\ +\infty[$.

(b) Age bins are: $B_1 = [0,\ 0.5[$, $B_2 = [0.5,\ 2.5[$, $B_3 = [2.5,\ 3.5[$, and $B_4 = [3.5,\ +\infty[$.

Figure 2.4: Illustrations of the bins method on a simple case. Adapted from Machel et al. [20]. Note that other choices of age bins are possible to obtain the same results.

2.4 Steady-state

2.4.1 Iterative method

The model developed by Machel et al. (2009) introduces an age discretisation to offer a better understanding of the age distribution in a WDN. As for all methods, there are drawbacks to the bins methods. Some of them are listed below.

- The age bins are fixed prior to the computation when the optimal configuration is unknown.
- Information is lost, not only if several components of the age distribution lie in the same bin, but also if there is a loss in information in any upstream nodes. Indeed, the age bins chosen in Fig. 2.4b should allow representing the age distribution at the last node correctly. It is not the case because there is an error introduced at the upstream node.
- The number of bins is the same for every computational point. This causes over-refinement for some nodes, whereas some are under-refined.

The method presented here tries to address some of these drawbacks. For clarity, this method deals with steady-state first. How to extend this method to unsteady-state will be addressed in the next Section. The main idea of this method is to not introduce any prefered discretisation for the age distribution before the computation.

Again, this section starts from Eq. 2.8 but assumes steady-state, $\frac{\partial \cdot}{\partial t} = 0$, (*i.e* both the flow is steady and the distributions of water age do not evolve with time)[1]

$$\frac{\partial C}{\partial \tau} = -U\frac{\partial C}{\partial x},$$

which is a transport equation through the spatial and the age space. To get a well-posed problem, one should specify the domain of the variables and the boundary conditions. Let us first consider a semi-infinite pipe connected to a zero-age inlet $C(\tau) = \delta(\tau)$. The extension to a WDN will be done later. Mathematically, it gives

$$\begin{cases} \dfrac{\partial C}{\partial \tau} + U\dfrac{\partial C}{\partial x} = 0 & \text{in }]0, +\infty[, \quad (2.16a) \\ C(0,\tau) = \delta(\tau) & \text{at } x = 0. \quad (2.16b) \end{cases}$$

This problem can be solved thanks to the method of the characteristics. Changing variables with

$$\eta = x + U\tau \qquad \xi = x - U\tau,$$

gives, for Eq. 2.16a,

$$\frac{\partial}{\partial \eta}\left(C(\eta, \xi)\right) = 0.$$

Which means that any distribution of the form $C\left(x - U\tau\right)$ satisfying Eq. 2.16b is a solution. Using Eq. 2.16b, one has

$$C^\star\left(-U\tau\right) = \delta\left(\tau\right) \Rightarrow C^\star\left(x - U\tau\right) = \delta\left(-\frac{x}{U} + \tau\right) \Rightarrow C(x,\tau) = \delta\left(\tau - \frac{x}{U}\right).$$

At the outlet of a pipe of finite length L, the water age distribution is approximated with the water age distribution computed for a semi-infinite pipe in $x = L$, which is

$$C(L,\tau) = \delta\left(\tau - \frac{L}{U}\right).$$

Because of the boundary condition 2.16b, this solution will be referred to 'the impulse response'.

This result will now be generalised to a WDN. First, the pipe is connected to an upstream junction with an arbitrary age $C_{\text{up}}(\tau)$. In that case, the problem writes

$$\begin{cases} \dfrac{\partial C}{\partial \tau} + U\dfrac{\partial C}{\partial x} = 0 & \text{in }]0, +\infty[, \quad (2.17a) \\ C(0,\tau) = C_{\text{up}}(\tau) & \text{at } x = 0. \quad (2.17b) \end{cases}$$

From the linear system theory, one can generalise the result obtained previously for the impulse response thanks to a convolution product

$$C(L,\tau) = \int_0^\tau C_{\text{up}}\left(\tau - \tau'\right)\delta\left(\tau' - \frac{L}{U}\right)d\tau' = C_{\text{up}}\left(\tau - \frac{L}{U}\right).$$

[1]Note that a different situation may occur if the flow is assumed steady ($\partial_t U = 0$) but the transient evolution of C is considered. In other words, assuming steady-flow is not a sufficient condition to ensure that age distributions are also steady.

From this result, the water age distribution at any nodes can be computed iteratively. For a junction j, assuming perfect mixing again, the water age distribution is

$$C_j = \frac{\sum_{i \in P^+} |Q_i| \, C_i \left(\tau - \frac{L}{U}\right)}{\sum_{i \in P^+} |Q_i|}. \tag{2.18}$$

Intuitively, this equation means that the water age distribution at a junction j, is equal to the perfect mixing of the age distributions of the upstream junctions but aged by the travel time needed for the water to go from the upstream junctions to the considered junction. Eq. 2.18 can then be repeated iteratively to determine the water age distribution at all junctions.

Numericaly, the age at the junction j is represented by two vectors: the age components $A_j = [\alpha_1, \alpha_2, \ldots, \alpha_N]$ and the fraction of water corresponding to each age component $F_j = [f_1, f_2, \ldots, f_N]$. The age distribution is thus written as

$$C_j(\tau) = \sum_{k=1}^{N} f_k \delta(\tau - \alpha_k), \quad \text{where } \sum_{k=1}^{N} f_k = 1. \tag{2.19}$$

Conversely to the bins method, no prefered discretisation is introduced before the computation.

The numerical algorithm that has been implemented, based on Eq. 2.18 and the discretisation of Eq. 2.19, is described hereafter.

1. Initialised all junctions to have a zero age: $A_j = [0]$, and $F_j = [1]$, for all junctions j.
2. Similarly to Eq. 2.18, apply complete mixing to update the age at the junctions based on its neighbour nodes.

   $$A_j = \left[A_p + T_p, A_q + T_q, \ldots, A_r + T_r\right], \quad \text{where } T_x = \frac{L_x}{U_x},$$
   $$F_j = \frac{1}{Q_T}\left[|Q_p| \, F_p, |Q_q| \, F_q, \ldots, |Q_r| \, F_r\right], \quad \text{where } Q_T = |Q_p| + |Q_q| + \cdots + |Q_r|,$$
   where $\{p, q, \ldots, r\} \equiv P^+$ is the set of junction with a flow entering the junction j.

3. Stage 2 has to be iterated until convergence.

These new methods have been implemented and will be compared to the one described previously in the next Section.

2.4.2 Illustrations and convergence

Two networks have been used to validate the written code: the network 'Net1' described in Fig. 1.2a, and the network from Jowitt et al. in Fig. 1.2c. A constant demand is applied at each junction, and the discharges are computed thanks to the software EPANET (more details on the simulation parameters are available in Appendix A). Since the demands are constant, the solutions found have constant discharges in each pipe. These discharges are taken as input for our simulations that neglected the transient part of C.

Once the discharges are known, the water age distributions are computed numerically and analytically. The analytical results for network Net1 are shown in Table 2.1 and numerical results obtained with the new implemented method are shown in Fig. 2.5. The same work has been done with the network from Jowitt et al., but the complete results are not shown here as it would be unusefully long. The detailed 'pen-and-paper' solution for the Jowit et al. network is shown in Appendix B. Two cumulative distribution functions are shown in Fig. 2.6 as examples for this network. Numerical and analytical results remarkably agree.

Pipe	U [km/h]	L [km]	D [cm]	T [h]
10	3.69	3.21	46	8.69×10^{-1}
11	4.24	1.61	35	3.79×10^{-1}
12	4.39×10^{-1}	1.61	25	3.65
21	1.57	1.61	25	1.02
22	4.73×10^{-1}	1.61	30	3.40
31	7.73×10^{-1}	1.61	15	2.08
111	2.97	1.61	25	5.41×10^{-1}
112	1.21×10^{-1}	1.61	30	1.33×10^{1}
113	-1.31×10^{-2}	1.61	20	-1.22×10^{2}
121	1.13	1.61	20	1.41
122	4.71×10^{-1}	1.61	15	3.40

Node	C [h^{-1}]
10	δ_0
11	$\delta_{0.87}$
12	$\delta_{1.25}$
21	$\delta_{1.41}$
31	$\delta_{2.83}$
22	$0.9\delta_{2.43} + 0.1\delta_{14.56}$
23	$0.9\delta_{5.84} + 0.1\delta_{17.97}$
13	$0.98\delta_{4.9} + 0.0169\delta_{128.1} + 0.0018\delta_{140.2}$
32	$0.62\delta_{4.9} + 0.342\delta_{5.84} + 0.038\delta_{18.0}$

Table 2.1: Analytical results for network Net1. See Fig. 1.2a for Pipe ID and Node ID. U is the velocity in the pipe, L its length, and D its diameter. The water age distribution, C, is shown in the last column for each node.

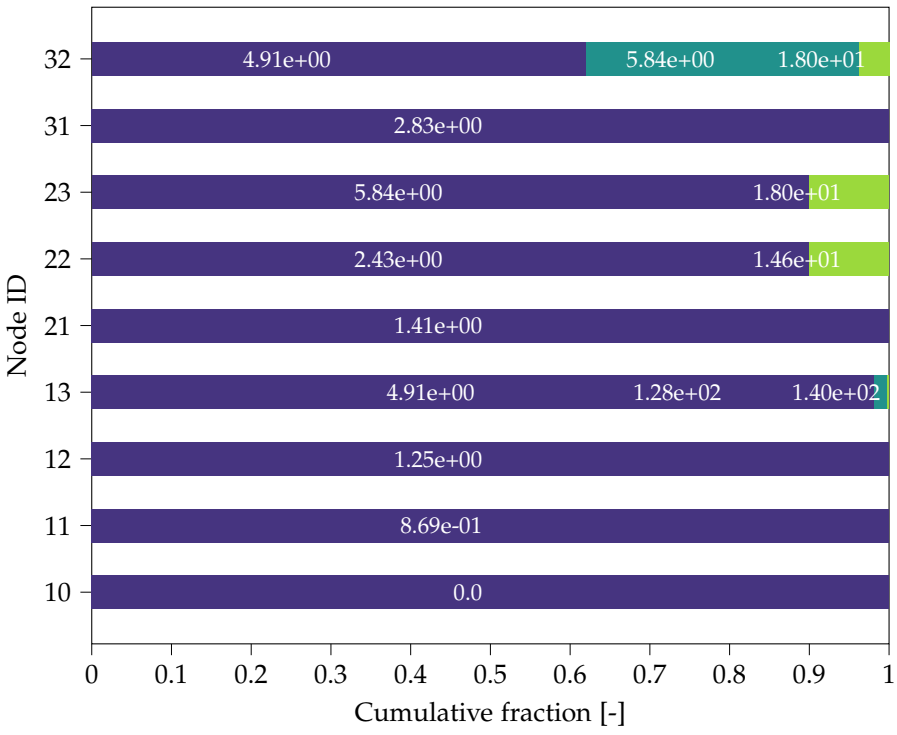


Figure 2.5: Water age distribution at each node of network Net1 computed thanks to the implemented code. Colours indicate different Dirac impulses in the distribution. The ages (in hours) are written inside each contribution (for visibility, some writings have been shifted to the left). To be compared with Table 2.1.

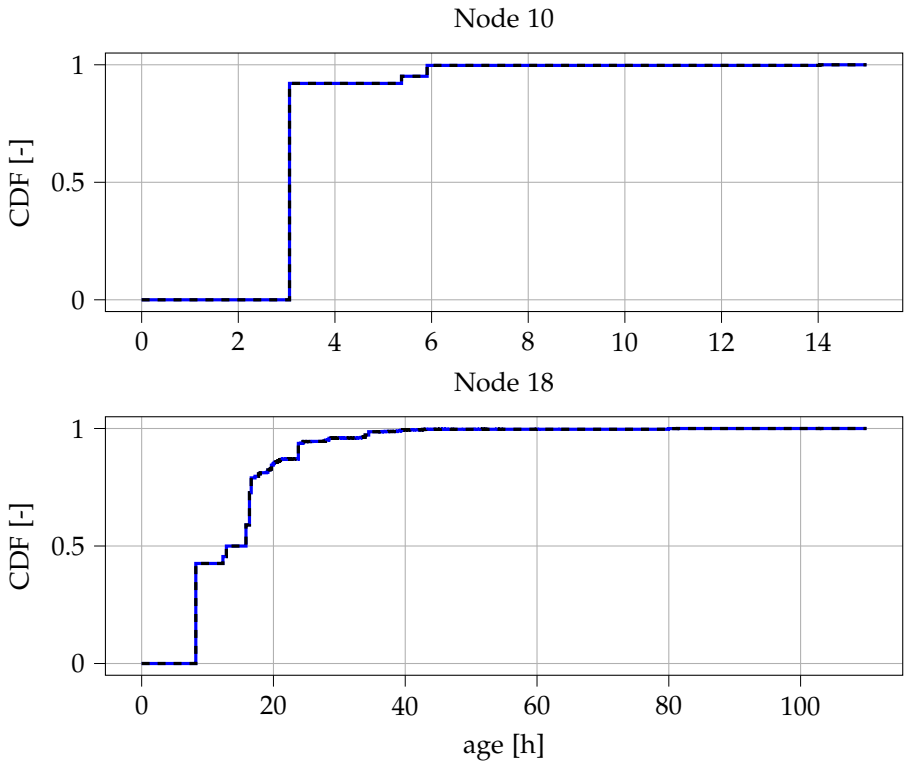


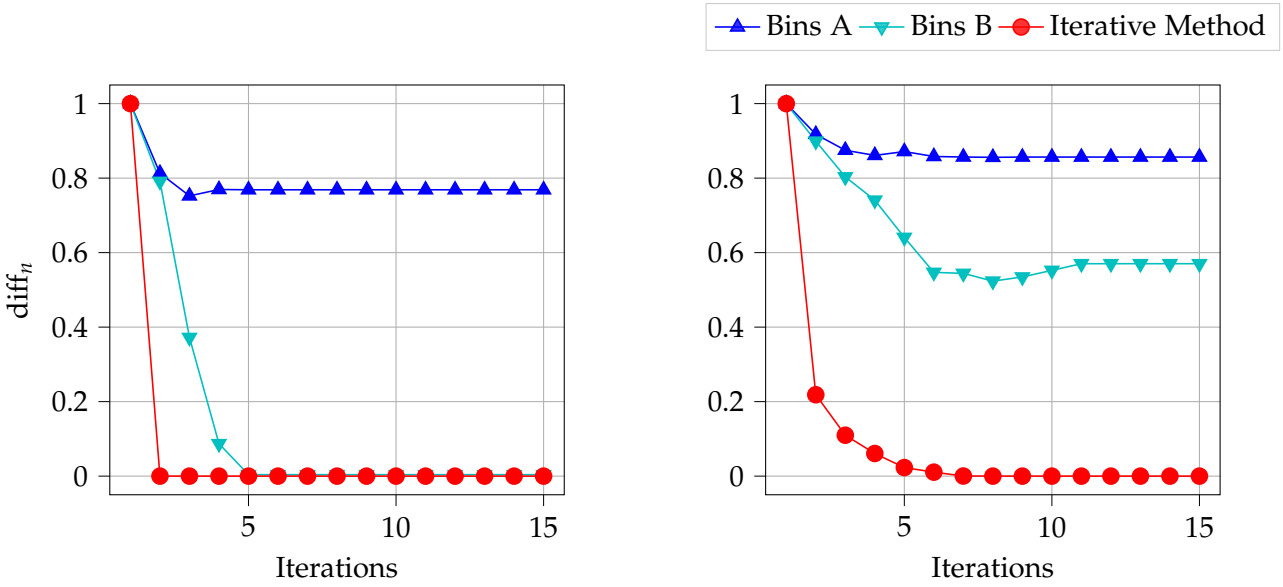
Figure 2.6: CDF of nodes 10 and 18 of the Jowitt et al. network. The solid blue line is computed with the numerical implementation, and the black dashed line is the analytical solution.

The agreement between numerical and analytical results is quantified with a difference function. The difference function, $\mathrm{diff}(C_1, C_2)$, between two distributions, C_1 and C_2, is defined as the area that separates the two cumulative water age distributions

$$\mathrm{diff}(C_1, C_2) = \int_0^\infty |C_1(\tau) - C_2(\tau)|\, d\tau. \tag{2.20}$$

The total difference function is defined as the sum of difference functions over all nodes of the network, $\mathrm{Diff}(C_1, C_2) = \sum_{\text{all junctions } j} \mathrm{diff}_j(C_1, C_2)$. For the sake of simplicity, a normalized difference function is introduced $\mathrm{diff}_n(C_1, C_2) = \mathrm{Diff}(C_1, C_2) / \max_{\text{all iterations}} (\mathrm{Diff}(C_1, C_2))$. It allows to compare convergence of the different methods on a single graph easily.

The difference functions, obtained with the implemented code and the reimplementation of the bins method, are compared in Fig. 2.7. For both networks, the implemented methods converge quickly towards the analytical solution. As predicted in the previous Section, convergence for the bins method is not guaranteed. Thanks to a wise choice of bins, convergence can be reached for the smaller network Net1. However, the existence of a particular choice of bins to ensure convergence is not guaranteed in any case. None of the chosen bins allows reaching the analytical solution on the larger network from Jowitt et al. It would required to choose at least 211 bins since it is the number of components in the age distribution at node 18. On should also be sure to not loose any information in any upstream nodes, which would required even more bins.



(a) Net1

(b) Jowitt et al.

Figure 2.7: Normalised difference functions for the bins method and the implemented iterative method. 'Bins A' lines use three bins: $\{[0,15[\,,[15,20[\,,[20,\infty[\}$ and 'Bins B' five: $\{[0,5[\,,[5,10[\,,[10,15[\,,[15,20[\,,[20,130[\,,[130,\infty[\}$ [h].

2.4.3 Application to the full-scale Jockgrim network

In the previous Section, the iterative method has been shown to be more accurate than the two already existing methods. These last two methods tolerate inaccuracy in order to remain computationally efficient. Being efficient is indeed a prior requirement to be applicable on full-scale networks, or in other words: to be useful for water quality monitoring in 'real-life' networks. This Section thus investigates the applicability of the developed method on the full-scale Jockgrim network.

A full-scale network is mainly characterised by a high number of junctions. The consequences are twofold:

- the number of paths for going from one reservoir to any junctions tends to increase, which increases the number of components in the age distribution at the junctions;
- The average distance between a reservoir and a junction increases, which increases the number of required iterations to converge. It is equivalent to say that the range of influence of all reservoirs increases.

The increase in the number of components causes each iteration to be more computationally expensive, while the number of required iterations also increases with the size of the network. This double negative effect renders convergence impossible for the junctions that are the farthest of the network.

A small modification in the previously described iterative method has been introduced to make convergence possible for all junctions of a full-scale network. It speeds up the compu-

tation at the cost of a loss of precision. The idea is to limit the maximum number of components in the age distributions to avoid the 'double punishment' of the increase in network size. When this maximum number is reached, the neighbours with the lowest contribution to the age distribution are grouped together to form a single value. To make it clear, let us write it explicitly for a junction j with $N_{\max}$ the maximum allowed number of components. If the update step of the iterative method leads to an illegal number of components

$$\begin{aligned} A_j &= [\alpha_1, \alpha_2, \ldots, \alpha_N], \quad \text{where } N = N_{\max} + 1, \\ F_j &= [f_1, f_2, \ldots, f_N], \end{aligned}$$

then the age and fraction vectors are modified as follows:

1. First, the sum of adjacent fractions, $s_\cdot$, is computed

$$F_j = \left[\underbrace{f_1}_{s_1},\ \overbrace{f_2}^{s_2=f_2+f_3},\ f_3,\ \ldots,\ \underbrace{f_x,\ f_y}_{s_x},\ \ldots,\ \overbrace{f_{N-1},\ f_N}^{s_{N-1}} \right].$$

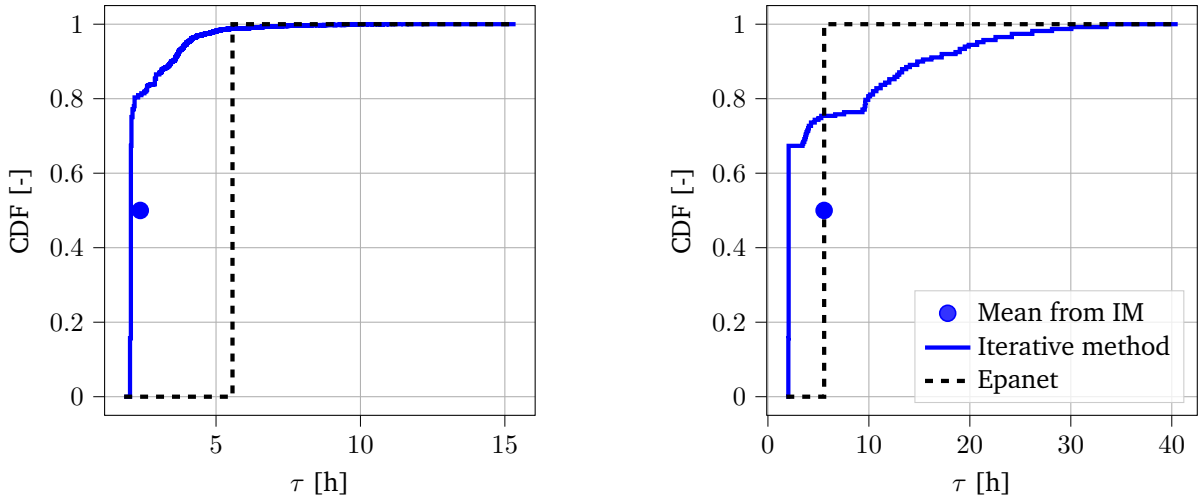
2. Then the fractions and the age with the smallest sum are grouped. For illustration, assume $s_x < s_i \ \ \forall i \in \{[0, x-1] \cup [x+1, N-1]\}$.

$$F_j = \left[f_1,\ f_2,\ f_3,\ \ldots,\ \underbrace{f_z}_{f_z=f_x+f_y},\ \ldots,\ f_{N-1},\ f_N \right],$$

$$A_j = \left[\alpha_1,\ \alpha_2,\ \alpha_3,\ \ldots,\ \underbrace{\alpha_z}_{\star},\ \ldots,\ \alpha_{N-1},\ a_N \right] \quad \text{where } \star = \frac{f_x \alpha_x + f_y \alpha_y}{f_x + f_y}.$$

With this modification, it is possible to obtain convergence for all nodes of the network. An example on the node KUNE0005 of the Jockgrim network is shown in Fig. 2.8. Note the agreement between the mean age computed thanks to software EPANET and the mean age computed from the age distribution resulting from the developed method. This result is expected and can be used to check the convergence of the iterative method.

It is interesting to use the new iterative method and the software EPANET simultaneously. Indeed, while the second method gives a rapid value for the mean age, the first one can give more accurate results. If one wants to determine the age at a given junction, it can easily check convergence by comparing mean age results and adapt the refinement of the iterative scheme to get the more accurate results possible, while ensuring convergence. The iterative method also offers advantages over the bins method in the sense that it does not require fixing an age discretisation prior to the computation, which would induce over/under-refinement at some nodes. This causes wasted computational power in empty bins that can not be used at other junctions where it may be necessary. With the iterative method, all the computational power, and only the needed computational power, will be used.



(a) Number of iteration: 30, $N_{\max} = +\infty$.

(b) Number of iteration: 200, $N_{\max} = 50$.

Figure 2.8: Convergence of the iterative method for the node KUNE0005 of the Jockgrim network. The limitation of the number of age components allows to increase the number of iteration and reach convergence. (The mean computed with the iterative method has been represented with a dot and not a Heaviside function for visibility.)

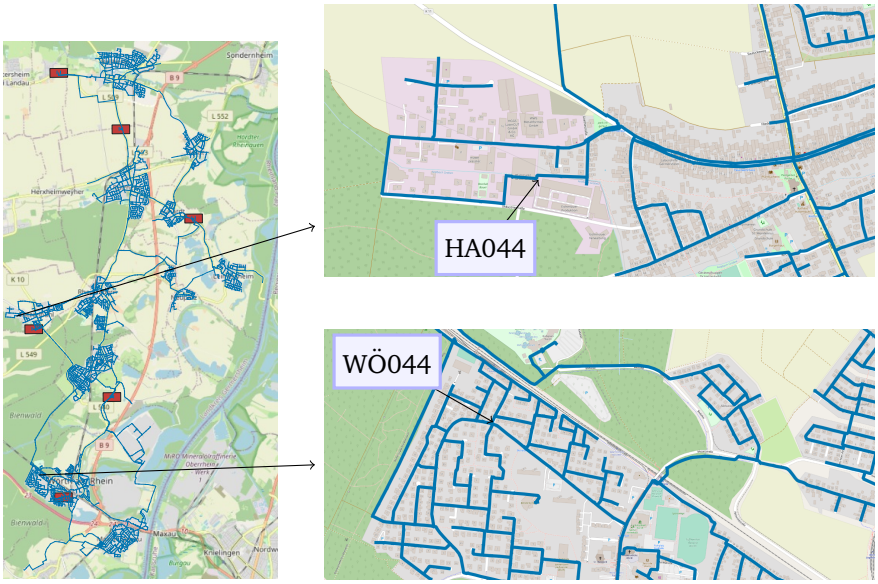
Results that can be obtained are shown in Fig. 2.9 for two junctions: HA044 in the municipality of Hatzenbühl, and WÖ044 in the municipality of Wörth am Rhein. Fig. 2.9 compares results obtained with mean age computation and the implemented iterative method. These two nodes have been selected as they are nice illustrations of the motivation of this master thesis. The mean value at node HA004 (Fig. 2.9b) is about 22 hours, but it hides the key information on the age distribution: 60% of the water is older than 30 hours. In a very figurative way, mixing 'green young' with 'red old' water gives a more complicated result than the 'yellow mid age' water obtained with EPANET. Going beyond mean age computation may also relocate the focus of attention. Whereas node WÖ044 was classified as safe 'green young junction', it is actually composed of more than 10% of water older than 30 hours.

2.4.4 Conclusion

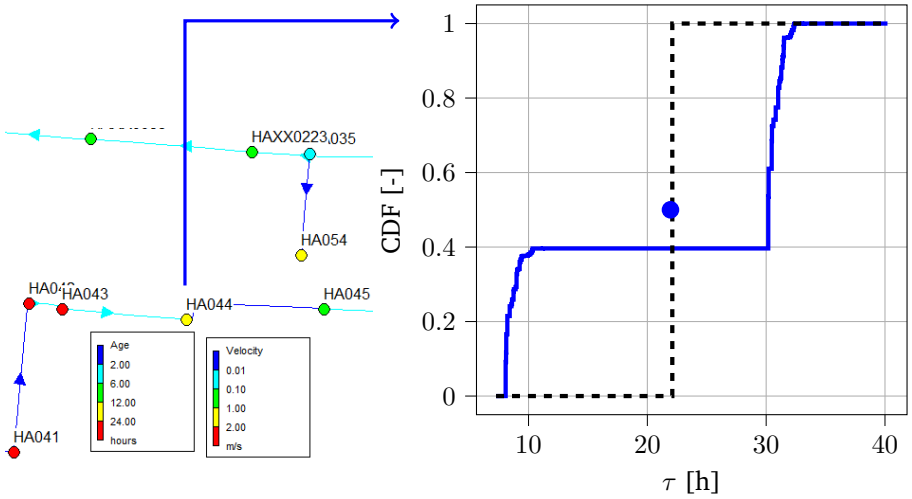
From these results, one can be convinced that more accurate results can be obtained thanks to the new method. The major distinction between this method and the bins method is the discretisation of the age distribution function. Whereas the bins method used a fixed number of bins and a fixed bins discretisation for all nodes, the implemented code uses two vectors which store only the necessary information: $[\alpha_1, \alpha_2, \ldots, \alpha_N]$ and $F_j = [f_1, f_2, \ldots, f_N]$

$$C_j(\tau) \approx \sum_{b=1}^{N_b} f_b \delta\left(\tau - \beta_b\right), \quad \text{where } \beta_b \in B_b \ \forall b \quad \longrightarrow \quad C_j(\tau) \approx \sum_{k=1}^{N_{\max}} f_k \delta(\tau - \alpha_k). \tag{2.21}$$

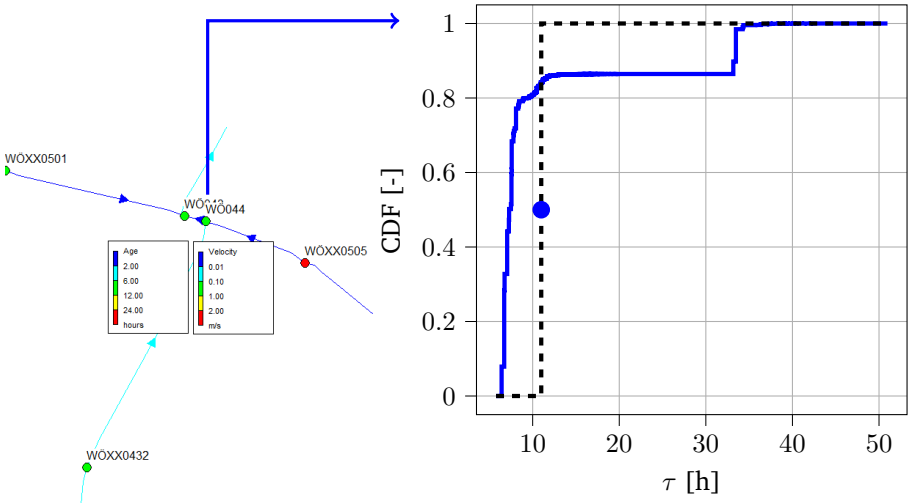
A limitation in length has been introduced to ensure convergence for all nodes. Note that for a maximum length equals to the number of bins, $N_b = N_{\max}$, the developed method always gives identical or better results since it does not fix the discretisation of the age distributions prior to the computation.



(a) Localisation of nodes HA044 and WÖ044 in the Jockgrim network.



(b) Result from EPANET (left and dashed line), and from the iterative method (blue line and dot) at node HA044.



(c) Result from EPANET (left and dashed line), and from the iterative method (blue line and dot) at node WÖ044.

Figure 2.9: Illustrations of results that can be obtained with the iterative method. Numerical parameters: $N_{\max} = +\infty$, and the number of iterations is 30.

2.5 Unsteady-state

The previous Section introduces a new method for water age computation in steady-state. However, for real configurations, the discharges in the pipes of WDNs are not constant, which means that the steady-state assumption is not valid. The goal of this Section is to investigate how unsteady-state can be simulated. A time-dependent scheme similar to the one used by EPANET and another one using a fix-grid has been implemented and compared. These methods are described in the next and the Section ends with some test cases.

For a time-dependent state, a temporal and spatial discretisation of the network will be necessary. In practice, the discharges are computed thanks to the software EPANET with a time-stepping Δt for N_t time steps. The velocity, U_n, in each pipe is known at each time-step, n, and is assumed to be constant during a time equals to Δt. If U_n is held constant during the time step Δt, the method of the characteristic can be applied. Restarting from Eq. 2.8

$$\frac{\partial C}{\partial t} + \frac{\partial C}{\partial \tau} = -U_n \frac{\partial C}{\partial x}, \quad \text{for } (n-1)\Delta t < t < n\Delta t.$$

Let us introduce the following change of variables

$$\eta = x + U_n t, \quad \xi = x - U_n t, \text{and} \quad \lambda = t - \tau.$$

The governing equation becomes,

$$\begin{aligned} &\partial_\lambda C + U_n \partial_\eta C - U_n \partial_\xi C - \partial_\lambda C = -U_n \partial_\eta C + -U_n \partial_\xi C, \\ &\Rightarrow \partial_\eta C = 0. \end{aligned}$$

which means that the solution has the form $C(\xi, \eta)$. In other words, for any fixed value of the set (ξ, λ), the solution is constant during the time step. The characteristic lines are thus defined by the set of lines

$$d \equiv \begin{cases} \xi = x - U_n t \\ \lambda = t - \tau \end{cases} \quad \forall (\xi, \lambda) \in \mathbb{R}^2.$$

Therefore, the following equality holds

$$C(t, x, \tau) = C(t - \Delta t, x - U_n \Delta t, \tau - \Delta t). \tag{2.22}$$

Shorter notations are introduced to facilitate the future developments

$$C_i^n(\tau) = C(n\Delta t, i\Delta x, \tau).$$

The age discretisation for the age distribution is done exactly in the same way as it has been done for the steady-state case, thanks to the two vectors $A_j = [\alpha_1, \alpha_2, \ldots, \alpha_N]$, and $F_j = [f_1, f_2, \ldots, f_N]$ (see Eq. 2.19). These two vectors depend on space and time

$$C_i^n(\tau) \approx \sum_{k=1}^{N} f_k(i\Delta x, n\Delta t)\delta\left(\tau - \alpha_k(i\Delta x, n\Delta t)\right). \tag{2.23}$$

2.5.1 Fix-grid scheme

This first method introduces a fix spatial discretisation in each pipe, Δx, that may be different from one pipe to another. From a practical standpoint, Eq. 2.22 means that the value of the age distribution in (t, x) should be updated thanks to the age distribution at the previous time step, $t - \Delta t$, at the localisation $x - U_n \Delta t$, but aged by Δt. Since it is unlikely that the point $x - U_n \Delta t$ falls right on a computation point, two approximations are possible to recover the value of the distribution in $x - U_n \Delta t$: a downstream approximation or an upstream approximation. An interpolated scheme is also possible and is discussed later.

$$C_i^n = \begin{cases} C_{i-l}^{n-1}\left(\tau - \Delta t + \chi \dfrac{\Delta x}{U_n}\right) & \text{(downstream)}, \\ C_{i-l-1}^{n-1}\left(\tau - \Delta t - (1-\chi)\dfrac{\Delta x}{U_n}\right) & \text{(upstream)}, \end{cases}$$

where

$$l = \lfloor U_n \Delta t / \Delta x \rfloor \quad \text{and,} \quad \chi = \frac{U_n \Delta t}{\Delta x} - l, \tag{2.24}$$

with $\lfloor \cdot \rfloor$ the floor funtion. These approximations are illustrated in Fig 2.10.

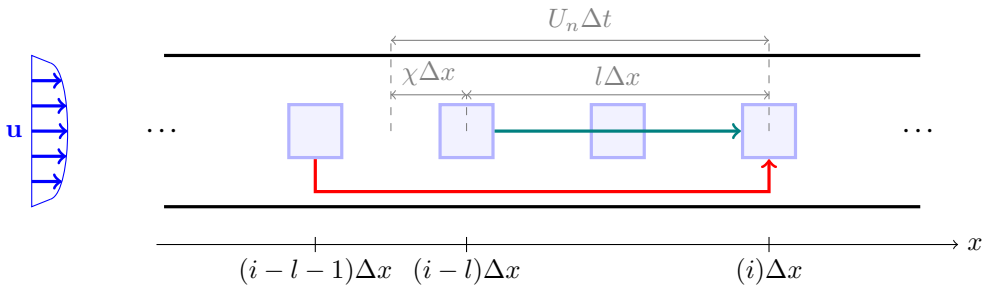


Figure 2.10: Sketch of the fix-grid numerical scheme. Blue arrows indicate the flow direction, grey arrows illustrate the text notations, red/teal-blue arrow shows the upstream/downstream approximation, and rectangles are computation nodes.

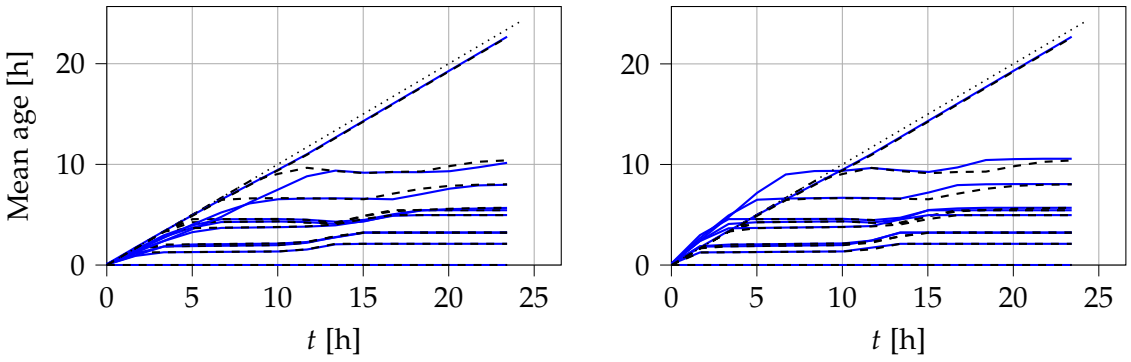
Exploiting the expressions of χ allows rewriting these update equations in a simpler form

$$C_i^n = \begin{cases} C_{i-l}^{n-1}\left(\tau - l \dfrac{\Delta x}{U_n}\right) & \text{(downstream)}, \\ C_{i-l-1}^{n-1}\left(\tau - (l+1)\dfrac{\Delta x}{U_n}\right) & \text{(upstream)}. \end{cases}$$

The two vectors representing the age distribution are thus updated as follows

$$A_i^n = \begin{cases} A_{i-l}^{n-1} + l \dfrac{\Delta x}{U_n} & \text{(downstream)} \\ A_{i-l-1}^{n-1} + (l+1)\dfrac{\Delta x}{U_n} & \text{(upstream)} \end{cases} \quad \text{and} \quad F_i^n = \begin{cases} F_{i-l}^{n-1} & \text{(downstream)} \\ F_{i-l-1}^{n-1} & \text{(upstream)} \end{cases}$$

If a downstream approximation is used, the system will tend to react too slowly. This effect is visible in Fig. 2.11a. The reverse is true for the upstream approximation, as shown in Fig. 2.11a. At each time step, the error on the propagation speed is (at most) $\Delta x/\Delta t$. As a consequence, the spatial discretisation should be small enough to model correctly all hydraulic data and to prevent significant errors in the dynamical behaviour of the system.



(a) Downstream (b) Upstream

Figure 2.11: Illustration of the upstream/downstream approximation on the fix-grid scheme for network Net1. The solid blue lines are mean ages at all nodes computed with the fix-grid method. Dashed black lines serve as references and are computed with the software EPANET.

2.5.2 Adaptive-grid scheme

A second numerical method has been implemented. Similar to the previous one, it relies on the method of the characteristics but does not use a fixed spatial grid. This method has been introduced to mimic the implementation of the software Epanet (see Section 2.3.1). It is based on lagrangian volumes of water created at the upstream node at each time step with a length $U_n\Delta t$ in which water age is equal to the upstream node age. All the already existing volumes inside the pipe are shifted towards the downstream nodes by $U_n\Delta t$. Water in these volumes is then aged by Δt. With this algorithm, the water age inside each segment represents the water age at the upstream part of the segment. The age distribution of the water at the downstream node is computed thanks to the last segment partially inside the pipe. It is thus an upstream approximation. This method is illustrated in Fig.2.12.

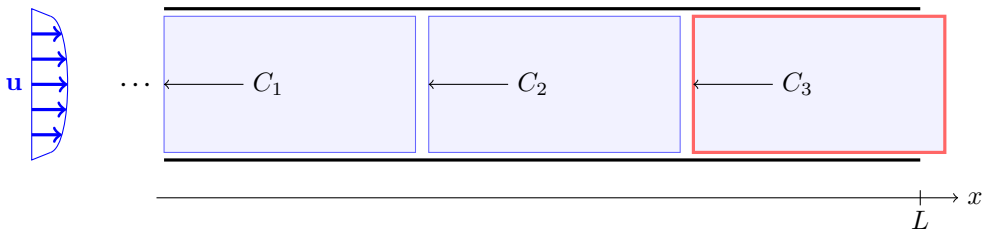


Figure 2.12: Sketch of the adaptive-grid numerical scheme. Blue rectangles are volumes of water, and the red rectangle shows the volumes used to compute the water age at the outlet.

This adaptive spatial discretisation offers a better representation of the dynamical behaviour than the fix-grid scheme since the downstream/upstream approximation of the fix-grid scheme causes the system to react too quickly/slowly. However, the adaptive-grid scheme introduces an error in the water age. To easily illustrate this point, let us consider a single pipe in a steady-state. Since the water is aged by Δt at each time step, the travel time will be approximate by

$$\Delta t \lfloor L/(U\Delta t) \rfloor$$

instead of L/U, because the only representable ages are the multiples of Δt. To ensure convergence, one should therefore ensure the time discretisation to be sufficiently small to represent the water age correctly.

2.5.3 Comparaison

At this point, it is interesting to compare both schemes. The fix-grid scheme accurately represents the travel time through pipes, whatever the time discretisation, provided that one node is at the beginning and another one at the end of the pipe. In contrast, the travel time is correctly represented in the adaptive scheme only if the time stepping is small enough. On the other hand, the dynamic of the network is correctly represented in the adaptive scheme, whereas the upstream/downstream approximation introduces non-causal/delay errors. This antagonist behavior has been summarized in Table 2.2.

	Fix-grid	Adaptive-grid
Accurate dynamic modelling	If $\Delta x \to 0$.	Yes.
Accurate travel-time representation	Yes.	If $\Delta t \to 0$.

Table 2.2: Comparaison of the fix-grid and adaptive-grid scheme to model unsteady-state.

2.5.4 Modified approximations

In this Section, some modifications are introduced to the previous schemes. The motivation behind this is to get better results for low-cost simulations. Indeed, water quality management may require performing several simulations on full-scale networks. It may thus be necessary to be able to obtain sufficiently accurate results at a minimal cost. Note that, for robust and accurate computation, one of the previous schemes should always be prefered.

Random fix-grid scheme The fix-grid scheme suffers from its downstream/upstream approximations when the spatial discretisation is not sufficiently refined. These approximations induce errors in the propagation speed. One way to mitigate this effect is to randomly choose between the upstream or the downstream approaches at each time step. To conserve the propagation speed, the random choice should be weighted according to the distance between the upstream and downstream node. In practice, the choice between the approximation follows a Bernouilli distribution. Noting the random variable $\mathcal{Y} \in \{\text{upstream}, \text{downstream}\}$, the probability mass function defining the choice is

$$f(\mathcal{Y}) = \begin{cases} \chi & \text{if } \mathcal{Y} = \text{downstream}, \\ 1-\chi & \text{if } \mathcal{Y} = \text{upstream}, \end{cases}$$

where χ was defined in Eq. 2.24 and in Fig. 2.10.

Interpolated fix-grid scheme To get even better results, one could imagine a linear reconstruction to retrieved the age exactly in $i\Delta x - U_n$ (see Fig. 2.10). Such interpolation has the form

$$A_i^n = \left(\chi A_{i-l}^{n-1} + (1-\chi)A_{i-l-1}^{n-1}\right) + \Delta t,$$
$$F_i^n = \chi F_{i-l}^{n-1} + (1-\chi)F_{i-l-1}^{n-1}.$$

which perfectly works when the two distributions have the same shape. There is, however, no suitable interpolation method when the distributions do not have the same shape, *i.e* when two neighbours computation nodes are characterised by distributions with a different number of components. In other words, neighbouring A and F vectors (see Eq. 2.23) should have the same length to perform the interpolation. In the implemented code, interpolation is done if the neighbour's distributions have the same shape. Otherwise, the random fix-grid scheme is used.

Interpolated adaptive-grid scheme The adaptive-grid scheme suffers from its inability to model the travel time through pipes accurately. For steady-states, this problem can be overcome by interpolating all the water outside the pipes instead of using an upstream approximation. The difference between the upstream approximation and the interpolated scheme is emphasised in Fig. 2.13. Again, the same problem as for the interpolated fix-grid scheme arises for the unsteady state. The problem is overcome in the same way: interpolation is done for distributions with the same shape, and an upstream approximation is used otherwise.

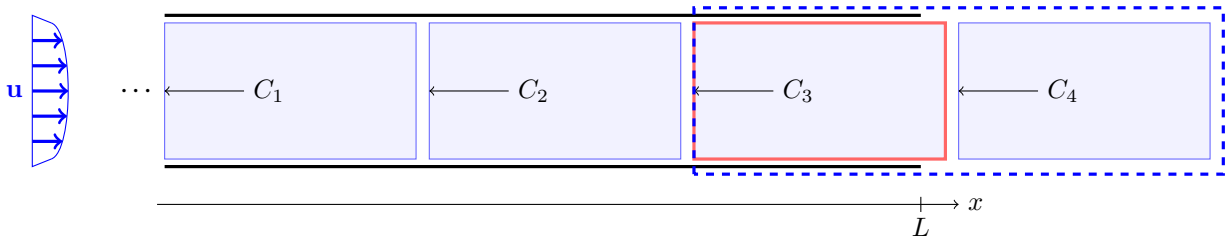


Figure 2.13: Modified lagrangian scheme. The blue box emphasises the volumes used for the interpolation.

2.5.5 Test cases and convergence

Tests are performed on the network from Jowitt et al. with a demand pattern shown in Appendix A applied to all nodes. First, the convergence of all numerical schemes is checked. Looking at Table 2.2, the fix-grid schemes should converge towards the exact solution when the spatial discretisation is refined. The number of discretisation points, N_p, in pipe p is

$$N_p = m\frac{L}{\min_{n\in[1,N_t]}(U_n)\Delta t},$$

where $m \in \mathbb{R}_0^+$ is called the multiplier. A reference solution is computed for a simulation time $T = 144$h, a time step $\Delta t = 1$h and $m = 160$ with the upstream scheme. The total difference

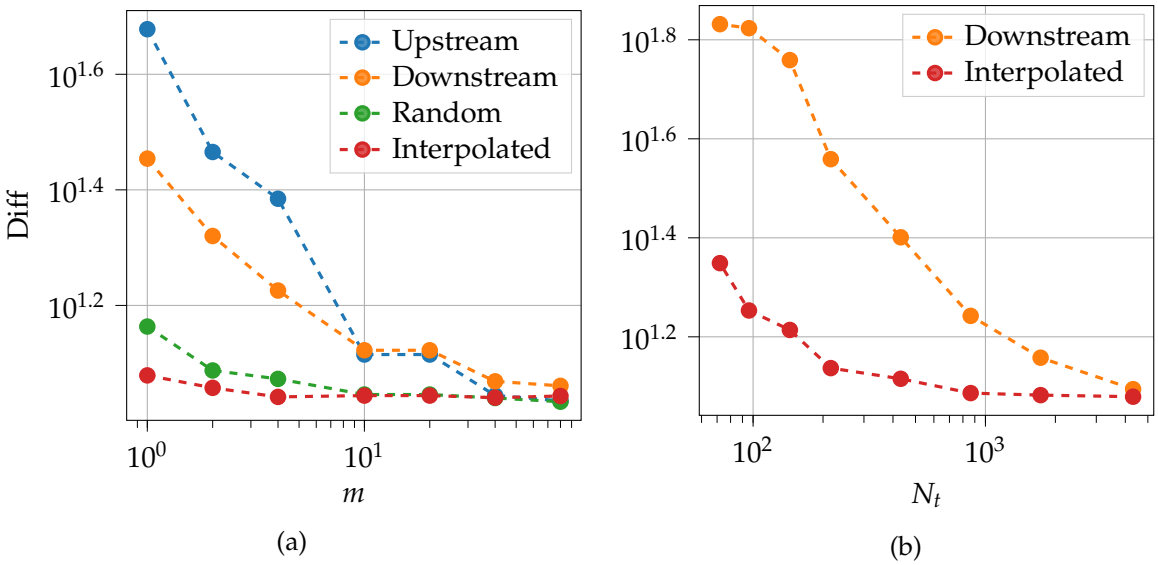


Figure 2.14: Convergence analysis for fix-grid scheme (a) and adaptive-grid scheme (b).

function (Eq. 2.20) is computed for different multipliers on the last time step. Results are shown in Fig. 2.14a. One can see that all methods converge towards the same results and that the modified schemes (random and interpolated schemes) give better results for coarse discretisations.

The convergence for the adaptive scheme is reached when the time stepping is sufficiently small. A reference solution is computed with $\Delta t = 1$min with the upstream approximation. Difference values are shown for various time steps in Fig. 2.14b. Once again, both methods give the same results for a refined time step, and the modified scheme (interpolated scheme) leads to improvement on rough time discretisation.

For illustration, let us first consider a quasi-static case on the same network. The discharges are constant but a time step Δt is introduced. At the time $t = 0$, the age distribution is $\delta(\tau = 0)$ everywhere in the network. The age distributions are aged at each time step and evolve according to the flows. Fig. 2.15a shows the evolution of the mean age at each node, computed with EPANET and the implemented code. Fig. 2.15c shows the evolution of the age distribution at node 10. The age distribution converges towards the static result of Fig. 2.6.

Figs. 2.15b and 2.15d show the results when using a variable demand (see Appendix A). Mean ages (Fig. 2.15b) and the age distribution at node 10 (Fig. 2.15d) do not reach a steady-state. However, a repetitve pattern is observed for $t \gtrsim 75$ hours which indicates convergence.

2.5.6 Conclusion

This Section has reimplemented and developed different numerical schemes to simulate time-dependent demands and thus time-dependent discharges. The different techniques have been shown to converge and to all give satisfactory results. Simple and robust scheme have

been first described and their key difference summarised in Table 2.2. Some modified scheme have been introduced to get slightly better results on rough discretisation.

These time-dependent schemes do not directly depend and how the age distribution is discretised. It has been coupled with the new method described in the previous Section, and this new method does not include any additional complication compare to other age distribution discretisation techniques.

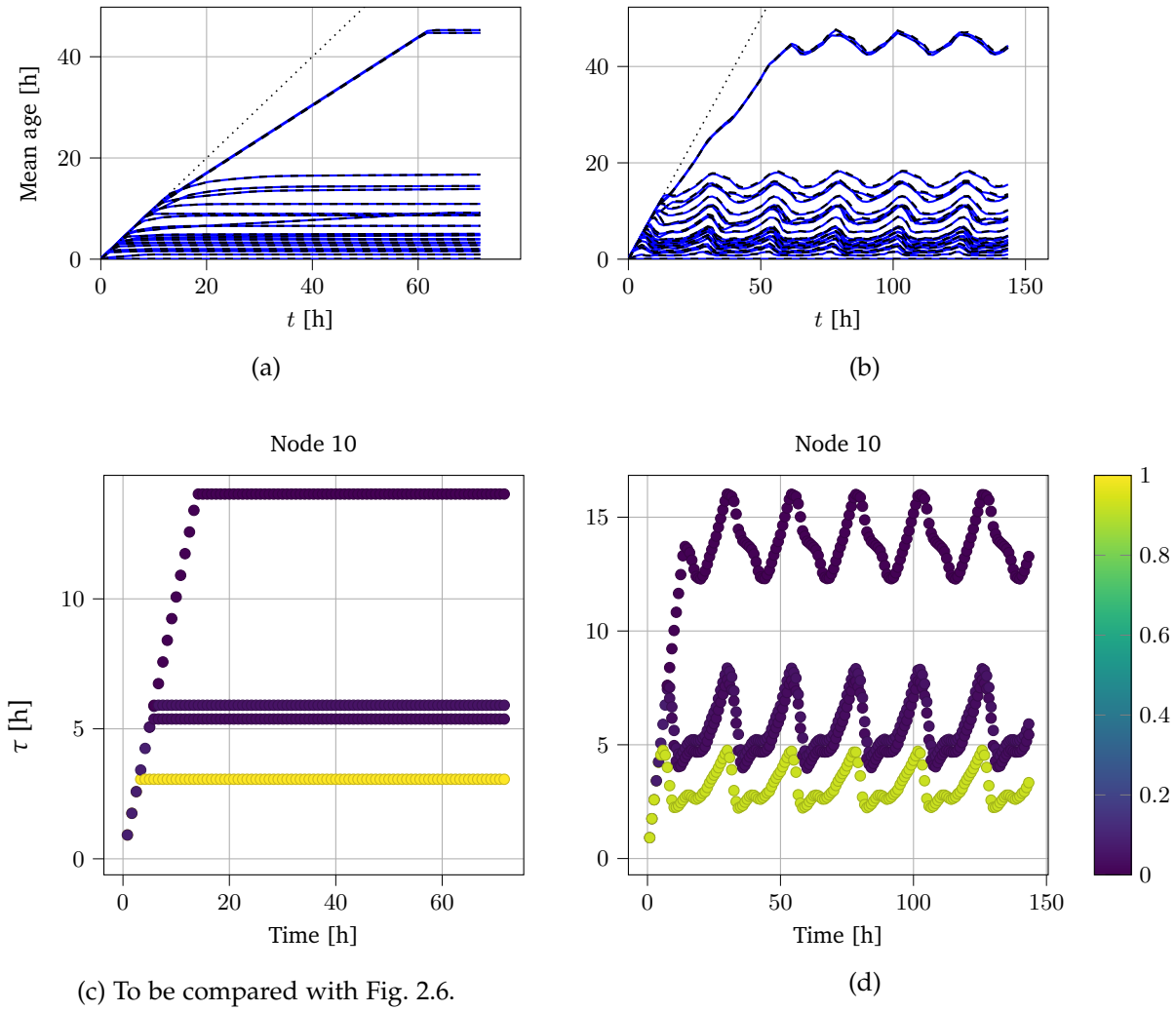


Figure 2.15: Figs. a and b show the mean age computed with the fix-grid interpolated method (solid blue lines) and returned by EPANET (dashed black lines) for the quasi-static case (a) and variable demand case (b) at each node of the network from Jowitt et al. Figs. c and d highlight the time dependence of age distribution at node 10 for both cases. The colours indicate the fraction of water in each branch.

— 3 —

Relaxing Assumptions

The previous chapter focuses on improving water age representation with common modelling assumptions. The goal of the following chapter is to investigate two underlying assumptions that are the plug flow assumption and the homogenous mixing. First, new models that do not make these assumptions are introduced. These models are then tested on some of the studied networks.

3.1 Dispersion

3.1.1 Introduction

Water quality monitoring in WDNs commonly uses the plug flow assumption ([24]). This means that dispersion effects are neglected. One can, however, be easily convinced that, because of the velocity profile and turbulent mixing, this assumption is not true and is only used because of its convenience for computation. Despite the impact that this hypothesis may have, only two studies on the subject have been found, both from Remero-Gomez et al.: [24], and [25]. These papers compared analytical, numerical, and experimental results for axial dispersion in a single pipe.

This Section aims at understanding the effects of dispersion in water quality computing. More precisely, this Section describes two techniques that have been implemented to model axial dispersion. The first one is based on an analytical result, whereas the second technique is based on the finite volume method. These methods will be explained first and then will be applied on a single pipe configuration and on the network 'Net1' (Fig. 1.2a) for illustrations.

The mathematical model of this Section restarts from Eq. 2.7, which is

$$\frac{\partial C}{\partial t} + \frac{\partial C}{\partial \tau} = -U\frac{\partial C}{\partial x} + \frac{\partial}{\partial x}\left(K\frac{\partial C}{\partial x}\right),$$

and assumes steady-state and the dispersion parameter K to be constant through the spatial dimension of the pipes. The governing equation thus takes the form of an advection-diffusion equation

$$\frac{\partial C}{\partial \tau} = -U\frac{\partial C}{\partial x} + K\frac{\partial^2 C}{\partial x^2}. \tag{3.1}$$

At the interior of the domain, no water has an age equal to zero since the age is defined as the time elapsed since the water particle has entered the domain ([8]), which gives the 'initial'

condition

$$C(x, \tau = 0) = 0. \tag{3.2}$$

The age at the upstream junction, C_0, is again computed thanks to perfect mixing from the upstream pipes. As it has been shown in Deleersnijder et al. ([6]), boundary conditions play a crucial role in water age computation. Here, the boundary conditions are chosen in accordance with Dewals et al. ([8]) and are illustrated in Fig. 3.1.

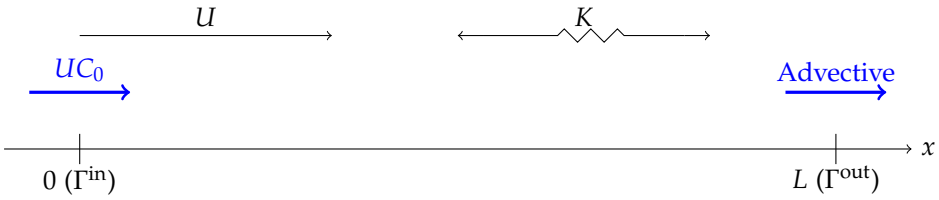


Figure 3.1: Schematic of the computational domain for a single-pipe and its boundary conditions. Figure adapted from [6].

The age of water particle entering the pipe through the departure boundary, Γ_{in}, is set to C_0. It results in a Robin boundary condition to set the incoming flux

$$\left[UC - K\frac{\partial C}{\partial x}\right]_{x\in\Gamma^{\text{in}}} = UC_0, \tag{3.3}$$

where the incoming flux is assumed purely advective. Note that not all water particles have an age equal to C_0 at the inlet because of the diffusion [6]. At the arrival boundary, Γ^{out}, flux is assumed to be purely advective

$$\left[K\frac{\partial C}{\partial x}\right]_{x\in\Gamma^{\text{out}}} = 0. \tag{3.4}$$

3.1.2 Modified Iterative Method

Hereafter is proposed a modified iterative method for dispersion (MIMD) (see the original method in Section 2.4). This method includes the effects of dispersion without introducing a spatial discretisation of the WDN. It is achieved by looking at the analytical solution for a semi-infinite pipe. First, the water entering the pipe is assumed to have a zero-age, which sets the inlet boundary condition. Such problem writes

$$\begin{cases} \dfrac{\partial C}{\partial \tau} = -U\dfrac{\partial C}{\partial x} + K\dfrac{\partial^2 C}{\partial x^2}, \\ C(x,0) = 0, \\ UC - K\dfrac{\partial C}{\partial x} = U\delta(\tau) \quad \text{at } x = 0, \end{cases} \tag{3.5}$$

which solution is (see [6])

$$C(x,\tau) = \frac{U}{\sqrt{\pi K\tau}} \exp\left[-\frac{(x-U\tau)^2}{4K\tau}\right] - \frac{U^2 e^{Ux/K}}{2K} \operatorname{erfc}\left[\sqrt{\frac{U^2\tau}{4K}} + \sqrt{\frac{x^2}{4K\tau}}\right], \tag{3.6}$$

where $\operatorname{erfc}(z)$ is the complementary error function

$$\operatorname{erfc}(z) = 1 - \operatorname{erf}(z), \text{ and } \quad \operatorname{erf} z = \frac{2}{\sqrt{\pi}} \int_0^z e^{-y^2} dy.$$

If the considered pipe is connected to a junction, with an age distribution C_0 rather than with zero-age water, the problem writes

$$\begin{cases} \dfrac{\partial C}{\partial \tau} = -U\dfrac{\partial C}{\partial x} + K\dfrac{\partial^2 C}{\partial x^2}, \\ C(x,0) = 0, \\ UC - K\dfrac{\partial C}{\partial x} = UC_0 \quad \text{at } x = 0. \end{cases} \tag{3.7}$$

The partial differential equation problems described in Eq. 3.5, and in Eq. 3.7 differ by their inlet boundary conditions. In the first case, the inlet boundary condition is described with a Dirac impulse, whereas it is defined with an arbitrary function C_0 in the second case. It is known, from the linear system theory, that the solution to the problem 3.7 can be rewritten thanks to the convolution product of the solution to problem 3.5 and the boundary condition C_0. Solution to problem 3.7 thus writes (see also [31])

$$C(x,\tau) = \int_0^\tau C_0(\tau-\tau') \left\{ \frac{U}{\sqrt{\pi K\tau'}} \exp\left[-\frac{(x-U\tau')^2}{4K\tau'}\right] - \frac{U^2 e^{Ux/K}}{2K} \operatorname{erfc}\left[\sqrt{\frac{U^2\tau'}{4K}} + \sqrt{\frac{x^2}{4K\tau'}}\right] \right\} d\tau'. \tag{3.8}$$

Strictly speaking, this solution is only valid for semi-infinite pipes. Though, it reveals to be a good approximation for flows of high Peclet numbers, as it will be shown in Section 3.1.4. An intuitive explanation can be found considering the arrival boundary condition described by Eq. 3.4. This boundary condition imposes the flux to be purely advective at the outlet. At high Peclet numbers, advection dominates, and the semi-infinite solution is a good approximation.

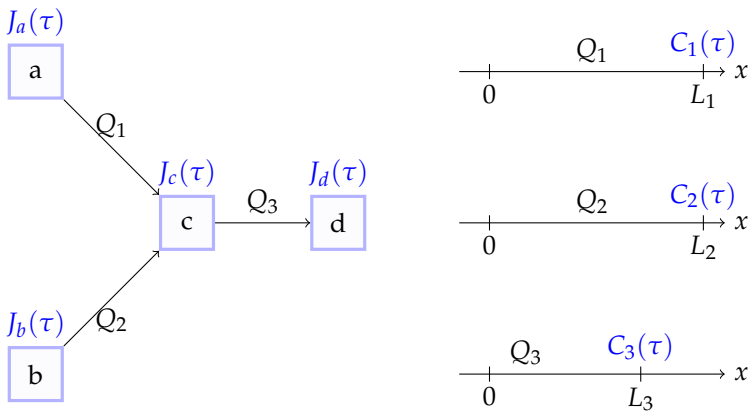


Figure 3.2: Schematic of a simple network to illustrate text notations. Squares indicate junctions, and arrows show flow directions.

In the following, the implemented algorithm to compute the age at the junctions in a WDN is described step-by-step. Numerically, the water age distribution is discretised in the age space with an age time step $\Delta\tau$

$$C^n(x) = C(x, n\Delta\tau), \quad \text{for } n = 1, \ldots, N_\tau.$$

Four steps described in the next are followed.

1. The age for the most downstream point of each pipe, *i.e* at $x = L_i$, is initialized thanks to Eq. 3.6

$$\begin{aligned} C_i^n = C(x = L_i, \tau = n\Delta\tau) = &\frac{U}{\sqrt{\pi K n\Delta\tau}} \exp\left[-\frac{(L_i - Un\Delta\tau)^2}{4Kn\Delta\tau}\right] \\ &- \frac{U^2 e^{UL_i/K}}{2K} \operatorname{erfc}\left[\sqrt{\frac{U^2 n\Delta\tau}{4K}} + \sqrt{\frac{L_i^2}{4Kn\Delta\tau}}\right], \end{aligned} \tag{3.9}$$

   where, for each pipe, the x-coordinate is the coordinate along the pipe in the streamwise direction. The x-coordinate is set to zero at the inlet of each pipe (see Fig. 3.2).

2. The age at each junction I, $J_I(\tau)$, is computed assuming perfect mixing (see Eq. 2.18).

$$J_I(n\Delta\tau) \approx J_I^n = \frac{1}{\sum_{i\in P^+} |Q_i|} \sum_{i\in P^+} |Q_i|\, C_i^n \tag{3.10}$$

3. At this stage, the age at the most downstream point of each pipe (C_i^n) will be updated. If the upstream junction of the pipe i is connected to a zero-age junction, Eq 3.9 from stage 1 is again used. Otherwise, the convolution product is computed in accordance with Eq. 3.8 thanks to the age at the upstream junction, J_{up} (Eq. 3.10).

For low Peclet values, the convolution product is directly computed. This can be done by trapezoidal integration

$$C_i^n = \sum_{l=0}^{l=n-1} \frac{\Delta\tau}{2} \left[J_{\text{up}}^{n-(l+1)} F_i((l+1)\Delta\tau) + J_{\text{up}}^{n-l} F_i(l\Delta\tau)\right]$$

$$\text{where } F_i(z) = \frac{U}{\sqrt{\pi K z}} \exp\left[-\frac{(L_i - Uz)^2}{4Kz}\right] - \frac{U^2 e^{UL_i/K}}{2K} \operatorname{erfc}\left[\sqrt{\frac{U^2 z}{4K}} + \sqrt{\frac{x^2}{4Kz}}\right].$$

The computation cost of this convolution is $\mathcal{O}(N_\tau^2)$ for each pipe. To speed up computation time, the code perfoms this computation using the fast Fourier transform method available in the package SciPy [13], which computational cost is $\mathcal{O}\left(N_\tau \log(N_\tau)\right)$.

For high Peclet values, an approximation is used. This is because the second exponential in Eq. 3.6 (*i.e* e^{Pe}) can be a higher number than the maximum representable float. This maximum will depend on the computer, the language used, and the details of the implementation. For this master thesis, this maximum was $1.79{\cdot}10^{308}$, which means that the maximum allowed Peclet is about 709. To go beyond this limit and be able to model very advective flow, an approximation of the second term of Eq 3.6 is used. The starting point is the approximation of the erfc$(\cdot)$ function. This approximation originates from [1]. A more accurate approximation described in [19] has been selected for this master thesis, which is

$$\begin{aligned}\operatorname{erfc} x &\approx \left(a_1 s + a_2 s^2 + a_3 s^3 + a_4 s^4 + a_5 s^5\right) e^{-x^2}, \quad s = \frac{1}{1+px},\\ \text{with } a_1 &= 0.254829592, \quad a_2 = -0.284496736, \quad a_3 = 1.42141741,\\ a_4 &= -1.453152027, \quad a_5 = 1.061405429, \text{ and } \quad p = 0.3275911.\end{aligned}$$

The second term of Eq. 3.8 can thus be approximated as

$$\begin{aligned}&\frac{U^2 e^{UL_i/K}}{2K} \operatorname{erfc}\left[\sqrt{\frac{U^2 t}{4K}} + \sqrt{\frac{L_i^2}{4Kt}}\right] \approx \frac{U^2}{2K}\left(a_1 s + a_2 s^2 + a_3 s^3 + a_4 s^4 + a_5 s^5\right) e^{\frac{UL_i}{K} - \left[\sqrt{\frac{U^2 t}{4K}} + \sqrt{\frac{L_i^2}{4Kt}}\right]^2}\\ &\text{with } s = \frac{1}{1 + p\left[\sqrt{\frac{U^2 t}{4K}} + \sqrt{\frac{L_i^2}{4Kt}}\right]},\end{aligned} \tag{3.11}$$

where the argument of the exponential has been reduced. This approximation may cause the age distribution not to be normalised as shown in Fig. 3.3. This would violate mass conservation, and thus the water age distribution is normalised to avoid this problem. This will introduce significant errors if the temporal domain of computation is too small compared to the real temporal domain of the age distribution. In other words, the fraction of water with age higher than the maximum representable age numerically (which is $N_\tau \Delta\tau$), should be as small as possible and convergence has to be checked.

4. Once the age at the most downstream point of each pipe has been updated, age at the junction can be recomputed according to step 2. Steps 2 and 3 must be iterated until convergence.

This numerical method based on an approximate analytical solution but with no spatial discretisation has been implemented and will be compared to others method in the last part of this Chapter.

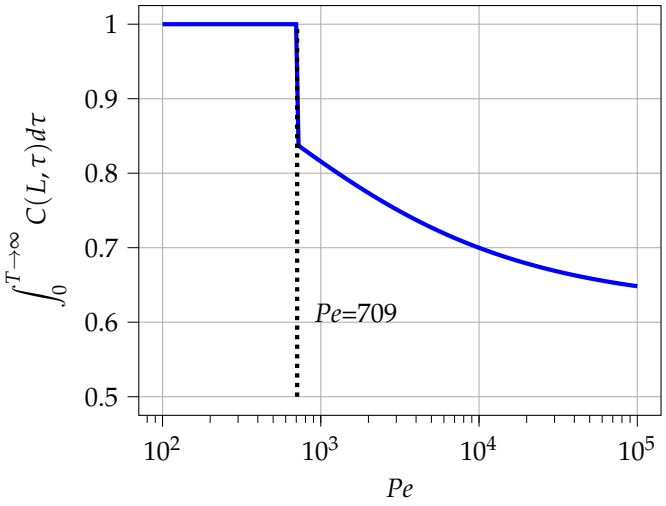


Figure 3.3: Numerical integral of Eq. 3.6. Unity is not reach for Pe>709, when approximation described in Eq. 3.11 is used. Here, the numerical parameters are: $L = 100$m, $U = 1$m/s, $N_\tau = 10^5$, $T = 10^4$s, $K = UL/\text{Pe}$. Numerical integration is done with the trapezoidal rule.

3.1.3 Finite Volume

As explained in the previous Section, the implemented method does not model correctly the output boundary condition. For the outlet boundary condition to be correctly represented, a finite volume scheme is described in this Section.

Constant Reconstruction

From Eq. 3.3, one has a Dirac impulse in the boundary condition if the age of water is assumed equal to zero at the pipe inlet. The Dirac impulse in the boundary condition may cause instabilities in the numerical simulation. To overcome this issue, Dewals et al. ([8]) propose to first compute the step response b, which means that the boundary condition for pipes connected to zero-age water are defined with the Heaviside function, $B_0 = H(\tau)$, and not a dirac impulse, $C_0 = \delta(\tau)$. If the pipe is not connected to an inlet of the network, the inlet boundary condition B_0 is computed thanks to the complete mixing assumption. For a given pipe of the network, the boundary condition is thus

$$B_0 = \begin{cases} H(\tau) & \text{if the upstream junction of the pipe is an entrance of the network,} \\ \dfrac{1}{\sum_{i\in P^+} |Q_i|} \displaystyle\sum_{i\in P^+} |Q_i|\, b_i(x = L_i, \tau) & \text{otherwise;} \end{cases}$$

where $b_i(x = L, \tau)$ is the step response at the downstream location of pipe i entering the studied pipe.

The impulse response can then be recovered from the step response derivative. Thus an equivalent formulation to Eqs. 3.1, 3.2, 3.4, and 3.3, but easier to solve numerically is

$$\begin{cases} \dfrac{\partial b}{\partial \tau} = -U\dfrac{\partial b}{\partial x} + K\dfrac{\partial^2 b}{\partial x^2}, \\ b(x,0) = 0, \\ Ub - K\dfrac{\partial b}{\partial x} = UB_0 \quad \text{at } x = 0, \\ Ub - K\dfrac{\partial b}{\partial x} = Ub \quad \text{at } x = L, \\ C = \dfrac{\partial b}{\partial \tau}. \end{cases}$$

Here is introduced a numerical scheme based on the finite volume method (FVM) with a spatial discretisation of the network. A cell-centred scheme is used. The mesh is taken uniform in space, as sketched in Fig. 3.4 for one pipe.

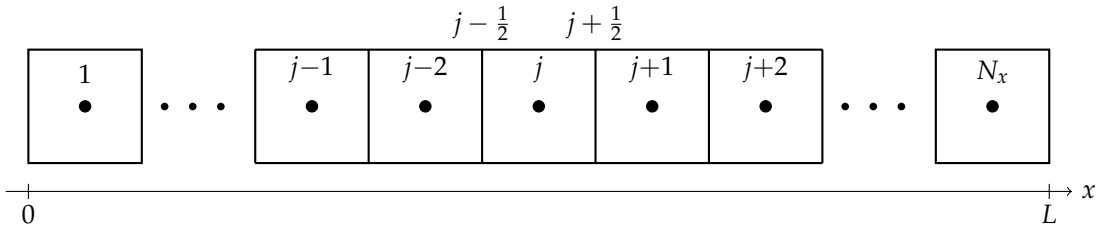


Figure 3.4: 1D cell-centred finite volume mesh with N_x uniform cells. Illustration of text notations.

The principle of a finite volume discretisation is first to integrate the partial differential equation over one cell

$$\int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \frac{\partial b}{\partial \tau} dx + \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \frac{\partial}{\partial x}\left(Ub - K\frac{\partial b}{\partial x}\right) dx = 0.$$

The first term, including the time derivative, is transformed based on the mean value theorem, while the next terms are straightforward to integrate

$$\frac{d\bar{b}_j}{d\tau}\Delta x + \left[Ub - K\frac{\partial b}{\partial x}\right]_{j-\frac{1}{2}}^{j+\frac{1}{2}} = 0,$$

where $\bar{b}_j$ refers to the average of b over the cell j and $\Delta x = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}}$. Hence, considering that so far U and K are assumed constant and uniform,

$$\frac{d\bar{b}_j}{d\tau} = -U\frac{[b]_{j+\frac{1}{2}} - [b]_{j-\frac{1}{2}}}{\Delta x} + K\frac{\left[\frac{\partial b}{\partial x}\right]_{j+\frac{1}{2}} - \left[\frac{\partial b}{\partial x}\right]_{j-\frac{1}{2}}}{\Delta x}. \tag{3.12}$$

The only remaining step to come up with a finite volume discretisation is to specify how the fluxes are evaluated at the cell edges $j-\frac{1}{2}$ and $j+\frac{1}{2}$. For the gradient of b in the diffusive flux, a centred finite difference is used

$$\left[\frac{\partial b}{\partial x}\right]_{j-\frac{1}{2}} \approx \frac{\bar{b}_j - \bar{b}_{j-1}}{\Delta x}, \quad \text{and} \quad \left[\frac{\partial b}{\partial x}\right]_{j+\frac{1}{2}} \approx \frac{\bar{b}_{j+1} - \bar{b}_j}{\Delta x}.$$

For the variable b in the advective flux, an upwind approximation is used (constant reconstruction, assuming $U \geq 0$)

$$[b]_{j-\frac{1}{2}} \approx \bar{b}_{j-1}, \quad \text{and} \quad [b]_{j+\frac{1}{2}} \approx \bar{b}_j. \tag{3.13}$$

By applying the specified boundary conditions at the inlet and outlet, the scheme writes as follows for the first and last cells

- for $j = 1$:

$$\frac{d\bar{b}_1}{d\tau} = -\frac{U\bar{b}_1 - UB_0}{\Delta x} + K\frac{\left[\frac{\partial b}{\partial x}\right]_{\frac{3}{2}}}{\Delta x} \approx -\frac{U\bar{b}_1 - UB_0}{\Delta x} + K\frac{\bar{b}_2 - \bar{b}_1}{\Delta x^2},$$

- for $j = N_x$:

$$\frac{d\bar{b}_{N_x}}{d\tau} = -\frac{U\bar{b}_{N_x} - U\bar{b}_{N_x-1}}{\Delta x} + K\frac{-\left[\frac{\partial b}{\partial x}\right]_{N_x-\frac{1}{2}}}{\Delta x} \approx -\frac{U\bar{b}_N - U\bar{b}_{N_x-1}}{\Delta x} + K\frac{\bar{b}_{N_x-1} - \bar{b}_{N_x}}{\Delta x^2}.$$

The time-stepping has been done thanks to an explicit Euler method. The water age is discretised with a time step $\Delta\tau$. Let us rewrite Eq. 3.12 at time $n\Delta\tau$

$$\frac{d\bar{b}_j^n}{d\tau} = -U\frac{[b]_{j+\frac{1}{2}}^n - [b]_{j-\frac{1}{2}}^n}{\Delta x} + K\frac{\left[\frac{\partial b}{\partial x}\right]_{j+\frac{1}{2}}^n - \left[\frac{\partial b}{\partial x}\right]_{j-\frac{1}{2}}^n}{\Delta x},$$

where the upper indices indicate the time step. The step response has thus to be updated according to

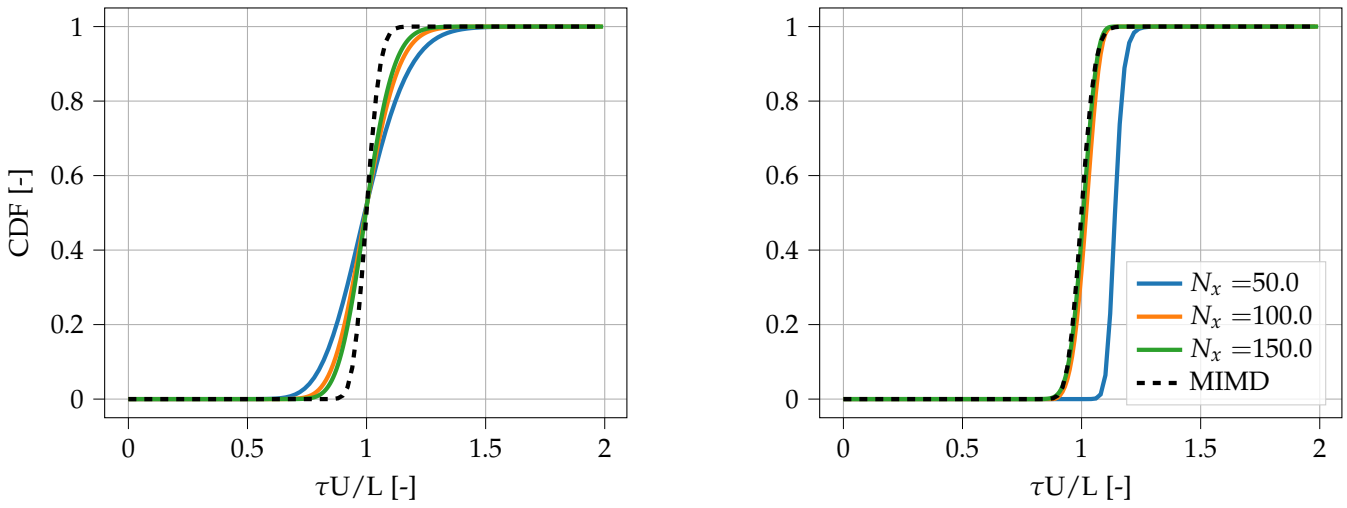
$$\bar{b}_j^{n+1} = \bar{b}_j^n + \Delta\tau \left\{ -U\frac{[b]_{j+\frac{1}{2}}^n - [b]_{j-\frac{1}{2}}^n}{\Delta x} + K\frac{\left[\frac{\partial b}{\partial x}\right]_{j+\frac{1}{2}}^n - \left[\frac{\partial b}{\partial x}\right]_{j-\frac{1}{2}}^n}{\Delta x} \right\}.$$

Grouping all terms, the constant reconstruction finite volume scheme writes (assuming $U \geq 0$)

$$\begin{aligned}
&\text{For } 1 < j < N_x: \ \bar{b}_j^{n+1} = \bar{b}_j^n + \Delta\tau \left[-U \frac{[b]_{j+\frac{1}{2}}^n - [b]_{j-\frac{1}{2}}^n}{\Delta x} + K \frac{\left[\frac{\partial b}{\partial x}\right]_{j+\frac{1}{2}}^n - \left[\frac{\partial b}{\partial x}\right]_{j-\frac{1}{2}}^n}{\Delta x} \right] \\
&\qquad \text{with } [b]_{j-\frac{1}{2}}^n = \bar{b}_{j-1}^n, \quad \text{and } [b]_{j+\frac{1}{2}}^n = \bar{b}_j^n \\
&\qquad \left[\frac{\partial b}{\partial x}\right]_{j-\frac{1}{2}}^n = \frac{\bar{b}_j^n - \bar{b}_{j-1}^n}{\Delta x}, \quad \text{and } \left[\frac{\partial b}{\partial x}\right]_{j+\frac{1}{2}}^n = \frac{\bar{b}_{j+1}^n - \bar{b}_j^n}{\Delta x} \\
&\text{For } j = 1: \bar{b}_1^{n+1} = \bar{b}_1^n + \Delta\tau \left[-\frac{U\bar{b}_1^n - UB_0^n}{\Delta x} + K \frac{\bar{b}_2^n - \bar{b}_1^n}{\Delta x^2} \right] \\
&\text{For } j = N_x: \bar{b}_{N_x}^{n+1} = \bar{b}_{N_x}^n + \Delta\tau \left[-\frac{U\bar{b}_{N_x}^n - U\bar{b}_{N_x-1}^n}{\Delta x} + K \frac{\bar{b}_{N_x-1}^n - \bar{b}_{N_x}^n}{\Delta x^2} \right].
\end{aligned} \tag{3.14}$$

Linear reconstruction

The previous numerical scheme, based on constant reconstruction (see Eq. 3.13), is simple and robust. However, it suffers from a major drawback that is numerical diffusion. It means that the solution tends to smear out because of the numerical scheme. The consequence is that dispersion can be seen in the solution even if $K \to 0$, as illustrated in Fig. 3.5a.



(a) Constant reconstruction.

(b) Linear reconstruction.

Figure 3.5: Comparaison of the constant (a) and linear (b) reconstruction scheme for high Peclet numbers on a single pipe. Numerical parameters are $K = 0.1\text{m}^2/\text{s}$, $L = 100\text{m}$, U=1m/s (Pe=10^3), $\alpha = 0.01$ (CFL condition, see later Eq. 3.18).

This numerical diffusion can be shown rigorously for the constant reconstruction scheme. Taking the constant reconstruction numerical scheme with $K = 0$, and for $1 < j < N_x$

$$\bar{b}_j^{n+1} = \bar{b}_j^n + \Delta\tau \left\{ -U \frac{\bar{b}_j^n - \bar{b}_{j-1}^n}{\Delta x} \right\},$$

and the Taylor expansions

$$\bar{b}_j^{n+1} = \bar{b}_j^n + \Delta t \left(\frac{\partial \bar{b}}{\partial t}\right)_j^n + \frac{(\Delta t)^2}{2} \left(\frac{\partial^2 \bar{b}}{\partial t^2}\right)_j^n + \frac{(\Delta t)^3}{6} \left(\frac{\partial^3 \bar{b}}{\partial t^3}\right)_j^n + \dots,$$
$$\bar{b}_{j-1}^n = \bar{b}_j^n - \Delta x \left(\frac{\partial \bar{b}}{\partial x}\right)_j^n + \frac{(\Delta x)^2}{2} \left(\frac{\partial^2 \bar{b}}{\partial x^2}\right)_j^n - \frac{(\Delta x)^3}{6} \left(\frac{\partial^3 \bar{b}}{\partial x^3}\right)_j^n + \dots,$$

one can show (see for example [17])

$$\frac{\partial \bar{b}}{\partial t} + U \frac{\partial \bar{b}}{\partial x} = \underbrace{\frac{U\Delta x}{2}(1-\phi)\frac{\partial^2 \bar{b}}{\partial x^2}}_{\text{numerical diffusion}} + \dots,$$

where $\phi = U\frac{\Delta\tau}{\Delta x}$ is the Courant–Friedrichs–Lewy (CFL) number for advection. This numerical diffusion term is not physical and renders the analysis of dispersion effects meaningless for a small value of the diffusion parameter K. Therefore, a higher-order reconstruction method is introduced to replace the constant reconstruction approximation (Eq. 3.13). A linear reconstruction can be written as

$$[b]_{j+\frac{1}{2}} \approx \bar{b}_j + \frac{\Delta x}{2}\sigma_j, \quad \text{and} \quad [b]_{j-\frac{1}{2}} \approx \bar{b}_j - \frac{\Delta x}{2}\sigma_j,$$

where the slope σ_j could be naively computed as

$$\sigma_j = \frac{\bar{b}_{j+1} - \bar{b}_{j-1}}{2\Delta x}. \tag{3.15}$$

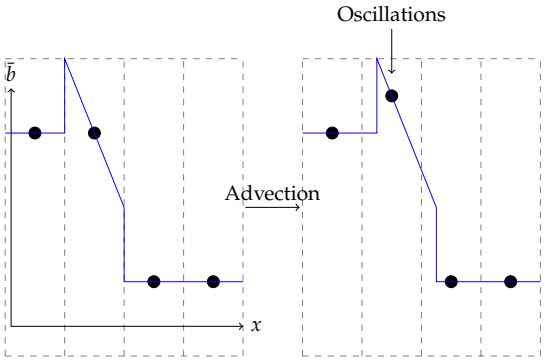
However, when using a linear reconstruction method, one has to introduce a slope-limiter to avoid numerical dispersions, *i.e* oscillations in the solution originating from the numerical scheme. The origin of the oscillations is shown in Fig.3.6. By limiting the slope, the slope-limiter ensures to avoid overshoot. Several slopes limiters exist, see for example [18]. Here a monotoned central-difference limiter (MC limiter) is used. The slope from Eq. 3.15 has to be replaced by

$$\sigma_j^n = \text{minmod}\left(\left(\frac{\bar{b}_{j+1}^n - \bar{b}_{j-1}^n}{2\Delta x}\right), 2\left(\frac{\bar{b}_j^n - \bar{b}_{j-1}^n}{\Delta x}\right), 2\left(\frac{\bar{b}_{j+1}^n - \bar{b}_j^n}{\Delta x}\right)\right), \tag{3.16}$$

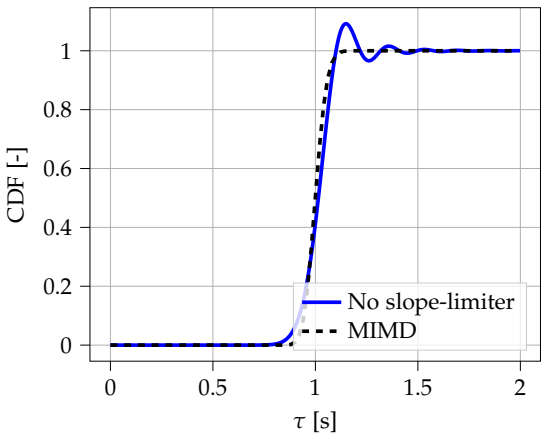
where the minmod function is

$$\text{minmod}(a,b,c) = \begin{cases} a & \text{if } |a| < |b| \text{ and } |a| < |c|, \\ b & \text{if } |b| < |a| \text{ and } |b| < |c|, \text{ and} \\ c & \text{if } |c| < |a| \text{ and } |c| < |b|. \end{cases}$$

The limited slope is depicted in Fig. 3.7. Thanks to this linear reconstruction, one can mitigate the numerical diffusion that may make any quality analysis wrong for low dispersion factor K, as illustrated in Fig. 3.5b.



(a) Illustration of the origin of the oscillations. If the slope estimation (Eq. 3.15) is not limited at each time step, overshoot may arise. Overshoot will be responsible for oscillations. Blue lines show the slope estimations, and the black dots are nodal values. Two successive time steps are shown here. Inspired from [28].



(b) Oscilations in the solution because of the absence of slope-limiter. Numerical parameters are the same as in Fig. 3.5.

Figure 3.6: Numerical oscillations in linear reconstruction finite volume scheme.

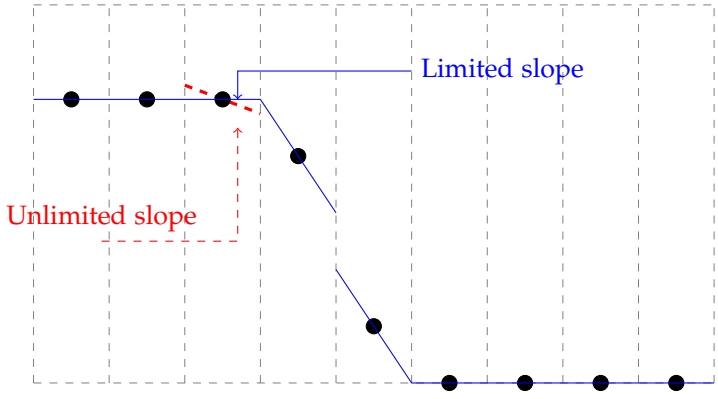


Figure 3.7: Example of limited slope computed with Eq. 3.16 to avoid oscillations.

Grouping all the terms together give for the finite volume scheme with linear reconstruction

$$
\begin{aligned}
&\text{For } 1 < j < N_x: \\
&\bar{b}_j^{n+1} = \bar{b}_j^n + \Delta\tau \left[-U \frac{[b]^n_{j+\frac{1}{2}} - [b]^n_{j-\frac{1}{2}}}{\Delta x} + K \frac{\left[\frac{\partial b}{\partial x}\right]^n_{j+\frac{1}{2}} - \left[\frac{\partial b}{\partial x}\right]^n_{j-\frac{1}{2}}}{\Delta x} \right] \\
&\text{with } [b]^n_{j+\frac{1}{2}} = \bar{b}_j^n + \frac{\Delta x}{2}\sigma_j^n, \quad \text{and } [b]^n_{j-\frac{1}{2}} = \bar{b}_j^n - \frac{\Delta x}{2}\sigma_j^n \\
&\sigma_j^n = \text{minmod}\left(\left(\frac{\bar{b}^n_{j+1} - \bar{b}^n_{j-1}}{2\Delta x} \right), 2\left(\frac{\bar{b}^n_j - \bar{b}^n_{j-1}}{\Delta x} \right), 2\left(\frac{\bar{b}^n_{j+1} - \bar{b}^n_j}{\Delta x} \right) \right) \\
&\left[\frac{\partial b}{\partial x}\right]^n_{j-\frac{1}{2}} = \frac{\bar{b}^n_j - \bar{b}^n_{j-1}}{\Delta x}, \quad \text{and} \quad \left[\frac{\partial b}{\partial x}\right]^n_{j+\frac{1}{2}} = \frac{\bar{b}^n_{j+1} - \bar{b}^n_j}{\Delta x} \\
&\text{For } j = 1: \bar{b}_1^{n+1} = \bar{b}_1^n + \Delta\tau \left[-\frac{U\bar{b}_1^n - UB_0^n}{\Delta x} + K\frac{\bar{b}_2^n - \bar{b}_1^n}{\Delta x^2} \right] \\
&\text{For } j = N_x: \bar{b}_{N_x}^{n+1} = \bar{b}_{N_x}^n + \Delta\tau \left[-\frac{U\bar{b}_{N_x}^n - U\bar{b}_{N_x-1}^n}{\Delta x} + K\frac{\bar{b}_{N_x-1}^n - \bar{b}_{N_x}^n}{\Delta x^2} \right].
\end{aligned}
\tag{3.17}
$$

CFL condition

A necessary condition to be stable for the two finite volume schemes is the so-called Courant-Friedrichs-Lewy (CFL) condition. The CFL condition states that the numerical speed should always be higher than the physical speed. In other words, at each age time step $\Delta\tau$, the travel distance (through advection or diffusion) should be smaller than the space grid (the range of influence of the scheme at the boundary is only one Δx)

$$\Delta x > U\Delta\tau \quad \text{, and} \quad \Delta x > \sqrt{2K\Delta\tau},$$

which can be rewritten as

$$\Delta\tau < \min\left(\frac{\Delta x}{U}, \frac{\Delta x^2}{2K}\right).$$

In this master thesis, the number of computational nodes N_x is taken constant in each pipe of the WDN. This choice has been made to ensure that all pipes are discretised with at least one node and to avoid unnecessary refined discretisation in long pipes since the pipe length can be very different from one pipe to another inside the same WDN. Once the number of computational points is fixed, and thus the spatial step in each pipe, the age temporal discretisation is taken to satisfy the CFL condition

$$\Delta\tau = \alpha \min_{\text{all pipes}}\left(\min\left(\frac{\Delta x}{U}, \frac{\Delta x^2}{2K}\right)\right), \tag{3.18}$$

where $0 < \alpha < 1$.

3.1.4 Single Pipe

In this Section, the numerical methods are illustrated on a simple single-pipe configuration. The pipe has a length $L = 100$m, and the water flows at 1m/s. From the computed water age distribution, the mean age can be computed by integration

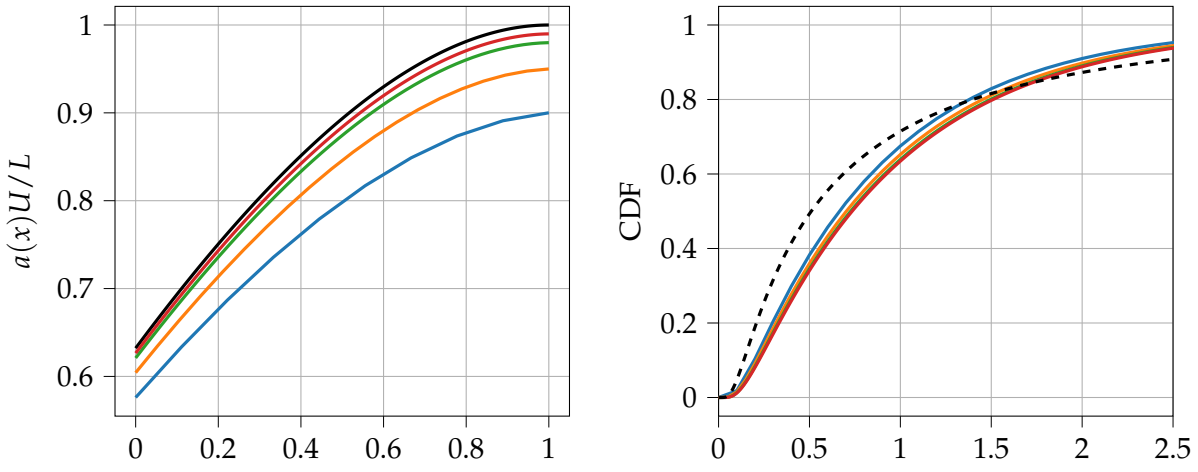
$$a(x) = \int_0^{+\infty} \tau C(\tau, x) d\tau.$$

Trapezoidal integration is used to numerically determined this integral. This numerical solution will be compared to results obtained by Dewals et al. For the very same problem (Eqs. 3.1, 3.2, 3.4, and 3.3), they established the water mean age profile through the pipe analyticaly

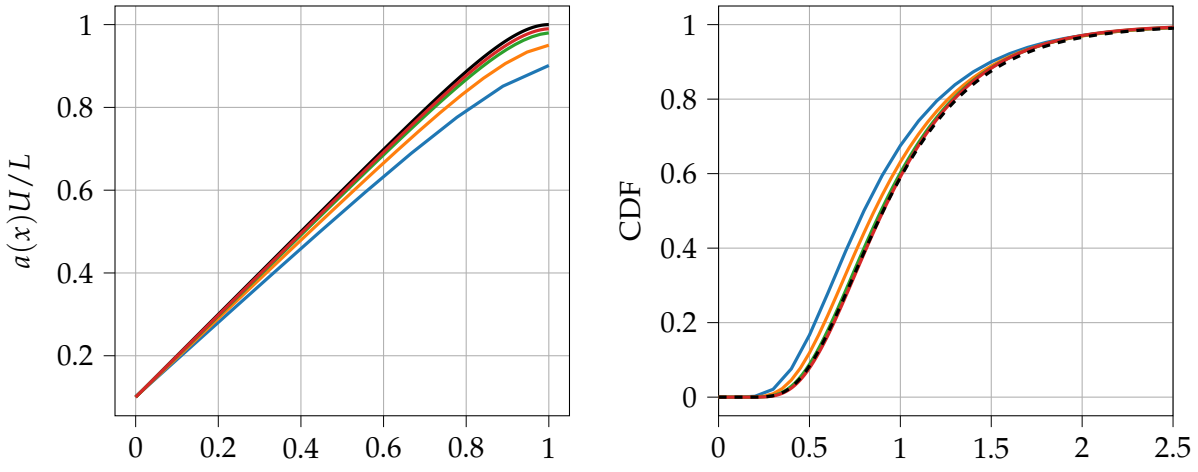
$$a(x) = \frac{L}{U}\left(\frac{x}{L} + \frac{1 - e^{-Pe(1-x/L)}}{Pe}\right). \tag{3.19}$$

This analytical result is compared to the numerical results from the finite volume method in the left column of Fig. 3.8 for various Peclet numbers. One can see the numerical results tend toward the analytical expression as the spatial grid is refined. The right column of Fig. 3.8 shows the CDFs at the outlet of the pipe. The numerical results obtained with the FVM are compared to the one obtained with the modified iterative scheme for dispersion (MIMD). For a single-pipe configuration with zero-age prescribed at the inlet, the MIMD reduces to Eq. 3.9. At high Peclet numbers ($Pe \gtrsim 10$), both methods agree, as predicted in the previous sections. The impact of the boundary condition is visible at lower Peclet values ($Pe \lesssim 10$) where assuming a semi-infinite pipe at the outlet is very different from our modelling assumption with a purely advective flow at the outlet (Eq. 3.4).

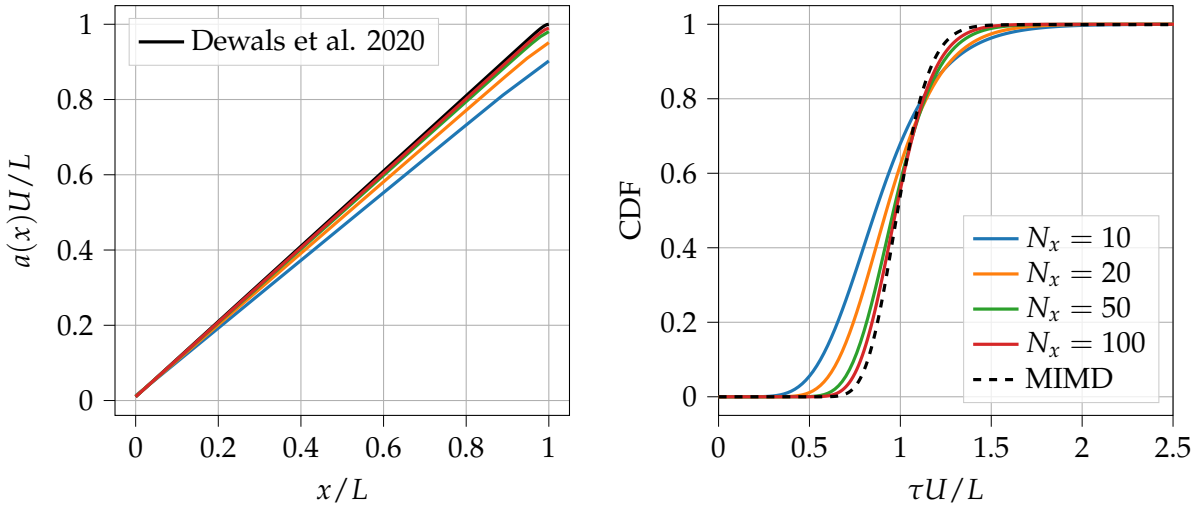
A simple experiment is possible to show that the mismatch between the two methods at low Peclet is due to the boundary condition. First, the FVM is applied on a longer pipe of length sL with $s > 1$, which allows reducing the effect of the boundary condition at the outlet. Then the result of the FVM in $x = L$ is compared to the one of the MIMD. This is shown in Fig. 3.9 and remarkable agreement can be seen between the two methods for $s \gtrsim 2$.



(a) Pe=1

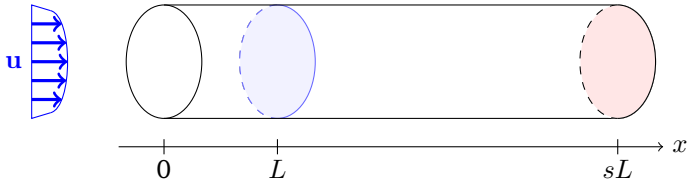


(b) Pe=10

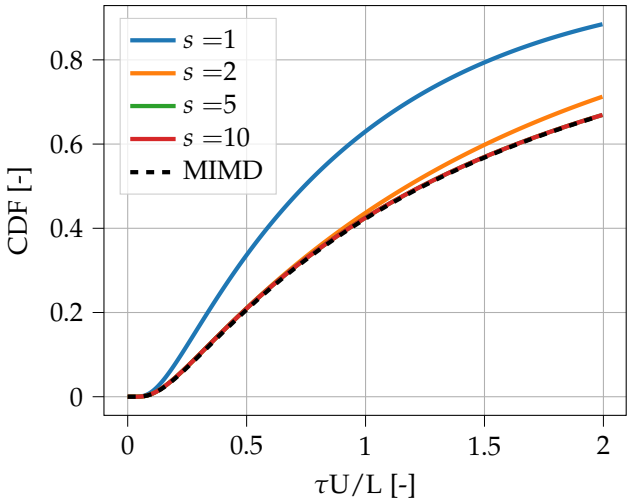


(c) Pe=100

Figure 3.8: Left: mean age profile computed analyticaly (Eq. 3.19) and numerically. Right: CDFs from Eq. 3.9 (MIMD), and the FVM. N_τ sets to satisfy the CFL condition with $\alpha = 0.05$ (Eq. 3.18).



(a) The red cross-section shows where the outlet boundary condition is applied. The blue cross-section shows where the FVM solution is retrieved. The FVM computational domain goes from 0 to sL.



(b)

Figure 3.9: FVM and MIMD at $Pe = 1$ for different extension of the FV computational domain.

3.1.5 Net1

The developed numerical methods have been tested on the network Net1 for various dispersion coefficients. Results are shown in Fig. 3.10 for node 23, for different mean Peclect value, $\overline{Pe}$, where the mean is taken over all pipes. For the highest Peclet value (≈ 15), the MIMD has been used, which does not require a spatial discretisation of the pipes. The result is similar to the one obtained when the dispersion was neglected (see Section 2.4). The finite volume scheme with linear reconstruction has been used for lower Pecelet values to ensure to satisfy the outlet boundary condition. The solutions behave as expected, with the distributions tending to smear out when the Peclet number decreases.

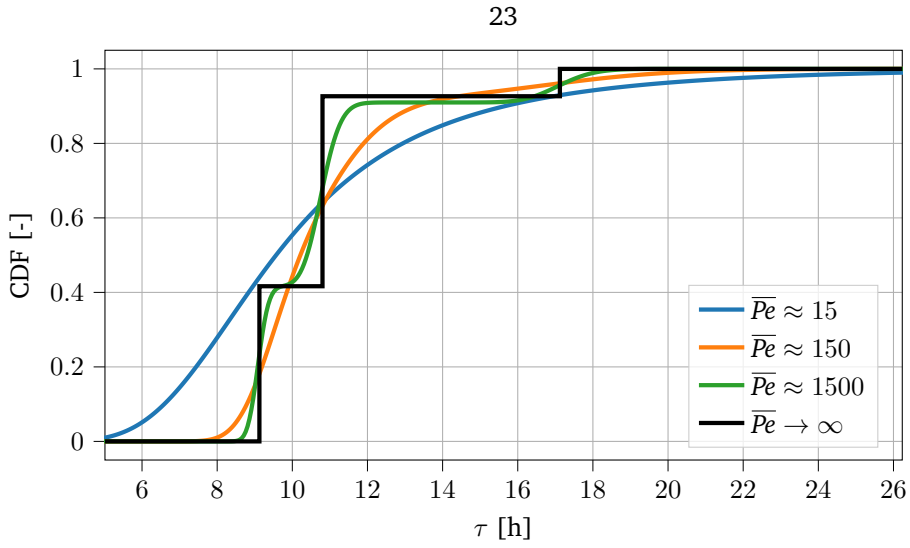


Figure 3.10: Dispersive effects at node 23 of network Net1. See text for the numerical methods used. Numerical parameters: $N_{\tau,\text{MIMD}} = 10^3$, $N_{x,\text{FVM}} = 20$, $\alpha = 0.01$, $T = N_\tau \Delta\tau = 50$h.

3.1.6 Conclusion and outlook

Computation of dispersion effects has been achieved on a single-pipe configuration and a small WDN thanks to two different techniques. The first one is based on an approximate analytical solution valid at high Peclet numbers. The underlying numerical method is a numerical integration of a convolution product. The great advantage of this method is that it does not require a spatial discretisation of the WDN. A finite volume scheme has been introduced to correctly represent the outlet boundary condition, taking care of distinguishing numerical diffusion and physical dispersion thanks to a linear reconstruction method. A finite volume method requires a complete discretisation of the WDN, which has been done.

For computational efficiency, it has been chosen to take the number of computational nodes constant in each pipe. This choice may not be optimal. It suffers from a major drawback that is: if one considers two identical flows (same velocity U, and diffusion parameter K) through two pipes having the same diameter but a different length, the two pipes will have a different space-grid whereas they model the exact same physic. Other approaches are possible and could be investigated in future work. Some possibilities are a spatial grid proportional to the pipe diameter or to first fix the age time step $\Delta\tau$ and then to take the spatial discretisation to be $\Delta x = \frac{1}{\alpha}\max(U\Delta\tau, \sqrt{2K\Delta\tau})$.

Unfortunately, the computational cost of these two methods makes, for now, computation of dispersion effects impossible for larger networks with a 'simple' laptop. At high Peclet numbers, convergence is unlikely to be reached on a traditional laptop with the FVM because of the CFL condition (Eq. 3.18) that requires the time step to scale as the square of the grid spacing. Relevant results may be found with the MIMD for this case. Conversely, for low Peclet numbers, the solution from the MIMD is invalid because it violates the outlet boundary condition. The optimal choice to get the most accurate possible solution will thus depend on the Peclet number of each pipe of the WDN. One solution to overcome this problem could be to combine different schemes. One could imagine each pipe of the network as a 'black box'. This black box would have one input that is the age at the upstream junction. This black box would then output the age at the outlet of the pipe, selecting the numerical scheme depending on the Peclet number. This may be a solution to achieve the computation of dispersion effects on traditional computers for larger networks.

3.2 Non-homogeneous Mixing

3.2.1 Introduction

Up to now, all models assume perfect mixing. Water entering the junctions is instantaneously mixed, and the water age is the same in any downstream pipes of the junction. It is, however, not true as it has been shown by several authors, and the results may be very different when non-homogeneous mixing is considered. The short residence time inside junctions, and the limited contact between incoming flows that it induces, renders complete mixing unrealistic, as demonstrated in the work of Austin et al. ([3]). In fact, if one considers cross junctions such as in Fig. 3.11 b, 'non-homogeneous mixing results from bifurcating inlet flows that reflect off of one another with minimal contact time', as it can be seen from the CFD simulations and experimental works of Romero et al. 2008 ([23]). The mixing assumption at the junctions has been shown to play a significant role in water quality monitoring. As an example, Romero et al. 2011 ([26]) show the impact of non-homogeneous mixing on sensor network design. The study shows different optimal sensor locations in the network depending on the mixing assumption for a network similar the network 'Net1' considered in this master thesis. Song et al. ([30]) reduces the solute concentration average prediction error by nearly 50% while going from complete mixing modelling towards more realistic mixing models. The studied networks were composed of only nine cross junctions.

Mixing at the junction is not trivial and depends, at least, on the type of junction, the junction geometry, the flows configuration, and the hydraulic variables. The complications arise from secondary currents, flow instabilities at the interface [29], and the great diversity of existing junction types and flow configurations. Fig. 3.11 shows some of the mixing configurations studied in literature. Unfortunately, these studied configurations do not necessarily represent all existing junctions (consider, for example, nodes 5, 6, 8, 12, 15, and 22 of the network of Jowitt et al. (Fig. 1.2c)).

This Section aims at understanding the impact of the mixing assumption for water wage computations. The focus is set on cross junctions as it is the most studied junction type in literature. First, three mixing models for single cross junctions are reviewed. Then, some of the mixing models are implemented to analyse the influence of the mixing assumption for the network Test rig. Eventually, application to a full-scale network is discussed with the network from Jockgrim.

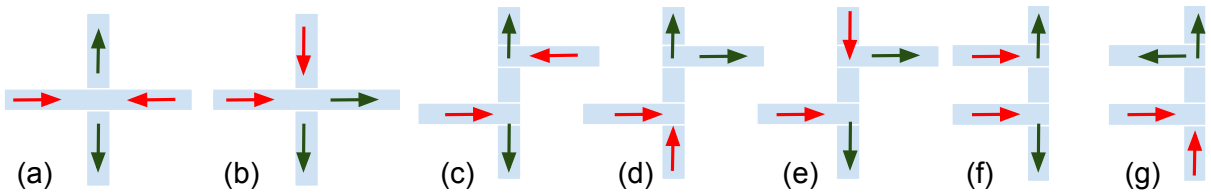


Figure 3.11: Different junction types and flow configurations. (a) and (b) are cross junctions, (c) to (e) are double-T junctions, and (f) and (g) are U junctions. Arrows indicate flow direction.

3.2.2 Mixing models

When the flow entering the junction comes from a unique pipe (Fig. 3.12a), mixing does not play a role. When the outcoming flow leaves a junction by a single pipe (Fig. 3.12b), mixing at the cross junction may be assumed perfect [29]. Complete mixing is also a good approximation for junctions where the incoming flows are facing each other (Fig. 3.12c)[29]. The relevant configuration to study is thus the case where two incoming flows enter the junction from adjacent legs (Fig. 3.12d). Pipe numbering is introduced here to remain consistent throughout the entire chapter. The incoming pipes are numbered 1 and 2. The opposite of pipe 1 (resp. 2) is labelled 3 (resp. 4).

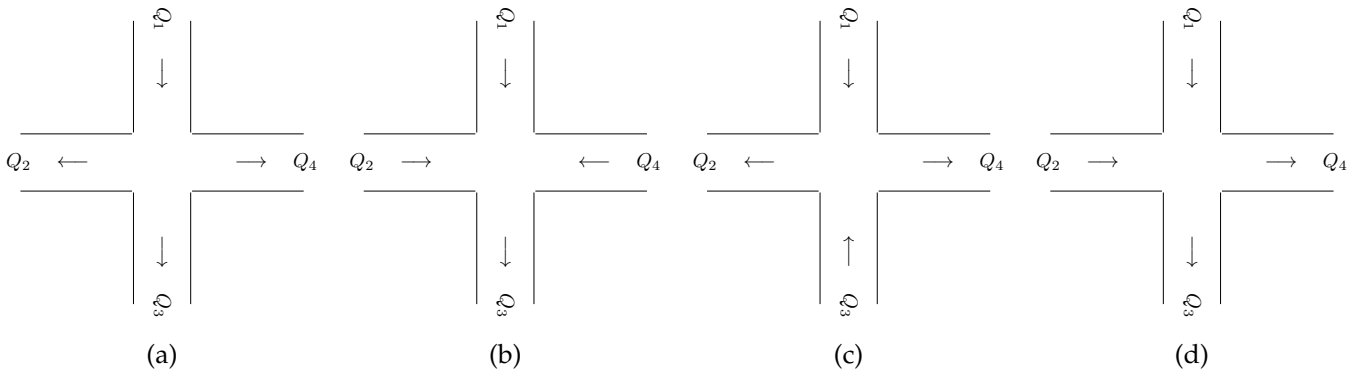


Figure 3.12: Possible flow configurations in a cross junction.

Azred

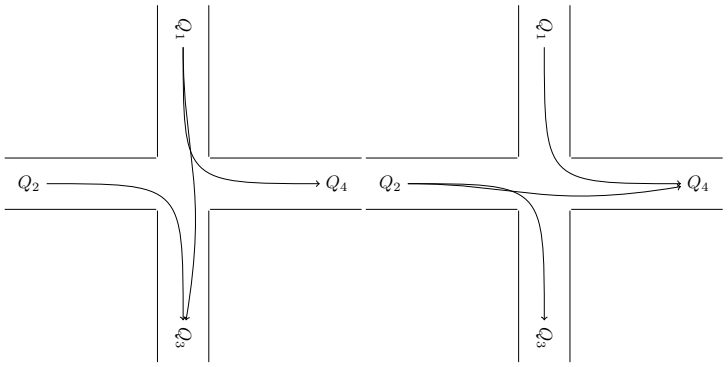
A plugin to Epanet, called Azred and developed at the University of Wisconsin-Madison by Choi et al. ([5]), can model incomplete-mixing for large scale WDNs. It is thus possible to compute the mean age at the junctions considering non-homogeneous mixing. The authors have studies cross, double-T (Figs. 3.11 c, d, e), U (Figs. 3.11 f, g), and Y (double-T with an angle different from 90°) junctions. According to the authors, the main parameters influencing the level of mixing in the junctions are the type of junction and the Reynolds number in each leg of the junction: $\mathrm{Re}_i = U_i L_i / \nu$. Mixing at the junctions is then represented in terms of the dimensionless concentration at one of the outlet, say outlet 3,

$$C_3^\star = \frac{C_3 - C_1}{C_1 - C_2}. \tag{3.20}$$

In practice, the software first scans the network to detect the occurrence of one of the studied junctions to apply non-homogeneous mixing. The cross junctions are easy to detect, double-T junctions are defined as two T-junctions separated by a pipe i with a maximum length $L_i < 10D_i$, where D_i is the diameter of the pipe. The non-homogeneous mixing is then applied. The code computes the incoming and outgoing Reynolds ratios. Then, thanks to an experimental database, the code links these ratios to dimensionless concentration (Eq. 3.20). Once the dimensionless concentration and the input concentrations are known, the output concentrations can be retrieved. Unfortunately, the authors did not publish the detailed code or the database. It is thus not possible to couple Azred with our code implementation to model non-homogeneous mixing.

Bulk-mixing model

This model has been developed by Ho et al. [10]. First, only cross junctions with pipes having all identical diameters are considered. The bulk-mixing model is a lower bound for the mixing. The hypothetical lowest achievable level of mixing would be a junction where incoming flows do not interact with each other. The momentum of the flows would then define the level of mixing. This lower bound for mixing is called *bulk-mixing* and is illustrated in Fig. 3.13 for a cross junction. Let derive the bulk-mixing model formally.



(a) Higher momentum in vertical branches.

(b) Higher momentum in horizontal branches.

Figure 3.13: Schematic representation of the bulk-mixing model.

If the momentum is higher in the vertical pipes (Fig. 3.13a), *i.e.* if

$$K_1 + K_3 > K_2 + K_4, \quad \text{where } K_i = Q_i^2 / A_i.$$

Bulk-mixing assumes that no water can flow from pipe 2 to pipe 4. The water age considering bulk-mixing for the downstream pipes would then be

$$C_{\text{bulk},3} = \frac{Q_2 C_2 + (Q_1 - Q_4)\, C_1}{Q_3}, \text{ and } \quad C_{\text{bulk, 4}} = C_1 \quad ,$$

because of mass conservation. Conversely, if $K_1 + K_3 < K_2 + K_4$ (Fig. 3.13b), no water can flow from pipe 1 to pipe 3 and the downstream water age are

$$C_{\text{bulk},3} = C_2 \text{ , and } \quad C_{\text{bulk, 4}} = \frac{Q_1 C_1 + (Q_2 - Q_3)\, C_2}{Q_4}.$$

It has to be contrasted with the perfect mixing model, used up to now, which is an upper bound for the mixing. Indeed, it is equivalent to the theoretical case where the incoming water would remain an infinite amount of time in the junction (while not ageing), would perfectly mix to eventually release the same water age distribution into all downstream pipes.

Thus, complete and bulk-mixing are respectively an upper and a lower bound for mixing at junctions. Exact mixing lies in between and can therefore be written as

$$C_{\text{exact},i} = C_{\text{bulk},i} + s\left(C_{\text{complete},i} - C_{\text{bulk},i}\right), \text{ with } 0 \leq s \leq 1.$$

In this bulk-mixing model, the complexity of the junctions is hidden in the dimensionless parameter s. This parameter depends mainly on the junction geometry and the momentum in the pipes connected to the junction. Ho et al. access its value through experimental data for a set of inlet flow ratios considering a specific cross junction. Of course, experimental characterisation of the dimensionless parameter s is not possible for all junctions on a full-scale network, and this method is not directly usable in practice. Though, bulk-mixing is easy to compute and can indicate the range of influence of the mixing assumption.

Further works have been done ([11]) to extend this model to cross junctions composed of different pipe diameters. This work only extends to the case where $D = D_2 = D_4$, and $d = D_1 = D_3$ as illustrated in Fig. 3.14. If the higher momentum is in the larger pipes, previous equations still hold since no water can flow from pipe 1 to pipe 3. In the other case, where the higher momentum is in the smaller pipes, Ho et al. have shown the wraparound effect. It is thus necessary to derive a new model to take into account this effect. This model is called the Bulk-Advective Mixing Wrap (BAM-WRAP) model. It assumes that a certain amount of flow coming from the larger diameter but smaller momentum is able to cross the higher momentum stream through the wraparound region (A_w), as shown in red in Fig 3.14.

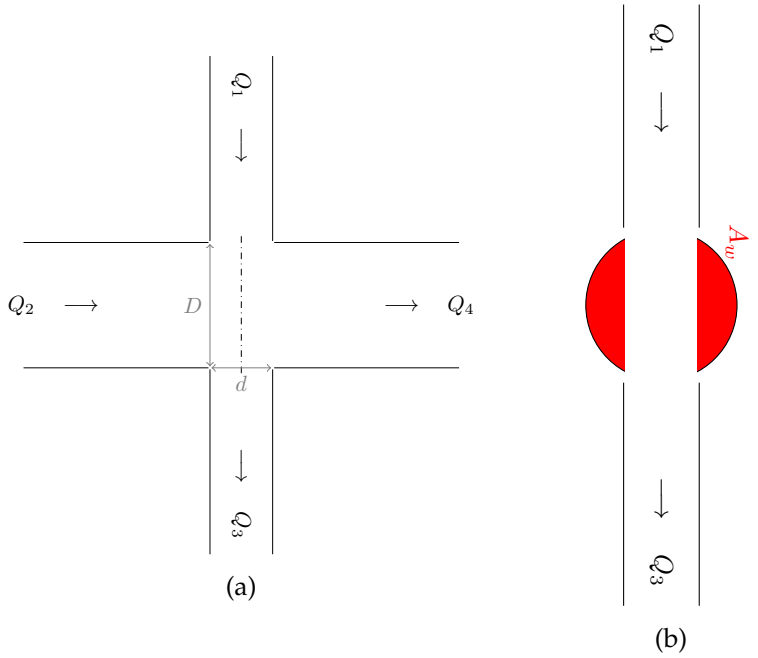


Figure 3.14: Schematic of a cross junction with pipes having different diameters for the BAM-WRAP model [11]. (a) shows a view from the top, whereas (b) is a cross-sectional view. The dash-dotted line in (a) shows the cross-sectional plane.

The authors assume this amount to be proportional to the wraparound region

$$A_w = (D/2)^2 (\theta - \sin\theta), \text{ where } \theta = 2\cos^{-1}(d/D).$$

The amount of flow going from pipe 2 to pipe 4 is thus

$$Q_w = \frac{A_w}{\pi D^2/4} Q_2.$$

From mass conservation, the BAM-WRAP model can then be recovered

$$C_4 = \frac{(Q_4 - Q_w)C_1 + Q_w C_2}{Q_4},$$
$$C_3 = \frac{(Q_3 - Q_2 + Q_w)C_1 + (Q_2 - Q_w)C_2}{Q_3}.$$

Flow distribution factors

The third method has been developed by Shao et al. [29]. The cross junction case is described here, but they proposed alternative equations to model double-T junctions. The main idea is to determine how the incoming flows distribute among the outcoming pipes. The flow distribution can be fully characterised for a cross junction thanks to one of the two parameters f_1 or f_2

$$\begin{cases} f_1 = \dfrac{Q_{14}}{Q_1}, \\ f_2 = \dfrac{Q_{23}}{Q_2}, \end{cases}$$

where Q_{ij} represents water going from pipe i into pipe j, as illustrated in Fig. 3.15.

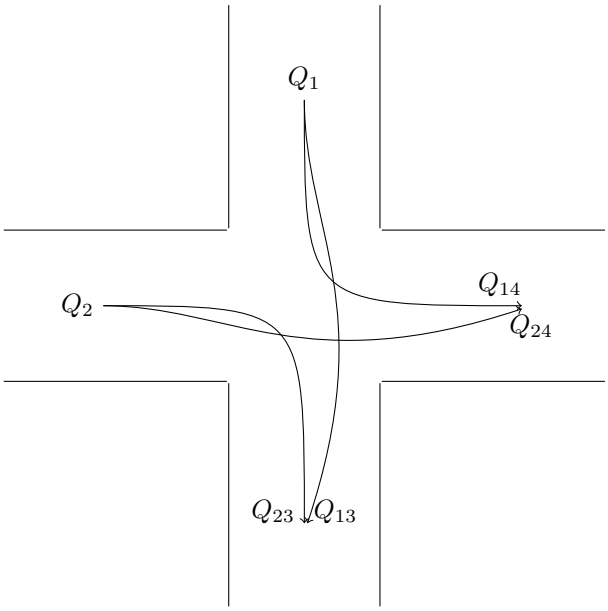


Figure 3.15: Schematic illustration of the flow distribution factors developed in Shao et al. ([29]).

From these parameters, the outcoming water age distribution can be deduced:

$$\begin{cases} C_3 = \dfrac{Q_1\,(1-f_1)\,C_1 + [Q_3 - Q_1\,(1-f_1)]\,C_2}{Q_3} \\ C_4 = \dfrac{Q_1 f_1 C_1 + (Q_4 - Q_1 f_1)\,C_2}{Q_4} \end{cases} \quad \text{or} \quad \begin{cases} C_3 = \dfrac{(Q_3 - Q_2 f_2)\,C_1 + Q_2 f_2 C_2}{Q_3} \\ C_4 = \dfrac{[Q_4 - Q_2\,(1-f_2)]\,C_1 + Q_2\,(1-f_2)\,C_2}{Q_4} \end{cases} \tag{3.21}$$

Shao et al. determine f_1 and f_2 thanks to experimental data and CFD calculations. These parameters only rely, for the cross junction case, on the flow momentum K_i in each pipe i: $K_i = Q_i^2 / A_i$. Here, it is assumed that there is a higher flow momentum coming from inlet 1 than coming from inlet 2, *i.e* $K_1 \geq K_2$. If it is not the case, pipes should be renamed to satisfy this condition. Once the pipes are named correctly, there are two possible cases. One where there is a higher output flow momentum in outlet 4 ($K_3 \leq K_4$, called higher momentum in the adjacent legs), and the other one where the higher momentum is in outlet 3 ($K_3 \geq K_4$, called higher momentum in the opposite legs). For the higher momentums in adjacent pipes, f_1 can be linked to the flow momentum ratio, $\eta = \dfrac{K_1 + K_3}{K_2 + K_4}$,

$$f_1 = \frac{1.0001}{\left[1 + \exp\left(1.0097 + 11.8279 \times \log_{10} \eta\right)\right]^{0.0747}},$$

whereas it is easier to link the flow momentum ratio to f_2 when the higher momentums locate in opposite pipes

$$f_2 = 0.990 \frac{\eta^{0.898}}{0.103 + \eta^{0.898}}.$$

Once one of the flow distribution factor, f_1 or f_2, is known; downstream CDF can be found thanks to Eq. 3.21.

Although [29] derive general expressions for pipes of different diameters, care should be taken while generalising since experimental verifications of the flow factors have only been performed on pipes with the same diameter.

The BAM-WRAP and flow distribution factors methods have been implemented in the iterative method developed in Section 2.4. They will be analysed and illustrated in the next Section.

3.2.3 Study cases

We now investigate the practical consequences of non-homogeneous mixing in our studied networks. Steady-state and perfectly advective flows are assumed. The analysis will be based on a modified Iterative Scheme for non-homogeneous mixing. The bulk-mixing, BAM-WRAP, and flow factors methods have been implemented and will be tested. Unfortunately, it is not possible to study the network Net1 since the only cross junction is composed of four pipes, all having a different diameter. The network Test rig will be investigated, and non-homogeneous mixing for a full-scale network will be discussed at the end on the Jockgrim network.

The demands at the nodes of the networks will be modified to set the demands to zero where one wants to model non-homogeneous mixing. This is the approach currently adopted in literature ([3], [30], [23]). It is a strong limitation for practical implementations. This emphasises the need for precise network geometry knowledge and demand point localisation. It may be necessary for future network descriptions to differentiate between simple junction and junction where water is extracted for the consumer.

In practice, the iterative scheme has been modified to model non-homogeneous mixing. The initial iterative scheme updates the value at each iteration thanks to the value of the direct upstream junctions. This is not directly applicable for non-homogeneous mixing since the age distribution leaving a junction is not the same for all downstream pipes. This situation is sketched in Fig. 3.16. It introduces some subtilities in the numerical code that are detailed in appendix C.

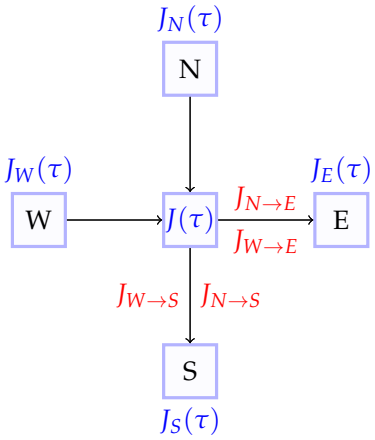


Figure 3.16: Modified iterative method for non-homogeneous mixing. Conversely to the complete mixing case, downstream junctions are not updated thanks to age at the direct upstream junction ($J(\tau)$) but are different in each downstream direction. More detailed in appendix C.

Test rig

Some results for the network Test rig are detailed in Fig. 3.17. Since all pipes have the same diameter, the bulk-mixing model and the flow factor distributions method have been used. The detailed results for cross junction 23 are shown in Fig. 3.17c. The CDFs from the upstream junctions (node 22 and 19) are compared to the CDFs from the downstream junctions (node 27 and 24). In this particular case, the flow distribution factors method tends to agree with the bulk-mixing model. As it has been predicted in the description of the mixing models, one can see that the results obtained with the flow distribution factors are always bound by the complete and bulk-mixing methods.

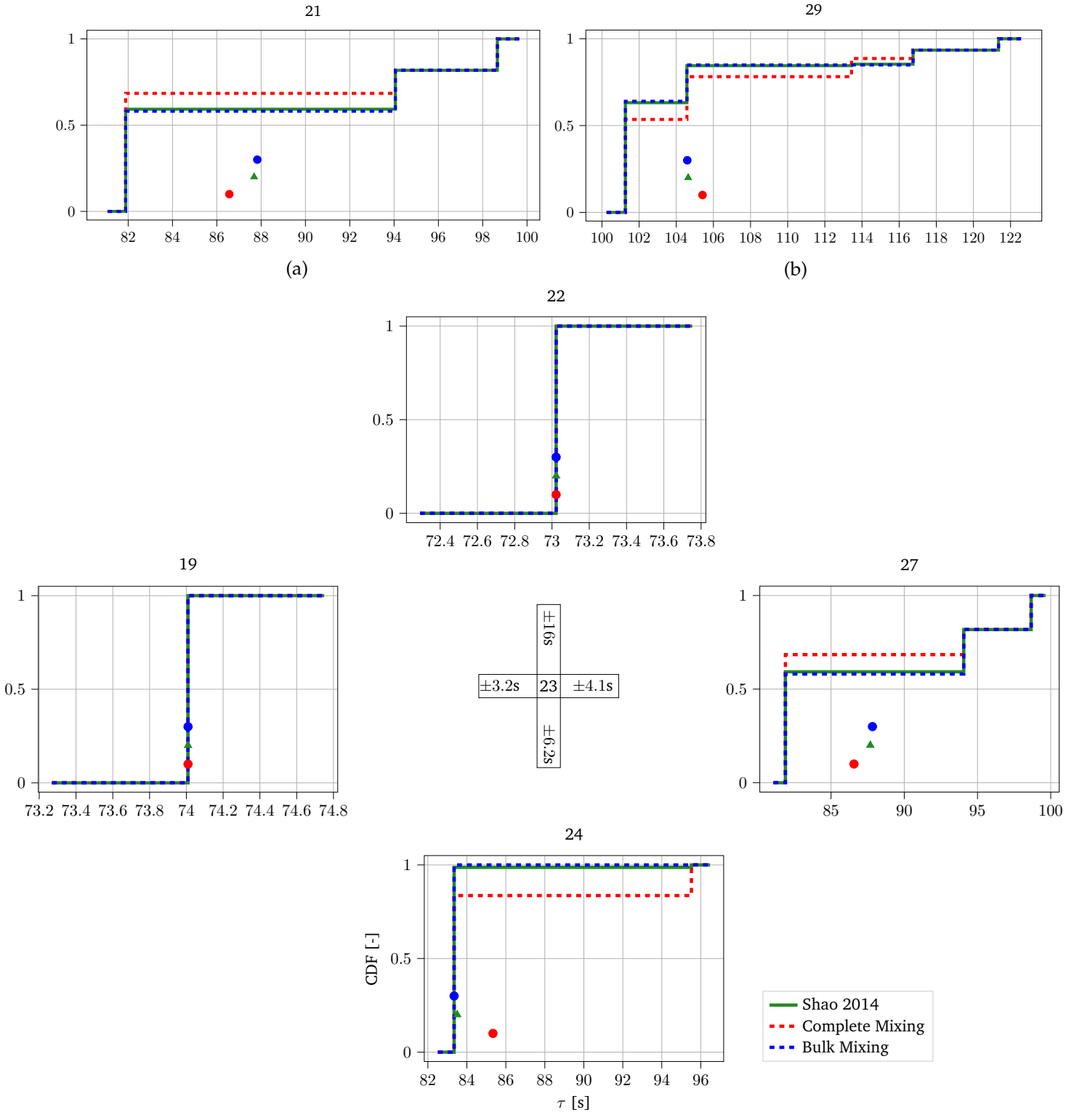


Figure 3.17: Non-homogeneous mixing applied to network Test Rig (Fig. 1.2b). Each figure shows results obtained with complete mixing, bulk-mixing, and flow distribution factors from Shao et al.([29]). Symbols indicates mean ages (at an arbitrary y-coordinate). Figs. 3.17a and 3.17b show the cumulative distribution for junction 21 and 29 respectively. Fig. 3.17c shows a detailed answer for junction 23 and its neighbours.

For node 24, the mixing assumption causes the older component to disappear. This may lead to different decisions in water quality management, which highlights the importance of the mixing assumption on the water age. Furthermore, the mixing assumption will have an impact on all its downstream nodes even if they are not directly connected to the cross junction, such as at node 29 in Figs. 3.17b.

Jockgrim

The full-scale Jockgrim network is now considered. For a full-scale network, one should determine an automatic detection of the junctions where non-homogeneous mixing has to be applied. An automatic determination of the junction geometry is also necessary. The method proposed here is based on the closest adjacent nodes. If a junction has exactly four connecting nodes, it is investigated as a potential cross junction. The adjacent nodes are brought back on a fictitious circle of unit radius, as shown in Table 3.1. The 'north' position is then assigned to the node with the highest y-coordinate, the 'west' position to the node with the lowest x-coordinate, and so on... Let us write it formally for the 'north' junction. Similar reasoning applies to others positioning. Noting a given cross junction J_0 with four neighbours $\{J_1, J_2, J_3, J_4\}$, which coordinate are noted (x_i, y_i) the 'north' position will be assigned to the junction $i \in \{1, 2, 3, 4\}$ such that

$$\frac{y_i - y_0}{\sqrt{(x_i - x_0)^2 + (y_i - y_0)^2}} \text{ is maxium.} \tag{3.22}$$

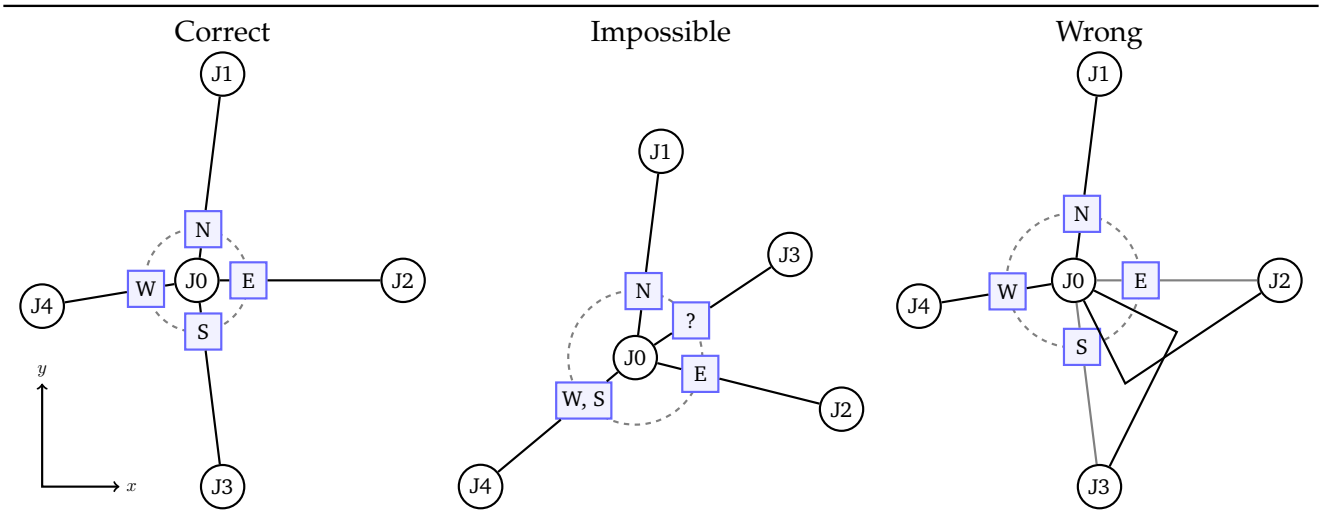


Table 3.1: Automatic detection of the geometry at a cross junction. $J0$ is the studied junction, JX are neighbour junctions, squares indicate the assignment (North, South, West, or East), solid black lines are pipes, and the dashed grey circles are the fictitious unit radius circle use for the automatic assignment.

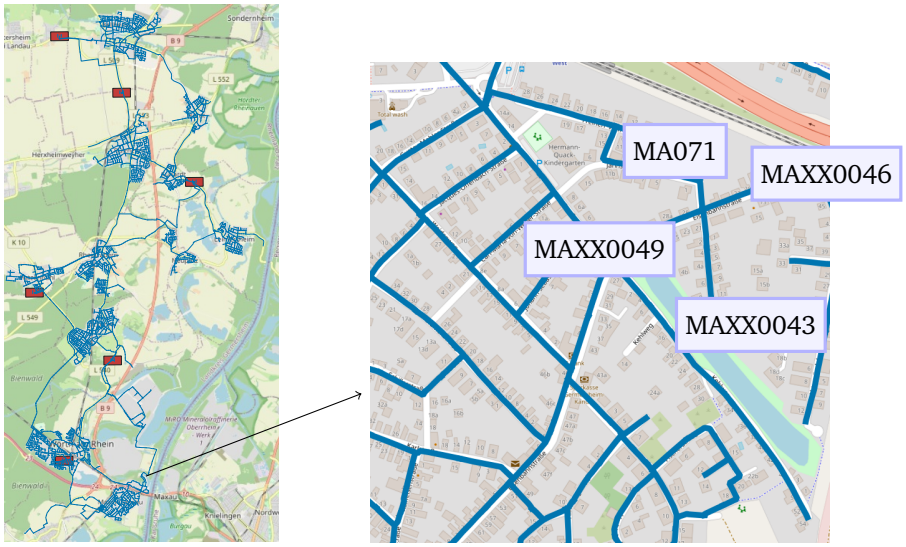
For very deform configurations, see for example 'Impossible assignment' in Table 3.1 or node 5 of Jowitt et al. network (Fig. 1.2c), this assignment may not be unique. In the latter

case, complete mixing is assumed because no work has been found for such junctions. The automatic assignment may also fail because of inaccurate junction descriptions (consider, for example, the last column of Table 3.1). If only the position of the nodes are known, but not the geometry of the pipes, a correct assignment is not guaranteed.

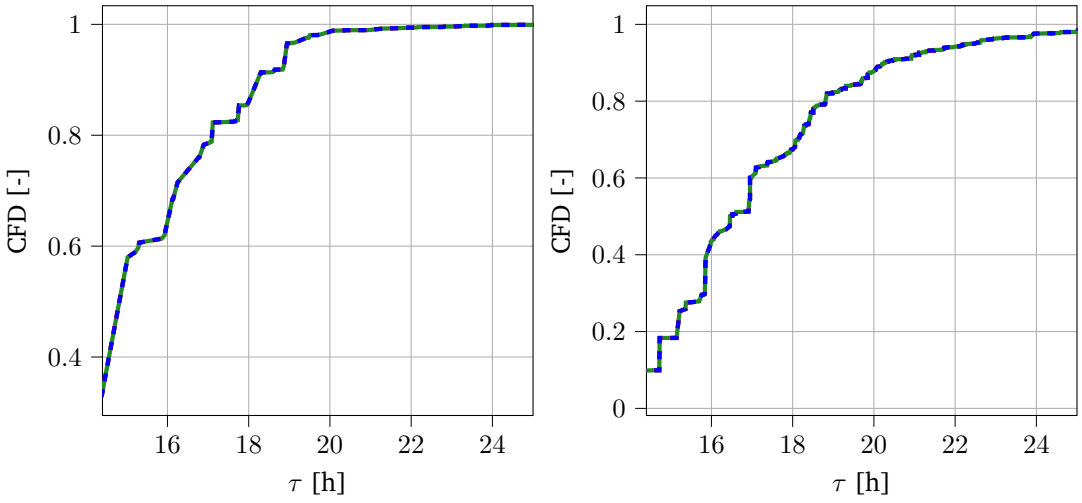
In the particular case of the Jockgrim network, only 64 junctions among the 4551 ones are connected to exactly four pipes, and 11 are double-T junctions. The same definition as in Romero et al. ([23]) has been taken for the definition of double-T junctions: two nodes with exactly three connections separated by a distance smaller than ten pipe diameters. An automatic assignment is not possible for 3 of the cross junctions ('Impossible' column of Table 3.1). Among the 61 remaining junctions, 14 junctions are composed of pipes with the same diameters, and 11 have only the opposite pipes with the same diameters. The 36 remaining junctions are composed of pipes with different diameters for which no study has been realised. In other words, non-homogeneous mixing can be applied confidently to only 25 junctions of the networks. The impact of the mixing assumption, in this case, is thus negligible. The vast majority of junctions is not influenced by the mixing assumption.

A detailed look at the cross junction MA072 in the municipality of Maximiliansau (south of Jockgrim) is shown in Fig. 3.18. Results are expected and similar to the ones that have been obtained on the network Test rig. The flow distribution factors method tends to agree with the bulk-mixing model but remains bound by this latter model and the complete mixing results.

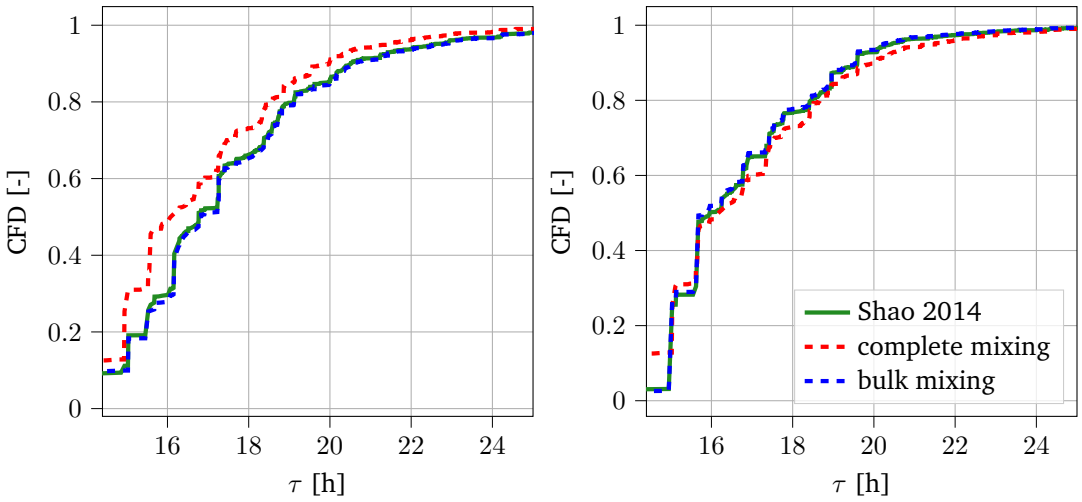
To really study the impact of non-homogeneous mixing on a network that is not majority composed of straight cross junctions, further studies are necessary to extend the non-homogeneous mixing on junctions of very different shapes.



(a) Relative positioning of the junctions around the cross junction MA072 of Jockgrim network.



(b) CDFs of junctions entering the cross junction.



(c) CDFs of junctions leaving the cross junction.

Figure 3.18: Examples of results that can be obtained for non-homogeneous mixing on a full-scale network.

3.2.4 Conclusion and outlook

Three mixing models have been reviewed from detailed studies that mainly focuses on single junctions. These models have been implemented with the model previously developed in Section 2.4. The implementation leads to similar results to the one obtained in the previous studies for the network Test rig. These results emphasised the impact of the mixing assumption on the water age. This Chapter then tries to extend these models for a full-scale network. A full-scale network required to automatically detect the geometry of the junctions, which has been done. The major obstacle to a usable implementation is the lack of models for junctions different from a traditional cross junction.

To model incomplete-mixing, one should know the junction types and the network geometry. The network geometry is not necessarily available for all networks. Furthermore, the junction type is sometimes simplified: double-T junctions are replaced with a unique cross-junction ([29]). A network can also be skeletonised to simplify hydraulic computation. These geometry simplifications are not significant for hydraulic modelling but are crucial for water quality monitoring. It would be a great advantage to access more data and more accurate junction descriptions to better model non-homogeneous mixing and improve water age computation in distribution networks. Further studies are necessary to increase the scope of junctions where non-homogeneous mixing can be applied. One way to improve water age representation in WDNs would be the availability of more complete data for each junction of the network. This would remove the need for automatic detection from adjacent junctions that will always lack reliability.

— 4 —

Software implementation

This chapter briefly explains the software architecture that has been used for the numerical implementations described in this master thesis. The numerical implementations have all been developed from scratch. Even if the goal of this chapter is not to describe extensively all numerical implementations, the developments are synthesised here to help potential future works on a similar subject. All the codes have been written in `Python 3.7.6`. The implemented code takes as input a connection matrix (used to describe the geometry of the network) and the discharges in the pipes. It outputs the age distributions as shown in Fig. 4.1. The only notable exception to this graph is for non-homogeneous mixing, for which the geographical coordinates of all junctions have to be input as well.

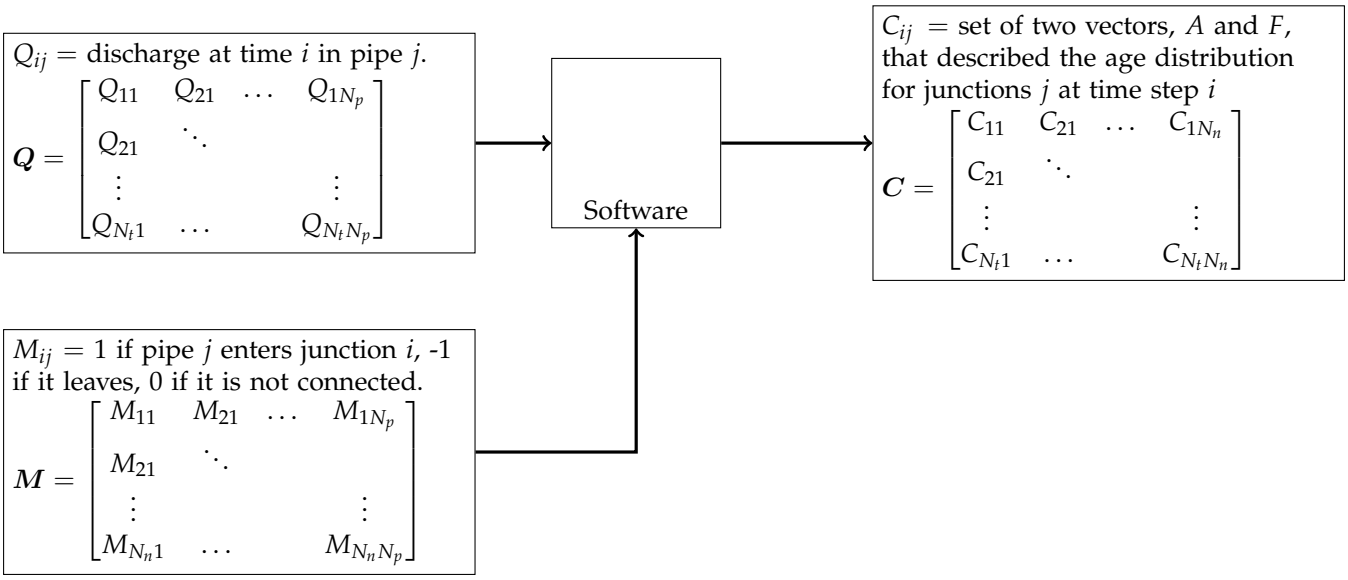


Figure 4.1: Input/Output schematic of the software implementation. N_n, N_p, and N_t are the number of pipes, junctions, and time-steps respectively.

To some extent, object-oriented programming has been used. The motivation was to be able to efficiently reused part of already implemented codes between the different Section of this work. Indeed, even if the different implementation may have very different characteristics: pipe spatially discretised or not, steady-state or not, sizes,...; it often converges in one or

several aspects: network topology described with connection matrix, often assumes complete mixing (notable exception of Section 3.2), age computation retrieved at the junctions,...

The adopted architecture is briefly presented in Fig. 4.2. A class named `Solver` instantiates one `Pipe` and one `Junction` for all pipes and junctions in the network. It takes care to correctly connected each `Junction` to all the relevant `Pipe`'s. The implementation of the classes `Junction` and `Pipe` depends on the numerical method chosen. However, the philosophy is always the same: each junction remembers its age distribution and the connected pipes. Thanks to the value of its upstream junction, each pipe is able to update the value at its downstream node. This result is, in turn, use to update all junctions.

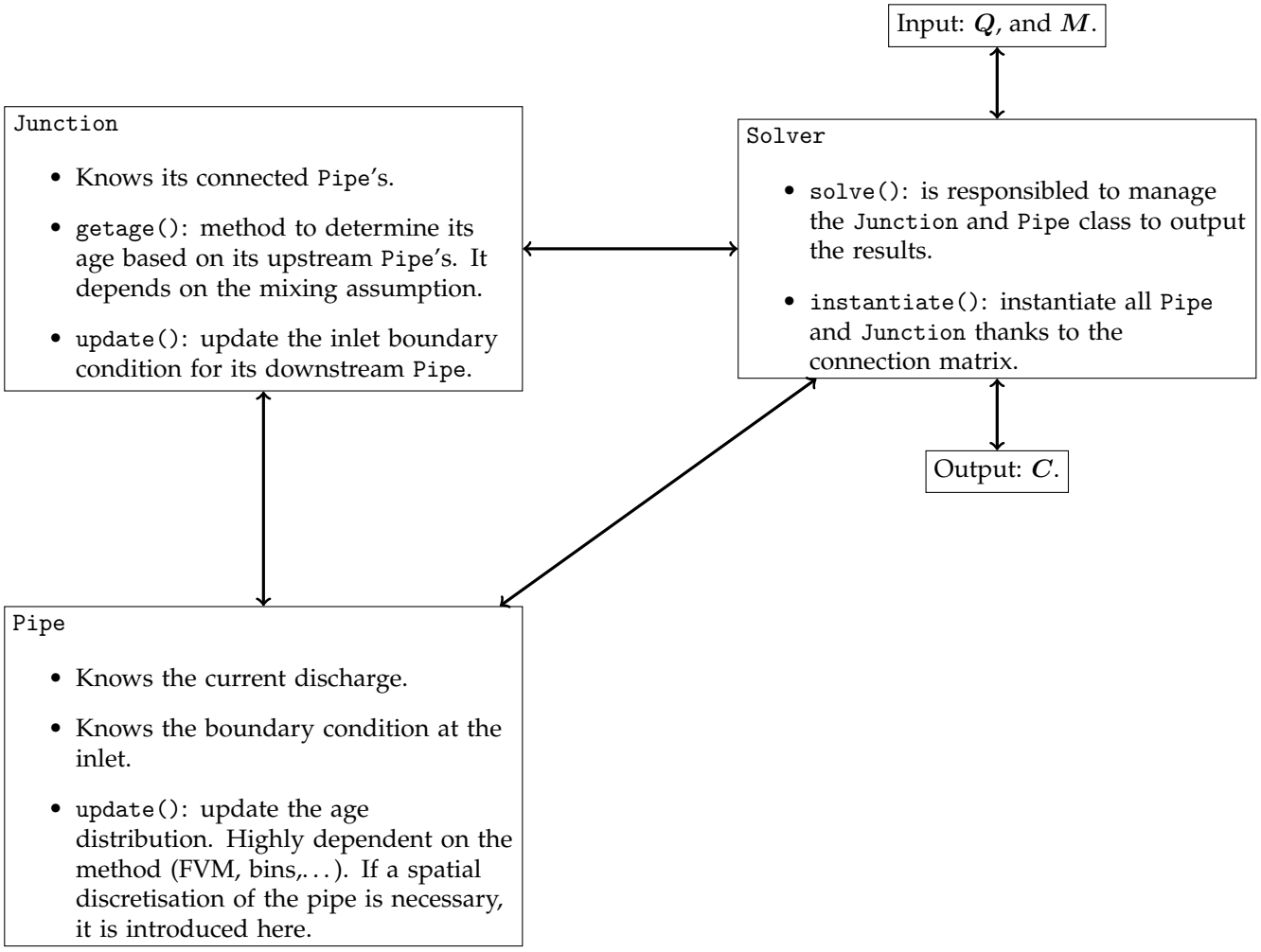


Figure 4.2: Architecture of the software implementation.

— 5 —

Conclusion and Further works

Based on the observation that computing water age distributions is of interest to assess the water quality, this master thesis tries to humbly contribute to the improvement of age computation in WDNs. The focus was set on two research directions, which correspond to the two main chapters:

- Reviewing and improving current modelling techniques for direct applications on full-scale network.
- Study how to include more complex phenomena, by relaxing assumptions on academic networks.

These works have been concretised thanks to the implementation of numerical methods written in `Pyhton`.

The first milestone was the reimplementation of existing methods, namely the one of EPANET and the one of Machel et al. (2009) [20]. The reimplementation of these methods leads to the development of a new method, first in stead-state, that was based on a modification of the water age distribution representation. The 'fix-bins' method developed by Machel et al. has been replaced with an 'adaptive' discretisation of the age distribution (see Eq. 2.21)

$$C_j(\tau) \approx \sum_{b=1}^{N_b} f_b \delta\left(\tau - \beta_b\right), \quad \text{where } \beta_b \in B_b \; \forall b \quad \longrightarrow \quad C_j(\tau) \approx \sum_{k=1}^{N_{\max}} f_k \delta(\tau - \alpha_k).$$

This allows significant improvements in accuracy and in computational efficiency. This gain in accuracy was shown on academic networks for which an analytical solution can be retrieved. This method was then applied on a full-scale network to prove the applicability of this new method for actual configurations. Since modelling only steady-states is unsatisfactory for real applications, some existing and new numerical methods were discussed in the next to model unsteady-state. These methods were coupled to the new age distribution representation, tested, and compared on academic networks.

Modelisation forces to make assumptions, the main ones being to neglect the dispersion effects and to assume perfect mixing. The second part of the present work thus focused on studying the impact of these two assumptions and how could these two effects be modelled for steady-state. The dispersion was investigated thanks to two different modelling techniques. First, an approximate analytical solution was used. This method did not require a discretisation and was shown to be a good approximation for high Peclet values when the

advective dominates the diffusive flux. Modelling dispersion was also achieved thanks to a finite volume numerical scheme, which required a spatial discretisation of the WDNs. A constant reconstruction scheme suffers from numerical dispersion and should only be used for low Peclet number flows. A linear reconstruction was shown to give better results and is useful when modelling highly advective flows but is more expensive from a computational point of view. The dispersion effects were investigated on a single pipe configuration and a simple academic network.

The second modelling assumption that has been questioned is homogeneous mixing. Several authors have studied mixing in a single junction and shown significant deviation from the perfect mixing assumption. They established non-homogeneous mixing models that have been reviewed and implemented for a full-scale network for junctions where it was possible.

There are many roads for future works to continue improving water age computation in WDNs. Some of them, inspired by this master thesis, are listed hereafter.

In this master thesis, the actual value of the diffusion parameter K was not addressed. This value should be investigated to orient further works. Indeed, modelling dispersion is computationally expensive, probably impossible nowadays on the traditional laptop for a full-scale network. However, as it has been shown, cheaper models can be applied when the properties of the flow are known. Investigating the value of K could thus be of great interest. This can be done by analytical analysis, stochastic methods, and experimental works such as in Romero et al. [25].

The implementation developed for non-homogeneous mixing was mainly limited because of the range of junctions to which non-homogeneous mixing can be applied confidently. The implementation of non-homogeneous mixing has been shown to be feasible on a full-scale network. The major remaining challenge is to develop non-homogeneous mixing models for a wider range of junctions. This can be done through experimental or computational fluid dynamic investigations.

A different direction is to develop a numerical scheme that can be parallelised (on CPUs or GPU). It may allow obtaining better results thanks to the increase in computational power.

Any of these works, or the ones that may have been forgotten, will be of great help to improve water age computation in WDNs. There is probably room for any engineering field that would like to contribute to the noble objective of ensuring good water quality to every consumer.

— A —

Detailed Water Distribution Networks

A.1 Net1

This network is one of the examples delivered with the software EPANET. It is shown in Fig. A.1 with its demand pattern applied to all nodes. Some of its key parameters are detailed in Table A.1. The exact EPANET input file used in this work is available here.

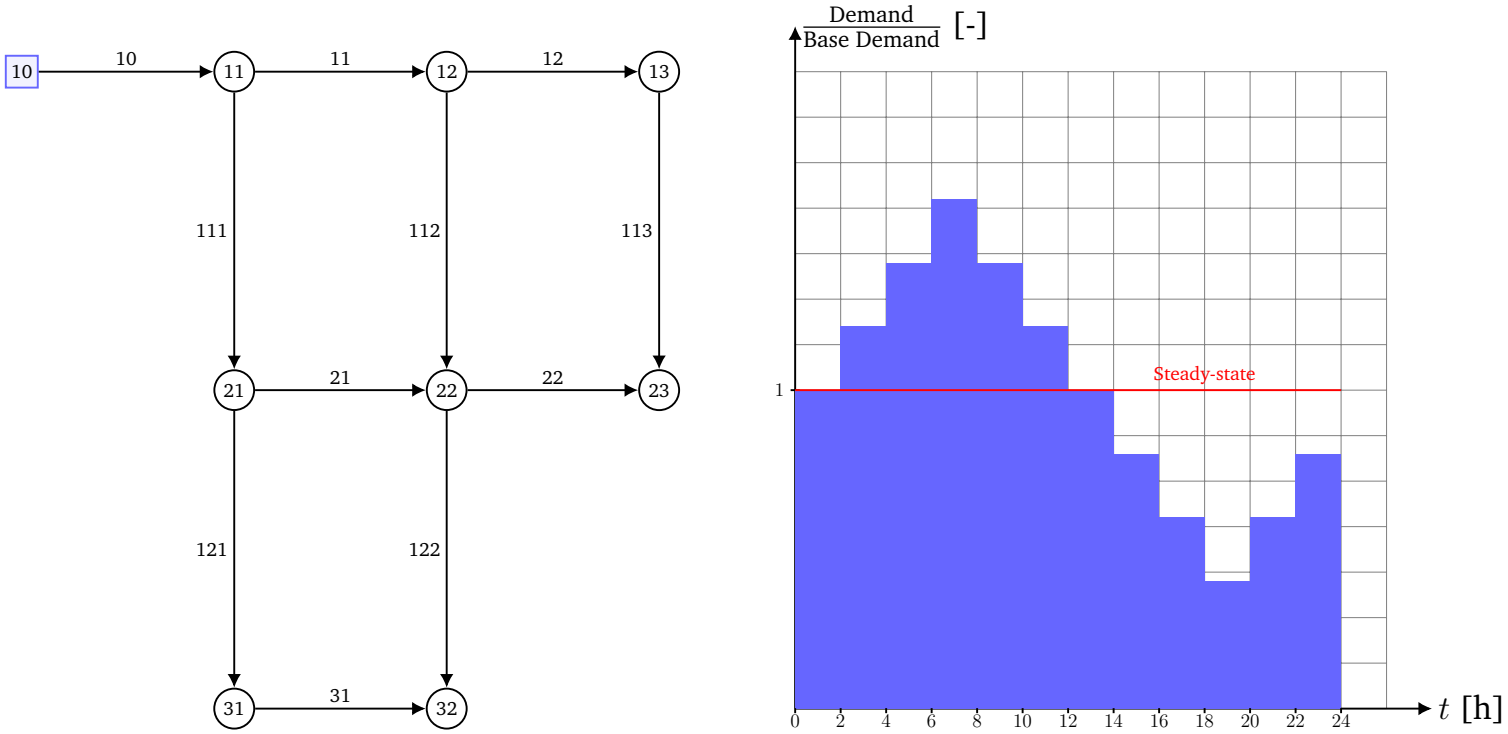


Figure A.1: Net1 network from EPANET [27] and its demand pattern. Arrows indicate conventional directions

Pipe ID	Length [foot]	Diameter [inch]
10	10 530	18
11	5280	14
12	5280	10
21	5280	10
22	5280	12
31	5280	6
110	200	18
111	5280	10
112	5280	12
113	5280	8
121	5280	8
122	5280	6

Node ID	Elevation [foot]	Demand [GPM]
10	710	0
11	710	150
12	700	150
13	695	100
21	700	150
22	695	200
23	690	150
31	700	100
32	710	100

Table A.1: Details of Net1 network.

A.2 Jowitt et al. 1990

This network comes from the paper of Jowitt et al. [14]. It is shown with its demand pattern in Fig. B.1. Some of its key parameters are detailed in Table A.2. The exact input file is available here.

A.3 Test Rig

Test rig network is shown in Fig. A.3. Some of its key parameters are detailed in Table A.3. The exact input file is available here. Some modifications have been made and are visible in the input file. The major one is the demands set to zero at cross junctions.

A.4 Jockgrim

This full-scale network has been kindly share by Mr. Ralf Friedmann and Markus Justen from Zweckverband für Wasserversorgung Germersheimer Südgruppe. Some modifications have been made to model steady-state: Node Hatz, Rulz, and Wör have been categorised as 'Reservoir's.

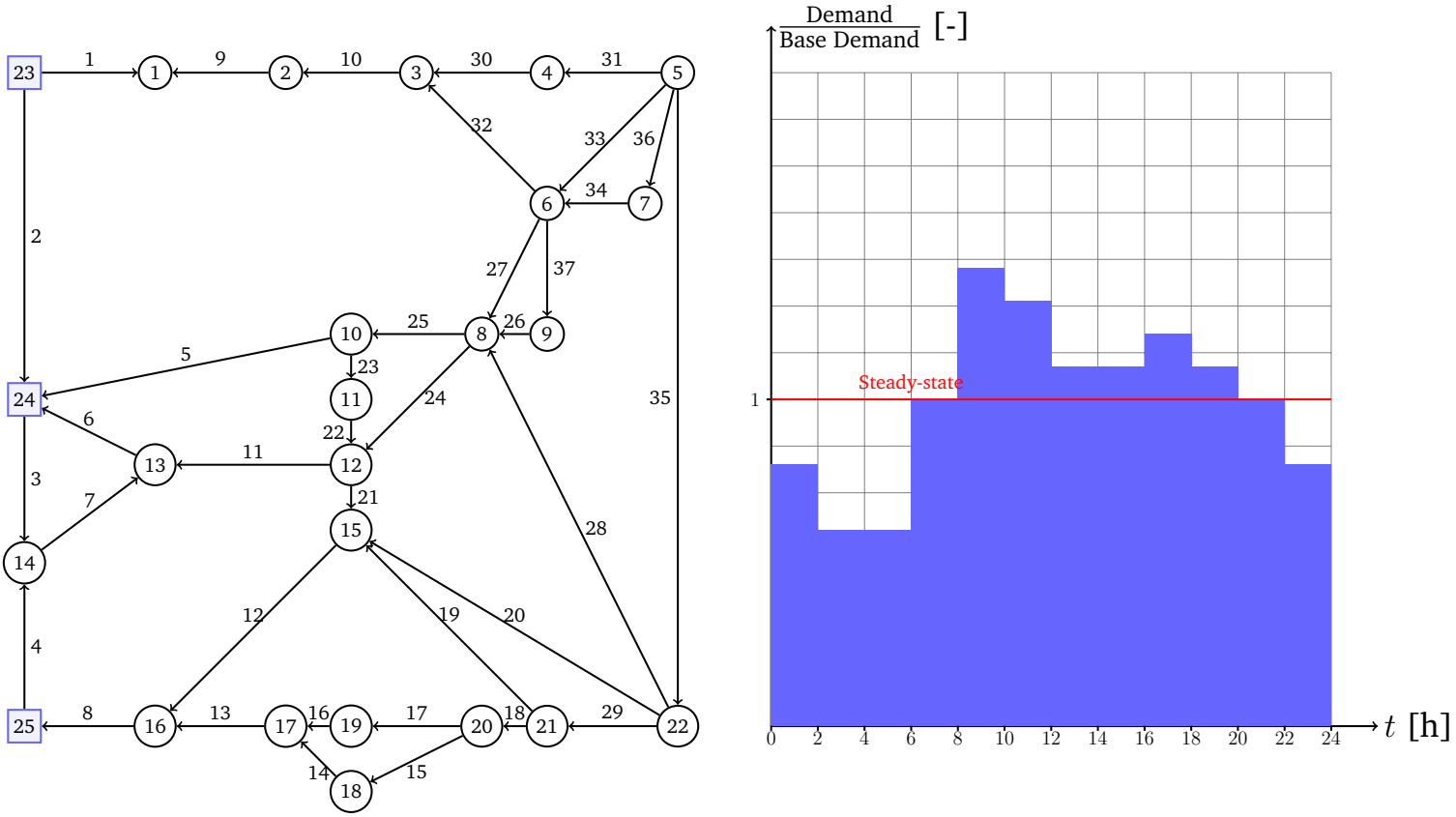


Figure A.2: Jowitt et al. network [14] and its demand pattern.

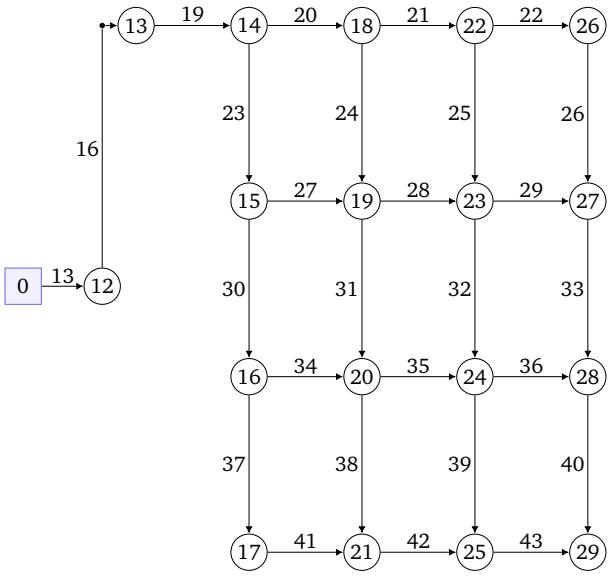


Figure A.3: Test rig network.

Pipe ID	Length [m]	Diameter [mm]
1	606	457
2	454	457
3	2782	229
4	304	381
5	3382	305
6	1767	475
7	1014	381
8	1097	381
9	1930	457
10	5150	305
11	762	457
12	914	229
13	822	305
14	411	152
15	701	229
16	1072	229
17	864	152
18	711	152
19	832	152
20	2334	229
21	1996	229
22	777	229
23	542	229
24	1600	457
25	249	305
26	443	229
27	743	381
28	931	229
29	2689	152
30	326	152
31	844	229
32	1274	152
33	1115	229
34	615	381
35	1408	152
36	500	381
37	300	229

Node ID	Elevation [m]	Demand [LPM]
1	18	300
2	18	600
3	14	0
4	12	5
5	14	1800
6	15	600
7	14.5	0
8	14	1200
9	14	0
10	15	300
11	12	600
12	15	0
13	23	0
14	20	300
15	8	1200
16	10	0
17	7	0
18	8	300
19	10	300
20	7	0
21	10	0
22	15	1200

Table A.2: Details of Jowitt et al. network.

Pipe ID	Length [m]	Diameter [mm]
13	10	115
16	30	115
19	1.5	50
20	1.5	50
21	1.5	50
22	1.5	50
23	1.5	50
24	1.5	50
25	1.5	50
26	1.5	50
27	1.5	50
28	1.5	50
29	1.5	50
30	1.5	50
31	1.5	50
32	1.5	50
33	1.5	50
34	1.5	50
35	1.5	50
36	1.5	50
37	1.5	50
38	1.5	50
39	1.5	50
40	1.5	50
41	1.5	50
42	1.5	50
43	1.5	50

Node ID	Elevation [m]	Demand [LPS]
12	0	0.00
13	0	0.00
14	0	0.50
15	0	0.50
16	0	0.50
17	0	0.50
18	0	0.50
19	0	0.00
20	0	0.00
21	0	0.50
22	0	0.50
23	0	0.00
24	0	0.00
25	0	0.50
26	0	0.50
27	0	0.50
28	0	0.50
29	0	0.50

Table A.3: Details of Test rig network.

— B —

Solution for Jowitt et al. network: pen-and-paper method

For steady-state, perfect mixing and no dispersion effects; a pen-and-paper solution can be found for the Jowitt et al. network. The Jowitt et al. network is represented in Fig. B.1 with the arrow in the direction of the flow. The details of the simulation parameters are available in appendix A and the velocity in each pipe of the network is shown in Table B.1.

Pipe ID	U [m/s]	Q [LPM]	L [m]
1	0.10	1019.45	606
2	0.19	1886.00	454
3	0.09	214.80	2782
4	0.36	2441.70	304
5	0.31	1345.87	3382
6	0.31	3294.48	1767
7	0.34	2356.50	1014
8	0.06	388.70	1097
9	0.07	719.45	1930
10	0.03	119.45	5150
11	0.57	5650.98	762
12	0.04	94.56	914
13	0.11	483.26	822
14	0.15	158.78	411
15	0.06	141.22	701
16	0.13	324.48	1072
17	0.02	24.48	864
18	0.11	116.74	711
19	0.01	9.74	832
20	0.14	338.60	2334
21	0.39	965.69	1996
22	0.29	715.91	777
23	0.05	115.91	542
24	0.40	3969.38	1600
25	0.27	1161.78	249
26	0.18	447.00	443
27	0.30	2084.27	743
28	0.57	1399.89	931
29	0.10	107.00	2689
30	0.16	178.01	326
31	0.07	173.01	844
32	0.05	58.56	1274
33	0.13	330.71	1115
34	0.23	1542.00	615
35	0.23	245.72	1408
36	0.23	1542.00	500
37	0.18	447.00	300

Table B.1: Velocity in each pipe of Jowitt et al. network for steady-state.

For each pipe i, the travel time T_i can be easily computed: $T_i = L_i/U_i$. Node 23, 24, and 25 are the entrance of the network where water has a zero-age

$$C_{23} = \delta(\tau), \quad C_{24} = \delta(\tau), \quad \text{and } C_{25} = \delta(\tau).$$

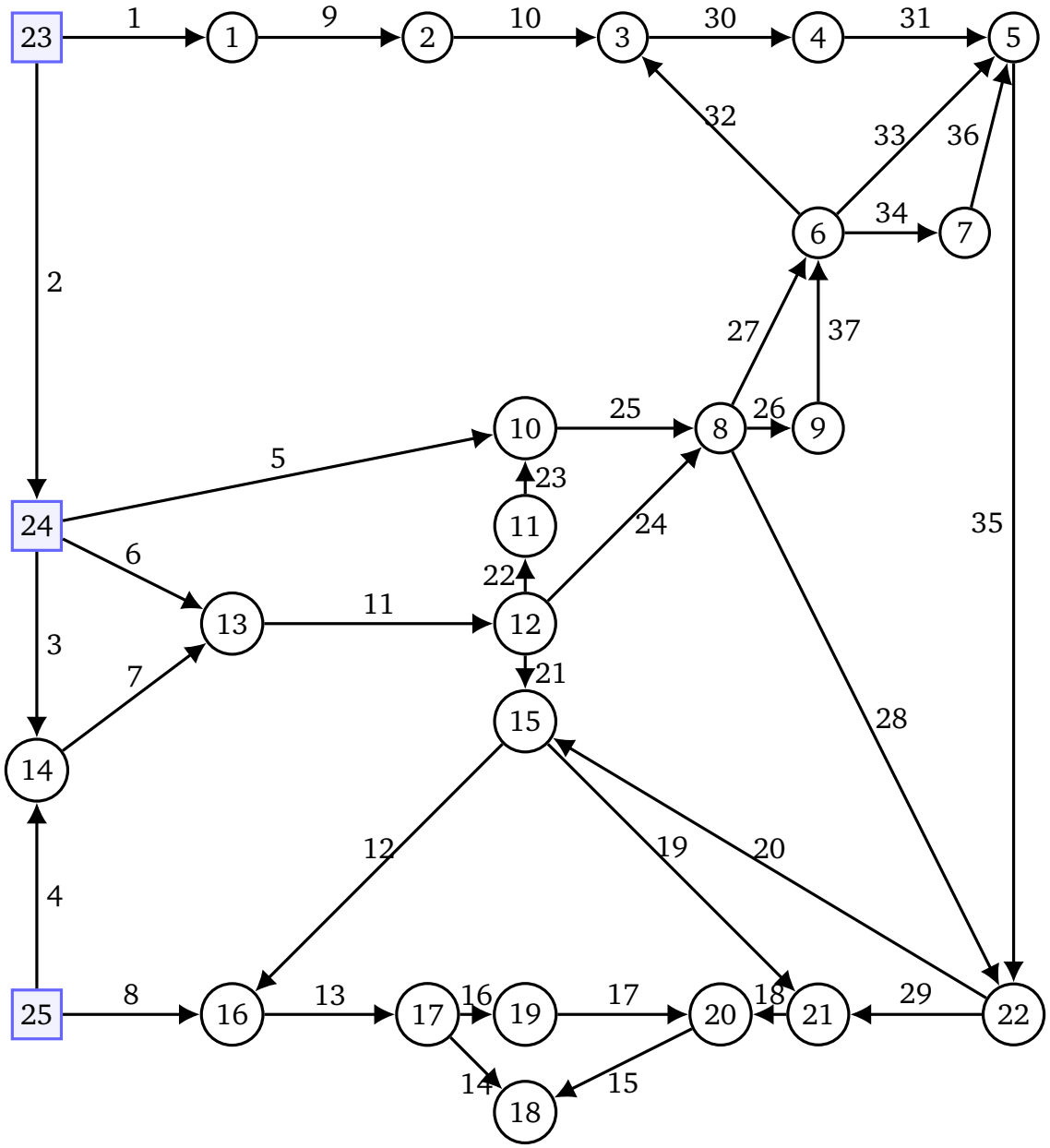


Figure B.1: Jowitt et al. network. Steady-state with arrows in the direction of the flow.

The age at junction 14 is

$$C_{14} = \left[Q_3\delta(\tau - T_3) + Q_4\delta(\tau - T_4)\right]\frac{1}{Q_3 + Q_4}.$$

Therefore the age at junction 13 is

$$C_{13} = \left[Q_7 C_{14}(\tau - T_7) + Q_6\delta(\tau - T_6)\right]\frac{1}{Q_6 + Q_7}.$$

Continuing with a similar reasoning, the distribution at all nodes can be computed. The order of computation matters, one should take care to always compute all upstream node before computing any given node. Starting from node 13

$$\begin{aligned}
C_{12} &= C_{13}(\tau - T_{11}),\\
C_{11} &= C_{12}(\tau - T_{22}),\\
C_{10} &= \left[Q_{23}C_{11}(\tau - T_{23}) + Q_5\delta(\tau - T_5)\right]\frac{1}{Q_{23} + Q_5},\\
C_8 &= \left[Q_{25}C_{10}(\tau - T_{25}) + Q_{24}C_{12}(\tau - T_{24})\right]\frac{1}{Q_{25} + Q_{24}},\\
C_9 &= C_8(\tau - T_{26}),\\
C_6 &= \left[Q_{27}C_8(\tau - T_{27}) + Q_{37}C_9(\tau - T_{37})\right]\frac{1}{Q_{27} + Q_{37}}, \text{ and}\\
C_7 &= C_6(\tau - T_{34}).
\end{aligned}$$

Restarting from node 1, $C_1 = \delta(\tau - T_1)$

$$\begin{aligned}
C_2 &= C_1(\tau - T_9),\\
C_3 &= \left[Q_{10}C_2(\tau - T_{10}) + Q_{32}C_6(\tau - T_{32})\right]\frac{1}{Q_{10} + Q_{32}},\\
C_4 &= C_3(\tau - T_{30}),\\
C_5 &= \left[Q_{31}C_4(\tau - T_{31}) + Q_{33}C_6(\tau - T_{33}) + Q_{36}C_7(\tau - T_{36})\right]\frac{1}{Q_{31} + Q_{33} + Q_{36}},\\
C_{22} &= \left[Q_{28}C_8(\tau - T_{28}) + Q_{35}C_5(\tau - T_{35})\right]\frac{1}{Q_{28} + Q_{35}},\\
C_{15} &= \left[Q_{20}C_{22}(\tau - T_{20}) + Q_{21}C_{12}(\tau - T_{21})\right]\frac{1}{Q_{20} + Q_{21}}, \text{ and}\\
C_{21} &= \left[Q_{29}C_{22}(\tau - T_{29}) + Q_{19}C_{15}(\tau - T_{19})\right]\frac{1}{Q_{29} + Q_{19}}.
\end{aligned}$$

Eventually, restarting from node 25

$$\begin{aligned}
C_{16} &= \left[Q_8\delta(\tau - T_8) + Q_{12}C_{15}(\tau - T_{12})\right]\frac{1}{Q_8 + Q_{12}},\\
C_{17} &= C_{16}(\tau - T_{13}),\\
C_{19} &= C_{17}(\tau - T_{16}),\\
C_{20} &= \left[Q_{17}C_{19}(\tau - T_{17}) + Q_{18}C_{21}(\tau - T_{18})\right]\frac{1}{Q_{17} + Q_{18}}, \text{ and}\\
C_{18} &= \left[Q_{14}C_{17}(\tau - T_{14}) + Q_{15}C_{20}(\tau - T_{15})\right]\frac{1}{Q_{14} + Q_{15}}.
\end{aligned}$$

— C —

Modified iterative scheme for non-homogeneous mixing

For a better understanding of this appendix, it is recommended to first read the chapter about the software implementation, Chapter 4. Let us first describe the implementation for the iterative method assuming perfect mixing. Each pipe of the network is represented by an instance of the class `Pipe`. The feature of interest for this Section of the `Pipe` class is its method `get_travel_time()`. This method returns the time needed for the water to go through the pipe. The second class of interest is the class `Junction` that is used to represent the nodes of the WDNs. Each instantiation of the class `Junction` is characterised by:

- a vector of the connected pipes, and
- a method `get_age()`, used to retrieve the age at the junction. As explained in Section 2.4, the age distribution is represented thanks to two vectors: the age components A, and the fraction of water for each age component, F.

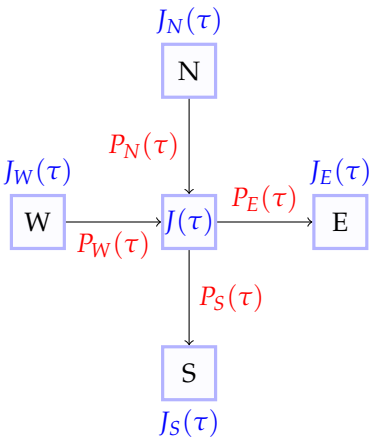


Figure C.1: Schematic of a cross junction. J_x (resp. P_x) is the age distribution at the node (resp. pipe) of the branch x.

Considering the cross junction in Fig. C.1, the south age distribution will be updated as follows:

- Retrieved the age distribution at the centred node thanks to the `get_age()` method:

$$[A, F] = J.\texttt{get\_age()}$$

- Determined the travel time to cross pipe P_S thanks to the method `get_travel_time()`:

$$T = P_S.\texttt{get\_travel\_time()}.$$

- Update the age distribution at the south junction

$$A_S = A + T,$$
$$F_S = F.$$

This algorithm can not be directly applied for non-homogeneous mixing since the age distribution in pipe P_S is not the one returned by the method `get_age()` at the junction J. To overcome this problem, a new method is implemented at each junction: `get_age_outlet(outlet)`. It returns the age distribution in '`outlet`' pipe of the junction. The `get_age_outlet(outlet)` method is a recursive method that determines the age distribution in the pipe '`outlet`' (P_S or P_E) given the age distribution in the upstream pipe of the junction (P_N and P_W). P_N and P_W are detrmined by applying the same `get_age_outlet(outlet)` method at node J_N and J_W.

This recursive algorithm is easy to implement. Its drawback is that the recursive implementation may be computationally expensive on networks where many cross junctions are connected. The easier way to avoid a recursive method would be to describe each junction with N_p distributions, where N_p is the number of pipes connected to the junction. There is one distribution for each pipe connected to the junction. There is never more than three cross junctions in a row for the studied networks. Therefore, only the recursive method has been implemented.

Bibliography

[1] Milton Abramowitz and Stegun. Handbook of mathematical functions with formulas, graphs, and mathematical tables, 1988.

[2] AWWA (American Water Works Association), Economic, and Inc Engineering Services. Effects of water age on distribution system water quality. Technical report, U.S. Environmental Protection Agency Office of Ground Water and Drinking Water Standards and Risk Management Division, August 2002.

[3] R. G. Austin, B. van Bloemen Waanders, S. McKenna, and C. Y. Choi. Mixing at cross junctions in water distribution systems. ii: Experimental study. *Journal of Water Resources Planning and Management*, 134(3):295–302, 2008.

[4] E.J. Mirjam Blokker, William R. Furnass, John Machell, Stephen R. Mounce, Peter G. Schaap, and Joby B. Boxall. Relating water quality and age in drinking water distribution systems using self-organising maps. 3(2), 2016.

[5] Christopher Y Choi, Jerry Y Shen, and Ryan G Austin. Development of a comprehensive solute mixing model (azred) for double-tee, cross, and wye junctions. In *Water Distribution Systems Analysis 2008*, pages 1–10. 2008.

[6] Eric Deleersnijder, Insaf Draoui, Jonathan Lambrechts, Vincent Legat, and Anne Mouchet. Consistent boundary conditions for age calculations. *Water*, 12(5):1274, 2020.

[7] Eric JM Delhez, Jean-Michel Campin, Anthony C Hirst, and Eric Deleersnijder. Toward a general theory of the age in ocean modelling. *Ocean Modelling*, 1(1):17–27, 1999.

[8] Benjamin Dewals, Pierre Archambeau, Martin Bruwier, Sebastien Erpicum, Michel Pirotton, Tom Adam, Eric Delhez, and Eric Deleersnijder. Age of water particles as a diagnosis of steady-state flows in shallow rectangular reservoirs. *Water*, 12(10):2819, 2020.

[9] Walter M. Grayman, Regan Murray, and Dragan A. Savic. *Effects of Redesign of Water Systems for Security and Water Quality Factors*, pages 1–11.

[10] Clifford K Ho. Solute mixing models for water-distribution pipe networks. *Journal of Hydraulic Engineering*, 134(9):1236–1244, 2008.

[11] Clifford K. Ho and Leslie O'Rear Jr. Evaluation of solute mixing in water distribution pipe junctions. *Journal AWWA*, 101(9):116–127, 2009.

[12] S. A. Imran, J. D. Dietz, G. Mutoti, J. S. Taylor, and A. A. Randall. Modified larsons ratio incorporating temperature, water age, and electroneutrality effects on red water release. *Journal of Environmental Engineering*, 131(11):1514–1520, 2005.

[13] Eric Jones, Travis Oliphant, Pearu Peterson, et al. SciPy: Open source scientific tools for Python: scipy.signal.fftconvolve, 2001–.

[14] Paul W Jowitt and Chengchao Xu. Optimal valve control in water-distribution networks. *Journal of Water Resources Planning and Management*, 116(4):455–472, 1990.

[15] Gregory J Kirmeyer. *Guidance manual for maintaining distribution system water quality*. American Water Works Association, 2000.

[16] Nikolaos Kourbasis, Menelaos Patelis, Stavroula Tsitsifli, and Vasilis Kanakoudis. Optimizing water age and pressure in drinking water distribution networks. In *Environmental Sciences Proceedings*, volume 2, page 51. Multidisciplinary Digital Publishing Institute, 2020.

[17] Dmitri Kuzmin. Lecture 10 in introduction to computational fluid dynamics. https://www.mathematik.uni-dortmund.de/~kuzmin/cfdintro/lecture10.pdf, , accessed May 2021., 2007.

[18] Randall J. LeVeque. *Finite Volume Methods for Hyperbolic Problems*. Cambridge Texts in Applied Mathematics. Cambridge University Press, 2002.

[19] Jin-Sheng Lin and Lynn M. Hildemann. A nonsteady-state analytical model to predict gaseous emissions of volatile organic compounds from landfills. *Journal of Hazardous Materials*, 40(3):271–295, 1995.

[20] John Machell, Joby Boxall, Adrian Saul, and Dylan Bramley. Improved representation of water age in distribution networks to inform water quality. *Journal of Water Resources Planning and Management*, 135(5):382–391, 2009.

[21] Sheldon Masters, Jeffrey Parks, Amrou Atassi, and Marc A Edwards. Distribution system water age can create premise plumbing corrosion hotspots. *Environmental monitoring and assessment*, 187(9):1–18, 2015.

[22] William J Rhoads, Amy Pruden, and Marc A Edwards. Survey of green building water systems reveals elevated water age and water quality concerns. *Environmental Science: Water Research & Technology*, 2(1):164–173, 2016.

[23] P Romero-Gomez, CK Ho, and CY Choi. Mixing at cross junctions in water distribution systems. i: Numerical study. *Journal of Water Resources Planning and Management*, 134(3):285–294, 2008.

[24] P Romero-Gomez, Z Li, CY Choi, SG Buchberger, Kevin E Lansey, and VT Tzatchkov. Axial dispersion in a pressurized pipe under various flow conditions. In *Water Distribution Systems Analysis 2008*, pages 1–10. 2008.

[25] Pedro Romero-Gomez and Christopher Y Choi. Axial dispersion coefficients in laminar flows of water-distribution systems. *Journal of Hydraulic Engineering*, 137(11):1500–1508, 2011.

[26] Pedro Romero-Gomez, Kevin E Lansey, and Christopher Y Choi. Impact of an incomplete solute mixing model on sensor network design. *Journal of Hydroinformatics*, 13(4):642–651, 2011.

[27] Lewis A. Rossman, Hyoungmin Woo, Michael Tryby, Feng Shang, Robert Janke, and Terranna Haxton. *EPANET 2.2 User Manual*. U.S. Environmental Protection Agency, https://epanet22.readthedocs.io/_/downloads/en/latest/pdf/, epa/600/r-20/133 edition.

[28] Benjamin Seibold. Lecture 19 in numerical methods for partial differential equations. https://ocw.mit.edu/courses/mathematics/18-336-numerical-methods-for-partial-differential-equations-spring-2009/lecture-notes/MIT18_336S09_lec19.pdf, , accessed May 2021., 2009.

[29] Yu Shao, Y Jeffrey Yang, Lijie Jiang, Tingchao Yu, and Cheng Shen. Experimental testing and modeling analysis of solute mixing at water distribution pipe junctions. *Water research*, 56:133–147, 2014.

[30] Inhong Song, Pedro Romero-Gomez, Manuel A Andrade, Mario Mondaca, and Christopher Y Choi. Mixing at junctions in water distribution systems: an experimental study. *Urban Water Journal*, 15(1):32–38, 2018.

[31] Martinus Th van Genuchten, Feike J Leij, Todd H Skaggs, Nobuo Toride, Scott A Bradford, and Elizabeth M Pontedeiro. Exact analytical solutions for contaminant transport in rivers 1. the equilibrium advection-dispersion equation. *Journal of Hydrology and Hydromechanics*, 61(2):146–160, 2013.