

Jonathan de Halleux

Membres du Jury:

Georges Bastin, (Promoteur)

Jean-Michel Coron,

Brigitte d'Andréa-Novel,

Denis Dochain,

Joseph Winkin,

Vincent Blondel (Président)

Boundary Control of Quasi-Linear Hyperbolic Initial Boundary-Value Problems

September 23, 2004

Contents

1	Introduction	3
1.1	Conference Papers	8
1.2	Articles	8
2	MIBVP	9
2.1	Quasilinear Hyperbolic Systems	9
2.2	Classical solutions	11
2.3	Hyperbolicity	12
2.4	Lopatinski condition	12
2.5	Existence of solutions for the MIBVP problem	13
2.6	Riemann invariants	14
2.6.1	Description	14
2.6.2	Finding the invariants	15
2.6.3	Characteristic form	17
2.7	Boundary Stabilization of MIBVP	18
2.7.1	Li's Theorem (94)	18
2.7.2	Illustration of the proof on a 2×2 system	20
2.7.3	Extension of Li's Theorem	22
2.8	Application to control	25
2.9	Conclusion	27
3	Stabilization of a cascade of 2×2 systems	29
3.1	Cascade of 2×2 Hyperbolic Systems	29
3.2	Graph theory and non-negative matrices	31
3.3	Graph representations	33
3.3.1	Examples	33
3.3.2	Graph of a 2×2 system	34
3.3.3	Graph of a cascade of two 2×2 system	35
3.4	Control design	36
3.4.1	Model	36
3.4.2	Properties of the <i>invariant graph</i>	37

3.4.3	Control Design using the <i>invariant graph</i>	37
3.5	Stability of 2×2 systems	39
3.5.1	Illustration for values of $m = 3, 4, 5$	40
3.6	Conclusion	45
4	Application to waterways with mobile spillways	47
4.1	Introduction	47
4.2	Model of a horizontal reach	48
4.2.1	Saint Venant equations	48
4.2.2	Steady-states	49
4.2.3	Statement of the control problem	49
4.2.4	Stability Analysis of the Steady-States and Control design	49
4.3	Stabilization of a cascade of reaches	53
4.3.1	Model	53
4.3.2	Main result	55
4.4	Practical case: Sambre river	62
4.4.1	Description and model	62
4.4.2	Stability condition	63
4.4.3	Simulation results	63
5	Control design based on Entropy	67
5.1	Introduction	67
5.2	Entropies	67
5.3	Boundary Control design methodology	68
5.4	Application to Saint Venant equation	69
5.4.1	Model	69
5.4.2	Steady-states	70
5.4.3	Statement of the control problem for the two reach channel	71
5.5	Entropies of shallow water equations	71
5.6	Lyapunov control design	72
5.6.1	Open loop control	75
5.7	Numerical results	76
5.7.1	Open loop stability	76
5.8	Conclusion	77
6	Control of a moving water tank	81
6.1	Introduction	81
6.2	System and control problem description	83
6.2.1	Model description	83
6.2.2	Steady states	83
6.2.3	Control problem	84
6.3	Lyapunov control design	85
6.3.1	Stabilization of the fluid state (H, V)	86

6.3.2	Stabilization of the fluid state (H, V) and of the tank speed S	89
6.3.3	Stabilization of the fluid-tank state	91
6.4	Numerical results	93
6.4.1	Observing stabilization in simulation	93
6.4.2	Stabilization with output feedback and full-state feedback	94
6.4.3	Importance of the nonlinear terms of the shallow water equations	96
6.5	Conclusion	96
7	Numerical Simulation of the Shallow Water Equation	101
7.1	Model	101
7.1.1	Saint-Venant equations	101
7.1.2	Hydraulic characteristics	103
7.1.3	Boundary conditions	103
7.1.4	Preissman scheme	104
7.2	General idea for the resolution	105
7.3	Model validation	105
7.4	Linearized Model	106
7.4.1	Saint-Venant equations	106
7.4.2	Boundary functions	108
7.5	Global system	110
7.6	Computing steady-states	112
7.6.1	Flow Q_i	112
7.6.2	Depth h_i	112
7.7	Details on $\mathbf{S}_i, \mathbf{Q}_i$	112
8	Conclusions and perspective	115
8.1	Conclusions	115
8.2	Perspectives	116
	References	117
	Mathematical preliminaries	121
	A.1 Implicit Function Theorem	121
	List of notations	123
	Saint-Venant equation simulator	125
C.1	Minimum requirements	125
C.2	Simulation configuration	125
C.2.1	General parameters - Figure C.1	125
C.2.2	Number of reaches - Figure C.2	126
C.2.3	Description of the geometry - Figure C.3	126
C.2.4	Detailed configuration of each reach	128

VIII Contents

C.2.5	Detailed configuration of each gate	130
C.2.6	Detailed configuration of each controller	132
C.3	Visualisation of the results	133
C.3.1	Animations	133
C.3.2	Flow and water depth.	133
C.3.3	3D display	133
C.3.4	Toolbars	133
C.3.5	Result analysis	135
 Data structure and algorithms for Flux graph and Invariant		
	graphs	139
D.1	Data structures	139
D.2	Algorithm	140

Acknowledgements

Je tiens à remercier le promoteur Georges Bastin ainsi que Jean-Michel Coron et Brigitte d'Andréa-Novel pour leur encadrement au cours de cette thèse. Je remercie les membres du Jury Joseph Winkin, Denis Dochain ainsi que Vincent Blondel.

Je tiens particulièrement à remercier Christophe Prieur pour sa participation active dans nos travaux de recherches Enfin, je remercie toute l'équipe des chercheurs et du corps enseignant du CESAME pour leur accueil chaleureux et leur convivialité au cours de ces quatre années.

Introduction

During the 19-th century, the field of fluid dynamics and hydraulics witness a major revolution. Famous mathematicians like Navier and Stokes wrote their name in history by deriving the equations for the motion of incompressible fluids. Navier was the first to publish them in 1821, while Stokes derived the equations afterward. In fact this duplication of results was not entirely an accident, but was rather brought about by the lack of knowledge of the work of continental mathematicians at Cambridge at that time. These equations remained as the Navier-Stokes equations.

Another French Mathematician, the engineer Jean Claude Baré de Saint-Venant, had also derived this result in 1843 (the reason why his name never became associated with those equations remains a mystery). Nevertheless, in 1871, Saint-Venant publishes a paper entitled *Théorie du mouvement non-permanent des eaux avec applications aux crues des rivières et à l'introduction des marées Dans leur lit* [63] where he derived the equations for non-steady flow in open channels. Those equations, named Saint-Venant equations in honor of his work, are the basis of the modern hydraulics. Oddly, these equations are also called *shallow water equations*. They are still taught in Hydraulic schools and used in simulation software to model the behavior of open channels.

The Saint-Venant equations are conservation laws. One equation represents the mass conservation and the second equation the conservation of the momentum. Conservation laws are described by hyperbolic partial differential equations. Partial differential equations are usually distinguished in three important families, hyperbolic, parabolic and elliptic equations. Each family is associated with a fundamental equation: wave, heat and Laplace. The stabilization of conservation laws is a difficult problem because of the complexity of the underlying mathematical model. In general, fully nonlinear and even quasilinear systems are hard to deal with because of the lack of classical solutions. Mathematicians are usually forced to drop details about the models such as viscosity and geometry of the system or even linearise them, in order to obtain mathematical models that can be studied in their framework.

While some systems such as rivers are naturally stable, they can become unstable when man starts to manipulate them. Modern canalised rivers have become canal-boat highways composed of reaches interconnected by hydraulic structures. To satisfy the navigation constraints, the hydraulic structures have been equipped with controllers to stabilize the water depth and flow. Bad tuning or design of such controllers usually creates a small amplification of the deviation of the water state with regards to the nominal state. Due to the cascade structure of navigation channels, those deviations sum up along the reaches and quickly become significant. Therefore, there exist a practical need for ways of tuning and designing controllers for such systems. During the work of the Thesis, an extensive study of the Sambre river was made as part as a contract with Ministry of Equipment and Transports (MET). Their engineers were facing a severe problem of flow oscillation amplification: from the Monceau gate to the Salzinnes gate 50 km downstream, the flow oscillations amplify from a few cubic meters to peaks as big as 80 cubic meters. In the worst cases, the flow would simply stop at the Salzinnes gate. It is interesting to note that natural rivers are naturally stable, this is not true anymore in canalised rivers as one can see from the hydrograms in Figure 1.1. The reason for this destabilisation are often a bad design or a bad tuning of the controllers. Nowadays, the rivers are separated in reaches by hydraulic gates. Each gate has controller that controls the water flow in order to stabilize each reach water depth and flow. If each of those controllers destabilizes slightly the system, their cumulative errors sum up, leading to results that could be observe at the Salzinnes gate.

The objective of this thesis is to explore methods and techniques for designing boundary controls and proving the stability of a system of conservation laws. In this study, a particular care was taken in order to obtain controls that could be implemented in an automaton in a real plant, such as the Sambre river. Therefore, a significant amount of work was made to develop a numerical simulator of the Saint-Venant equation (see Appendix C) where the different control design approaches could be compared in terms of robustness and efficiency.

In Chapter 2, we give the mathematical background on quasilinear hyperbolic systems. We define the classical formulation of conservation laws and important definitions and results as the definition of hyperbolicity and Theorems for the existence of solutions. In Section 2.6, we introduce the concept of Riemann invariant for quasi-linear hyperbolic systems. Riemann invariants are quantities that have the interesting property of remaining constant along special trajectories called characteristic curves. This invariance property is crucial in the control design. We present a stability result due Greenberg and Li Ta-Tsien in [34], later on generalized in by Li in [49]. In the continuity of their work, we generalize further their result in Theorem 2.28 which was published in [41] and used in a conference paper [23]. Finally, we show how the stability Theorem 2.28 can be used to design controllers for conservative laws.

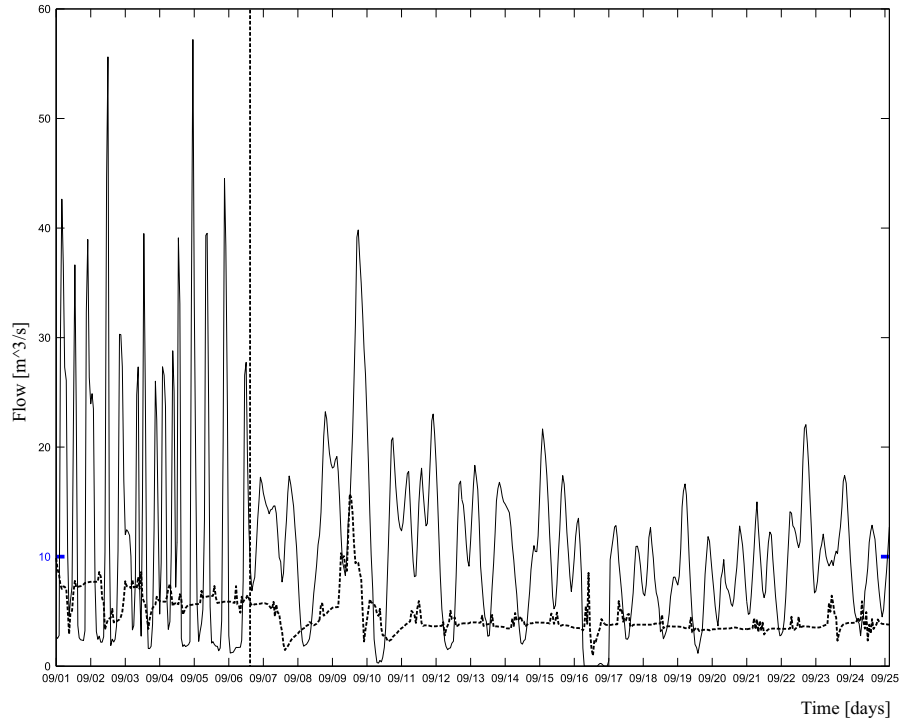


Fig. 1.1. Measured water flows in September 2002 at Monceau (dotted curve) and Salzines (plain curve) gates in m^3/s . The vertical dotted line denotes a change of parameters in the controllers of the Sambre that drastically reduced the oscillation of the flow. Before that time, one could observe periods where the flow of the river would actually be stopped.

In Chapter 3, a design methodology, called Riemann approach, enables to design control laws using the Theorem 2.28. This approach uses a new representation of the system characteristic curves called the invariant graph. This graph represents the possible "travelling" of the Riemann invariants between the different boundary laws of the system. On the basis of this topologically rich formalism, a methodology has been developed for the particular case of network composed of a cascade of 2×2 systems. The objective of the method is to find a boundary control law (if possible, easily tunable and implementable) that stabilizes the system. Following the idea of having simple and implementable control laws, this design approach is focused on deriving control laws that are analytically simple. The conference paper [25] is related to this topic.

In Chapter 4, we apply the design approach proposed in Chapter 3 to systems composed of a cascade of reaches modelled by the Saint-Venant equations (shallow water equations) separated by spillways. We show that by using the

proposed methodology one can stabilize the system using simple boundary local control laws. The theoretical results are illustrated by numerical simulation using the Simulator (see Appendix C).

In Chapter 5, another design technique based on entropies and Lyapunov functions is also studied. This approach was introduced by Coron & al. in [18] for a single reach and has been studied in various publications [24, 23, 19] and [41]. Entropies are scalar quantities that are conserved along the smooth trajectories of the system. As we show in Chapter 5, such a property makes the entropy an attractive candidate for a Lyapunov function. Using a classic Lyapunov control design approach, this methodology yields new controllers. It is important to note that this is not sufficient to prove the stability. Therefore, the stability Theorem 2.28 can be used to prove that fact.

In Chapter 6, the Lyapunov approach, coupled with backtracking, was applied to the stabilization of moving water tanks and published in [62]. This problem is often referred to sloshing where a moving vehicle contains a quantity of fluid. Depending on the situation, sloshing of the fluid can induce oscillation of the vehicle, fuel consumption increase, etc... In fact, sloshing problems are encountered in many industrial fields such as fuel trucks, agro-alimentary liquid packaging or even spatial ships. Following the methodology developed in Chapter 5, stabilizing controllers are designed for a sequence of problems with increasing complexity. At each step, the previous problem is augmented with a new state to stabilize and the Lyapunov function is updated accordingly. This gives an intuitive and iterative method to design control laws for such systems. The conference paper [59, 61, 60] is related to this topic and was followed by a regular paper [62].

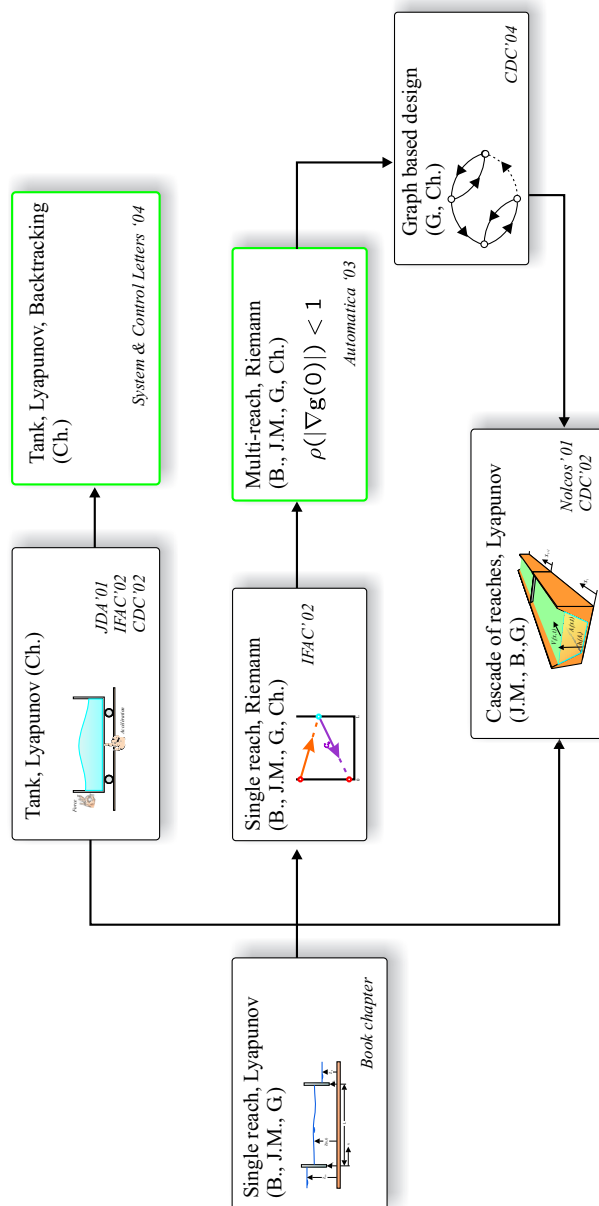


Fig. 1.2. This scheme shows the relationships between the different studied problems, the publications made on the subject and the "road" that led to them. In each box, one can find the initials of the collaborators (J.M. for Jean-Michel Coron, B. for Brigitte d'Andréa-Novel, G. for Georges Bastin and Ch. for Christophe Prieur) and the various publications made in conferences and journals on the topic.

1.1 Conference Papers

The Figure 1.2 depicts the temporal and causal relationships between the different results that were presented during the thesis. Each box represents a particular result and contains the conference and journals where those results have been presented and published.

- [24] J. de Halleux, B. d'Andréa-Novel, J.-M. Coron, and G. Bastin. A Lyapunov approach for the control of multi reach channels modelled by Saint-Venant equations. In *CD-Rom Proceedings, pg. 1515-1520*, St-Petersburg, Russia, June 2001. NOLCOS'01.
- [59] C. Prieur and J. de Halleux. Complete stabilization of a tank. In *Journées Doctorales d'Automatique 2001*, 2001.
- [23] J. de Halleux and G. Bastin. Stabilization of Saint-Venant equations using Riemann invariants: Application to waterways with mobile spillways. In *CD-Rom proceedings*, Barcelona, Spain, July 2002.
- [60] Ch. Prieur and J. de Halleux. Complete stabilization of a tank. In *CD-Rom proceedings*, Barcelona, Spain, July 2002.
- [19] J.-M. Coron, J. de Halleux, G. Bastin, and B. d'Andréa Novel. On boundary control design for quasilinear hyperbolic systems with entropies as lyapunov functions. In *41st IEEE Conference on Decision and Control (CDC02), Las Vegas, USA, December, 2002.*, 2002.
- [61] Ch. Prieur and J. de Halleux. Stabilization of a tank via output feedback. In *Conf. on Dec. and Cont.*, 2002.
- [25] J. de Halleux, Ch. Prieur, and G. Bastin. Boundary control design for cascades of hyperbolic 2×2 pde systems via graph theory. In *43rd IEEE Conference on Decision and Control*, 2004.

1.2 Articles

- [41] de Halleux J., Prieur C., Coron J.-M., d'Andréa-Novel B., and Bastin G. Boundary feedback control in networks of open channels. *Automatica*, 39(8):1365–1376, August 2003.
- [62] Christophe Prieur and Jonathan de Halleux. Stabilization of a 1-d tank containing a fluid modeled by the shallow water equations. *Systems & Control Letters*, 52(3-4):167–178, July 2004.

The mixed Initial-Boundary value problem for hyperbolic systems in one space dimension

This chapter gives a short introduction on quasi-linear hyperbolic systems. It starts by stating the problem and giving the definition of hyperbolicity in Section 2.4. The conditions and the results for the existence of solutions of such systems is then described in Sections 2.4 and 2.5. An introduction to the Riemann invariants, as well as the method to compute them, is given in Section 2.6. Section 2.7 recalls the existing stability results. The methodology used to prove these results is illustrated on a simple system. A new stability result is given in Theorem 2.28. At last, application to control of the stability results presented in Section 2.7 is given in Section 2.8.

Among the numerous publications on the subject, the material from this chapter is mainly extracted from the following reference textbooks [48], [64] and [10].

2.1 Quasilinear Hyperbolic Systems

In this section we give a brief introduction on quasi-linear hyperbolic systems in one dimension space and restrict us to classical solutions.

Let n be a positive integer. Let $\mathcal{M}_n$ be the set of linear mappings $\mathbb{R}^n \mapsto \mathbb{R}^n$, let $T > 0$ and $\mathcal{U}$ be an open non-empty convex set of $\mathbb{R}^n$. Let $\mathbb{C}^k$ denotes the space of functions which are k times continuously differentiable, with uniformly bounded derivatives up to order k .

We define the applications $A : \mathcal{U} \mapsto \mathcal{M}_n$. We consider a class of quasilinear hyperbolic systems of order n defined as follows: Let $u(x, t) \in \mathcal{U}$,

$$\partial_t u + A(u) \partial_x u = 0 . \quad (2.1)$$

where ∂_t (resp. ∂_x) denotes the partial derivative with respect to t (resp. to x). The spatial variable x belongs to the bounded interval $[0, 1]$, while the time variable t belongs to the bounded interval $[0, T]$. The application A is also called *characteristic matrix* of (2.1).

The Cauchy condition

$$u(x, 0) = u^\natural(x), \quad (2.2)$$

where $u^\natural : [0, 1] \mapsto \mathbb{R}^n$ is given but is usually not enough to ensure the uniqueness of the solutions of the Cauchy problem (2.1)-(2.2). Therefore, boundary conditions are added at $x = 0$ and $x = 1$. We impose

$$\begin{aligned} B_0(u(0, t)) &= 0, \\ B_1(u(1, t)) &= 0. \end{aligned} \quad (2.3)$$

where

$$B_0 : [0, T] \times \mathcal{U} \rightarrow \mathbb{R}^{m_0} \text{ and } B_1 : [0, T] \times \mathcal{U} \rightarrow \mathbb{R}^{m_1}. \quad (2.4)$$

Remark 2.1. In Section 2.7.3, the boundary conditions will be (2.3) are generalized to the following form:

$$B(u(0, t), u(1, t)) = 0. \quad (2.5)$$

Example 2.2 (Saint-Venant and Shallow Water equations). In order to illustrate the definition and concepts introduced in this Chapter, we shall give a short overview of the *Saint-Venant equations* (alias Shallow Water Equations (SWE)). These equations are a well-known example of quasilinear hyperbolic system and have a straightforward physical interpretation which facilitates the understanding of the chapter. These SWE will be further described and discussed in Chapter 4, 5 and 6.

Let us consider a one-dimensional, horizontal, friction-less open channel delimited by two underflow gates. Let $V(x, t)$, the water velocity and $H(x, t)$ the water depth at position x and time t . Let H_{up} (resp. H_{down}) be respectively the constant upstream (resp. downstream) water level. Using the notation introduced in this section, (2.1) is written as follows:

$$u(x, t) = \begin{pmatrix} H(x, t) \\ V(x, t) \end{pmatrix}, \quad (2.6)$$

$$A(u) = \begin{pmatrix} V & H \\ g & V \end{pmatrix}. \quad (2.7)$$

Sluice gates are well-known hydraulic structures (see [33, Section 4.4.3]). The discharge relationship for the sluice gates is given in the form of (2.3) as follows:

$$\begin{aligned} B_0(u(0, t)) &= H(0, t)V(0, t) - K_v \sqrt{2g(H_{up} - H(0, t))} = 0 \\ B_1(u(1, t)) &= H(1, t)V(1, t) - K_v \sqrt{2g(H(1, t) - H_{down})} = 0 \end{aligned} \quad (2.8)$$

where K_v is a physical coefficient and g is the gravity.

Following Remark 2.1, one can also write the discharge relationship in the (2.5) form:

$$B(u(0, t), u(1, t)) = \begin{pmatrix} H(0, t)V(0, t) - K_v \sqrt{2g(H_{up} - H(0, t))} \\ H(1, t)V(1, t) - K_v \sqrt{2g(H(1, t) - H_{down})} \end{pmatrix}. \quad (2.9)$$

2.2 Classical solutions

We need to introduce the notion of *strict hyperbolicity* and the *Lopatinskiĭ* conditions in order to ensure the unicity of the Cauchy problem (2.1)-(2.3)-(2.2) which is usually called *Mixed Initial-Boundary Value Problem (MIBVP)*. We shall only consider classical solutions defined as follows:

Definition 2.3 (Classical solution). *By a classical solution of (2.1)-(2.2)-(2.3) we mean a continuously differentiable function $u = u(x, t)$ which satisfies (2.1) at every point of its domain and (2.3) at the boundaries.*

We shall use the following norm:

$$\|u(t, \cdot)\|_{\mathbb{C}^1} = \|u(t, \cdot)\|_{\mathbb{C}^0} + \|\partial_x u(t, \cdot)\|_{\mathbb{C}^0} . \quad (2.10)$$

Remark 2.4 (Non-homogenous quasilinear equations). Given an application $h : [0, 1] \times [0, T] \times \mathcal{U} \rightarrow \mathbb{R}^n$, a more general class of quasi-linear hyperbolic systems in one dimension space is as follows:

$$\partial_t u + A(u) \partial_x u = h(x, t, u) . \quad (2.11)$$

This class of systems will not be considered in this thesis. Such systems have been studied numerically using the simulator described in Appendix C.

Remark 2.5 (Systems of Conservation Laws). Many physical systems can be expressed in terms of conservation laws in the sense that they can be written as follows:

$$\partial_t u + \partial_x f(x, t, u) = 0 \quad (2.12)$$

where $f : [0, 1] \times [0, T] \times \mathcal{U} \mapsto \mathbb{R}^n$ is called the *flux* and u is the *conserved quantity*. Integrating (2.12) over the interval $[0, 1]$ one obtains

$$\begin{aligned} \partial_t \int_0^1 u(x, t) dx &= \int_0^1 \partial_t u(x, t) dx \\ &= - \int_0^1 \partial_x f(x, t, u) dx \\ &= f(0, t, u(0, t)) - f(1, t, u(1, t)) \\ &= [\text{inflow at } 0] - [\text{outflow at } 1] . \end{aligned} \quad (2.13)$$

In other words, the quantity u is neither created nor destroyed: the total amount of u contained within the interval $[0, 1]$ can change only due to the flow of u across the boundaries.

If the flux function $f(u)$ is regular enough, the system (2.1) can be deduced from (2.12) by defining:

$$A(u) = \nabla_u f(u) . \quad (2.14)$$

2.3 Hyperbolicity

We have the following classic definition of hyperbolicity (see [64]):

Definition 2.6 (Hyperbolicity). *The system (2.1) is hyperbolic if, for every $u \in \mathcal{U}$, the matrix $A(u)$ has n real eigenvalues $\lambda_1(u) \leq \dots \leq \lambda_n(u)$. Moreover, if all the eigenvalues are distinct, $\lambda_1(u) < \dots < \lambda_n(u)$, the system is said strictly hyperbolic.*

2.4 Lopatinski condition

We assume that system (2.1) is strictly hyperbolic (see Definition 2.6). We consider that the boundaries $x = 0$ and $x = 1$ are non-characteristic for the system (2.1), i.e. the characteristic matrix $A(u)$ of (2.1) remains full rank at the boundaries:

$$\begin{aligned} \det A(u(0, t)) &\neq 0, \forall u \in \mathcal{U}, \\ \det A(u(1, t)) &\neq 0, \forall u \in \mathcal{U}. \end{aligned} \quad (2.15)$$

Since we assume that $A(u)$ is continuous, the number of positive eigenvalues of $A(u(0, t))$ is independent of $t \in [0, T]$ and $u \in \mathcal{U}$ (recall that $\mathcal{U}$ is convex). We call m_+^0 this number. Similarly, the number of negative eigenvalues for $A(u(1, t))$ is independent of $t \in [0, T]$ and $u \in \mathcal{U}$. We denote m_-^1 this number.

For $u \in \mathcal{U}$, we denote $E_+(u)$ (resp. $E_-(u)$) the invariant subspace of $A(u)$ in $\mathbb{R}^n$ generated by the eigenvectors of $A(u)$ associated with the positive (resp. negative) or zero eigenvalues:

$$\begin{aligned} E_+(u) &= \sum_{\lambda \geq 0} \ker(A(u) - \lambda I), \\ E_-(u) &= \sum_{\lambda \leq 0} \ker(A(u) - \lambda I). \end{aligned} \quad (2.16)$$

With these notations, we can state the Lopatinskii condition which assumes that B is differentiable with respect to u :

Definition 2.7 (Lopatinskii Condition). *For the strictly hyperbolic system (2.1), we say that the boundary conditions (2.3) satisfy the **Lopatinskii condition** if (2.15) is verified, if m_0, m_1 defined in (2.4) satisfy*

$$\begin{aligned} m_0 &= m_+^0, \\ m_1 &= m_-^1, \end{aligned} \quad (2.17)$$

where m_+^0 (resp. m_-^1) is the number of positive eigenvalues of $A(u(0, t))$ (resp. $A(u(1, t))$) and if, for all t in $[0, T]$ and for all u in $\mathcal{U}$,

$$\begin{aligned} B_0(u(0, t)) = 0 &\Rightarrow \ker [\partial_u B_0(u(0, t))] \cap E_+(u(0, t)) = \{0\}, \\ B_1(u(1, t)) = 0 &\Rightarrow \ker [\partial_u B_1(u(1, t))] \cap E_-(u(1, t)) = \{0\}. \end{aligned} \quad (2.18)$$

2.5 Existence of solutions for the MIBVP problem

The objective of this section is to show that, for strictly hyperbolic systems with boundary conditions satisfying Lopatinskii conditions, the non-homogeneous Cauchy problem (2.1)-(2.2)-(2.3) is well-posed, at least for a finite interval of time.

We have the following regularity assumptions:

$$\begin{aligned} A &\in \mathbb{C}^1(\mathcal{U}), \\ B_0 &\in \mathbb{C}^1(\mathcal{U}), \\ B_1 &\in \mathbb{C}^1(\mathcal{U}), \\ u^\natural &\in \mathbb{C}^1([0, 1]) . \end{aligned} \tag{2.19}$$

Under these assumptions, the following Theorem from Li Ta-Tsien and Yu Wen-Ci [49] holds:

Theorem 2.8 (Existence of solution). *Assume that system (2.1) is strictly hyperbolic and that the boundary conditions (2.3) satisfy the Lopatinskii condition. If the initial condition $u^\natural(x)$ satisfy the following "first order" compatibility conditions:*

$$\begin{aligned} B_0(u^\natural(0)) &= 0 , \\ B_1(u^\natural(1)) &= 0 , \\ \nabla_u B_0(u^\natural(0))A(u^\natural(0))\partial_x u^\natural(0) &= 0 , \\ \nabla_u B_1(u^\natural(1))A(u^\natural(1))\partial_x u^\natural(1) &= 0 , \end{aligned} \tag{2.20}$$

then there exists a time instant T_0 such that the Cauchy problem (2.1)-(2.2)-(2.3) admits a unique $\mathbb{C}^1$ solution on $\mathcal{U}$.

Proof. See [49, Theorem 4.2, p.96]. $\square$

Remark 2.9 (Methodology to compute the compatibility conditions). The compatibility conditions are computed by differentiating the boundary condition with respect to time. The $\partial_t u$ terms are then eliminated by using (2.1). For example, let us develop the computation of the compatibility condition for B_1 :

$$\begin{aligned} B_1(u^\natural(0)) &= 0 \\ \partial_t B_1(u^\natural(0)) &= 0 \\ \nabla_u B_1(u^\natural(0))\partial_t u^\natural(0) &= 0 \\ \nabla_u B_1(u^\natural(0))A(u)\partial_x u^\natural(0) &= 0 \end{aligned}$$

Example 2.10 (Compatibility conditions for sluice gates). The compatibility conditions on the derivative of u in the case of the sluice gates (2.8) is written as follows:

$$\begin{aligned} \nabla_u B_0(u^\natural(0))A(u^\natural(0)) &= \\ &\left(\begin{aligned} &\left(V^\natural(0) + 1/2 \frac{K\sqrt{2}g}{\sqrt{g(H_{up} - H^\natural(0))}} \right) V^\natural(0) + H^\natural(0)g \\ &\left(V^\natural(0) + 1/2 \frac{K\sqrt{2}g}{\sqrt{g(H_{up} - H^\natural(0))}} \right) H^\natural(0) + H^\natural(0)V^\natural(0) \end{aligned} \right) \end{aligned}$$

$$\nabla_u B_1(u^\natural(1))A(u^\natural(1)) = \begin{pmatrix} \left(V^\natural(1) - 1/2 \frac{K\sqrt{2}g}{\sqrt{g(H^\natural(1)-H_{down})}} \right) V^\natural(1) + H^\natural(1)g \\ \left(V^\natural(1) - 1/2 \frac{K\sqrt{2}g}{\sqrt{g(H^\natural(1)-H_{down})}} \right) H^\natural(1) + H^\natural(1)V^\natural(1) \end{pmatrix}$$

Note that when the initial state is chosen with null derivatives at the boundaries (flat water and velocity line at the boundaries), those conditions are trivially satisfied.

It is well known that in general, one cannot take $T_0 = T$ because the solution blows up (e.g. [20, 40, 64]). Two types of phenomena can occur before T :

1. u tends to the boundary of $\mathcal{U}$ or blows up to infinity when $t \rightarrow T_0 < T$,
2. the $\mathbb{C}^1$ norm of u blows up to infinity when $t \rightarrow T_0 < T$.

More precisely, the following theorem can be deduced from Theorem 2.8:

Theorem 2.11 (Blow up of solutions). *Under the assumption of Theorem 2.8, if the Cauchy problem (2.1)-(2.2)-(2.3) has a unique $\mathbb{C}^1$ solution u in the open domain $[0, 1] \times [0, T_0[$ but does not have a solution of class $\mathbb{C}^1$ in $[0, 1] \times [0, T_0]$, then at least one of the two following properties are fulfilled:*

$$\lim_{t \rightarrow T_0^-} \max\{d(u(x, t)) : x \in [0, 1]\} = +\infty, \quad (2.21)$$

$$\lim_{t \rightarrow T_0^-} \|u(\cdot, t)\|_{\mathbb{C}^1([0, 1])} = +\infty, \quad (2.22)$$

where the application $d(u)$ is defined as

$$d(u) = \|u\| + \frac{1}{\text{distance}(u, \mathbb{R}^n - \mathcal{U})}, \quad (2.23)$$

and

$$\frac{1}{\text{distance}(u, \emptyset)} = 0. \quad (2.24)$$

Remark 2.12. The case of (2.22) usually arises when the derivative of u with respect to x goes to infinity. This phenomenon is commonly called a shock or jump

2.6 Riemann invariants

2.6.1 Description

For some hyperbolic systems, there exists a bijection $\xi(u)$ such that (2.1) can be transformed into a system of coupled transport equations

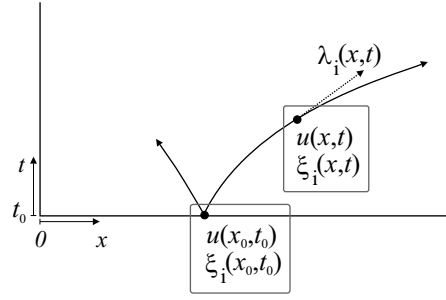


Fig. 2.1. The representation of the characteristic curves. While it is unlikely that $u(x_0, t_0)$ equals $u(x, t)$, the Riemann invariant ξ remains constant along the curve and thus, $\xi_i(x_0, t_0) = \xi_i(x, t)$.

$$\partial_t \xi_i(x, t) + \lambda_i(\xi(x, t)) \partial_x \xi_i(x, t) = 0 \quad \text{for } i = 1, \dots, n$$

One can observe that the quantity ξ_i can be seen as the total derivative $d\xi_i/dt$ of $\xi_i(x, t)$ at point (x, t) of the plane, along a curve defined by the differential equation

$$\frac{dx}{dt} = \lambda_i(\xi(x, t)) . \quad (2.25)$$

This curve is called the *characteristic curve* and the solution $\xi_i(x, t)$ is called the *characteristic solution*. Since $d\xi_i/dt = 0$ along the characteristic curve, it follows that ξ_i is constant (or invariant) along the characteristic curve. This property, illustrated in Figure 2.1, is very important and, in fact, it is a crucial tool for the results that will be developed in the following chapters.

The quantities ξ_i are called **Riemann invariants** in honor of the seminal work that Riemann did about the isothermal Euler equation in [8].

Remark 2.13. There is a straightforward physical interpretation for the characteristic curves in the context of Shallow water equations. When one throws a rock in a liquid, one can see that two waves travelling backward and forward of the stream as the negative ξ_1 and the positive ξ_2 Riemann invariant. Both waves are travelling at the corresponding characteristic velocity λ_1 and λ_2 and "carrying" the "information" of the rock hitting the water. It is important to note that the influence of the rock on the water surface is limited to the position of the two travelling waves. This is an important feature of the hyperbolic system which is usually referred as the *cone* of influence.

2.6.2 Finding the invariants

In this section, we describe the classic method for computing the Riemann invariants of a system.

Assume that (2.1) is strictly hyperbolic and introduce an eigensystem of $A(u)$ composed of:

- the set of real eigenvalues: $\lambda_1(u) < \dots < \lambda_n(u)$,
- the set $(l_1(u), \dots, l_n(u))$ of left eigenvectors satisfying:

$$l_k(u)A(u) = \lambda_k(u)l_k(u), \quad \text{for } k = 1, \dots, n,$$

- the set $(r_1(u), \dots, r_n(u))$ of right eigenvectors satisfying:

$$A(u)r_k(u) = \lambda_k(u)r_k(u), \quad \text{for } k = 1, \dots, n,$$

- the following normalization: $(k, p = 1, \dots, n)$

$$l_k(u)r_p(u) = \delta_{kp}.$$

Remark 2.14. From the implicit function theorem (see Theorem A.1), if the entries of $A(u)$ are Lipschitz continuous, or k -times continuously differentiable as functions of u , then the same is true of the functions $\lambda_i, l_i, r_i, i = 1, \dots, n$.

The classical notion of Riemann invariants is formalized in the following definition (see e.g. [64, 31]):

Definition 2.15. A function ξ from $\mathcal{U}$ into $\mathbb{R}$ is a Riemann invariant for (2.1) provided that for every $u \in \mathcal{U}$, the vector $\nabla_u \xi(u)$ is a left eigenvector of $A(u) : \exists \lambda : \mathcal{U} \mapsto \mathbb{R}$ such that

$$\nabla_u \xi(u)A(u) = \lambda(u)\nabla_u \xi(u) \quad (2.26)$$

where $\nabla_u = (\partial_{u_1}, \dots, \partial_{u_n})$.

In the case where one can find n independent Riemann invariants $\xi_1, \dots, \xi_n$, it is possible to diagonalize (2.1):

$$\partial_t \xi_k(u) + \lambda_k(u) \partial_x \xi_k(u) = 0. \quad (2.27)$$

The above equation can be verified by pre-multiplying (2.1) with $\nabla_u \xi(u)$:

$$\begin{aligned} & \nabla_u \xi(u) \partial_t u + \nabla_u \xi(u) A(u) \partial_x u = 0 \\ \Leftrightarrow & \nabla_u \xi(u) \partial_t u + \lambda(u) \nabla_u \xi(u) \partial_x u = 0 \\ \Leftrightarrow & \partial_t \xi(u) + \lambda(u) \partial_x \xi(u) = 0 \end{aligned}$$

Remark 2.16. The 2×2 quasi-linear systems always have Riemann invariants. Moreover, such systems are generalized by *rich* systems (see [64, Volume II, Chapter 12]. Rich systems generalize the class of 2×2 systems while preserving the essential properties: diagonalisation using Riemann invariants and infinite dimension of the space of entropies (see Section 5.2).

Example 2.17 (Riemann invariants for SWE). The Riemann invariants for the SWE (2.7) and (2.6) is rewritten as follows:

$$A(u) = \begin{pmatrix} V - \sqrt{gH} & 0 \\ 0 & V + \sqrt{gH} \end{pmatrix}, \quad (2.28)$$

$$\begin{aligned}\xi_1(H, V) &= V - 2\sqrt{gH} \\ \xi_2(H, V) &= V + 2\sqrt{gH} .\end{aligned}\quad (2.29)$$

The inverse transformation can also be computed from (2.29):

$$\begin{aligned}H &= \frac{(\xi_2 - \xi_1)^2}{16g} \\ V &= \frac{\xi_1 + \xi_2}{2}\end{aligned}\quad (2.30)$$

2.6.3 Characteristic form

We introduce matrix-vector notations in order to simplify the description of MIBVP (2.1)-(2.2)-(2.3) in the characteristic form. We also assume that the system (2.1) has strictly non-zero eigenvalues.

Let Λ , the vector of characteristic velocities,

$$\Lambda = \text{diag}(\lambda_1, \dots, \lambda_n) ,$$

where the λ_i are the real eigenvalues of A in (2.1), $\lambda_i(0) < 0$, $\forall i \in \{1, \dots, m\}$ and $\lambda_i(0) > 0$, $\forall i \in \{m+1, \dots, n\}$.

Let $\Lambda_-: \mathbb{R}^n \rightarrow \mathbb{R}^{m \times m}$ and $\Lambda_+: \mathbb{R}^n \rightarrow \mathbb{R}^{(n-m) \times (n-m)}$ two functions defined by

$$\Lambda_- = \text{diag}(\lambda_1, \dots, \lambda_m) , \quad \Lambda_+ = \text{diag}(\lambda_{m+1}, \dots, \lambda_n) .$$

such that

$$\Lambda = \begin{pmatrix} \Lambda_- & \\ & \Lambda_+ \end{pmatrix}$$

We also define the vector of Riemann invariants

$$\boldsymbol{\xi} = (\xi_1, \dots, \xi_n)^T \in \mathbb{R}^n$$

We separate $\boldsymbol{\xi}$ into a vector of Riemann invariants associated to negative characteristic velocities, also known as *backward invariants*,

$$\boldsymbol{\xi}_- = (\xi_1, \dots, \xi_m)^T \in \mathbb{R}^m ,$$

and a vector of Riemann invariants associated to positive characteristic velocities, also known as *forward invariants*,

$$\boldsymbol{\xi}_+ = (\xi_{m+1}, \dots, \xi_n)^T \in \mathbb{R}^{n-m} .$$

Under this notations, (2.27) rewrites

$$\frac{\partial \boldsymbol{\xi}}{\partial t} + \Lambda(\boldsymbol{\xi}) \frac{\partial \boldsymbol{\xi}}{\partial x} = 0 , \quad (2.31)$$

and the initial condition

$$\boldsymbol{\xi}(\cdot, 0) = \boldsymbol{\xi}^\natural \quad (2.32)$$

where $\boldsymbol{\xi}^\natural(x) = (\xi_1^\natural(x), \dots, \xi_n^\natural(x))$.

2.7 Boundary Stabilization of MIBVP

For Mixed Initial-Boundary Value Problem, the classical solution must blow up in a finite time, provided that the initial data is small (see Theorem 2.11). The situation changes, however, if there is a so-called *boundary damping*.

This section presents results that were first introduced by Greenberg and Li in [34] for the Euler equations and later by Li to any first order quasilinear hyperbolic systems in [48, Chapter 5].

In the following, we shall consider the *Local Asymptotic Stability* of systems (2.1).

This stability concept is formally defined as follows: The MIBVP (2.31)-(2.32) is said *Locally Asymptotically Stable* if, for sufficiently small initial conditions $\xi^\natural$, the state ξ decays asymptotically to zero. The technical definition is give as follows:

Definition 2.18 (Local Asymptotic Stability). *The MIBVP (2.1)-(2.2)-(2.3) in the characteristic form (2.31)-(2.32) is said to be Locally Asymptotically Stable if there exists $\varepsilon_0 > 0$ so small that for any fixed $\varepsilon (0 < \varepsilon \leq \varepsilon_0)$ there exists $\delta = \delta(\varepsilon) > 0$ such that if*

$$\|\xi_i^\natural\|_{\mathbb{C}^1} \leq \delta, \quad i = 1, \dots, n, \quad (2.33)$$

then the problem (2.1)-(2.2)-(2.3) admits a unique $\mathbb{C}^1$ solution $\xi = \xi(x, t)$ on the domain $\mathcal{U}$; moreover,

$$\|\xi(t, \cdot)\|_{\mathbb{C}^1} \leq K\varepsilon \exp^{-bt}, \quad \forall t \geq 0, \quad (2.34)$$

where $b > 0$ is any fixed small positive number, K is a positive constant.

2.7.1 Li's Theorem (94)

We consider the MIBVP (2.1)-(2.2)-(2.3) in the characteristic form (2.31)-(2.32) together with boundary conditions of the form:

$$\begin{aligned} \xi_-(1, t) &= \mathbf{G}_2(\xi_+(1, t)) \\ \xi_+(0, t) &= \mathbf{G}_1(\xi_-(0, t)) \end{aligned} \quad (2.35)$$

where $\mathbf{G}_1, \mathbf{G}_2: \mathbb{R}^m \rightarrow \mathbb{R}^m$ are two continuously differentiable functions in a neighborhood of $0 \in \mathbb{R}^m$ satisfying $\mathbf{G}_i(0) = 0$.

Remark 2.19. One cannot usually compute the explicit expression of $\mathbf{G}_1$ and $\mathbf{G}_2$ from (2.3) because ξ is a nonlinear change of variable that makes the form of $\mathbf{G}_1$ and $\mathbf{G}_2$ analytically complex. Nevertheless, using the Local Inverse Theorem (see Theorem A.1), you can compute the gradient $\nabla \mathbf{G}_1$ and $\nabla \mathbf{G}_2$ in a neighborhood of zero:

$$\begin{aligned}\nabla \mathbf{G}_1 &= - \left(\nabla_{\xi_+(0,t)} B_0(0) \right)^{-1} \left(\nabla_{\xi_-(0,t)} B_0(0) \right) \\ \nabla \mathbf{G}_2 &= - \left(\nabla_{\xi_-(1,t)} B_1(0) \right)^{-1} \left(\nabla_{\xi_+(1,t)} B_1(0) \right)\end{aligned}\quad (2.36)$$

We shall usually refer to $\nabla_{[\xi_-(1,t), \xi_+(0,t)]}$ as the Jacobian with respect to the vector $[\xi_-(1,t), \xi_+(0,t)]$:

$$\nabla_{[\xi_-(1,t), \xi_+(0,t)]} B = \begin{pmatrix} \partial_{xi_1(1,t)} B_1 \cdots \partial_{xi_m(1,t)} B_1 & \partial_{xi_{m+1}(0,t)} B_1 \cdots \partial_{xi_n(0,t)} B_1 \\ \vdots & \vdots \\ \partial_{xi_1(1,t)} B_n \cdots \partial_{xi_m(1,t)} B_n & \partial_{xi_{m+1}(0,t)} B_n \cdots \partial_{xi_n(0,t)} B_n \end{pmatrix} \quad (2.37)$$

We define the compatibility condition for (2.35)-(2.32):

Definition 2.20 (Compatibility conditions for (2.35)-(2.32)).

The function $\xi^\# \in C^1([0, L]; \mathbb{R}^n)$ satisfies the compatibility condition (C_1) if

$$\begin{aligned}\xi_-^\#(1) &= \mathbf{G}_2(\xi_+^\#(1)) , \\ \xi_+^\#(0) &= \mathbf{G}_1(\xi_-^\#(0)) , \\ \Lambda_-(\xi^\#(1)) \partial_x \xi_-^\#(1) &= \nabla \mathbf{G}_2(\xi_+^\#(1)) \Lambda_+(\xi^\#(1)) \partial_x \xi_+^\#(1) \\ \Lambda_+(\xi^\#(0)) \partial_x \xi_+^\#(0) &= \nabla \mathbf{G}_1(\xi_-^\#(0)) \Lambda_-(\xi^\#(0)) \partial_x \xi_-^\#(0)\end{aligned}$$

We give the following hypotheses:

(H1) On the domain under consideration, λ_i , a_i and $\xi^\natural \in \mathbb{C}^1$ for $i = 1, \dots, n$, $\mathbf{G}_0$ and $\mathbf{G}_1 \in \mathbb{C}^2$.

(H2) The system (2.31) is strictly hyperbolic at the origin, with non-zero velocities, namely

$$\lambda_r(0) < 0 < \lambda_s(0), \quad r = 1, \dots, m, \quad s = m+1, \dots, n. \quad (2.38)$$

Let the gradient matrix $\nabla \mathbf{G}$, at the origin be defined as:

$$\nabla \mathbf{G} = \begin{pmatrix} \mathbf{0} & \nabla_{\xi_+(1)} G_1 \\ \nabla_{\xi_-(0)} G_0 & \mathbf{0} \end{pmatrix}. \quad (2.39)$$

By denoting $\rho(A)$ the spectral radius of a matrix A , we can then state the following stability condition theorem:

Theorem 2.21 (Boundary Asymptotic Stability for MIBVP). Under hypotheses $(C1)$, $(H1)$, $(H2)$, if

$$\rho(\text{abs}(\nabla \mathbf{G}(0))) < 1, \quad (2.40)$$

there exists $\varepsilon_0 > 0$ so small that for any fixed $\varepsilon (0 < \varepsilon \leq \varepsilon_0)$ there exists $\delta = \delta(\varepsilon) > 0$ such that if

$$\|\xi_i^h\|_{\mathbb{C}^1} \leq \delta, \quad i = 1, \dots, n, \quad (2.41)$$

then the problem (2.1)-(2.2)-(2.3) admits a unique global $\mathbb{C}^1$ solution $\xi = \xi(x, t)$ on the domain $\mathcal{U}$; moreover,

$$\|\xi(t, \cdot)\|_{\mathbb{C}^1} \leq K\varepsilon \exp^{-bt}, \quad \forall t \geq 0, \quad (2.42)$$

where $b > 0$ is any fixed small positive number, K is a positive constant and

$$\|\xi(t, \cdot)\|_{\mathbb{C}^1} = \|\xi(t, \cdot)\|_{\mathbb{C}^0} + \|\partial_x \xi(t, \cdot)\|_{\mathbb{C}^0}. \quad (2.43)$$

Proof. See [48, Theorem 1.1 and Theorem 1.3, Chapter 5] . $\square$

The proof of Theorem 2.21 is a proof based on characteristics. This technique will be explained later and we show that it can also be used to design stabilizing control laws for such systems.

Remark 2.22. Let us point out that with the definition of $\|A\|_{min}$ given in [48, p. 170] one has $\|A\|_{min} = \|\text{abs}(A)\|_{min} = \rho(\text{abs}(A))$. See Lemma 2.4 in [48, p. 146].

Remark 2.23. The Theorem of Li in [48] is a more general version of the Theorem 2.21 because it applies to MIBVP (2.1)-(2.2)-(2.3) with a non-homogenous form:

$$\partial_t \xi_i + \lambda_i(\xi) \partial_x \xi_i = \mu_i(\xi), \quad i = \{1, \dots, n\} \quad (2.44)$$

where $\mu_i \in \mathbb{C}^2$ and

$$\mu_i(0) = \frac{\partial \mu_i(0)}{\partial \xi_j} = 0, \quad i, j = 1, \dots, n. \quad (2.45)$$

2.7.2 Illustration of the proof on a 2×2 system

As mentioned above, Li uses a proof based on tracking the Riemann invariants along the characteristic trajectories for Theorem 2.21. This technique was first introduced in his paper with Greenberg [34]. In this section, we shall briefly illustrate the main idea of the proof on a simple 2×2 hyperbolic system.

Remark 2.24. According to [64, Volume II, Section 9.3], a 2×2 system is a system with two state variables $u_1(x, t)$, $u_2(x, t)$ and two independent variables (x, t) . We adopt this terminology in the thesis.

Let us consider a 2×2 linear system, as depicted in Figure 2.2, under the characteristic form

$$\partial_t \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} + \begin{pmatrix} \lambda_1 & 0 \\ 0 & \lambda_2 \end{pmatrix} \partial_x \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (2.46)$$

where $\lambda_1 < 0 < \lambda_2$.

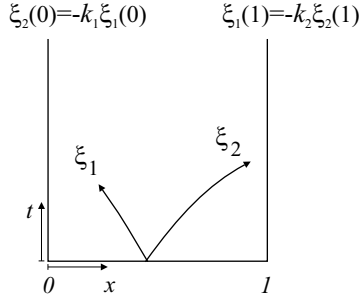


Fig. 2.2. Representation of the trajectories of ξ_1 and ξ_2 in the (x, t) plane.

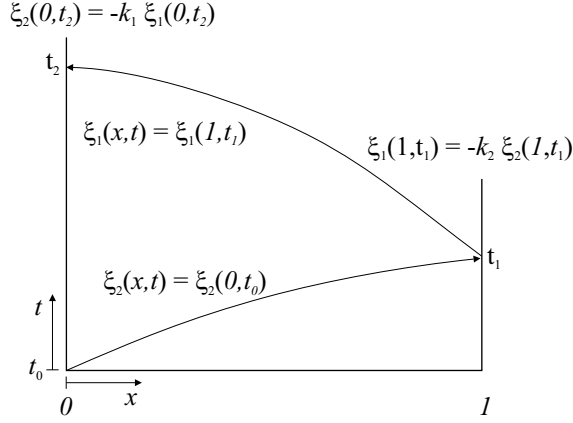


Fig. 2.3. Illustration of Theorem 2.21. The curves represent characteristic trajectories of the system.

We choose the following linear boundary conditions:

$$\begin{aligned}\xi_1(1, t) &= -k_2\xi_2(1, t), \\ \xi_2(0, t) &= -k_1\xi_1(0, t),\end{aligned}\tag{2.47}$$

and an initial condition ξ_1^h, ξ_2^h that satisfies $(C1) - (H1) - (H2)$.

Let us now follow a Riemann invariant along its characteristic curve and across the different rebounds at the boundaries. This process is illustrated on the Figure 2.3.

1. We start by choosing a value in the initial condition ξ_2^h at any position $x_0 \in [0, 1]$ and at time $t_0 = 0$.
2. By the invariance property, there exists obviously a time instant $t_1 > t_0$ such that $\xi_2(1, t_1) = \xi_2(x_0, t_0)$.
3. Suppose now that we are able to apply a boundary control at $x = 1$ such that $\xi_1(1, t_1) = -k_2\xi_2(1, t_1)$.
4. Obviously, there exists a time instant $t_2 > t_1$ such that $\xi_1(0, t_2) = \xi_1(1, t_1)$.

5. We now apply a boundary control at $x = 0$ such that $\xi_2(0, t_2) = -k_1 \xi_1(0, t_2)$ and so on.

This clearly implies that, for any arbitrary t_0 there is a monotonically increasing sequence of time instants $t_i, i = 0, 1, 2, \dots$ such that

$$\begin{aligned}\xi_2(0, t_{2j}) &= (k_1 k_2)^j \xi_2(0, t_0), \\ \xi_1(1, t_{2j+1}) &= (k_1 k_2)^j \xi_1(1, t_1), \quad j = 1, 2, \dots\end{aligned}$$

If we choose k_1 and k_2 such that

$$0 \leq |k_1 k_2| < 1. \quad (2.48)$$

This allows to understand why the boundary control laws (2.47) will guarantee the convergence of $\xi_1(x, t)$ and $\xi_2(x, t)$ to zero. As a matter of fact, if we apply Theorem 2.21 to the MIBVP (2.46)-(2.47), we see that (2.48) and (2.40) are equivalent.

Remark 2.25. With equations (2.47) and (2.48), we have all the ingredients to design stabilizing boundary control laws (we emphasize that the control is located only at the boundaries). If the k_1, k_2 coefficients are the control variables, then the design objective will be to make them satisfy $k_1 k_2 < 1$.

2.7.3 Extension of Li's Theorem

In [41], we have extended Theorem 2.21 by considering a more general form of the boundary condition (2.35) as follows:

$$\begin{pmatrix} \xi_-(1, t) \\ \xi_+(0, t) \end{pmatrix} = \mathbf{G} \begin{pmatrix} \xi_-(0, t) \\ \xi_+(1, t) \end{pmatrix}, \quad (2.49)$$

In (2.35), $\xi_-(1, t)$ is a function of $\xi_+(1, t)$ only while in (2.49) $\xi_-(1, t)$ is a function of *both* $\xi_+(1, t)$ and $\xi_-(0, t)$. Similarly, $\xi_+(0, t)$ is a function of both $\xi_-(0, t)$ and $\xi_+(1, t)$.

Remark 2.26. The form of (2.49) implies that the boundary conditions (2.3) are generalized to

$$B(u(0, t), u(1, t)) = 0. \quad (2.50)$$

This boundary law can also be expressed in Riemann invariants as follows:

$$B(\xi_-(0, t), \xi_+(1, t), \xi_-(1, t), \xi_+(0, t)) = 0 \quad (2.51)$$

Similarly to Remark 2.19, one cannot usually compute the explicit expression of $\mathbf{G}$ from (2.50). Again, using the Local Inverse Theorem (see Theorem A.1), you can compute the gradient $\nabla \mathbf{G}$ in a neighborhood of zero:

$$\nabla \mathbf{G}(0) = - \left(\nabla_{\xi_-(1, t), \xi_+(0, t)} B(0, 0) \right)^{-1} \left(\nabla_{\xi_-(0, t), \xi_+(1, t)} B(0, 0) \right) \quad (2.52)$$

We define the compatibility condition for (2.49)-(2.32):

Definition 2.27 (Compatibility conditions for (2.49)-(2.32)).

The function $\xi^\# \in C^1([0, L]; \mathbb{R}^n)$ satisfies the compatibility condition (C_2) if

$$\begin{pmatrix} \xi_-^\#(1) \\ \xi_+^\#(0) \end{pmatrix} = \mathbf{G} \begin{pmatrix} \xi_-^\#(0) \\ \xi_+^\#(1) \end{pmatrix},$$

$$\begin{pmatrix} \Lambda_- (\xi^\#(1)) \frac{\partial \xi_-^\#}{\partial x}(1) \\ \Lambda_+ (\xi^\#(0)) \frac{\partial \xi_+^\#}{\partial x}(0) \end{pmatrix} = \nabla \mathbf{G} \begin{pmatrix} \xi_-^\#(0) \\ \xi_+^\#(1) \end{pmatrix}$$

$$\begin{pmatrix} \Lambda_- (\xi^\#(0)) \frac{\partial \xi_-^\#}{\partial x}(0) \\ \Lambda_+ (\xi^\#(1)) \frac{\partial \xi_+^\#}{\partial x}(1) \end{pmatrix}.$$

We now give the extension of the local asymptotic stability, defined in Definition 2.18, of the Theorem 2.21:

Theorem 2.28 (Local Asymptotic Stability). *If*

$$\rho(\text{abs}(\nabla \mathbf{G}(0))) < 1, \quad (2.53)$$

then there exists $\varepsilon > 0$, $\mu > 0$ and $C > 0$ such that, for every $\xi^\# \in C^1([0, L]; \mathbb{R}^n)$ satisfying condition (C_2) and such that

$$\|\xi^\#\|_{C^1([0, L])} \leq \varepsilon,$$

there exists one and only one function $\xi \in C^1([0, L] \times [0, +\infty); \mathbb{R}^n)$ satisfying (2.31), (2.49) and

$$\xi(x, 0) = \xi^\#(x), \quad \forall x \in [0, L]. \quad (2.54)$$

Moreover, this function ξ satisfies

$$\|\xi(\cdot, t)\|_{C^1([0, L])} \leq C e^{-\mu t} \|\xi^\#\|_{C^1([0, L])}, \quad \forall t \geq 0. \quad (2.55)$$

Remark 2.29. The boundary condition in u corresponding to (2.49) takes a form slightly different from (2.3):

$$B(u(0, t), u(1, t)) = 0 \quad (2.56)$$

Proof of Theorem 2.28

We explain how this proof is a special case of the proof of Theorem 2.21 if the boundary condition (2.49) has the following particular form

$$\begin{pmatrix} \xi_-(1, t) \\ \xi_+(0, t) \end{pmatrix} = \begin{pmatrix} \mathbf{G}_1(\xi_+(1, t)) \\ \mathbf{G}_2(\xi_-(0, t)) \end{pmatrix}, \quad (2.57)$$

where $\mathbf{G}_1 : \mathbb{R}^{n-m} \rightarrow \mathbb{R}^n$ and $\mathbf{G}_2 : \mathbb{R}^m \rightarrow \mathbb{R}^n$ are of class C^1 on neighborhoods of $0 \in \mathbb{R}^{n-m}$ and of $0 \in \mathbb{R}^m$ respectively.

But one can use 2.21 to prove Theorem 2.28 even if (2.57) does not hold by doubling the size of the state as follows. Consider the hyperbolic system

$$\frac{\partial \tilde{\xi}}{\partial t} + \tilde{\Lambda}(\tilde{\xi}) \frac{\partial \tilde{\xi}}{\partial x} = 0, \quad (2.58)$$

with

$$\tilde{\xi} = (\xi_{1-}^T, \xi_{2-}^T, \xi_{1+}^T, \xi_{2+}^T)^T,$$

where $\xi_{1-} \in \mathbb{R}^m$, $\xi_{2-} \in \mathbb{R}^{n-m}$, $\xi_{1+} \in \mathbb{R}^{n-m}$, $\xi_{2+} \in \mathbb{R}^m$ and $\tilde{\Lambda}: \mathbb{R}^{2n} \rightarrow \mathbb{R}^{2n \times 2n}$ is defined by

$$\tilde{\Lambda}(\tilde{\xi}) = \text{diag} \begin{pmatrix} \Lambda_-((\xi_{1-}^T, \xi_{1+}^T)^T) \\ -\Lambda_+((\xi_{2+}^T, \xi_{2-}^T)^T) \\ \Lambda_+((\xi_{1-}^T, \xi_{1+}^T)^T) \\ -\Lambda_-((\xi_{2+}^T, \xi_{2-}^T)^T) \end{pmatrix}.$$

The boundary condition for (2.58) is defined by

$$\begin{pmatrix} \xi_{1-}(1, t) \\ \xi_{2-}(1, t) \end{pmatrix} = \mathbf{G} \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} \xi_{1+}(1, t) \\ \xi_{2+}(1, t) \end{pmatrix}, \\ \begin{pmatrix} \xi_{1+}(0, t) \\ \xi_{2+}(0, t) \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \mathbf{G} \begin{pmatrix} \xi_{1-}(0, t) \\ \xi_{2-}(0, t) \end{pmatrix}.$$

This boundary condition can be written in the following form

$$\begin{pmatrix} \tilde{\xi}_-(1, t) \\ \tilde{\xi}_+(0, t) \end{pmatrix} = \tilde{\mathbf{G}} \begin{pmatrix} \tilde{\xi}_+(1, t) \\ \tilde{\xi}_-(0, t) \end{pmatrix} = \begin{pmatrix} \tilde{\mathbf{G}}_1(\tilde{\xi}_+(1, t)) \\ \tilde{\mathbf{G}}_2(\tilde{\xi}_-(0, t)) \end{pmatrix}, \quad (2.59)$$

with

$$\tilde{\xi}_- = (\xi_{1-}^T, \xi_{2-}^T)^T, \quad \tilde{\xi}_+ = (\xi_{1+}^T, \xi_{2+}^T)^T, \quad (2.60)$$

$$\tilde{\mathbf{G}}_1 = \mathbf{G} \circ \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix}, \quad \tilde{\mathbf{G}}_2 = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \circ \mathbf{G}. \quad (2.61)$$

In particular the boundary condition for $\tilde{\xi}$ has the special form required by Theorem 1.3 in [48, Chapter 5] and

$$\rho(\text{abs}(\nabla \tilde{\mathbf{G}}(0))) = \rho(\text{abs}(\nabla \mathbf{G}(0)))^2. \quad (2.62)$$

Let $\xi^\# \in C^1([0, L]; \mathbb{R}^n)$ satisfying condition (C) and such that $|\xi^\#|_{C^1([0, L])}$ is small enough. We choose as initial condition for $\tilde{\xi}$ at $t = 0$,

$$\begin{aligned} \xi_{1-}^\#(x) &= \xi_{1-}^\#(x), & \xi_{2-}^\#(x) &= \xi_{1+}^\#(1-x), \\ \xi_{1+}^\#(x) &= \xi_{1+}^\#(x), & \xi_{2+}^\#(x) &= \xi_{2-}^\#(1-x). \end{aligned} \quad (2.63)$$

One easily sees that $\tilde{\xi}^\# := (\xi_{1-}^{\#T}, \xi_{2-}^{\#T}, \xi_{1+}^{\#T}, \xi_{2+}^{\#T})^T$ satisfies the compatibility condition associated to (2.58) and (2.59). Hence there exists a unique C^1 -solution $\tilde{\xi}$ of (2.58) and (2.59) such that

$$\tilde{\xi}(x, 0) = \tilde{\xi}^\#(x). \quad (2.64)$$

Let

$$\tilde{\xi}^*(x, t) = \begin{pmatrix} \xi_{2+}(1-x, t)^T \\ \xi_{1+}(1-x, t)^T \\ \xi_{2-}(1-x, t)^T \\ \xi_{1-}(1-x, t)^T \end{pmatrix}.$$

Then, as one easily checks, $\tilde{\xi}^*$ satisfies as $\tilde{\xi}$ the hyperbolic system (2.58), the boundary condition (2.59) and the initial condition (2.64). Hence by the uniqueness of the C^1 -solution of the Cauchy problem associated to (2.58) and (2.59), one has

$$\tilde{\xi}^* = \tilde{\xi}.$$

In particular

$$\begin{aligned} \xi_{1-}(x, t) &= \xi_{2+}(1-x, t), \\ \xi_{1+}(x, t) &= \xi_{2-}(1-x, t). \end{aligned} \quad (2.65)$$

Hence, if

$$\xi_-(x, t) := \xi_{1-}(x, t), \quad \xi_+(x, t) := \xi_{1+}(x, t),$$

then $\xi = (\xi_-^T, \xi_+^T)^T$ satisfies (2.31), (2.49) and (2.54).

Conversely, if $\xi = (\xi_-^T, \xi_+^T)^T$ satisfies (2.31), (2.49) and (2.54), then $\tilde{\xi}$ defined by

$$\begin{aligned} \xi_{1-}(x, t) &:= \xi_-(x, t), & \xi_{1+}(x, t) &:= \xi_+(x, t), \\ \xi_{2+}(x, t) &:= \xi_-(1-x, t), & \xi_{2-}(x, t) &:= \xi_+(1-x, t), \end{aligned}$$

satisfies the hyperbolic system (2.58), the boundary condition (2.59) and the initial condition (2.64). Hence, see also (2.62), Theorem 2.28 for the hyperbolic system (2.31) and the boundary condition (2.49) is a consequence of (the proof of) Theorem 2.21 for the hyperbolic system (2.58) and the boundary condition (2.59).

2.8 Application to control

In this section, we shall see how the stability condition of Theorem 2.28 can be used to design stabilizing boundary -i.e. the control variables are located at the boundaries - control laws for the system (2.1)-(2.56). In fact, it is crucial to derive control laws that depend on the boundary state only, since it is usually impossible to measure or observe the entire state of the system.

The control law design problem is stated as follows:

Definition 2.30 (Control Design Problem). *Design a control law w that is a function that depends continuously on the state at the boundaries:*

$$w = w(u(0, t), u(1, t)) \quad (2.66)$$

and that stabilizes the system (2.1) with the boundary conditions

$$B(u(0, t), u(1, t), w(u(0, t), u(1, t))) = 0 . \quad (2.67)$$

to the set point $\bar{u}$.

Note that (2.67) is a modified version of (2.56) where the control variable w has been added to the list of arguments. We can also express the control problem in terms of Riemann invariants:

Definition 2.31 (Control Design Problem in terms of Riemann invariants). *Design a control law w that is a function that depends continuously on the ξ at the boundaries:*

$$w = w(\xi(0, t), \xi(1, t)) \quad (2.68)$$

and that stabilizes the system (2.31) with the boundary conditions

$$B(\xi(0, t), \xi(1, t), w(\xi(0, t), \xi(1, t))) = 0 . \quad (2.69)$$

to the set point 0.

As in Remark 2.19, the gradient matrix $\nabla \mathbf{G}(0)$ of (2.67) can be computed using the Local Inverse Theorem,

$$\nabla \mathbf{G}(0, 0, w(0, 0)) = - \left(\nabla_{\xi_-(1, t), \xi_+(0, t)} B(0, 0, w(0, 0)) \right)^{-1} \left(\nabla_{\xi_-(0, t), \xi_+(1, t)} B(0, 0, w(0, 0)) \right) . \quad (2.70)$$

Therefore, our objective is to design a feedback control law

$$w(\xi(0, t), \xi(1, t))$$

such that $\nabla \mathbf{G}(0, 0, w(0, 0))$ satisfies the stability condition (2.53):

$$\rho(\text{abs}(\nabla \mathbf{G}(0, 0, w(0, 0)))) < 1 .$$

Example 2.32 (Shallow water equations). In (2.9), we have given the discharge relationship of the sluice gates. These equations can be rewritten under the form of (2.67) where the control action w is the opening of the gate:

$$B(u(0, t), u(1, t), w) = \left(\frac{H(0, t)V(0, t) - K_v(w)\sqrt{2g(H_{up} - H(0, t))}}{H(1, t)V(1, t) - K_v(w)\sqrt{2g(H(1, t) - H_{down})}} \right) \quad (2.71)$$

where $K_v(w)$ is a positive non linear relation that has been identified by experiment. For the sake of simplicity, let us assume that this relation is linear, i.e. $K_v(w) = K_v w$. Using (2.30), one can express the above equation in terms of Riemann invariants. Below is the formulation of B_0 in terms of Riemann invariants truncated to square terms in ξ :

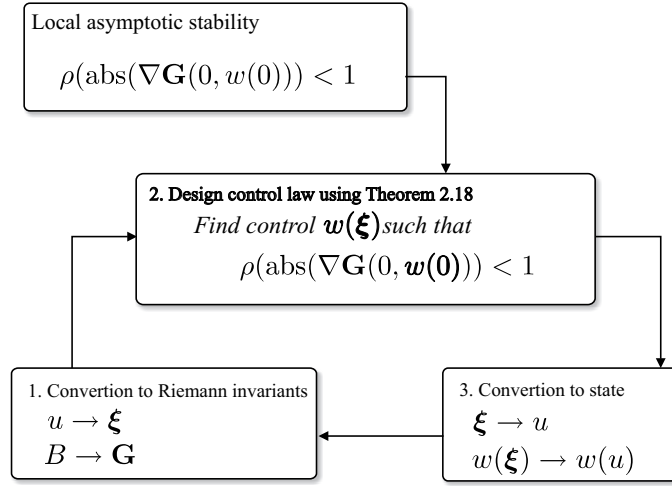


Fig. 2.4. Using Theorem 2.28 to design control laws

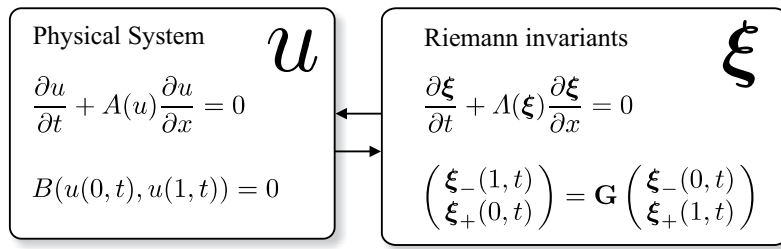
$$\begin{aligned}
 B_0(\xi_1(0, t), \xi_2(0, t), w) = \\
 -K_v w \sqrt{2} \sqrt{g H_{up} - 1/16 \xi_1(0, t)^2 - 1/8 \xi_1(0, t) \xi_2(0, t) - 1/16 \xi_2(0, t)^2} \\
 + O(\xi^3) .
 \end{aligned}
 \tag{2.72}$$

We see the fact that the conversion of the boundary laws into Riemann invariants increases their "analytical complexity". This is due to the nonlinearity of the transformation. This fact motivates the study of design techniques in the next Chapter to keep this "complexity hit" as low as possible.

2.9 Conclusion

In this Chapter, we have done a short introduction on conservation laws and more generally on MIBVP. We have introduced the concept of Riemann invariant as depicted in Figure 2.5. This formulation has a crucial importance in the control design because the stability results, such as Theorem 2.28 are expressed in that formulation. In fact, the design of a controller for a quasi-linear system usually involves a round trip from the state u formulation to the Riemann invariant ξ formulation.

Finally, we show how the control at the boundaries can be used to stabilize the system through the use of the stability results shown in Theorem 2.28.

**Fig. 2.5.** MIBVP in u and ξ coordinates.

Stabilization of a cascade of 2×2 systems

In this chapter, we provide a control design technique that keeps the stability condition simple. To do so, the constructive proof based on Riemann invariants is formalized by introducing the *invariant graph*, which is a weighted directed graph that represents an hyperbolic system. By taking advantage of the topological properties of the *invariant graph*, a stabilizing control design technique is built.

Taking advantage of the structure of a system through graph theory has already been done in the field of mine ventilation networks in [43].

Two graph representations of an hyperbolic system, namely the *invariant graph* and the *flux graph*, are given in Section 3.3. The control design is presented in Section 3.4. In Section 3.5, an application of this design to a cascade of 2×2 systems is given.

3.1 Cascade of 2×2 Hyperbolic Systems

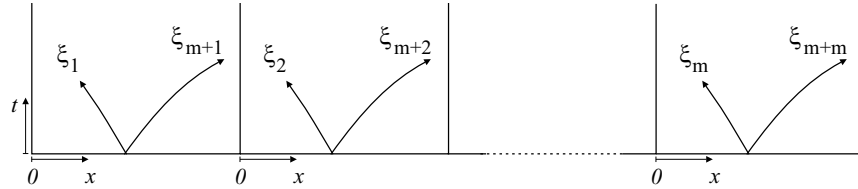


Fig. 3.1. A cascade of m 2×2 hyperbolic systems in the (x, t) plane.

In this section, we consider a cascade of m 2×2 hyperbolic systems that are connected in series by *linear* boundary conditions as depicted in Figure 3.1. The system in characteristic form is written as follows:

$$\partial_t \xi(x, t) + \Lambda(\xi(x, t)) \partial_x \xi(x, t) = 0, \quad (3.1)$$

$$\begin{pmatrix} \xi_-(1, t) \\ \xi_+(0, t) \end{pmatrix} = -K \begin{pmatrix} \xi_-(0, t) \\ \xi_+(1, t) \end{pmatrix}, \quad (3.2)$$

where $K = [k_{ij}] \in M_{2m, 2m}$.

Remark 3.1. The restriction to linear boundary conditions can be weakened by considering K as the gradient of nonlinear boundary conditions linearised around the origin. For instance, using (2.49),

$$\begin{pmatrix} \xi_-(1, t) \\ \xi_+(0, t) \end{pmatrix} = \mathbf{G} \begin{pmatrix} \xi_-(0, t) \\ \xi_+(1, t) \end{pmatrix}$$

the matrix K is computed as follows

$$K = -\nabla_{\xi}(\mathbf{G}(0))$$

where

$$\nabla_{\xi} = (\partial_{\xi_1}, \dots, \partial_{\xi_n}).$$

The matrix K has the following special structure:

$$K = \begin{pmatrix} T & Q \\ R & S \end{pmatrix} = \left(\begin{array}{cccc|cccc} 0 & k_{1,2} & & & k_{1,m+1} & & & \\ & 0 & k_{2,3} & & & k_{2,m+2} & & \\ & & \ddots & k_{m,m-1} & & & \ddots & \\ & & & 0 & & & & k_{m,2m} \\ \hline k_{m+1,1} & & & & 0 & & & \\ & k_{m+2,2} & & & k_{m+2,m+1} & 0 & & \\ & & \ddots & & & \ddots & \ddots & \\ & & & k_{2m,m} & & & k_{2m,2m-1} & 0 \end{array} \right) \quad (3.3)$$

where $Q, R, S, T \in M_{m,m}$, Q and R are diagonal matrices, S is a lower diagonal matrix and T is an upper diagonal matrix. With this definition, the system can indeed be viewed as a cascade of m 2×2 systems interconnected by the boundary conditions with $\xi_1, \dots, \xi_m$, the *backward* invariants and $\xi_{m+1}, \dots, \xi_{2m}$, the *forward* invariants.

From a control design point of view, the non-zero coefficients k_{ij} for the matrix K can be separated in two categories:

- *imposed coefficients* that are usually constrained by the physics of the boundary laws,
- *control coefficients* that can be assigned from the control law. These are used as control variables.

Our concern is to design a boundary control in order to stabilize the system at a particular desirable steady-state. Without loss of generality, this steady state is always chosen at the origin:

$$\bar{\xi} = 0. \quad (3.4)$$

3.2 Graph theory and non-negative matrices

In this section, we state some Definitions and Theorems from the framework of the theory of Nonnegative Matrices, see [1, Chapter 2].

A *nonnegative matrix* K is a matrix having nonnegative entries. It is denoted $K \geq 0$.

Definition 3.2. A $n \times n$ $K \geq 0$, $n > 1$, matrix is *cogredient* to a matrix E if for some permutation matrix P , $PKP^T = E$. K is *reducible* if it is cogredient to

$$E = \begin{bmatrix} B & 0 \\ C & D \end{bmatrix},$$

where B and D are square matrices. Otherwise, K is *irreducible*.

There exists a straightforward relation between directed graphs and non-negative matrices as stated in the definition below:

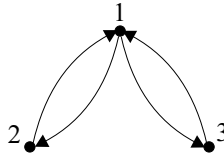
Definition 3.3. The associated directed graph, $G(K)$, of an $n \times n$ matrix K , consists of n vertices $P_1, P_2, \dots, P_n$ where an edge leads from P_i to P_j if and only if $k_{ij} \neq 0$

Example 3.4. Let

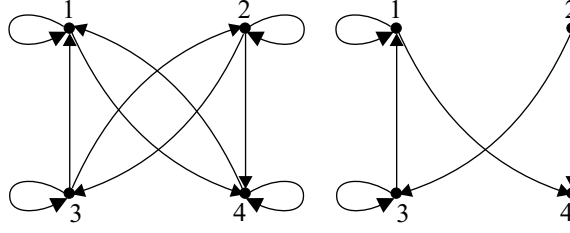
$$A = \begin{bmatrix} 0 & 1 & 1 \\ 1 & 0 & 0 \\ 1 & 0 & 0 \end{bmatrix},$$

$$B = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 1 \end{bmatrix}.$$

Then $G(A)$ is



$G(B)$ and $G(C)$ are respectively



The following theorem is the first part of the classical Perron-Frobenius theorem for positive matrices:

- Theorem 3.5.** • (a) If K is positive, then $\rho(K)$ is a simple eigenvalue, greater than the magnitude of any other eigenvalue.
- (b) If $K \geq 0$, then $\rho(K)$ is a simple eigenvalue, any eigenvalue of K of the same modulus is also simple, K has a positive eigenvector v corresponding to $\rho(K)$, and any nonnegative eigenvector of K is a multiple of v .

The second part of the theorem applies to irreducible matrices:

- Theorem 3.6.** • (a) If an irreducible $A \geq 0$ has h eigenvalues,

$$\lambda_0 = re^{i\theta_0}, \lambda_1 = re^{i\theta_1}, \dots, \lambda_{h-1} = re^{i\theta_{h-1}}$$

of modulus $r = \rho(K)$,

$$0 = \theta_0 < \theta_1 < \dots < \theta_{h-1} < 2\pi,$$

then these numbers are the distinct roots of $\lambda^h - r^h = 0$.

- (b) More generally, the whole spectrum $S = \{\lambda_0, \lambda_1, \dots, \lambda_{h-1}\}$ of K goes over into itself under a rotation of the complex plane by $2\pi/h$.

We shall now define the strong connectivity in a directed graph:

Definition 3.7. A directed graph G is strongly connected if for any ordered pair of vertices (u, v) , there exists a sequence of edges (a path) that leads from u to v .

Theorem 3.8. A matrix $K \geq 0$ is irreducible if and only if the associated directed graph $G(K)$ is strongly connected.

Definition 3.9. Let $K \geq 0$ be irreducible. The number h of eigenvalues of K of modulus $\rho(K)$ is called the index of cyclicity of K . If h is greater than one, K is said to be cyclic of index h .

Theorem 3.10. Let $K \geq 0$ be an irreducible matrix of order n . Let S_i be the set of all the circuits lengths m_u of in $G(K)$, through u . Let

$$h_u = \gcd_{m_u \in S_u} \{m_u\}.$$

Then $h_1 = h_2 = \dots = h_n = h$ and h is the index of cyclicity of K .

3.3 Graph representations

In this section, two graph representations are given for the system (3.1)-(3.2).

The *flux graph* represents the network of 2×2 systems where each edge represents one system and the nodes represent the coupling between those systems through the boundary conditions.

The *invariant graph* represents the interaction of the invariants ξ_i through the boundary conditions.

We now give a formal definition of the *flux graph*:

Definition 3.11. *The flux graph (G_F) is a directed graph defined by $G_F = (V_F, E_F)$ where V_F is the flux vertex-set and E_F is the flux edge-set. A flux vertex represents the coupling of systems through boundary conditions. A flux edge represents a 2×2 hyperbolic PDE system. The direction of the edge is chosen as the direction of the positive Riemann invariant.*

The *invariant graph* is a weighted directed graph (e.g. [2, Chapter 1], [1, Chapter 2]) based on the $\text{abs}(K)$ matrix.

Definition 3.12. *The invariant graph (G_I) is a weighted directed graph defined by $(V_I, E_I, \text{abs}(K))$ where V_I is the invariant vertex-set, E_I the invariant edge-set associated to the K matrix and $\text{abs}(K)$ the invariant edge weights. An invariant vertex represents a boundary condition in (2.49), and an invariant edge (u, v) connects a vertex u to a vertex v if the corresponding entry k_{uv} in K is non zero. Each edge is weighted by the absolute value of the corresponding entry in the K matrix.*

Note that the *invariant graph* can be seen as the *associated directed graph* of the non-negative matrix $\text{abs}(A)$ (e.g. [1, Chapter 2]). Another way of interpreting the *invariant graph* is to consider it as a network flow graph (e.g. [2, Chapter 4]) where the edge weights are defined by the matrix $\text{abs}(K)$.

3.3.1 Examples

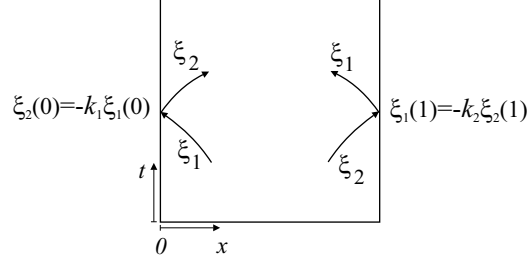
This section illustrates the definitions of the *Flux Graph* and the *Invariant Graph* with two simple networks composed respectively with one and two cascaded 2×2 systems.

For each example, the following representations are successively given:

- the representation of the invariants in the (t, x) plane,
- the K matrix,
- the *Flux Graph*,
- the *Invariant Graph*.

3.3.2 Graph of a 2×2 system

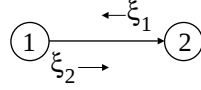
- (t, x) view:



- K matrix:

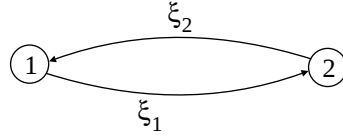
$$K = \begin{pmatrix} 0 & k_{12} \\ k_{21} & 0 \end{pmatrix}$$

- *Flux graph*:



$$\begin{aligned} V_F &= \{1, 2\} , \\ E_F &= \{(1, 2)\} . \end{aligned}$$

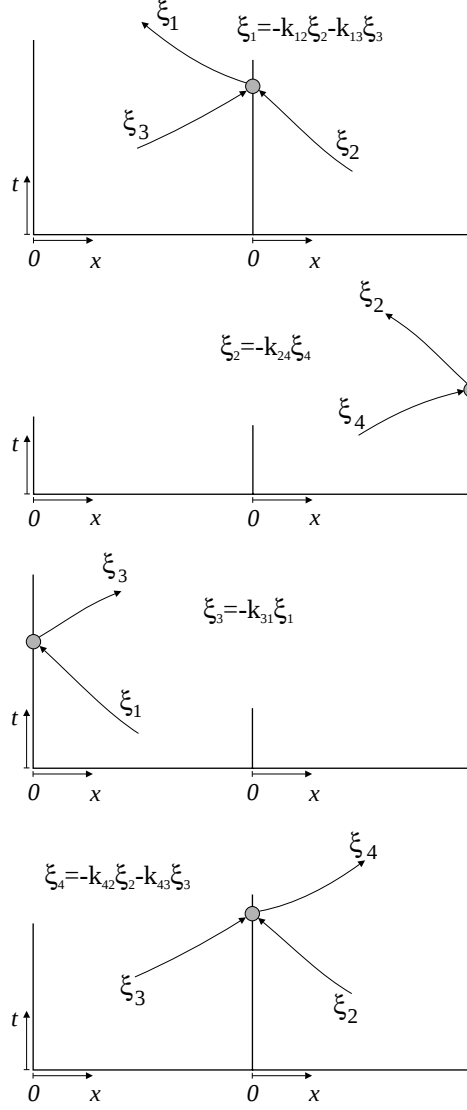
- *Invariant graph*:



$$\begin{aligned} V_I &= \{1, 2\} \\ E_I &= \{(1, 2), (2, 1)\} \end{aligned}$$

3.3.3 Graph of a cascade of two 2×2 system

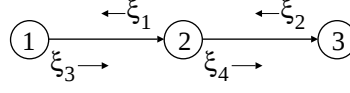
- (t, x) view:



- K matrix:

$$K = \begin{pmatrix} 0 & k_{12} & k_{13} & 0 \\ 0 & 0 & 0 & k_{24} \\ k_{31} & 0 & 0 & 0 \\ 0 & k_{42} & k_{43} & 0 \end{pmatrix}$$

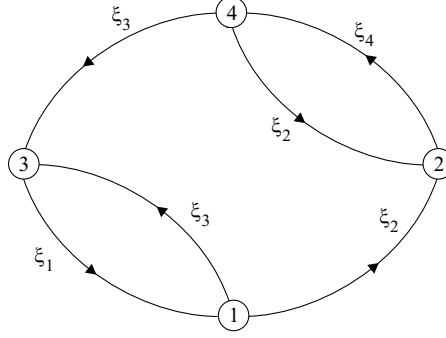
- *Flux graph:*



$$V_F = \{1, 2, 3\} ,$$

$$E_F = \{(1, 2), (2, 3)\} .$$

- *Invariant graph:*



$$V_I = \{1, 2, 3, 4\}$$

$$E_I = \{(1, 2), (1, 3), (2, 4), (3, 1), (4, 2), (4, 3)\} ,$$

3.4 Control design for a cascade of two 2×2 systems

3.4.1 Model

In this section, we shall consider the control design for the simple network composed of two systems in cascade as described in Section 3.3.3. The network is intentionally restricted to a cascade of two systems in order to simplify the notations and give the reader a clear understanding of the design methodology. A general result for a cascade of m systems is given in Section 3.5.

For this system, the boundary conditions (3.2) are written:

$$\begin{pmatrix} \xi_1(1, t) \\ \xi_2(1, t) \\ \xi_3(0, t) \\ \xi_4(0, t) \end{pmatrix} = - \begin{pmatrix} 0 & k_{12} & k_{13} & 0 \\ 0 & 0 & 0 & k_{24} \\ k_{31} & 0 & 0 & 0 \\ 0 & k_{42} & k_{43} & 0 \end{pmatrix} \begin{pmatrix} \xi_1(0, t) \\ \xi_2(0, t) \\ \xi_3(1, t) \\ \xi_4(1, t) \end{pmatrix} \quad (3.5)$$

3.4.2 Properties of the *invariant graph*

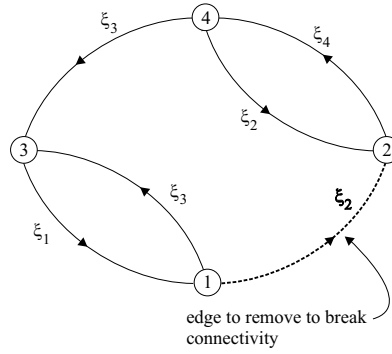
The *invariant graph* and its associated matrix $\text{abs}(K)$ defined in (3.3) have several interesting properties that come from the framework of the theory of Nonnegative Matrices (see [1, Chapter 2]).

First of all, it is trivial to see that the *invariant graph* is *strongly connected* (see Definition 3.7). In fact, since all the invariants come in "pairs" (a positive and a negative), there is always forward and backward path between each vertex of the invariant graph. Therefore, the graph is strongly connected.

This property implies that $\text{abs}(A)$ is *irreducible* (see Definition 3.2 and Theorem 3.8).

The strong connectivity implies that there is always at least one circuit passing through any *invariant vertex*. Moreover, the circuit lengths are always a multiple of two since the "information" should always travel through two *invariant edges* before returning to the initial vertex. Therefore, the index of cyclicity of $\text{abs}(A)$ is 2 (see Definition 3.9 and Theorem 3.10).

3.4.3 Control Design using the *invariant graph*



$$K = \begin{pmatrix} 0 & k_{12} & k_{13} & 0 \\ 0 & 0 & 0 & k_{24} \\ k_{31} & 0 & 0 & 0 \\ 0 & k_{42} & k_{43} & 0 \end{pmatrix}$$

Fig. 3.2. The *invariant graph* of two 2×2 systems in cascade. If the edge $(1, 2)$ has been removed, the connectivity of the graph is broken.

The control goal is to select the control coefficients k_{ij} in order to satisfy the stability condition of Theorem 2.28:

$$\rho(\text{abs}(K)) < 1 .$$

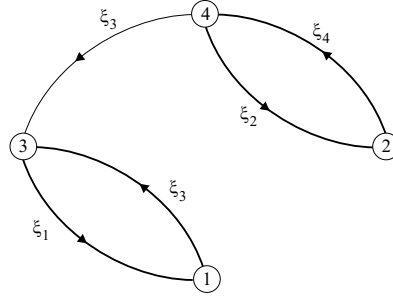
Our purpose is to show that the structure of the invariant graph can be used in order to simplify the computation of the control coefficients that guarantee the satisfaction of the stability condition.

Recall that the proof relies on the tracking of the Riemann invariants along their trajectories and bounces at the boundaries. The approach can be formalised with the *invariant graph* where to follow the trajectories is equivalent to follow all the possible paths in the graph.

The idea of the control design is to break the connectivity of the graph such that the maximum circuit length in the *invariant graph* is 2. Therefore, the *invariant graph* appears as a sequence of locally stabilizing loops. For example, in Figure 3.2, removing the edge $(1, 2)$ breaks the circuit $(1, 2, 4, 3)$ and the two remaining circuit have length two. The resulting graph is depicted in Figure 3.3. Analytically, this connectivity breaking is obtained by setting $k_{12} = 0$ in (3.5).

It turns out in the general case that the connectivity breaking is also equivalent to setting T to zero in (3.3), which is a key property that shall be used to compute the stability condition. By symmetry, we could also choose to set Q to zero but both are equivalent by relabelling the systems.

In the next section, we prove that breaking the connectivity in a cascade of 2×2 systems leads to a global stability condition based on the stability of the local loops.



$$\begin{aligned}
 V_I &= \{1, 2, 3, 4\} \\
 E_I &= \{(1, 3), (2, 4), (3, 1), (4, 2), (4, 3)\} , \\
 K &= \begin{pmatrix} 0 & 0 & k_{13} & 0 \\ 0 & 0 & 0 & k_{24} \\ k_{31} & 0 & 0 & 0 \\ 0 & k_{42} & k_{43} & 0 \end{pmatrix}
 \end{aligned}$$

Fig. 3.3. The *invariant graph* of a two 2×2 systems in cascade where the edge $(1, 2)$ has been removed.

3.5 Stability condition for cascades of 2×2 systems with particular boundary conditions

Let us come back to the formulation of the K matrix introduced in (3.3),

$$K = \begin{pmatrix} T & Q \\ R & S \end{pmatrix}. \quad (3.6)$$

The control design for $m \times 2$ systems is a straightforward generalization of the 2-dimensional system case. It consists of setting the T matrix in (3.3) to zero, by control design or due to a particular configuration of the physical devices represented by the boundary conditions. Hence, K is written as follows:

$$K = \begin{pmatrix} \mathbf{0} & Q \\ R & S \end{pmatrix}, \quad (3.7)$$

where $\mathbf{0}$ is a $(m \times m)$ matrix of zeros.

In graph theory terms, this is equivalent to remove edges until the index of cyclicity of the *invariant graph* is equal to 2.

Of course, the assumption that T could be set to zero can seem to be restrictive but this is not always the case. In fact, there are physical systems that naturally have this property. For example, open channels separated by spillway gates, an hydraulic structure often used in open channel regulation, have this property (e.g. [16, 33]).

In this particular case, the form of K let us compute a simple analytical stability condition for the system as stated in the proposition below.

Proposition 3.13. *If*

$$\rho(\text{abs}(K)) = \max_{i \in [1, p]} \{k_{n+i, i} k_{i, n+i}\} < 1. \quad (3.8)$$

then there exists $\varepsilon > 0$, $\mu > 0$ and $C > 0$ such that, for every $\xi^\# \in C^1([0, L]; \mathbb{R}^n)$ satisfying the condition (C_2) and such that

$$\|\xi^\#\|_{C^1([0, L])} \leq \varepsilon,$$

there exists one and only one function $\xi \in C^1([0, L] \times [0, +\infty); \mathbb{R}^n)$ satisfying (3.1), (3.2) and

$$\xi(x, 0) = \xi^\#(x), \quad \forall x \in [0, L]. \quad (3.9)$$

Moreover, this function ξ satisfies

$$\|\xi(\cdot, t)\|_{C^1([0, L])} \leq C e^{-\mu t} \|\xi^\#\|_{C^1([0, L])}, \quad \forall t \geq 0. \quad (3.10)$$

In order to prove Proposition 3.13 by using Theorem 2.28, we need the following lemma to compute the eigenvalues of K .

Lemma 3.14. *Let $n > 0$, $K \in M_{2n,2n}$, $Q, R, S \in M_{n,n}$ such that*

$$K = \begin{bmatrix} 0 & Q \\ R & S \end{bmatrix}$$

where Q and R are diagonal matrices and S is a lower diagonal matrix. Then, the eigenvalues λ_i of K are

$$\lambda_i(N) = \pm \sqrt{q_{ii}r_{ii}} \quad \text{for } i = 1, \dots, n.$$

Proof (Proof of Proposition 3.13). Using Lemma 3.14, it follows that the spectral radius of $\text{abs}(K)$ where K defined in (3.7) is equal to:

$$\rho(\text{abs}(K)) = \max_{i \in [1,p]} \{k_{n+i,i} k_{i,n+i}\}. \quad (3.11)$$

Using a direct application of Theorem 2.28, the stability condition of the system with boundary condition is rewritten as (3.8). This completes the proof. $\square$

We shall now give a proof of the Lemma 3.14.

Proof (Proof of Lemma 3.14). We look for the values λ such that

$$\det \begin{pmatrix} \lambda I & -Q \\ -R & \lambda I - S \end{pmatrix} = 0. \quad (3.12)$$

Taking the *Schur complement* of (3.12), we have

$$\begin{aligned} \det \begin{pmatrix} \lambda I & -Q \\ -R & \lambda I - S \end{pmatrix} &= \det(\lambda I) \det(\lambda I - S - R(\lambda I)^{-1}Q) \\ &= \det(\lambda^2 I - \lambda S - RQ) \\ &= \det \begin{pmatrix} \lambda_1^2 - q_{1,1}r_{1,1} & & \\ -\lambda_2 s_{2,1} & \lambda_2^2 - q_{2,2}r_{2,2} & \\ & \ddots & \ddots \\ & & -\lambda_n s_{n,n-1} & \lambda_n^2 - q_{n,n}r_{n,n} \end{pmatrix} \end{aligned}$$

Hence, this determinant is zero if

$$\lambda_i^2 - q_{i,i}r_{i,i} = 0 \quad \text{for } i = 1, \dots, n.$$

This completes the proof. $\square$

3.5.1 Illustration for values of $m = 3, 4, 5$

A software was developed to model such networks. It was used to produce the following graphs.

The figures represent the invariant graphs of network of cascade of 3, 4 and 5 2×2 systems in Figures 3.4, 3.5 and 3.6. For each system, the left figure represents the invariant graph when the sub-matrix T in K , defined in (3.3), is non zero, while the right figure represents the invariant graph in when T is zero. On the left figure, the invariant edges that have to be removed are denoted by dotted lines.

The data structures and algorithm that have been used to build the Figures 3.4, 3.5 and 3.6 are described in Appendix D.

$m = 3$

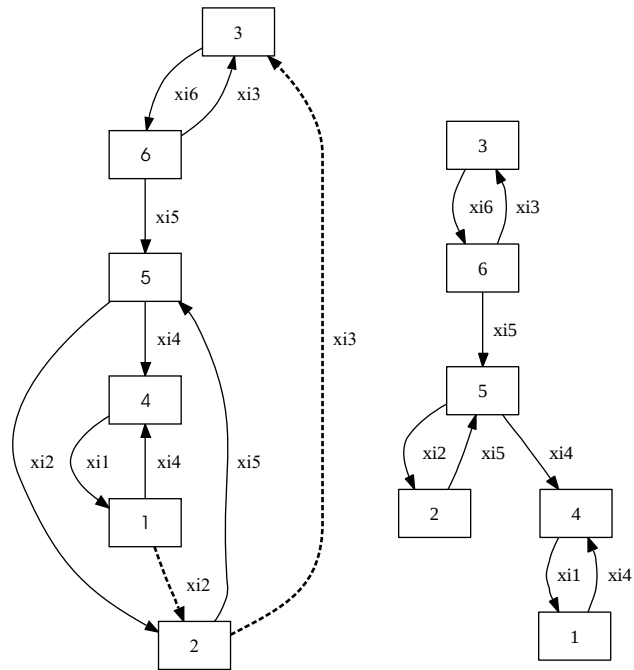


Fig. 3.4. Flux graph and invariant graph for a 3-reach channel.

$m = 4$

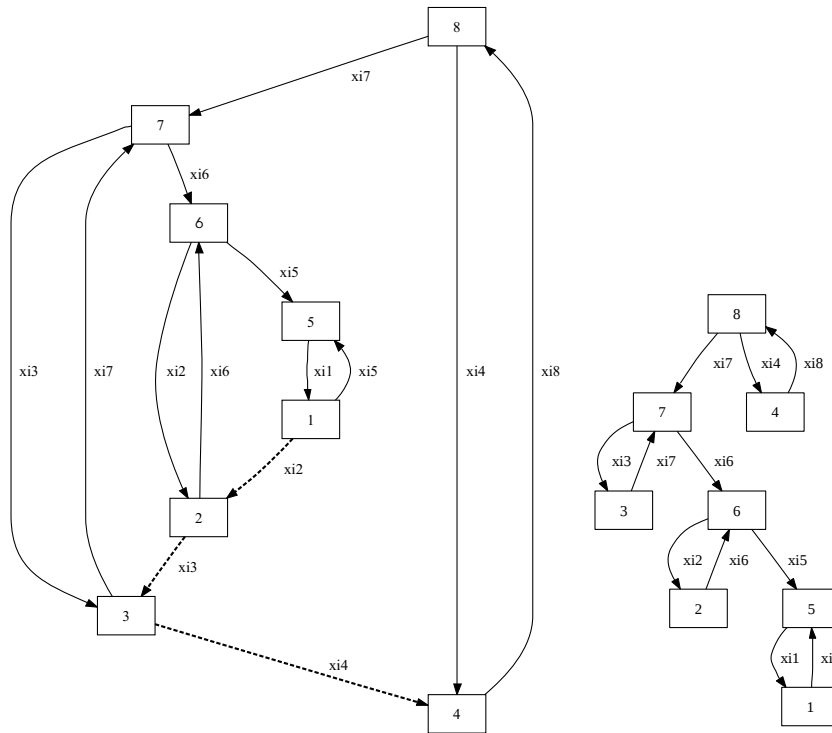


Fig. 3.5. Flux graph and invariant graph for a 4-reach channel.

$m = 5$

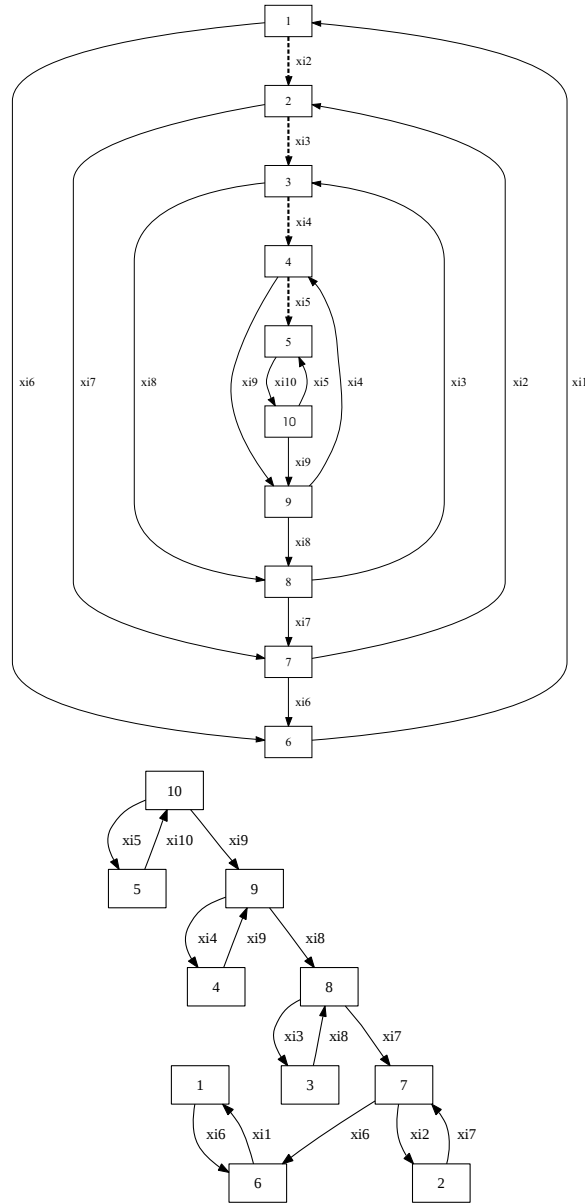


Fig. 3.6. Flux graph and invariant graph for a 5-reach channel.

3.6 Conclusion

In this chapter, we have proposed a control design for a cascade of $n \times 2$ hyperbolic systems. The design is based on breaking the connectivity of the *invariant graph* of the system in order to lower of index of cyclicity to 2. This step leads to an simple analytical stability condition.

During the chapter, we have described a number of representation of the boundary conditions that are tightly linked together as depicted in the Figure 3.7 for a single 2×2 system. In that Figure, the backward invariants are represented with dotted line (ξ_1), forward invariant with a plain line (ξ_2). Each boundary condition is denoted with a corresponding shape (circle or square) and this shape is used to show the relationships between the different views.

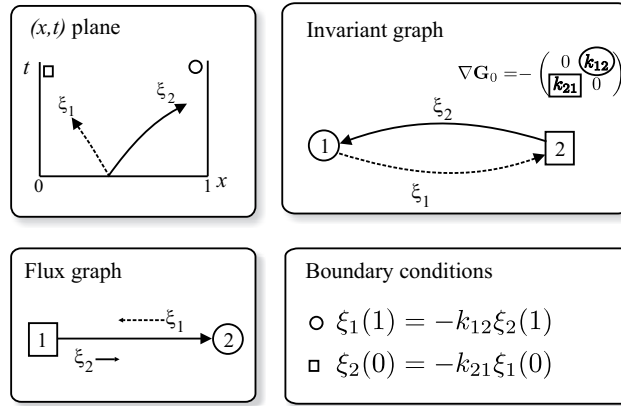


Fig. 3.7. Different graph representations.

Application to waterways with mobile spillways

4.1 Introduction

Canalised waterways are often made up of a cascade of reaches separated by hydraulic structures such as mobile spillways (see Fig. 4.1). When the water flow rate is slow enough, the water passes over the nappe and the discharge depends on the upstream lip but does not depend on the downstream. In order to guarantee the navigation, an important issue is to stabilize the water level in the reaches in spite of variations of the natural flow rate of the river. In this

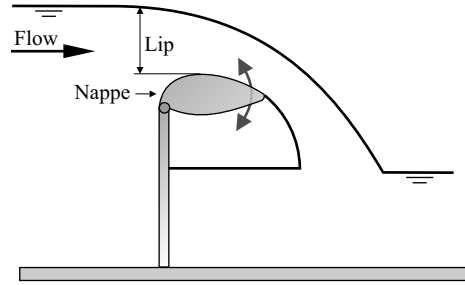


Fig. 4.1. A spillway.

chapter, the flow is modelled using the Saint Venant PDEs. The stabilization problem is solved by using boundary control based on Riemann invariants as introduced in Chapter 2 and Chapter 3. The control performance is illustrated with a realistic simulation experiment for the Sambre river in Belgium. The chapter is organized as follows.

In Section 4.2, we present the Saint Venant partial differential equations for the modelling of an horizontal reach. The steady-state of the system is calculated and the control objective is formulated. Section 4.2.4 is devoted to the stability analysis of the steady-states and the control design. The model

is reformulated in terms of Riemann invariants which are more convenient for our purpose. The stability of the steady-states is analyzed by using Theorem 2.28. On this basis, a control law based on Riemann invariants is proposed. Its efficiency is illustrated with a simulation experiment.

Finally in Section 4.4, we investigate the applicability of this control approach to a part of the Sambre river (Belgium) composed of a cascade of 7 reaches for a total length of about 50 km.

4.2 Model of a horizontal reach

4.2.1 Saint Venant equations

We consider a one-dimensional portion of canal as represented in Fig. 4.2. The dynamics of the system are described by Saint Venant equations (see e.g. [51, 16, 33, 30]). We restrict our attention to the case of an horizontal reach with a prismatic section, without viscous friction terms and we suppose that the flow is sub-critical, so that the dynamical equations simplify as follows :

Mass conservation :

$$\partial_t H + \partial_x (VH) = 0. \quad (4.1)$$

Quantity of movement conservation :

$$\partial_t V + \partial_x \left(gH + \frac{V^2}{2} \right) = 0, \quad (4.2)$$

where $x \in [0, L]$ is the space coordinate, $t \in [0, T]$ is time, ∂_x and ∂_t are the partial derivative w.r.t. x and t respectively, L is the reach length, $V(x, t)$ is the water velocity (at point x and time t), $H(x, t)$ is the water level (at point x and time t) and g is the gravity constant. The sub-critical flow condition is :

$$V < \sqrt{gH} \quad (4.3)$$

The water flow rate is defined as (we suppose an unitary width):

$$Q(x, t) = V(x, t)H(x, t) \quad (4.4)$$

The inflow rate at $x = 0$ is therefore:

$$Q(0, t) = V(0, t)H(0, t) \quad (4.5)$$

The control action is provided by one weir gate located at the right end ($x = L$) of the reach (see Fig. 1). The gate position is denoted u . A standard discharge relationship of weir gates (see e.g. [33, Chapter 4]) is as follows :

$$V(L, t)H(L, t) = k(H(L, t) - u)^m \quad (4.6)$$

where $k > 0$ and $m \in [1, 3/2]$ are constant parameters.

Equations (4.5) and (4.6) are the boundary conditions at $x = 0$ and $x = L$, associated with the PDEs (4.1)-(4.2).

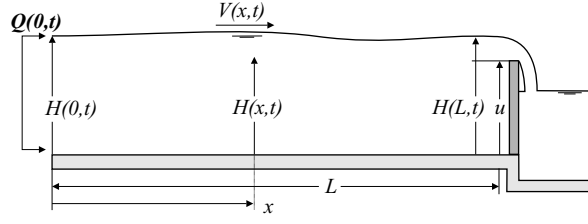


Fig. 4.2. The horizontal reach.

4.2.2 Steady-states

For a given constant gate position $\bar{u}$ and a given constant inflow rate $\bar{Q}$, there exists a steady-state solution $(\bar{V}, \bar{H})$ for equations (4.1), (4.2), (4.5) and (4.6). These satisfy the following relations :

$$\bar{H} = u + \left(\frac{\bar{Q}}{k} \right)^{1/m} \quad \bar{V} = \frac{\bar{Q}}{\left(u + \frac{\bar{Q}}{k} \right)^{1/m}} \quad (4.7)$$

4.2.3 Statement of the control problem

The control objective is to stabilize the system (4.1), (4.2), (4.5), (4.6) around a set point $(\bar{H}, \bar{Q})$. The control action is the gate opening u . The water level $H(L, t)$ supposed to be measured on line at each time instant t . The water level set point $\bar{H}$ is selected in order to satisfy navigability requirements.

4.2.4 Stability Analysis of the Steady-States and Control design

Characteristic velocities

We can rewrite system (4.1)-(4.2) in a matrix-vector form :

$$\partial_t \begin{pmatrix} H \\ V \end{pmatrix} + A(H, V) \partial_x \begin{pmatrix} H \\ V \end{pmatrix} = 0 \quad (4.8)$$

with the *characteristic* matrix :

$$A(H, V) = \begin{pmatrix} V & H \\ g & V \end{pmatrix} \quad (4.9)$$

The eigenvalues of this matrix:

$$\lambda_1(H, V) = V - \sqrt{gH}, \lambda_2(H, V) = V + \sqrt{gH} \quad (4.10)$$

are called the characteristic velocities. The subcritical flow hypothesis (4.3) implies that:

$$\lambda_1(H, V) < 0 < \lambda_2(H, V) \quad (4.11)$$

which enforces that the system is strictly hyperbolic (see Definition 2.6).

The model in terms of Riemann invariants

Let us now consider the following change of coordinates:

$$\xi_1 = V - \bar{V} - 2(\sqrt{g\bar{H}} - \sqrt{g\bar{H}}) \quad (4.12)$$

$$\xi_2 = V - \bar{V} + 2(\sqrt{g\bar{H}} - \sqrt{g\bar{H}}). \quad (4.13)$$

where $(\bar{H}, \bar{V})$ is an arbitrary steady-state.

With these new coordinates (ξ_1, ξ_2) , the system (4.1)-(4.2) is rewritten into the following diagonal form:

$$\partial_t \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} + \begin{pmatrix} \lambda_1(\xi_1, \xi_2) & 0 \\ 0 & \lambda_2(\xi_1, \xi_2) \end{pmatrix} \partial_x \begin{pmatrix} \xi_1 \\ \xi_2 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \quad (4.14)$$

where λ_1 and λ_2 are the characteristic velocities now expressed in terms of ξ_1, ξ_2 :

$$\lambda_1(\xi_1, \xi_2) = \frac{1}{4}\xi_2 + \frac{3}{4}\xi_1 + \bar{V} - \sqrt{g\bar{H}} \quad (4.15)$$

$$\lambda_2(\xi_1, \xi_2) = \frac{3}{4}\xi_2 + \frac{1}{4}\xi_1 + \bar{V} + \sqrt{g\bar{H}} \quad (4.16)$$

Since the change of coordinates (4.12)-(4.13) is a mapping, H and V can be expressed in terms of Riemann invariants :

$$H = \frac{(\xi_2 - \xi_1 + 4\sqrt{g\bar{H}})^2}{16g} \quad (4.17)$$

$$V = \frac{\xi_2 + \xi_1}{2} + \bar{V}. \quad (4.18)$$

Stability Theorem

It is obvious that the equilibrium $\bar{H}, \bar{V}$ expressed in the ξ_1, ξ_2 coordinates is:

$$\bar{\xi}_1 = 0 \quad \bar{\xi}_2 = 0 \quad (4.19)$$

The stability of the flow in a neighborhood of this steady state in a single reach can be analyzed with the following theorem. The theorem is stated here in a rather general form because it will be used also later for the closed loop system stability analysis.

We consider the shallow water equations (4.14), expressed in (ξ_1, ξ_2) coordinates, defined on the domain $(x, t) \in [0, L] \times [0, \infty)$. The boundary conditions are supposed to be given in the following general form:

$$f_0(\xi_1(0, t), \xi_2(0, t)) = 0 \quad f_L(\xi_1(L, t), \xi_2(L, t)) = 0 \quad (4.20)$$

with the functions $f_0(\xi_1, \xi_2)$ and $f_L(\xi_1, \xi_2)$ being of class C^1 . By differentiating these boundary conditions with respect to time and by using equations (4.14), we have the following so-called *boundary compatibility conditions* at the initial instant $t = 0$:

$$\begin{aligned} & \lambda_1(\xi_1, \xi_2)(0, 0) \partial_{\xi_1} f_0(\xi_1, \xi_2)(0, 0) \partial_x \xi_1(0, 0) + \\ & \lambda_2(\xi_1, \xi_2)(0, 0) \partial_{\xi_2} f_0(\xi_1, \xi_2)(0, 0) \partial_x \xi_2(0, 0) = 0 \\ & \lambda_1(\xi_1, \xi_2)(L, 0) \partial_{\xi_1} f_L(\xi_1, \xi_2)(L, 0) \partial_x \xi_1(L, 0) + \\ & \lambda_2(\xi_1, \xi_2)(L, 0) \partial_{\xi_2} f_L(\xi_1, \xi_2)(L, 0) \partial_x \xi_2(L, 0) = 0 \end{aligned} \quad (4.21)$$

Proposition 4.1. *Assume that the initial conditions $\xi_1^\sharp, \xi_2^\sharp$ in $C^1([0, L])^2$ satisfy the boundary compatibility conditions (4.21) and that the following inequality holds:*

$$A_1 A_2 < 1 \quad (4.22)$$

with

$$A_1 = \left| \frac{\partial_{\xi_1} f_0(0, 0)}{\partial_{\xi_2} f_0(0, 0)} \right| \quad \text{and} \quad A_2 = \left| \frac{\partial_{\xi_2} f_L(0, 0)}{\partial_{\xi_1} f_L(0, 0)} \right|. \quad (4.23)$$

Then, there exist positive constants ϵ , M , μ such that, if the initial condition is small enough:

$$|\xi_1^\sharp|_{C_1[0, L]} + |\xi_2^\sharp|_{C_1[0, L]} \leq \epsilon \quad (4.24)$$

there is a unique solution $\xi_1(x, t)$, $\xi_2(x, t)$ of class C^1 on $[0, L] \times [0, \infty)$ which decays to zero with an exponential rate:

$$|\xi_1(x, t)|_{C_1[0, L]} + |\xi_2(x, t)|_{C_1[0, L]} \leq M e^{-\mu t} \quad (4.25)$$

Proof. This theorem is a direct application of Theorem 2.28.

Control design with Riemann invariants

Let us assume that the inflow rate is constant: $Q(0, t) = \bar{Q}$. This implies that the boundary condition at $x = 0$ is written

$$\begin{aligned} f_0(\xi_1(0, t), \xi_2(0, t)) &= \frac{\xi_1(0, t) + \xi_2(0, t) + 2\bar{V}}{2} \frac{(\xi_2(0, t) - \xi_1(0, t) + 4\sqrt{g\bar{H}})^2}{16g} - \bar{Q} \\ &= 0 \end{aligned} \quad (4.26)$$

The control law, i.e. the gate position u , at $x = L$ is selected in order to have a linear relationship between $\xi_1(L, t)$ and $\xi_2(L, t)$:

$$\xi_1(L, t) = -\gamma \xi_2(L, t)$$

which implies

$$f_L(\xi_1(L, t), \xi_2(L, t)) = \xi_1(L, t) + \gamma \xi_2(L, t) \quad (4.27)$$

where $\gamma \geq 0$ is a tuning parameter.

It follows that the stability condition $A_1 A_2 < 1$ becomes:

$$\left| \frac{\sqrt{g\bar{H}} - \bar{V}}{\sqrt{g\bar{H}} + \bar{V}} \right| |\gamma| < 1 \quad (4.28)$$

Using the subcritical flow condition (4.3) in Theorem 2.21, we then have the following sufficient condition for locally exponential convergence of the solution to the equilibrium point :

$$0 \leq \gamma < \left| \frac{\sqrt{g\bar{H}} + \bar{V}}{\sqrt{g\bar{H}} - \bar{V}} \right|. \quad (4.29)$$

By inverting the change of coordinates using (4.17)-(4.18) and (4.6), we get an explicit expression of the control u :

$$u = H(L, t) - \left(\frac{(\frac{\bar{Q}}{\bar{H}} + e_v)H(L, t)}{k} \right)^{1/m} \quad (4.30)$$

where e_v is the error on the water velocity :

$$e_v = V - \bar{V} = 2 \frac{1-\gamma}{1+\gamma} (\sqrt{g\bar{H}} - \sqrt{g\bar{H}}) \quad (4.31)$$

From an engineering viewpoint, this control law has several advantages : the control law (4.30) is local -no communication with other gates is needed- and it depends only on water depth measurements -neither water velocity nor flow measurements are needed-.

Simulation result

The goal of the simulation is to test the robustness of the control with respect to inflow perturbations.

We consider a reach of length $L = 5000\text{m}$ and 40m width, the initial state and steady state are:

$$(Q, H)_{t=0} = (\bar{Q}, \bar{H}) = (10\text{m}^3/\text{s}, 4\text{m}).$$

The tuning parameter γ has been set to 0.1, so the product $A_1 A_2 = 0.098$. The inflow is disturbed by a flood wave as depicted in Figure 4.3. The simulation has been made using a semi-implicit Preissman scheme with a time step of 30s and a spatial step of 100 m. The proposed control law (4.30) is compared to a "no control" situation where u is simply constant, $u = \bar{u}$. The global

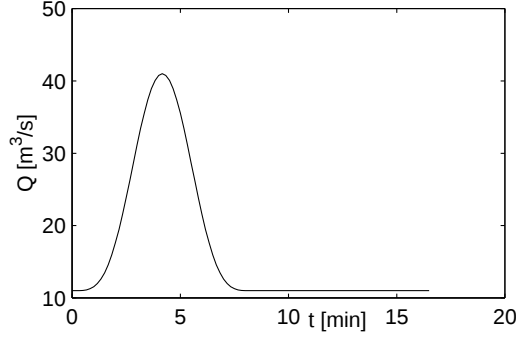


Fig. 4.3. Inflow at $x = 0$.

deviation of the canal state with respect to the equilibrium is measured by the entropy of the fluid, R :

$$R = \int_0^L H \frac{(V - \bar{V})^2}{2} + g \frac{(H - \bar{H})^2}{2} dx \quad (4.32)$$

In Figure 4.4, we see $H(., t)$ for different simulation times (13 min, 17.5 min and 22.5 min). One can see that the wave is widely damped by the proposed control law. In Figure 4.5, we see that the control law (4.30) asymptotically stabilizes the channel even for a large deviation (see Figure 4.3) of the inflow state.

4.3 Stabilization of a cascade of reaches

In this section, we generalize the design method of Section 4.2.4 to a cascade of reaches.

4.3.1 Model

Let us consider a cascade of $m > 1$ reaches in cascade indexed from 1 to m as depicted in Figure 4.6.

Let us define the vectors of Riemann invariants ξ_- , ξ_+ and ξ as defined in Section 2.6.3:

$$\xi_- = (\xi_1, \dots, \xi_n)^T, \quad \xi_+ = (\xi_{n+1}, \dots, \xi_{2n})^T, \quad \xi = \begin{pmatrix} \xi_- \\ \xi_+ \end{pmatrix} \quad (4.33)$$

In Riemann coordinates, the shallow water equations for the network are

$$\partial_t \xi + A(\xi) \partial_x \xi = 0. \quad (4.34)$$

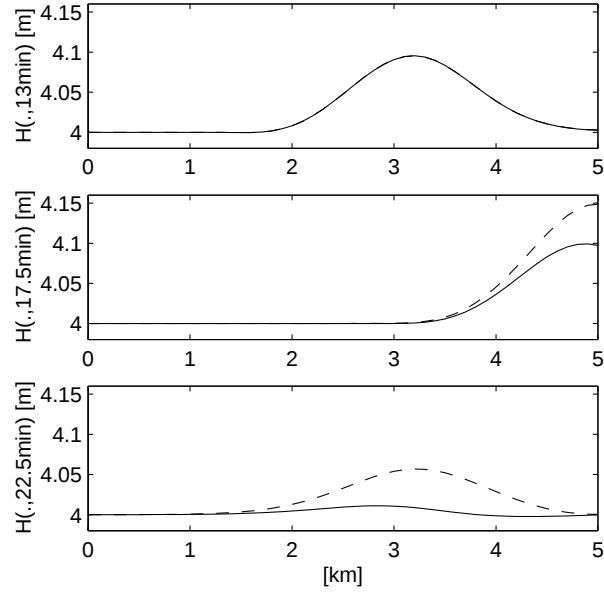


Fig. 4.4. $H(.,t)$ at different simulation time. Plain curve for our control, dotted curve for open loop.

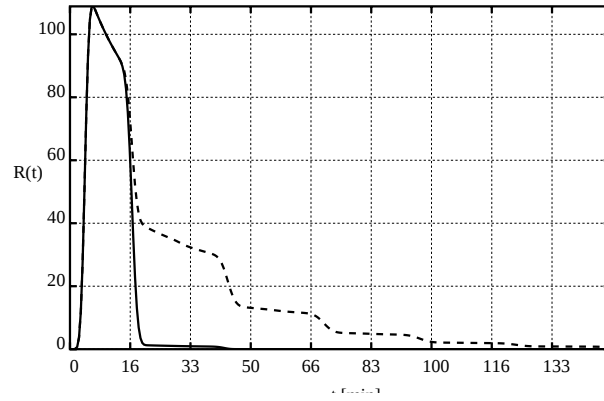


Fig. 4.5. Entropy R . Plain curve, Riemann based control. Dotted curve, open loop control.

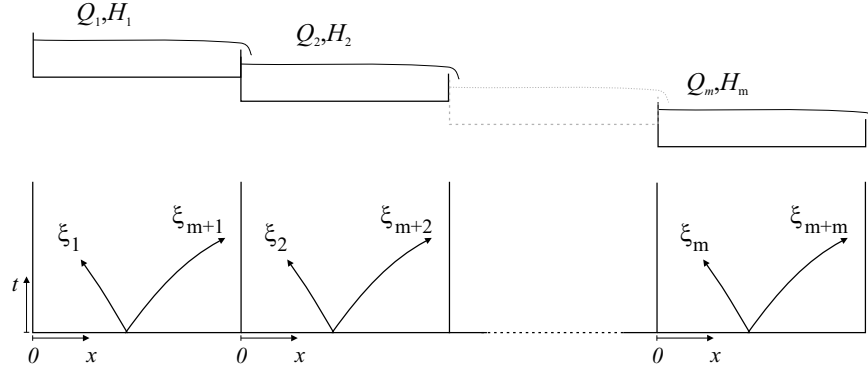


Fig. 4.6. A cascade of reaches and it's representation in the Riemann invariants variables.

The boundary condition in terms of Riemann invariants have of the following generic form:

$$\mathbf{f} \begin{pmatrix} \xi_{-}(0, t) \\ \xi_{-}(L, t) \\ \xi_{+}(0, t) \\ \xi_{+}(L, t) \end{pmatrix} = \begin{pmatrix} f_1(\xi_1(0, t), \xi_{n+1}(0, t)) \\ f_2(\xi_2(0, t), \xi_1(L, t), \xi_{n+2}(0, t), \xi_{n+1}(L, t)) \\ f_3(\xi_1(L, t), \xi_{n+1}(L, t)) \\ \vdots \\ f_{2i}(\xi_{i+1}(0, t), \xi_i(L, t), \xi_{n+i+1}(0, t), \xi_{n+i}(L, t)) \\ f_{2i+1}(\xi_i(L, t), \xi_{n+i}(L, t)) \\ \vdots \\ f_{2n-2}(\xi_n(0, t), \xi_{n-1}(L, t), \xi_{2n}(0, t), \xi_{2n-1}(L, t)) \\ f_{2n-1}(\xi_{n-1}(L, t), \xi_{2n-1}(L, t)) \\ f_{2n}(\xi_n(L, t), \xi_{2n}(L, t)) \end{pmatrix} = 0 \quad (4.35)$$

where f_{2i} are flow conservation conditions, similarly to (4.26) and f_{2i+1} are discharge relationships, similarly to (4.27).

Note that the steady state corresponds to vanishing Riemann invariants:

$$\bar{\xi} = 0.$$

4.3.2 Main result

In order to derive a stability condition, we must compute the eigenvalues of the following matrix:

$$-\left(\nabla_{[\xi_{-}(L, t), \xi_{+}(0, t)]} \mathbf{f}(0)\right)^{-1} \nabla_{[\xi_{-}(0, t), \xi_{+}(L, t)]} \mathbf{f}(0). \quad (4.36)$$

The particular type of channel network (canals in cascade) and control laws that do not depend on the downstream water state lead to a sparse

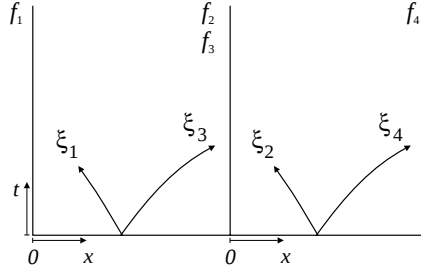


Fig. 4.7. Representing ξ_1, ξ_2, ξ_3 and ξ_4

form of $\mathbf{f}$ which leads to compact stability conditions such that we can apply Lemma 3.14.

It is crucial that the control laws f_{2i+1} depend only the upstream water state.

We can state our main result:

Proposition 4.2. *Assume that the initial conditions ξ^h in $C^1([0, L])^n$ satisfy the boundary compatibility conditions and the steady state $\bar{\xi} = 0$ satisfies the following conditions,*

$$\begin{aligned} \partial_{\xi_{n+1}(0,t)} f_1(0) &\neq 0, \\ \partial_{\xi_n(L,t)} f_{2n}(0) &\neq 0, \\ \partial_{\xi_{n+i+1}(0,t)} f_{2i}(0) \partial_{\xi_i(L,t)} f_{2i+1}(0) &\neq 0, \quad \text{for } i = 1, \dots, n-1, \end{aligned} \quad (4.37)$$

and that the following inequality holds:

$$\max \left\{ \begin{aligned} &\left| \frac{\partial_{\xi_1(0,t)} f_1(0)}{\partial_{\xi_{n+1}(0,t)} f_1(0)} \frac{\partial_{\xi_{n+1}(L,t)} f_3(0)}{\partial_{\xi_1(L,t)} f_3(0)} \right|, \\ &\left| \frac{\partial_{\xi_{2n}(L,t)} f_{2n}(0)}{\partial_{\xi_n(L,t)} f_{2n}(0)} \frac{\partial_{\xi_n(0,t)} f_{2n-2}(0)}{\partial_{\xi_{2n}(0,t)} f_{2n-2}(0)} \right|, \\ &\left| \frac{\partial_{\xi_{i+1}(0,t)} f_{2i}(0)}{\partial_{\xi_{n+i+1}(0,t)} f_{2i}(0)} \frac{\partial_{\xi_{n+i+1}(L,t)} f_{2i+3}(0)}{\partial_{\xi_{i+1}(L,t)} f_{2i+3}(0)} \right|; \quad i = 1, \dots, n-2 \end{aligned} \right\} < 1. \quad (4.38)$$

Then, there exist positive constants ϵ , M , μ such that, if the initial condition is small enough:

$$\|\xi^h(\cdot)\|_{C^1} \leq \epsilon \quad (4.39)$$

there is a unique solution $\xi(x, t)$ of class C^1 on $[0, L] \times [0, \infty)$ which decays to zero with an exponential rate:

$$\|\xi(t, \cdot)\|_{C^1} \leq M e^{-\mu t} \quad (4.40)$$

Proof. Note first that

$$\mathbf{f}(0) = 0.$$

Let $A \in M_{2n, 2n}$ such that

$$A = \nabla_{[\xi_-(L,t), \xi_+(0,t)]} \mathbf{f}(0) .$$

The non-zero elements of $A = (a_{ij})$ are

$$\begin{aligned} a_{1,n+1} &= \partial_{\xi_{n+1}(0,t)} f_1(0), \\ a_{2n,n} &= \partial_{\xi_n(L,t)} f_{2n}(0), \\ a_{2i,i} &= \partial_{\xi_i(L,t)} f_{2i}(0), \quad a_{2i,n+i+1} = \partial_{\xi_{n+i+1}(0,t)} f_{2i}(0) \\ a_{2i+1,i} &= \partial_{\xi_i(L,t)} f_{2i+1}(0). \end{aligned} \quad (4.41)$$

Due to the particular form of A , it can be easily diagonalized. Indeed let $U \in M_{2n,2n}$ such that $U = (u_{ij})$ where non-zero u_{ij} are defined by

$$\begin{aligned} u_{n+1,1} &= 1, \\ u_{n,2n} &= 1, \\ u_{i,2i} &= 1 \quad \text{for } i = 1, \dots, n-1, \\ u_{n+i+1,2i+1} &= 1 \text{ for } i = 1, \dots, n-1. \end{aligned} \quad (4.42)$$

One can see that U is orthogonal:

$$U^T U = I.$$

It follows from the definition of U that AU is a block diagonal matrix:

$$AU = \begin{pmatrix} D_0 & & \\ & \ddots & \\ & & D_n \end{pmatrix}$$

where D_i are defined by

$$\begin{aligned} D_0 &= a_{1,n+1}, \\ D_i &= \begin{pmatrix} a_{2i,i} & a_{2i,n+i+1} \\ a_{2i+1,i} & 0 \end{pmatrix} \quad \text{for } i = 1, \dots, n-1, \\ D_n &= a_{2n,n}. \end{aligned}$$

Since $\det U = -1$, the condition for A to apply the Inverse Function theorem can be rewritten

$$\det A = \det(AU) \neq 0 \Leftrightarrow \det(D_i) \neq 0 \quad \text{for } i = 0, \dots, n,$$

or, by using (4.41), this leads to (4.37).

Let $B \in M_{2n,2n}$ such that

$$B = \nabla_{[\xi_-(0), \xi_+(L)]} \mathbf{f}(0) .$$

The non-zero elements of B are

$$\begin{aligned} b_{1,1} &= \partial_{\xi_1(0,t)} f_1(0), \\ b_{2i,i+1} &= \partial_{\xi_{i+1}(0,t)} f_{2i}(0), \quad b_{2i,n+i} = \partial_{\xi_{n+i}(L,t)} f_{2i}(0), \\ b_{2i+1,n+i} &= \partial_{\xi_{n+i}(L,t)} f_{2i+1}(0), \\ b_{2n,2n} &= \partial_{\xi_{2n}(L,t)} f_{2n}(0) \end{aligned} \quad (4.43)$$

Remark 4.3. We have laid out $b_{2i,n+i}$, $b_{2i+1,n+i}$ and $b_{2n,2n}$ in a separate column to emphasize that these terms are in the right part ($b_{i,n+\dots}$) of the matrix.

Again, B can be diagonalized. Let $V \in M_{2n,2n}$ such that $V = (v_{ij})$ where non-zero v_{ij} are defined by

$$\begin{aligned} v_{1,1} &= 1, \\ v_{2n,2n} &= 1 \\ v_{n+i,2i} &= 1 \quad \text{for } i = 1, \dots, n-1, \\ v_{i+1,2i+1} &= 1 \quad \text{for } i = 1, \dots, n-1. \end{aligned} \quad (4.44)$$

One can see that V is orthogonal. It follows from the definition of V that BV is a block diagonal matrix:

$$BV = \begin{pmatrix} E_0 & & \\ & \ddots & \\ & & E_n \end{pmatrix}$$

where E_i are defined by

$$\begin{aligned} E_0 &= b_{1,1}, \\ E_i &= \begin{pmatrix} b_{2i,n+i} & b_{2i,i+1} \\ b_{2i+1,n+i} & 0 \end{pmatrix} \quad \text{for } i = 1, \dots, n-1, \\ E_n &= b_{2n,2n}. \end{aligned}$$

Let us estimate the product $(AU)^{-1}(BV)$ by using the diagonalized versions of A and B ,

$$C = (AU)^{-1}(BV) = \begin{pmatrix} D_0^{-1}E_0 & & \\ & \ddots & \\ & & D_n^{-1}E_n \end{pmatrix} \quad (4.45)$$

where the products $D_i^{-1}E_i$ are

$$\begin{aligned} D_0^{-1}E_0 &= \frac{b_{1,1}}{a_{1,n+1}}, \\ D_n^{-1}E_n &= \frac{b_{2n,2n}}{a_{2n,n}}, \\ D_i^{-1}E_i &= \frac{-1}{a_{2i+1,i}a_{2i,n+i+1}} \begin{pmatrix} 0 & -a_{2i,n+i+1} \\ -a_{2i+1,i} & a_{2i,i} \end{pmatrix} \begin{pmatrix} b_{2i,n+i} & b_{2i,i+1} \\ b_{2i+1,n+i} & 0 \end{pmatrix} \\ &= \begin{pmatrix} \frac{b_{2i+1,n+i}}{a_{2i+1,i}} & 0 \\ \frac{a_{2i+1,i}b_{2i,n+i} - a_{2i,i}b_{2i+1,n+i}}{a_{2i+1,i}a_{2i,n+i+1}} & \frac{b_{2i,i+1}}{a_{2i,n+i+1}} \end{pmatrix}. \end{aligned} \quad (4.46)$$

By using (4.45), the inverse theorem and the fact that U and V are orthogonal matrices, we have

$$\nabla \mathbf{G}(0) = -A^{-1}B = -UCV^T. \quad (4.47)$$

where

$$\nabla \mathbf{G} = \begin{pmatrix} 0 & Q \\ R & S \end{pmatrix} \quad (4.48)$$

where $Q, R, S \in M_{n,n}$, Q and R are diagonal matrices and S is a lower diagonal matrix. By using (4.48), (4.45), (4.42) and (4.44), one can see that Q, R, S are defined by

$$\begin{aligned} Q &= \text{diag}(c_{2,2}, \dots, c_{2i,2i}, \dots, c_{2n,2n}), \\ R &= \text{diag}(c_{1,1}, \dots, c_{2i-1,2i-1}, \dots, c_{2n-1,2n-1}), \\ S &= \begin{pmatrix} 0 & & & & \\ c_{3,2} & 0 & & & \\ & \ddots & \ddots & & \\ & & c_{2i+1,2i} & 0 & \\ & & & \ddots & \ddots \\ & & & & c_{2n,2n-1} & 0 \end{pmatrix}. \end{aligned} \quad (4.49)$$

By the definition of the matrix product and by using (4.47), we have

$$\nabla \mathbf{G}_{ij}(0) = - \sum_{k=1}^{2n} \sum_{l=1}^{2n} u_{ik} c_{kl} v_{jl},$$

Hence, by using (4.42) and (4.44),

$$\begin{aligned} r_{1,1} &= \nabla \mathbf{G}_{n+1,1} \\ &= u_{n+1,1} c_{1,1} v_{1,1} = c_{1,1}, \\ r_{i,i} &= \nabla \mathbf{G}_{n+i+1,i+1} \\ &= (0 \dots 0 u_{n+i+1,2i+1} 0 \dots 0) C (0 \dots 0 v_{i+1,2i+1} 0 \dots 0)^T \\ &= c_{2i+1,2i+1}, \quad \text{for } i = 1, \dots, n-1, \\ q_{i,i} &= \nabla \mathbf{G}_{i,n+i} \\ &= (0 \dots 0 u_{i,2i} 0 \dots 0) C (0 \dots 0 v_{n+i,2i} 0 \dots 0)^T \\ &= c_{2i,2i}, \quad \text{for } i = 1, \dots, n-1, \\ q_{n,n} &= \nabla \mathbf{G}_{n,2n} \\ &= u_{n,2n} c_{2n,2n} v_{2n,2n} = c_{2n,2n}, \\ s_{i+1,i} &= \nabla \mathbf{G}_{n+i+1,n+i} \\ &= (0 \dots 0 u_{n+i+1,2i+1} 0 \dots 0) C (0 \dots 0 v_{n+i,2i} 0 \dots 0)^T \\ &= c_{2i+1,2i}, \quad \text{for } i = 1, \dots, n-1. \end{aligned}$$

By using Lemma 3.14 and (4.49), the spectral radius of $\text{abs}(\nabla \mathbf{G}(0))$ can be computed as follows:

$$\rho(\text{abs}(\nabla \mathbf{G}(0))) = \max\{ |c_{2i-1,2i-1} c_{2i,2i}|^{1/2} ; i = 1, \dots, n\},$$

or, by using (4.45) and (4.46),

$$\rho(\text{abs}(\nabla \mathbf{G}(0))) = \max \left\{ \left| \frac{b_{1,1}}{a_{1,n+1}} \frac{b_{3,n+1}}{a_{3,1}} \right|^{1/2}, \left| \frac{b_{2n,2n}}{a_{2n,n}} \frac{b_{2n-2,n}}{a_{2n-2,2n}} \right|^{1/2}, \left| \frac{b_{2i,i+1}}{a_{2i,n+i+1}} \frac{b_{2i+3,n+i+1}}{a_{2i+3,i+1}} \right|^{1/2}; i = 1, \dots, n-2 \right\} \quad (4.50)$$

Using (4.41), (4.43) and (4.38), we have

$$\rho(\text{abs}(\nabla \mathbf{G}(0))) < 1$$

The asymptotic stability is a direct application of Theorem 2.28.

This ends the proof. $\square$

Example 4.4. This example illustrates the above proof.

Let us consider a system composed of two reaches in cascade as depicted in Figure 4.7. In this case, (4.33) is:

$$\xi_- = (\xi_1, \xi_2)^T, \quad \xi_+ = (\xi_3, \xi_4)^T \quad (4.51)$$

and the boundary conditions (4.35),

$$\mathbf{f} \begin{pmatrix} \xi_-(0, t) \\ \xi_-(L, t) \\ \xi_+(0, t) \\ \xi_+(L, t) \end{pmatrix} = \begin{pmatrix} f_1(\xi_1(0, t), \xi_3(0, t)) \\ f_2(\xi_2(0, t), \xi_1(L, t), \xi_4(0, t), \xi_3(L, t)) \\ f_3(\xi_1(L, t), \xi_3(L, t)) \\ f_4(\xi_2(L, t), \xi_4(L, t)) \end{pmatrix} = 0 \quad (4.52)$$

Let us explicitly compute (4.36) for the example system:

$$\begin{aligned} \nabla_{[\xi_-(L, t), \xi_+(0, t)]} \mathbf{f}(0) &= \begin{pmatrix} 0 & 0 & \partial_{\xi_3(0)} f_1(0) & 0 \\ \partial_{\xi_1(L)} f_2(0) & 0 & 0 & \partial_{\xi_4(0)} f_2(0) \\ \partial_{\xi_1(L)} f_3(0) & 0 & 0 & 0 \\ 0 & \partial_{\xi_2(L)} f_4(0) & 0 & 0 \end{pmatrix}, \\ \nabla_{[\xi_-(0, t), \xi_+(L, t)]} \mathbf{f}(0) &= \begin{pmatrix} \partial_{\xi_1(0)} f_1(0) & 0 & 0 & 0 \\ 0 & \partial_{\xi_2(0)} f_2(0) & \partial_{\xi_3(L)} f_2(0) & 0 \\ 0 & 0 & \partial_{\xi_3(L)} f_3(0) & 0 \\ 0 & 0 & 0 & \partial_{\xi_4(L)} f_4(0) \end{pmatrix} \end{aligned} \quad (4.53)$$

Let us substitute (4.62) in (4.52),

$$\begin{pmatrix} f_1(\xi_1(0, t), \xi_3(0, t)) \\ f_2(\xi_2(0, t), \xi_1(L, t), \xi_4(0, t), \xi_3(L, t)) \\ f_3(\xi_1(L, t), \xi_3(L, t)) \\ f_4(\xi_2(L, t), \xi_4(L, t)) \end{pmatrix} = \begin{pmatrix} \xi_3(0, t) + k_0 \xi_1(0, t) \\ f_2(\xi_2(0, t), \xi_1(L, t), \xi_4(0, t), \xi_3(L, t)) \\ \xi_1(L, t) + k_1 \xi_3(L, t) \\ \xi_2(L, t) + k_2 \xi_4(L, t) \end{pmatrix} \quad (4.54)$$

and (4.53) becomes

$$\begin{aligned} A &= \nabla_{[\xi_{-}(L,t), \xi_{+}(0,t)]} \mathbf{f}(0) = \begin{pmatrix} 0 & 0 & 1 & 0 \\ a_{21} & 0 & 0 & a_{24} \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix}, \\ B &= \nabla_{[\xi_{-}(0,t), \xi_{+}(L,t)]} \mathbf{f}(0) = \begin{pmatrix} k_0 & 0 & 0 & 0 \\ 0 & b_{22} & b_{23} & 0 \\ 0 & 0 & k_1 & 0 \\ 0 & 0 & 0 & k_2 \end{pmatrix} \end{aligned} \quad (4.55)$$

where $a_{21} = \partial_{\xi_1(L)} f_2(0)$, $a_{24} = \partial_{\xi_4(0)} f_2(0)$, $b_{22} = \partial_{\xi_2(0)} f_2(0)$, $b_{23} = \partial_{\xi_3(L)} f_2(0)$.

By using (4.41), U and V are computed as

$$U = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix}, \quad D = AU = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & a_{21} & a_{24} & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (4.56)$$

and the suitability conditions (4.37) rewrites as follows:

$$a_{24} = \partial_{\xi_4(0)} f_2(0) \neq 0. \quad (4.57)$$

Similarly, V and E are computed as follows

$$V = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad E = BV = \begin{pmatrix} k_0 & 0 & 0 & 0 \\ 0 & b_{23} & b_{22} & 0 \\ 0 & k_1 & 0 & 0 \\ 0 & 0 & 0 & k_2 \end{pmatrix}. \quad (4.58)$$

Hence, by using (4.45), C is computed as

$$C = D^{-1}E = \begin{pmatrix} k_0 & & & \\ & E_1 & & \\ & & & k_2 \end{pmatrix} \quad (4.59)$$

where

$$E_1 = \frac{-1}{a_{24}} \begin{pmatrix} 0 & -a_{24} \\ -1 & a_{22} \end{pmatrix} \begin{pmatrix} b_{23} & b_{22} \\ k_1 & 0 \end{pmatrix} = \frac{1}{a_{24}} \begin{pmatrix} a_{24}k_1 & 0 \\ b_{23}a_{24} - k_1a_{22} & b_{22} \end{pmatrix}.$$

Substituting (4.56), (4.58) and (4.59) in (4.47) leads to

$$\nabla \mathbf{G}(0) = - \left(\begin{array}{cc|cc} 0 & 0 & k_1 & 0 \\ 0 & 0 & 0 & k_2 \\ \hline k_0 & 0 & 0 & 0 \\ 0 & \frac{b_{22}}{a_{24}} & \frac{b_{23}a_{24} - k_1a_{22}}{a_{24}} & 0 \end{array} \right) \quad (4.60)$$

where Q, R and S of (4.49) are clearly identified. Hence, using Lemma 3.14, the eigenvalues of $\nabla \mathbf{G}(0)$ are

$$\Lambda = \left\{ -k_0 k_1, -\frac{a_{24}}{b_{22}} k_2 \right\},$$

which is (4.38) for this particular system.

4.4 Practical case: Sambre river

The purpose of this last section is to investigate the applicability of this control law to the Sambre river in Belgium

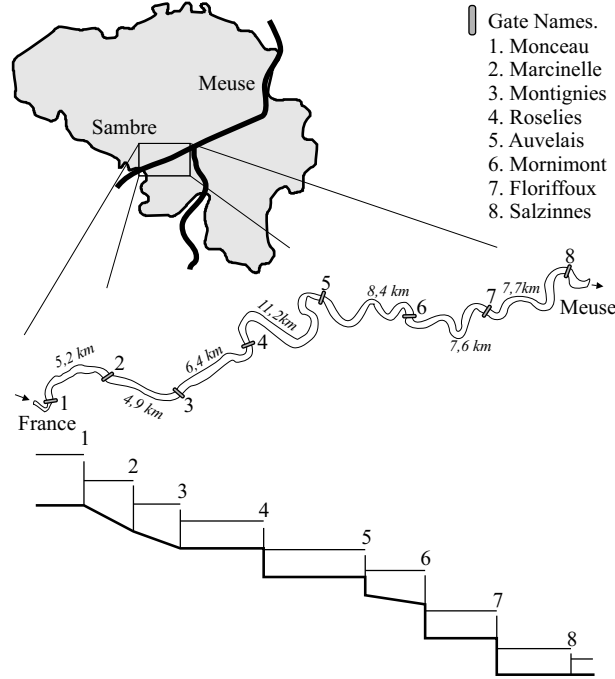


Fig. 4.8. Schematic view of Sambre river from *Monceau* to *Salzinnes* gate.

4.4.1 Description and model

The Sambre is a class IV (up to 1350T boat) water-way that starts in France and is a tributary of the *Meuse* river in Belgium as depicted in Figure 4.8. The studied part of the Sambre - from Monceau to Salzinnes - is composed of 7 reaches separated by spillways with an average width of 40m. The river is modelled by a cascade of prismatic reaches as depicted in the Figure 4.8.

4.4.2 Stability condition

Let us consider the system of channels in cascade described in

By design, we choose simple boundary conditions, linear in the ξ coordinates:

$$\begin{aligned} f_1 &= \xi_{n+1}(0, t) + k_0 \xi_1(0, t), \\ f_{2i+1} &= \xi_i(L, t) + k_i \xi_{n+i}(L, t), \quad \text{for } i = 1, \dots, n. \end{aligned} \quad (4.61)$$

where k_i are positive constraints used for the tuning of the controller.

The stability conditions of these control law turns out to be surprisingly simple as this can be seen in the following result:

Proposition 4.5. *For control law described in (4.61), the stability condition (4.38) particularizes*

$$\max \left\{ \left| \frac{a_{2i, n+i+1}}{b_{2i, 2i}} \right| k_i, \quad k_1 k_0 \quad \text{for } i = 2, \dots, n \right\} < 1 \quad (4.62)$$

where a_{ij} is defined in (4.41).

Remark 4.6. The stability condition (4.62) has several interesting properties. First, the terms in the maximum imply coupled gain like $k_0 k_1$. This implies that k_0 is a constant that cannot be modified, one can still use k_1 to stabilize the system. In the general case, it usually turns out that one of the parameter is fixed by the physics of the problem (e.g. k_0) and only one parameter is available as a control action. This fact makes the "coupling" property crucial in most situations.

The other interesting property comes from the fact that the stability condition is normalized "normalized": A control designer can tune its controllers from a stability condition equal to 0 for "lazy" controllers, to 1 for "nervous" controllers.

4.4.3 Simulation results

In order to get a more realistic model, the reaches are described by Saint Venant equations with friction terms of the Manning-Strickler type, $K = 50$, (see (7.5) and a slight slope for some of them.

A flood wave is injected at the first gate (*Monceau*). The simulation results for open loop and closed loop strategies are presented at Figures 4.9, 4.10, 4.11.

Figure 4.9 shows the hydrogram of each gate - the water flow passing through each gate -. One can see that the flood wave is constantly dampened for the open loop case, whereas in the closed loop case, the flood wave crosses the gates with little modification. As this will be emphasized in the next paragraph, the open loop case "pays the price" of such behavior by producing

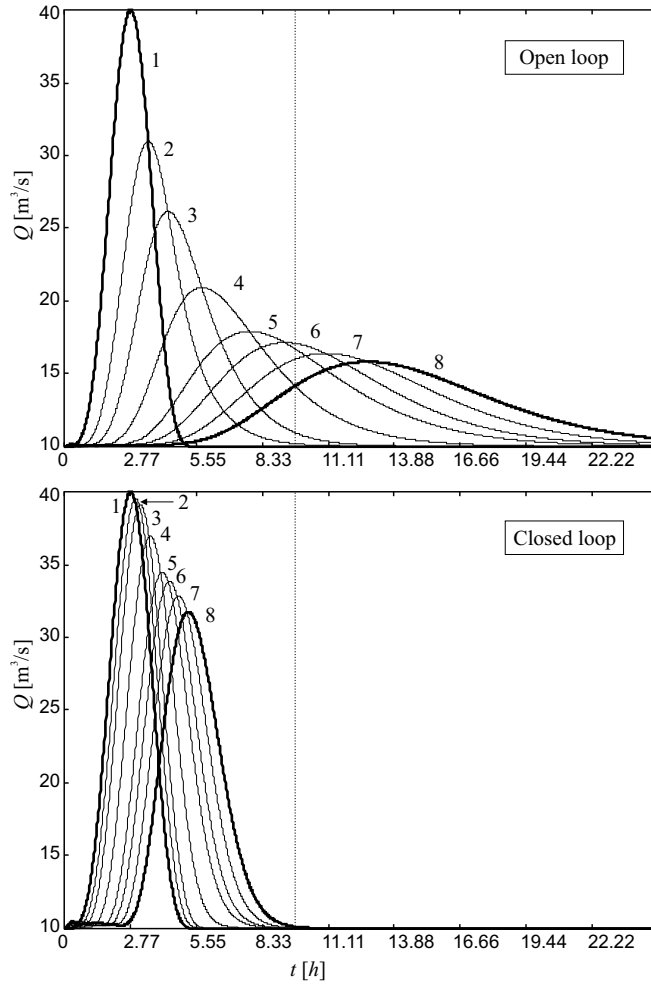


Fig. 4.9. Hydrogram for gates 1 to 8.

greater deviation of the water depth than the closed loop case. We can also appreciate the acceleration of convergence for the closed loop case.

Figure 4.10 shows the limnigram of each gate - the deviation of the water depth at each gate -. The open loop, by dampening the water flow, generates high deviation of the water depth (up to 40cm). The closed loop, which does not hold the flow, yields to smaller deviation of the water depth (up to 12cm). Again, one can see that the equilibrium is reached much faster in the closed loop case.

The acceleration of the convergence can also be assessed in Fig. 4.11 where the entropy is represented. Indeed the response time in closed loop (9 hours) is much smaller than in open loop (20 hours).

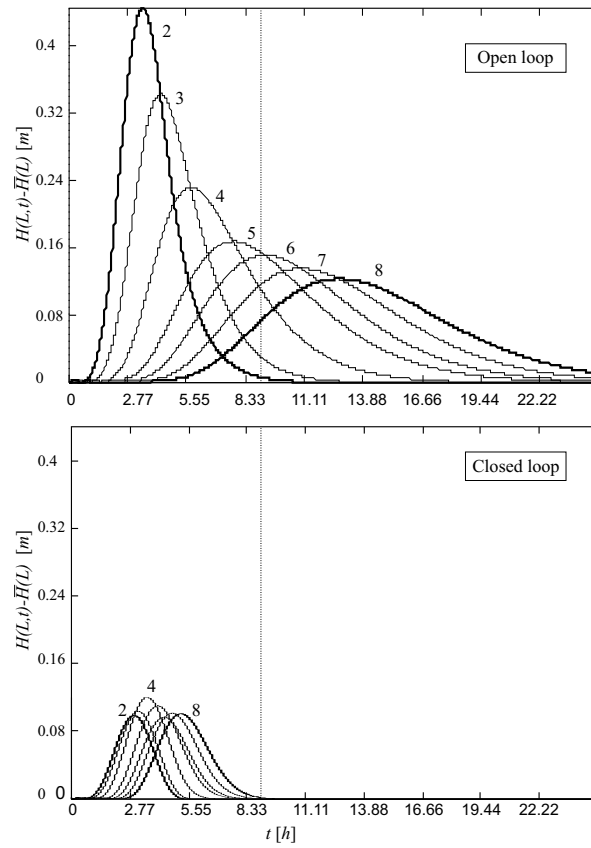


Fig. 4.10. Limnigram for gates 2 to 8.

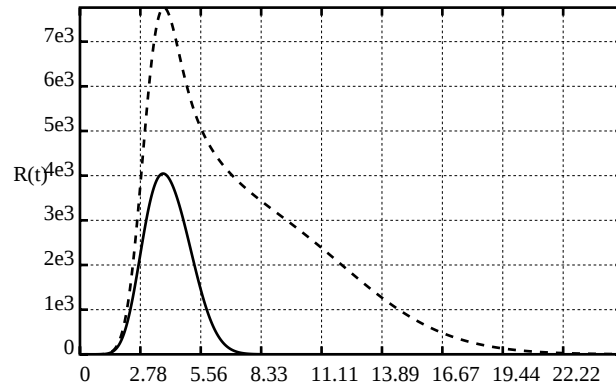


Fig. 4.11. Entropy R for Sambre river. Plain curve, Riemann based control. Dotted curve, open loop control.

Control design based on Entropy

5.1 Introduction

The purpose of this chapter is to show how an entropy function can be used to build a Lyapunov function candidate in order to design boundary controllers for systems that are described by quasilinear hyperbolic partial differential equations. The approach is illustrated with a case study, namely the flow control in open-channels described by Saint-Venant equations.

5.2 Entropies

Physically based hyperbolic systems are usually conservative. Many of them also admit an additional convex conservation law under the form

$$\partial_t E(u) + \partial_x F(u) = 0 \quad (5.1)$$

where $E(u)$ is an *entropy* function and $F(u)$ its associated *entropy flux*.

Entropies have been thoroughly studied in the following articles and books: [64, Volume I, Chapter 3.4] and [47, 39, 15, 10].

Remark 5.1 (Entropy and weak solutions). As outlined in Theorem 2.11, it is well known that the Cauchy problem (2.1)-(2.2)-(2.3) does not have globally defined smooth solutions; hence only discontinuous solutions may exist in the large. One of the main features of conservation laws is that uniqueness is lost within the class of the discontinuous solutions in the sense of distributions; many weak solutions may share the same initial data. Thus the problem arises of identifying an appropriate class of weak solutions, entropy solutions, to single out physically relevant solutions.

In the following, the directional derivative of a function $\varphi = \varphi(u)$, in the direction of the vector $r(u)$, is written

$$D_r \varphi(u) \doteq D\varphi(u) \cdot r(u) = \lim_{\varepsilon \rightarrow 0} \frac{\varphi(u + \varepsilon r(u)) - \varphi(u)}{\varepsilon} \quad (5.2)$$

Definition 5.2. A continuously differentiable function $E : \mathbb{R}^n \mapsto \mathbb{R}$ is called an **entropy** for the system (2.1), with **entropy flux** $F : \mathbb{R}^n \mapsto \mathbb{R}$, if

$$D_u E(u)A(u) = D_u F(u), \quad u \in \mathbb{R}^n. \quad (5.3)$$

The pair $(E(u), F(u))$ is called the **entropy pair**. The entropy $E(u)$ is convex on the domain $K \in \mathbb{R}^n$ if the Hessian matrix $H = H(u) = D^2 E(u) \geq 0$, for $u \in K$. The entropy is strictly convex on the domain K if $H > 0$, for $u \in K$.

Observe that (5.3) implies that (5.1). Indeed, (2.1) yields

$$D_u E(u)u_t + D_u F(u)u_x = D_u E(u)(-A(u)u_x) + D_u F(u)u_x = 0. \quad (5.4)$$

In other words, whenever we have a smooth solution to (2.1), not only the quantities $u_1, \dots, u_n$ are conserved, but the additional conservation law (5.1) holds as well. However, one should be aware that when u is discontinuous, in general, it does not provide a weak solution to (5.1), i.e. $E = E(u)$ is not a conserved quantity.

For a 2×2 strictly hyperbolic system, Lax's theorem (see [46, 64]) indicates that, given any bounded domain $K \subset \mathbb{R}^2$, there exists a strictly convex entropy pair $(E(u), F(u))$ on the domain K .

For $m \leq 3$, the system (5.3) is over-determined, thereby generally preventing the existence of nontrivial entropies.

5.3 Boundary Control design methodology

Assume now that we have the objective of designing boundary controls in order to regulate the state $u(x, t)$ of system (2.1) at a constant set point $\bar{u}$. The idea is to select a suitable entropy $E(u)$ such that the integral of the entropy over the space interval $[0, 1]$:

$$R = \int_0^1 E(u) dx, \quad (5.5)$$

is a Lyapunov function candidate, i.e. a positive definite function which is zero only when $u = \bar{u}$.

Along the smooth solutions of (2.1), the time derivative of R is then, using (5.1) :

$$\dot{R} = -[F(u)]_0^1. \quad (5.6)$$

The key point for the control design is that $\dot{R}$ depends only on the values of u at the boundaries $x = 0$ and $x = 1$. It is then quite natural to use these boundary conditions as the control actions to make $\dot{R}$ negative.

5.4 Application to Saint Venant equation

In this section, we shall illustrate the feasibility of this control strategy by considering, as a case study, the problem of the flow control in an open channel whose dynamics are described by the Saint-Venant equations (also called shallow water equations).

The special case of a single reach has been treated previously in [18]. Here, we shall consider the more general and more difficult situation of two reaches in cascade as depicted in Fig. 5.1. Note that we are considering another type of hydraulic gates from the Spillway described in Chapter 4, namely *sluice gates* which lets the water flow under itself.

From a mathematical point of view, a sluice gate discharge relationship induces more coupling of the system because they depend on the upstream and downstream water states. Recall that a spill way discharge relationships depends on the flow and upstream water depth.

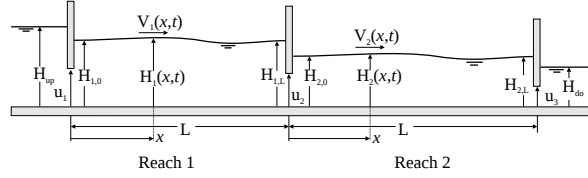


Fig. 5.1. The two horizontal reaches

5.4.1 Model

The dynamics of each reach are described by Saint-Venant equations. The beds of the two reaches are supposed to be horizontal and the friction effects are neglected. The following notations are introduced :

- For the sake of simplicity, we assume that both reaches have the same length L ,
- $H_i(x, t)$ is the water level in the i -th reach ($i = 1, 2$) :

$$H_{i,0} = H_i(0, t) \quad \text{and} \quad H_{i,L} = H_i(L, t)$$

- $V_i(x, t)$ is the water speed in the i -th reach :

$$V_{i,0} = V_i(0, t) \quad \text{and} \quad V_{i,L} = V_i(L, t)$$

- H_{up} and H_{do} are the left and right water levels outside the canal.

For each reach ($i=1,2$), the Saint-Venant equations are written :

$$\partial_t \begin{pmatrix} H_i \\ V_i \end{pmatrix} + \begin{pmatrix} V_i & H_i \\ g & V_i \end{pmatrix} \partial_x \begin{pmatrix} H_i \\ V_i \end{pmatrix} = 0 \quad (5.7)$$

This model is clearly a special case of the general hyperbolic system (2.1) with $n = 4$, $u = (H_1, V_1, H_2, V_2)$ and the bloc diagonal matrix

$$A(Y) = \text{diag} \left\{ \begin{pmatrix} V_i & H_i \\ g & V_i \end{pmatrix}, i = 1, 2 \right\} .$$

The flows are assumed to be subcritical:

$$V_i < \sqrt{gH_i} \quad \text{for } i = 1, 2 . \quad (5.8)$$

The control actions are provided by the three gate openings denoted w_1 for the left gate, w_2 for the intermediate gate and w_3 for the right gate.

The discharge relationships for the three gates are given by the following standard expression from hydraulic engineering (e.g. [33, Chapter 4.4]) :

$$\begin{aligned} f_1 &= (V_{1,0}H_{1,0})^2 - w_1(H_{\text{up}} - H_{1,0}) = 0 \\ f_2 &= (V_{1,L}H_{1,L})^2 - w_2(H_{1,L} - H_{2,0}) = 0 \\ f_4 &= (V_{2,L}H_{2,L})^2 - w_3(H_{2,L} - H_{\text{do}}) = 0 \end{aligned} \quad (5.9)$$

An equation of flow conservation is added at the intermediate gate for model completion :

$$f_3 = V_{1,L}H_{1,L} - V_{2,0}H_{2,0} = 0 \quad (5.10)$$

Equations (5.9) and (5.10) constitute the boundary conditions at $x = 0$, $x = L$ associated to the PDEs (5.7).

5.4.2 Steady-states

For given gate openings $\bar{w}_1$, $\bar{w}_2$, $\bar{w}_3$ there exists a steady-state solution $(\bar{V}_1, \bar{H}_1, \bar{V}_2, \bar{H}_2)$ of the Saint-Venant equations which satisfies, from (5.9) and (5.10), the following relations :

$$\begin{aligned} \bar{H}_1 &= \frac{\bar{w}_1 H_{\text{up}} (\bar{w}_2 + \bar{w}_3) + \bar{w}_3 H_{\text{do}} \bar{w}_2}{\bar{w}_1 \bar{w}_2 + \bar{w}_2 \bar{w}_3 + \bar{w}_1 \bar{w}_3}, \\ \bar{H}_2 &= \frac{\bar{w}_3 H_{\text{do}} (\bar{w}_1 + \bar{w}_2) + \bar{w}_1 H_{\text{up}} \bar{w}_2}{\bar{w}_1 \bar{w}_2 + \bar{w}_2 \bar{w}_3 + \bar{w}_1 \bar{w}_3}, \\ \bar{V}_1 &= \frac{\sqrt{\bar{w}_1 (H_{\text{up}} - \bar{H}_1)}}{\bar{H}_1}, \\ \bar{V}_2 &= \frac{\sqrt{\bar{w}_3 (\bar{H}_2 - H_{\text{do}})}}{\bar{H}_2}. \end{aligned} \quad (5.11)$$

Observe that these steady state values satisfy the flow conservation condition $\bar{V}_1 \bar{H}_1 = \bar{V}_2 \bar{H}_2$. Of course for $(\bar{V}_1, \bar{H}_1, \bar{V}_2, \bar{H}_2)$ to make sense, we must have :

$$H_{\text{up}} > \bar{H}_1 > \bar{H}_2 > H_{\text{do}} \quad \text{and} \quad \bar{V}_1 < \bar{V}_2 . \quad (5.12)$$

5.4.3 Statement of the control problem for the two reach channel

The control objective is to stabilize the water levels H_1 and H_2 and the water velocities V_1 and V_2 at given set points $(\bar{V}_1, \bar{H}_1, \bar{V}_2, \bar{H}_2)$ with the constraint $\bar{V}_1 \bar{H}_1 = \bar{V}_2 \bar{H}_2$. The control actions are the three gate openings u_1 , u_2 and u_3 . The levels $H_{1,0}(t)$, $H_{1,L}(t)$, $H_{2,0}(t)$ and $H_{2,L}(t)$ are supposed to be measured on line at each time instant t . The external water levels H_{up} and H_{do} are known.

5.5 Entropies of shallow water equations

It is interesting to note that shallow water equations admit an infinite number of entropy and entropy flux pairs satisfying (5.1):

$$\partial_t E(H, V) + \partial_x F(H, V) = 0 . \quad (5.13)$$

Given the shallow water equation (5.7), it is only necessary that

$$\begin{aligned} \partial_H F &= g \partial_V E + V \partial_H E \\ \partial_V F &= V \partial_H E + H \partial_V E . \end{aligned}$$

Thus, any solution of

$$g \frac{\partial^2}{\partial V^2} E + H \frac{\partial^2}{\partial H^2} E$$

will lead to a conservation equation. The most interesting entropies are polynomials in H and V . For instance, some may be obtained consistently by taking

$$P = \sum_{m=0}^n p_m(V) H^m , \quad (5.14)$$

from which it follows that

$$\begin{aligned} g p_{m-1}'' &= m(m-1) p_m , & m &= 2, \dots, n , \\ p_0'' &= 0 , \\ p_n'' &= 0 . \end{aligned} \quad (5.15)$$

The first few are:

Interpretation	E	F
	V	$\frac{1}{2} V^2 + gH$
Mass	H	HV
Momentum	HV	$V^2 H + \frac{1}{2} g H^2$
Energy	$\frac{1}{2} V^2 H + \frac{1}{2} g H^2$	$\frac{1}{2} V^3 H + V g H$

5.6 Lyapunov control design

We choose a modified version of the "energy" entropy for a single reach. This entropy was "shifted" in order to have be zero at the steady state, $R(\bar{V}, \bar{E}) = 0$,

$$E(V_i, H_i) = H_i \frac{(V_i - \bar{V}_i)^2}{2} + g \frac{(H_i - \bar{H}_i)^2}{2} . \quad (5.16)$$

The corresponding entropy flux is

$$F(V_i, H_i) = V_i H_i \frac{(V_i - \bar{V}_i)^2}{2} + g V_i H_i (H_i - \bar{H}_i) - g \bar{V}_i \frac{H_i^2}{2} . \quad (5.17)$$

Remark 5.3. Note that E and F defined above still satisfy (5.13).

As described in Section 5.3, a natural choice of a Lyapunov candidate is the integral of the entropy along the spatial domain, as defined in (5.5):

$$R = \int_0^L E(u) dx .$$

To make $\dot{R}$ negative, we may use $V_{1,0}$ and $V_{2,L}$ as control variables at the first and third gate, since we know from (5.9) that $V_{1,0}$ and $V_{2,L}$ are related to the control inputs w_1 and w_3 via the following relations :

$$w_1 = \frac{(V_{1,0} H_{1,0})^2}{H_{up} - H_{1,0}} \quad \text{and} \quad w_3 = \frac{(V_{2,L} H_{2,L})^2}{H_{2,L} - H_{do}} . \quad (5.18)$$

For the intermediate gate, we consider the water flow $Q_{1,L}$ as control variable :

$$w_2 = \frac{Q_{1,L}^2}{H_{1,L} - H_{2,0}} . \quad (5.19)$$

In fact, we shall consider small deviations around the equilibrium, so that, for the Lyapunov analysis we introduce the following control variables :

$$v_1 = V_{1,0} - \bar{V}_1, \quad v_3 = V_{2,L} - \bar{V}_2, \quad q_2 = H_{1,L} V_{1,L} - \bar{Q} \quad (5.20)$$

where $\bar{Q} = \bar{H}_1 \bar{V}_1 = \bar{H}_2 \bar{V}_2$.

Let us introduce also the following deviations of the water depths at the intermediate gate as shown in Figure 5.2 :

$$h_1 = H_{1,L} - \bar{H}_1, \quad \text{and} \quad h_2 = H_{2,0} - \bar{H}_2 . \quad (5.21)$$

Using the above notation, $\dot{R}$, defined in (5.6), is as follows:

$$\begin{aligned} \dot{R} &= -[F(u)]_0^L \\ &= F(V_{1,0}, H_{1,0}) \\ &\quad - k F(V_{2,0}, H_{2,0}) - F(V_{1,L}, H_{1,L}) \\ &\quad - k F(V_{2,L}, H_{2,L}) . \end{aligned} \quad (5.22)$$

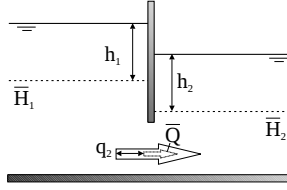


Fig. 5.2. Deviation variables at intermediate gate.

We can now propose a class of boundary controllers making R decrease, as stated in the following proposition. This class of controllers is an extension of the single reach case studied in [18].

Proposition 5.4. *If (5.8) holds then there exist $\delta > 0$, such that the boundary control laws w_1 , w_2 and w_3 defined by (5.18) and (5.19) where v_1 , v_3 and q_2 are given by :*

$$\begin{aligned} v_1 &:= -\left(1 - \frac{\bar{H}_1}{H_{1,0}}\right)\left(\frac{\bar{V}_1}{2} + \lambda_1\right), \\ v_3 &:= -\left(1 - \frac{\bar{H}_2}{H_{2,L}}\right)\left(\frac{\bar{V}_2}{2} - \lambda_3\right), \end{aligned} \quad (5.23)$$

$$q_2 := \alpha_1 h_1 - \alpha_2 h_2, \quad (5.24)$$

make $\dot{R}$ decrease, i.e. $\dot{R} < 0$ in a neighborhood of the equilibrium, for λ_1 , λ_3 , k , α_1 and α_2 such that :

$$\left\{ \begin{array}{l} \lambda_1 > 0 \in [r_1, r_2] \text{ with} \\ r_1 = \frac{2g\bar{H}_1(1 - \sqrt{1 - \bar{V}^2/(g\bar{H})}) - \bar{V}^2}{2\bar{V}}, \\ r_2 = \frac{2g\bar{H}_1(1 + \sqrt{1 - \bar{V}^2/(g\bar{H})}) - \bar{V}^2}{2\bar{V}}, \\ \lambda_3 > 0, \\ 0 < \alpha_2 < \frac{a_4}{2a_6}, \\ \alpha_1 > \max \left\{ \begin{array}{l} \frac{-a_3 + \sqrt{a_3^2 + 4a_1(a_5 - \delta)}}{2(a_5 - \delta)} \\ \frac{(4a_5 + a_3)\alpha_2}{2(\sqrt{\delta}\mu_2)} \end{array} \right\}, \\ k = \frac{2a_5}{a_4 - 2a_6\alpha_2}, \end{array} \right. \quad (5.25)$$

where $a_1, \dots, a_6$ and μ_2 are defined in (5.28) and (5.30). Moreover, $\dot{R} = 0$ if and only if $V_{1,0} = \bar{V}_1$, $H_{1,0} = \bar{H}_1$, $V_{1,L} = \bar{V}_1$, $H_{1,L} = \bar{H}_1$, $V_{2,0} = \bar{V}_2$, $H_{2,0} = \bar{H}_2$, $V_{2,L} = \bar{V}_2$ and $H_{2,L} = \bar{H}_2$.

Remark 5.5. Choosing $k = 1$ does not necessarily make $E(Y)$ a suitable Lyapunov candidate. k depends solely on the choice of α_2 (see demonstration):

$$k = \frac{2a_5\alpha_2}{a_4 - 2a_6\alpha_2}$$

As one can see $k = +\infty$ if $\alpha_2 = \frac{g\bar{H}_2 - \bar{V}_2^2}{2\bar{H}_2\bar{V}_2}$ and $k = 1$ if $\alpha_2 = \frac{g\bar{H}_2 - \bar{V}_2^2}{2\bar{H}_2(\bar{V}_1 + \bar{V}_2)}$.

Proof. Let us rewrite (5.22) as follows :

$$\begin{aligned} \dot{R} = & \underbrace{F(V_{1,0}, H_{1,0}) - kF(V_{2,L}, H_{2,L})}_{S_1} \\ & + \underbrace{kF(V_{2,0}, H_{2,0}) - F(V_{1,L}, H_{1,L})}_{S_2} . \end{aligned} \quad (5.26)$$

It is a straightforward extension of the proof of [18, Proposition 1] to show that the control laws (5.23) make $S_1 < 0$ in a neighborhood of the equilibrium and $S_1 = 0$ if $V_{1,0} = \bar{V}_1$, $H_{1,0} = \bar{H}_1$, $V_{2,L} = \bar{V}_2$ and $H_{2,L} = \bar{H}_2$.

Let us consider the second order expansion of S_2 :

$$S_2 = a_1 h_1^2 + a_2 k h_2^2 + q_2(-a_3 h_1 + a_4 k h_2) - (a_5 - k a_6) q_2^2 , \quad (5.27)$$

where

$$\begin{aligned} a_1 &= \frac{\bar{V}_1 g}{2} - \frac{\bar{Q}^3}{2\bar{H}_1^4}, \quad a_2 = \frac{\bar{Q}^3}{2\bar{H}_2^4} - \frac{\bar{V}_2 g}{2}, \\ a_3 &= -\frac{\bar{Q}^2}{\bar{H}_1^3} + g, \quad a_4 = -\frac{\bar{Q}^2}{\bar{H}_2^3} + g, \\ a_5 &= \frac{\bar{Q}}{2\bar{H}_1^2}, \quad a_6 = \frac{\bar{Q}}{2\bar{H}_2^2}. \end{aligned} \quad (5.28)$$

Using (5.24), (5.27) is rewritten

$$\begin{aligned} S_2 = & h_1^2 (a_1 - a_3 \alpha_1 + k a_6 \alpha_1^2) - h_1^2 a_5 \alpha_1^2 \\ & + k h_2^2 (a_2 - a_4 \alpha_2 + a_6 \alpha_2^2) - h_2^2 a_5 \alpha_2^2 \\ & + h_1 h_2 (-k a_4 \alpha_1 - a_3 \alpha_2 - 2(a_5 - k a_6) \alpha_1 \alpha_2) . \end{aligned} \quad (5.29)$$

From the subcritical flow condition (5.8), it can be shown that $a_4^2 > 4a_6 a_2$. Hence,

$$\exists \alpha_2 \in \mathbb{R} \text{ s.t. } a_2 - a_4 \alpha_2 + a_6 \alpha_2^2 = -\mu_2 < 0. \quad (5.30)$$

Moreover, since $a_5 > 0$, for $\alpha_1^* = \frac{-a_3 + \sqrt{a_3^2 + 4a_1(a_5 - \delta)}}{2(a_5 - \delta)}$,

$$\alpha_1 \geq \alpha_1^* \Rightarrow a_1 - a_3 \alpha_1 - a_5 \alpha_1^2 < -\delta \alpha_1^2 \quad (5.31)$$

for δ sufficiently small.

In the following, let us consider $\alpha_2 > 0$ satisfying (5.30) and $\alpha_1 \geq \alpha_1^*$. Hence, (5.29) is rewritten :

$$\begin{aligned} S_2 \leq & -\delta \alpha_1^2 h_1^2 - \mu_2 h_2^2 \\ & + h_1 h_2 [k a_4 \alpha_1 + a_3 \alpha_2 + 2(a_5 - k a_6) \alpha_1 \alpha_2] . \end{aligned}$$

Hence, in order to have S_2 negative, we must ensure, using (5.30) and (5.31) in (5.29), that :

$$[-k a_4 \alpha_1 - a_3 \alpha_2 - 2(a_5 - k a_6) \alpha_1 \alpha_2]^2 < 4\delta \mu_2 \alpha_1^2. \quad (5.32)$$

In order that to have (5.30), let $\alpha_2 > \alpha_2^*$ where α_2^* is defined by

$$\alpha_2^* = \frac{a_4 - \sqrt{a_4^2 - 4a_2a_6}}{2a_6} . \quad (5.33)$$

In fact, α_2^* is always positive so that $\alpha_2 > 0$ is sufficient. Noting that,

$$\alpha_2 < \frac{a_4 + \sqrt{a_4^2 - 4a_2a_6}}{2a_6} ,$$

let us add the following condition on α_2 :

$$2a_6\alpha_2 < a_4 . \quad (5.34)$$

Let us define k by

$$k = \frac{2a_5\alpha_2}{a_4 - 2a_6\alpha_2} . \quad (5.35)$$

Using the above, (5.32) is rewritten

$$(4a_5 + a_3)^2 \alpha_2^2 < 4\delta\mu_2 \alpha_1^2 ,$$

and thus this inequality can be ensured by an appropriate choice of α_1 .

Moreover using (5.29), we see that S_2 is zero if and only if $H_{1,L} = \bar{H}_1$ and $H_{2,0} = \bar{H}_2$, which from (5.21) implies $h_1 = 0$ and $h_2 = 0$ and using (5.24), this leads to $H_{1,L}V_{1,L} = H_{2,0}V_{2,0} = \bar{Q}$.

This ends the proof. $\square$

5.6.1 Open loop control

The open loop control is modelled by a step function. At time $t < 0$, the gate openings are set to match the system initial state. At $t \geq 0$, the values are updated to match the desired steady state. In a more formal way, for a single reach channel, the open loops control w_1 and w_2 are expressed as follows:

$$w_1 = \begin{cases} \frac{V(0,0)^2 H(0,0)^2}{H_{up} - H(0,0)} & \text{for } t < 0, \\ \frac{\bar{V}^2 \bar{H}^2}{H_{up} - \bar{H}} & \text{for } t \geq 0 \end{cases} \quad (5.36)$$

$$w_2 = \begin{cases} \frac{V(L,0)^2 H(L,0)^2}{H(L,0) - H_{down}} & \text{for } t < 0, \\ \frac{\bar{V}^2 \bar{H}^2}{\bar{H} - H_{down}} & \text{for } t \geq 0 \end{cases} \quad (5.37)$$

5.7 Numerical results

We consider two reaches of length $L = 40m$ and unit width. The initial state and the steady state are:

$$\begin{aligned}(\bar{H}_1, \bar{H}_2) &= (2, 1)[m], \bar{Q} = 0.63[m^3s^{-1}], \\ (H_1, H_2)(x, 0) &= (2.5, 1.5)[m], \\ Q(x, 0) &= 0.63[m^3s^{-1}].\end{aligned}$$

The tuning parameters λ_1 and λ_3 have been set to 2 and 1.5. The parameters α_1 and α_2 have been set to 1 and 0.1. For these settings, $k = 0.002$. The simulation has been made using a semi-implicit Preissman scheme with a time step of 1 s and a spatial step of 2 m. The control laws (5.18)-(5.19) are compared to an constant open loop strategy where u is simply constant, $u_i = \bar{u}_i$ for $i = 1, 2, 3$. The entropy R is depicted in Figure 5.3. One can see the

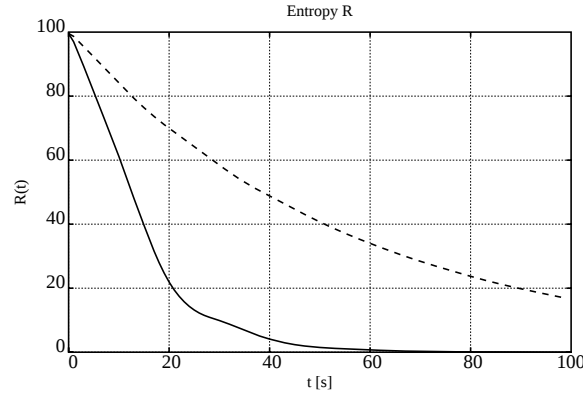


Fig. 5.3. Evolution of entropy $R(t)$. Solid line : feedback control. Dotted line: open loop

asymptotic stabilizing behavior of the feedback control and the acceleration of convergence with respect to the open loop control. In Figure 5.4, we see the deviation $H(x, t) - \bar{H}$ for different simulations times. One can see that the wave is almost totally dampened by the proposed control law.

5.7.1 Open loop stability

In the case of the open loop (5.36) and (5.37), the stability condition (4.22) is written as follows:

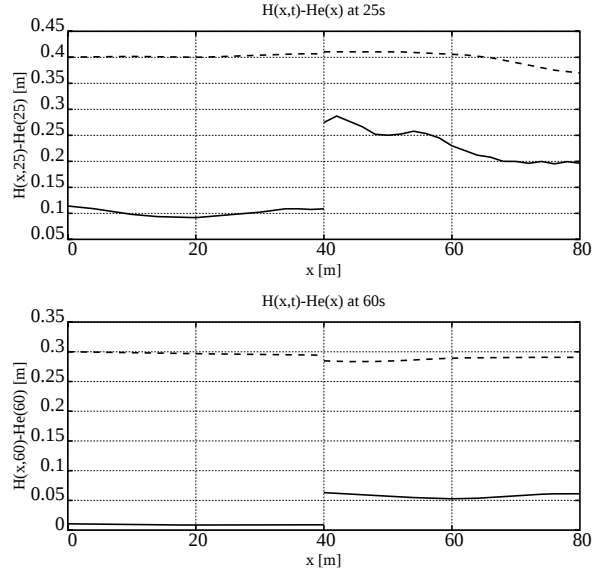


Fig. 5.4. Water depth deviation at 25s and 60s. Solid line : feedback control. Dotted line: open loop

$$A_1 A_2 = \left| \frac{\bar{V} \bar{H}^2 - \bar{V}^2 \bar{H}^{3/2} / \sqrt{g} - \bar{w}_1 \sqrt{\bar{H}} / 2\sqrt{g}}{\bar{V} \bar{H}^2 + \bar{V}^2 \bar{H}^{3/2} / \sqrt{g} + \bar{w}_1 \sqrt{\bar{H}} / 2\sqrt{g}} \right| \left| \frac{\bar{V} \bar{H}^2 + \bar{V}^2 \bar{H}^{3/2} / \sqrt{g} - \bar{w}_2 \sqrt{\bar{H}} / 2\sqrt{g}}{\bar{V} \bar{H}^2 - \bar{V}^2 \bar{H}^{3/2} / \sqrt{g} + \bar{w}_2 \sqrt{\bar{H}} / 2\sqrt{g}} \right|.$$

It can be easily shown that $A_1 A_2$ is strictly less than 1 if:

$$\bar{V} \bar{H}^2 - \bar{V}^2 \bar{H}^{3/2} / \sqrt{g} < \bar{w}_1 \sqrt{\bar{H}} / 2\sqrt{g}$$

and

$$\bar{V} \bar{H}^2 + \bar{V}^2 \bar{H}^{3/2} / \sqrt{g} > \bar{w}_2 \sqrt{\bar{H}} / 2\sqrt{g}.$$

However, for particular values of $\bar{V}$, $\bar{H}$, $\bar{H}_{up}$ and $\bar{H}_{down}$, that product can be superior to 1. In that case the system diverges with the open loop control although it can be stabilized using closed loop controls (5.23). This is illustrated numerically on a single reach simulation on Figure 5.5.

5.8 Conclusion

The main contribution for this chapter has been to propose a Lyapunov control design strategy, based on entropy functions, for quasi linear hyperbolic

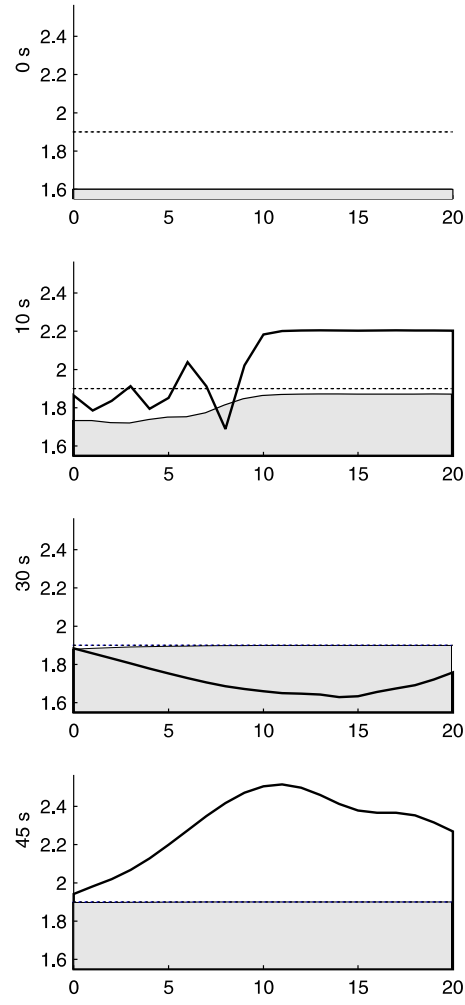
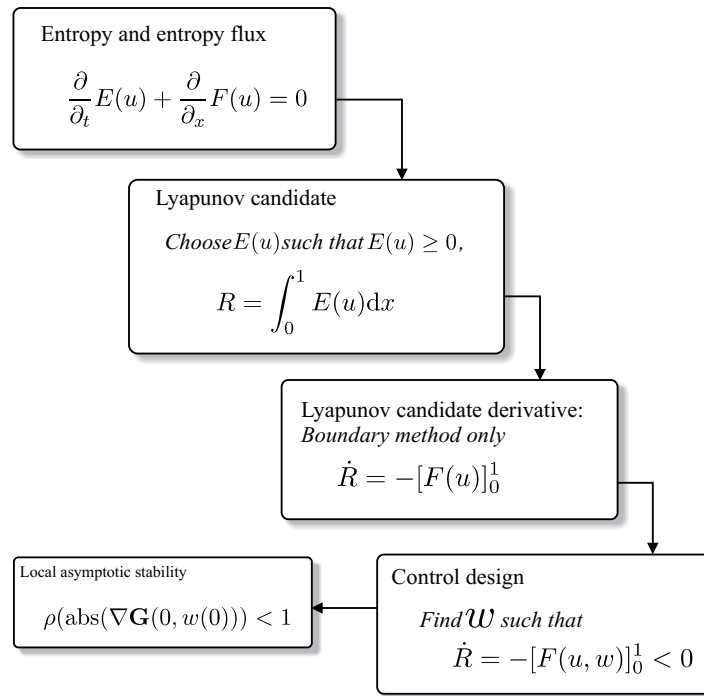


Fig. 5.5. Case when A_1A_2 is superior to 1 for the open loop. Different profiles of water depth at 0, 10, 30 and 45 seconds for a canal with the following properties: $L = 20m$, $H_{up} = 2m$, $H_{down} = 0.1m$, $\bar{V} = 2.5m/s$ and $\bar{H} = 1.9m$.

systems. The strategy has been illustrated with an application to the water flow control in open channels.

However, it must be emphasized that, although Proposition 5.4 provides an efficient control law, it does not yield a complete stabilization proof. However, we know that the closed loop stability is guaranteed with the inputs of Proposition 5.4 as it can be showed by using Theorem 2.28 as depicted in Figure 5.6.

**Fig. 5.6.** Lyapunov based control law design

Control of a moving water tank

"At the end of the yaw maneuver, you can really see in the descent film how dramatic the propellant slosh problem is becoming. As the propellant level gets to near fifty percent, there is more room for the fluid to swirl and slosh yet still enough mass for that movement to really torque the LM out of its tight deadband. This is the reason that there is much more thruster activity than Nil has been expecting. In the absence of slosh, slow gimbaling of the engine would be keeping them level. The effects of the sloshing reaches its peak right about here, with the spacecraft wiggling 2 to 3 degrees back-and-forth and side-to-side every couple of seconds. Later, more seriously, the spacecraft motions caused by slosh will render the LPD essentially useless and then, at 102:44:45, will cause the propellant low-level sensor to latch nearly 30 seconds early", comments by Paul Fjeld on the Apollo 11 mission.

The following chapter contains the material from [62].

6.1 Introduction

Consider a tank containing a non viscous incompressible fluid as described in Figure 6.1. The tank is subject to a one-dimensional horizontal movement. To move such a tank we need to take the motion of the fluid into account. This problem is commonly called *tank sloshing* and has been studied in many industrial applications:

- Ship tanks: sloshing has been studied for the evaluation of the liquid sloshing loads, its influence on stability, structure integrity[53, 68, 12, 54]
- Oscillation reduction and anti-roll using sloshing liquid dampers, see [53, 5]
- Fuel trucks: it is well that fuel are much more subject to roll danger due to the moving mass of fuel, see [65].
- Spatial and airplane fuel tanks: Since the Apollo missions, NASA has been aware of that sloshing in the fuel tank could destabilize spatial vessels, see [14, 3, 4].
- Liquid packaging: [35, 37, 36]

Other "exotic" applications imply the sloshing in the ice-fishing problem (see [44]).

Extensive studies of the modelling and the numerical simulation of sloshing can be found numerous publications: [7, 22, 29, 70, 67, 37].

This chapter is a first attempt to study the stabilization problem with the model of the shallow water equations.

Our main concern is the fluid state stabilization problem (level and speed relative to the tank) and the tracking problem of the tank state (position, speed and acceleration) to a prescribed trajectory (e.g. a prescribed final position of the tank) using the acceleration as the control variable.

Stabilizing feedback laws are designed using a Lyapunov approach and backstepping (see e.g. [45] for an introduction to this technique). The design process is repeated iteratively on control problems that have increasing complexity. For each control problem, an augmented Lyapunov function is built from the previous (simpler) problem and the corresponding stabilizing control laws are deduced. More specifically, the control "sub"-problems are:

- fluid state stabilization (Section 6.3.1),
- fluid state and tank speed stabilization (Section 6.3.2),
- fluid-tank state stabilization (Section 6.3.3) where a forward approach (see [52]) is used to design the Lyapunov function.

Two classes of stabilizing control laws are investigated: 1) time-varying full state feedbacks and 2) output feedbacks, where the output is defined by the trajectory of the tank, the level of the fluid at the boundary of the tank and the time. Practical and industrial motivations can be found in [55, 36, 37] for restricting ourselves to output feedbacks.

Some results can be found in [55] concerning the problem of the stabilization of a tank, but the input is defined as a flexible or a rigid wave generator and the equations are linearised around the equilibrium. Here, a different model of the control system is chosen. Moreover, the linearised shallow water are not stabilizable even locally (see [28]), thus a study of the nonlinear equations is needed. For these nonlinear equations, it is proved in [17] that one has a local controllability property.

Several other configurations are studied in [56] and it is proved that, for each configuration under consideration, the linear approximation is steady-state controllable. For our configuration, the linear approximation is not controllable, and thus we need to consider the nonlinear equations to study the stabilization problem.

The chapter is organized as follows. The shallow water equations are described in Section 6.2.1, the steady states in Section 6.2.2 and the stabilization problem in 6.2.3. The existence of Lyapunov functions and feedbacks are investigated in Section 6.3. At last, numerical simulation are used to check that asymptotic stabilization is achieved in Section 6.4.

6.2 System and control problem description

6.2.1 Model description

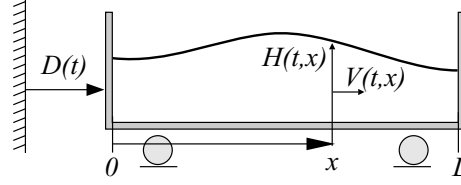


Fig. 6.1. A tank of length L containing a fluid.

Let us consider a moving 1-D tank containing a non viscous incompressible irrotational fluid as depicted in Figure 6.1. The tank is subject to a one-dimensional horizontal move. Let us assume that the acceleration is small compared with the gravity constant and that the level of the fluid is small compared with the length of the tank. Hence, the dynamics of the fluid are described by the shallow water equations (see [26, Section 4.2], see also [56]):

$$\frac{\partial H}{\partial t}(x, t) + \frac{\partial}{\partial x}(HV)(x, t) = 0, \quad (6.1)$$

$$\frac{\partial V}{\partial t}(x, t) + \frac{\partial}{\partial x}\left(gH + \frac{V^2}{2}\right)(x, t) = -A(t), \quad (6.2)$$

$$\dot{S}(t) = A(t), \quad \dot{D}(t) = S(t), \quad (6.3)$$

where

- $x \in [0, L]$ is the spatial coordinate attached to the tank of length L ,
- $t \in [0, T]$ is the time coordinate, $T > 0$,
- g is the gravity constant,
- $H(x, t)$ denotes the level of the liquid,
- $V(x, t)$ denotes the horizontal speed of the fluid in the referential attached to the tank,
- D , S and A denote respectively the position, the speed and the acceleration of the tank in the world coordinates.

The boundary conditions are given by, for all t in $[0, T]$,

$$V(0, t) = 0, \quad V(L, t) = 0. \quad (6.4)$$

6.2.2 Steady states

Let us now describe the set of equilibria:

$((\bar{H}, \bar{V}, \bar{S}, \bar{D}), \bar{A})$ in $((\mathbb{C}^1([0, L]))^2 \times \mathbb{R}^2) \times \mathbb{R}$ is said to be an equilibrium of (6.1)-(6.3) if it is a time-invariant solution of (6.1)-(6.3). This implies that

$$\frac{\partial}{\partial x}(\bar{H}\bar{V}) = 0, \quad \frac{\partial}{\partial x}(g\bar{H} + \frac{\bar{V}^2}{2}) = -\bar{A},$$

which, by using (6.1) and (6.2), can be easily rewritten as follows

$$\bar{V}(x) = 0, \quad \bar{H}(x) = \bar{H}(\frac{L}{2}) - (x - \frac{L}{2})\frac{\bar{A}}{g}. \quad (6.5)$$

By integrating (6.1) on $[0, L]$ and by using the boundary condition (6.4) together with an integration by parts, one gets

$$\frac{d}{dt}(\int_0^L H(x, t)dx) = 0. \quad (6.6)$$

This condition expresses the conservation of the mass of fluid in the tank. Moreover, it follows from (6.1) and (6.4) that

$$\frac{\partial H}{\partial x}(0, t) = \frac{\partial H}{\partial x}(L) \quad \left(= -\frac{u(t)}{g} \right). \quad (6.7)$$

Therefore the state-space $\mathbb{X} = \{X = (H, V, S, D)\}$ is defined as the affine subspace of $(\mathcal{C}^1([0, L]))^2 \times \mathbb{R}^2$ such that the following holds,

$$\text{Vol} = \int_0^L H(x)dx, \quad \frac{dH}{dx}(0) = \frac{dH}{dx}(L), \quad V(0) = V(L) = 0,$$

where the (constant) volume of the liquid in the tank is $\text{Vol} := \int_0^L \bar{H}(x)dx = L\bar{H}(\frac{L}{2})$.

6.2.3 Control problem

Assume that the fluid level at the boundary of the tank can be measured. The control system is described as follows:

- the state at time t is $X(t) = (H(., t), V(., t), S(t), D(t))$,
- the control at time t is $u(t) = A(t)$,
- the output at time t is $Y(t) = (H(0, t), H(L, t), S(t), D(t), t)$.

Let $|\cdot|$ be the usual norm of $\mathbb{R}$ and $|\cdot|_1$ be the norm on $\mathcal{C}^1([0, L])$ defined by, for all f in $\mathcal{C}^1([0, L])$,

$$|f|_1 = \max_{x \in [0, L]} |f(x)| + \max_{x \in [0, L]} \left| \frac{df}{dx}(x) \right|.$$

The set $\mathbb{X}$ is equipped with the following norm, for all $X = (H, V, S, D)$ in $\mathbb{X}$:

$$|X|_{\mathbb{X}} = |H|_1 + |V|_1 + |S| + |D|.$$

The control problem is stated as follows:

Definition 6.1 (Tank Control Problem). Find a time-varying full state feedback (resp. an output feedback) u satisfying (6.5) that locally stabilizes the system $(X(t), A(t))$ to the equilibrium $(\bar{X}, \bar{A})$.

i.e.

Definition 6.2 (Tank Control Problem, detailed). Find a function $u : \mathbb{X} \times [0, +\infty) \rightarrow \mathbb{R}$ (resp. $u : \mathbb{R}^5 \rightarrow \mathbb{R}$) such that for all $\varepsilon > 0$, there exists $C > 0$ such that, for all initial conditions (H_0, V_0, S_0, D_0) in $\mathbb{X}$ satisfying

$$|(H_0, V_0, S_0, D_0) - (\bar{H}, \bar{V}, \bar{S}, \bar{D})|_{\mathbb{X}} \leq C, \quad (6.8)$$

the following three properties hold on the system:

Existence and uniqueness of solutions: there exists one and only one $X : [0, +\infty) \rightarrow \mathbb{X}$ such that, (6.1)-(6.3) hold where, for all $t \geq 0$,

$$\begin{aligned} A(t) &= u(X(t), t) = u(H(\cdot, t), V(\cdot, t), S(t), D(t), t) \\ (\text{resp. } A(t) &= u(Y(t)) = u(H(0, t), H(L, t), S(t), D(t), t), \end{aligned} \quad (6.9)$$

and such that the following holds

$$H(\cdot, 0) = H_0, V(\cdot, 0) = V_0, S(0) = S_0, D(0) = D_0. \quad (6.10)$$

Attractivity property: for all $t \geq 0$,

$$|(H(\cdot, t), V(\cdot, t), S(t), D(t)) - (\bar{H}, \bar{V}, \bar{S}, \bar{D})|_{\mathbb{X}} \rightarrow_{t \rightarrow +\infty} 0.$$

Stability property: for all $t \geq 0$,

$$|(H(\cdot, t), V(\cdot, t), S(t), D(t)) - (\bar{H}, \bar{V}, \bar{S}, \bar{D})|_{\mathbb{X}} \leq \varepsilon.$$

In the following sections, a Lyapunov control law design that solves this problem is proposed. Stabilization is illustrated numerically in Section 6.4.

6.3 Lyapunov control design

The objective of the design is to build a Lyapunov function for the stabilization problem via a full state feedback and an other one for the output feedback. As mentioned in the Introduction, Lyapunov functions are built for control problems with increasing complexity.

At first, a Lyapunov function for the fluid state (H, V) (i.e. a function $R_1 : (\mathbb{C}^1([0, L]))^2 \rightarrow \mathbb{R}$ positive and null only at the point $(H, V) = (\bar{H}, \bar{V})$) and a full state feedback, making the Lyapunov function decrease, are designed.

Remark 6.3. For the sake of simplicity the design analysis below assumes that the method R_1 is Frechet-differentiable in order to use the methodology based the Lemma 6.4. As stated in Remark 6.8, we are presenting a design methodology without actually proving the stability of each controller. Therefore, this assumption has no significant impact on the quality of the results. For an introduction of the Frechet-differentiation see e.g. [21, Appendix A].

This full state feedback is used to derive an output feedback which is a good candidate to stabilize the fluid state (H, V) (see Section 6.3.1).

The fluid state Lyapunov function is augmented to obtain a Lyapunov function for the fluid-tank state in Sections 6.3.2 and 6.3.3.

6.3.1 Stabilization of the fluid state (H, V)

Let us consider first the stabilization of the fluid state.

A Lyapunov function for the variables (H, V) can be derived from the fluid entropy $E: [0, L] \times \mathbb{R}^2 \rightarrow \mathbb{R}$ and the corresponding entropic flux $F: [0, L] \times \mathbb{R}^2 \rightarrow \mathbb{R}$ (as in [18] for another stabilization problem). Note there is an infinite number of entropies for the shallow water equations (see [64, Volume II, Section 9.3]).

Let us check briefly that the following functions, derived from the moments of the fluid and defined, for all $(H, V) \in \mathbb{R}^2$, by

$$E(x, H, V) = H \frac{V^2}{2} + g \frac{(H - \bar{H}(x))^2}{2}, \quad (6.11)$$

$$F(x, H, V) = H \frac{V^3}{2} + gVH(H - \bar{H}(x)), \quad (6.12)$$

are a couple of entropy and entropic flux respectively.

Indeed, denoting $\frac{D}{Dt}$ and $\frac{D}{Dx}$ the total derivative with respect to t and x respectively, it follows that, along the solutions of (6.1)-(6.3),

$$\begin{aligned} & \frac{D}{Dt} E(x, H, V) + \frac{D}{Dx} F(x, H, V) \\ &= \partial_H E \partial_t H + \partial_V E \partial_t V + \partial_H F \partial_x H + \partial_V F \partial_x V + \partial_x F \\ &= - \left(\frac{V^2}{2} + g(H - \bar{H}) \right) \frac{\partial}{\partial x} (HV) - HV \frac{\partial}{\partial x} \left(gH + \frac{V^2}{2} + u \right) \\ & \quad + \left(\frac{V^3}{2} + gV(H - \bar{H}) + gVH \right) \partial_x H + \left(H \frac{3V^2}{2} + gH(H - \bar{H}) \right) \partial_x V \\ & \quad - gHV \partial_x \bar{H}, \\ &= -\partial_x \left(HV \left(\frac{V^2}{2} + g(H - \bar{H}) \right) \right) - gHV \partial_x \bar{H} - HVu \\ & \quad + \partial_x \left(HV \left(\frac{V^2}{2} + g(H - \bar{H}) \right) \right) + gHV \partial_x \bar{H} + gHV \frac{\bar{A}}{g}, \\ &= -HV(u - \bar{A}), \end{aligned} \quad (6.13)$$

which vanishes if $u = \bar{A}$. Thus (E, F) is a couple of entropy - entropic flux.

Let the function $R_1 : (\mathbb{C}^1([0, L]))^2 \rightarrow \mathbb{R}$, for all $(H, V) \in (\mathbb{C}^1([0, L]))^2$, be defined by

$$R_1(H, V) = \lambda_1 \int_0^L E(x, H(t, x), V(t, x)) dx, \quad (6.14)$$

where $\lambda_1 > 0$ is a tuning parameter. Note that R_1 is positive and is zero only at the point $(H, V) = (\bar{H}, \bar{V})$. By differentiating (6.14) with respect to t and by using (6.13) it follows that, along the solutions of (6.1)-(4.2),

$$\dot{R}_1 = -(u - \bar{A})\lambda_1 \left(\int_0^L HV dx \right) - \lambda_1 [F]_0^L . \quad (6.15)$$

Using (6.4) and (6.12), it can be seen that $[F]_0^L = 0$. Hence, a natural controller is $u : (\mathbb{C}^1([0, L]))^2 \rightarrow \mathbb{R}$, defined by

$$u(H, V) = \lambda_1 \int_0^L HV dx + \bar{A} . \quad (6.16)$$

This control law is a full state feedback law, but one can prove that the time-derivative of $\int_0^L HV dx$ can be expressed in terms of output variables. More precisely, using (6.1) and (6.2), it follows that

$$\frac{\partial HV}{\partial t} = -\frac{\partial}{\partial x} \left(g \frac{H^2}{2} + HV^2 \right) - Hu , \quad (6.17)$$

and thus, by using the boundary conditions (6.4) together with an integration by parts,

$$\frac{d}{dt} \left(\int_0^L HV dx \right) = -\frac{g}{2} ((H(L, t))^2 - (H(0, t))^2) - \text{Vol } u , \quad (6.18)$$

which is a function of the output $Y(t)$ only.

This allow us to use a backstepping approach which is usual in finite dimensional control theory (see e.g. [45, 57]). In this context, let us consider u as a new state variable and define the dynamics of u as

$$\dot{u} = v , \quad (6.19)$$

where v is the control law.

To understand how an output feedback can be derived from (6.15), let us prove the following result.

Lemma 6.4. *Let $\mathcal{X}$, $\mathcal{Y}$ and $\mathcal{U}$ be three Banach spaces called respectively the state space, the output space and the control space. Let $\mathcal{F} : \mathcal{X} \times \mathcal{U} \rightarrow \mathcal{X}$ and $\mathcal{G} : \mathcal{X} \rightarrow \mathcal{Y}$ be two Frechet-differentiable functions. Let us consider the control system*

$$\dot{X} = \mathcal{F}(X, U) , \quad Y = \mathcal{G}(X) , X(0) \in \mathcal{X} \quad (6.20)$$

Let us consider an equilibrium of (6.20), i.e. a state 0 in $\mathcal{X}$ and a control 0 in $\mathcal{U}$ such that $\mathcal{F}(0, 0) = 0$ and $\mathcal{G}(0) = 0$. Suppose that there exists a Lyapunov function $R : \mathcal{X} \rightarrow \mathbb{R}$ (i.e. a Frechet-differentiable function positive and zero only at the point $X = 0$) such that

$$\dot{R} = (U + k(Y))l(X, t) , \quad \overbrace{l(X, t)}^{\dot{}} = \tilde{l}(Y, U) , \quad (6.21)$$

along the solutions of (6.20), where $k : \mathcal{X} \rightarrow \mathbb{R}$, $l : \mathcal{Y} \times [0, +\infty) \rightarrow \mathbb{R}$ and $\tilde{l} : \mathcal{Y} \times \mathcal{U} \rightarrow \mathbb{R}$ are three Frechet-differentiable functions.

Then, for all $\lambda > 0$, we can design a Lyapunov function $\tilde{R}: \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}$, for the state variables (X, U) , and an output feedback making $\tilde{R}$ decrease along the solutions of (6.20). This output feedback is defined by

$$\dot{U} = -\overbrace{k(Y)}^{\cdot} - \lambda \tilde{l}(Y, U) - \frac{1}{\lambda \tilde{\lambda}}(U + k(Y)) \quad , \quad U(t=0) \in \mathcal{U} . \quad (6.22)$$

Remark 6.5. Although the Lemma 6.4 does not technically apply to the system we are studying because the functions that we are studying are not totally Frechet-differentiable, it is a valuable tool that can be used to design control laws.

Remark 6.6. Note that the dynamical equation of U must be a function of the output only, and, for $\tilde{R}$, a Lyapunov function for the whole state, i.e. the couple (X, U) . Moreover, note that, in order to prove Lemma 6.4, only properties of the Lyapunov function R i.e. (6.21) and any property of the control system (6.20) are used. Thus the lemma can be applied to any control problem having a Lyapunov function satisfying (6.21).

Proof. Note first that, due to (6.21), with the continuous feedback $U: \mathcal{X} \rightarrow \mathcal{U}$ defined by $U(X) = -\lambda l(X, t) - k(Y)$, for all $\lambda > 0$, the function R has a non-positive time-derivative. Let us define the Frechet-differentiable function $\tilde{R}: \mathcal{X} \times \mathcal{U} \rightarrow \mathbb{R}$ by $\tilde{R}(X, U) = R(X) + \frac{\tilde{\lambda}}{2}(U + k(Y) + \lambda l(X, t))^2$, where $\tilde{\lambda} > 0$. By using (6.21), the time derivative of $\tilde{R}$ along the solution of (6.20) is

$$\begin{aligned} \dot{\tilde{R}} &= l(X, t) (U + k(Y)) + \tilde{\lambda} (U + k(Y) + \lambda l(X, t)) (\dot{U} + \overbrace{k(Y)}^{\cdot}) + \lambda \tilde{l}(Y, U) , \\ &= l(X, t) \left(U + k(Y) + \lambda \tilde{\lambda} \dot{U} + \lambda \tilde{\lambda} \overbrace{k(Y)}^{\cdot} + \lambda^2 \tilde{\lambda} \tilde{l}(Y, U) \right) \\ &\quad + \tilde{\lambda} (U + k(Y)) \left(\dot{U} + \overbrace{k(Y)}^{\cdot} + \lambda \tilde{l}(Y, U) \right) . \end{aligned}$$

Thus, by defining U as

$$U + k(Y) + \lambda \tilde{\lambda} \dot{U} + \lambda \tilde{\lambda} \overbrace{k(Y)}^{\cdot} + \lambda^2 \tilde{\lambda} \tilde{l}(Y, U) = 0 \quad (6.23)$$

the first derivative of $\tilde{R}$ is $\dot{\tilde{R}} = -\frac{1}{\lambda}(U + k(Y))^2$, and thus $\tilde{R}$ is a Lyapunov function for the system (6.20). To achieve the proof of Lemma 6.4, note that (6.23) is equivalent to (6.22), and U is indeed an dynamic output feedback. $\square$

It is now possible to apply Lemma 6.4 to the system (6.1)-(6.3) and to the Lyapunov function R_1 with $\mathcal{U} = \mathbb{R}$, $\mathcal{Y} = \mathbb{R}^4$, $\mathcal{X} = (\mathbb{C}^1([0, L]))^2$, $k(Y) = -\bar{A}$, $R = R_1$, $\lambda = 1$, $l(X) = -\lambda_1 \int_0^L HV$ and

$\tilde{l}(Y, u) = \lambda_1 \frac{g}{2} ((H(L, t))^2 - (H(0, t))^2) + \lambda_1 \text{Vol } u$. Thus let us define $\tilde{R}_1 : (\mathbb{C}^1([0, L]))^2 \times \mathcal{U} \rightarrow \mathbb{R}$ by, for all $(h, v, u) \in (\mathbb{C}^1([0, L]))^2 \times \mathcal{U}$,

$$\tilde{R}_1(H, V, u) = R_1(H, V) + \frac{\tilde{\lambda}_1}{2} \left(u - \lambda_1 \int_0^L HV dx \right)^2, \quad (6.24)$$

where $\tilde{\lambda}_1 > 0$.

The above Lemma leads to the following result.

Theorem 6.7. • *For any positive gain λ_1 , there exist a Lyapunov function R_1 for the variables (H, V) and a full state feedback $u_1 : \mathbb{C}^1([0, L])^2 \rightarrow \mathbb{R}$, defined by, for all $(H, V) \in \mathbb{C}^1([0, L])^2$*

$$u_1(H, V) := \lambda_1 \int_0^L HV dx. \quad (6.25)$$

such that R_1 has a non-positive time-derivative.

- *For any positive gains λ_1 and $\tilde{\lambda}_1$, there exist a Lyapunov function $\tilde{R}_1$ for the variables (H, V, u) and an output feedback $\tilde{u}_1 : \mathbb{C}^1([0, +\infty), \mathbb{R}^5) \times [0, +\infty) \rightarrow \mathbb{R}$, defined, for all $Y : [0, +\infty) \rightarrow \mathbb{R}^5$, $Y(t) = (H(0, t), H(L, t), S(t), D(t), t)$, by $u_1(t = 0) \in \mathbb{R}$ and*

$$\dot{\tilde{u}}_1(Y) := -\left(\frac{1}{\tilde{\lambda}_1} + \lambda_1 \text{Vol}\right) \tilde{u}_1(Y) - \lambda_1 \frac{g}{2} ((H(L, t))^2 - (H(0, t))^2) \quad (6.26)$$

such that $\tilde{R}_1$ has a non-positive time-derivative.

Remark 6.8. To prove the stabilization, the LaSalle Theorem must be applied. More precisely, it must be proved that the equality $\frac{D}{Dt}(\tilde{R}_1(t)) = 0$, $\forall t \geq 0$, yields $(H, V) = (\bar{H}, \bar{V})$. Note moreover that in an infinite dimensional space of functions, a suitable compactness property must also be proved (see e.g. [38]).

6.3.2 Stabilization of the fluid state (H, V) and of the tank speed S

In this section, the control problem is augmented with the stabilization of the tank speed $\dot{D}$ around $\bar{S} + \bar{A}t$. In order to achieve this, a modified “kinetic energy” of the tank is introduced in (6.14), i.e. the Frechet-differentiable function $R_2 : (\mathbb{C}^1([0, L]))^2 \times \mathbb{R} \rightarrow \mathbb{R}$ defined by

$$R_2(H, V, S) = R_1(H, V) + \frac{\lambda_2}{2} (S(t) - \bar{S} - \bar{A}t)^2, \quad (6.27)$$

where R_1 is defined by (6.14) and λ_2 is a positive constant introduced for the tuning of the controller. Note that R_2 is positive and is zero only at the point $(H, V, S) = (\bar{H}, \bar{V}, \bar{S} + \bar{A}t)$. Due to (6.3) and (6.15),

$$\dot{R}_2 = (u - \bar{A}) \left(-\lambda_1 \int_0^L HV dx + \lambda_2(S - \bar{S} - \bar{A}t) \right). \quad (6.28)$$

Thus a control law candidate to stabilize the variables H , V and S is $u_2 : \mathbb{C}^1([0, L]) \times \mathbb{R} \rightarrow \mathbb{R}$ defined, for all $(H, V, S) \in \mathbb{C}^1([0, L]) \times \mathbb{R}$, by

$$u_2 = \left(\lambda_1 \int_0^L HV dx - \lambda_2(S - \bar{S} - \bar{A}t) \right) + \bar{A}.$$

This is a full state feedback. As in the previous case, it is possible to apply Lemma 6.4 with $\mathcal{X} = (\mathbb{C}^1([0, L]))^2 \times \mathbb{R}$, $k(Y) = -\bar{A}$, $R = R_2$, $\lambda = 1$, $l(X, t) = -\lambda_1 \int_0^L HV dx + \lambda_2(S(t) - \bar{S} - \bar{A}t)$ and $\tilde{l}(Y, u) = \lambda_1 \frac{g}{2}((H(L, t)^2 - H(0, t)^2) + \lambda_1 \text{Vol } u + \lambda_2(u - \bar{A}))$. This motivates the introduction of a new Lyapunov function (see the proof of Proposition 6.4): $\tilde{R}_2 : (\mathbb{C}^1([0, L]))^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$ defined by

$$\tilde{R}_2(H, V, S, u) = R_2(H, V, S) + \frac{\tilde{\lambda}_2}{2} \left(u - \lambda_1 \int_0^L HV dx + \lambda_2(S - \bar{S} - \bar{A}t) \right)^2 \quad (6.29)$$

where $\tilde{\lambda}_2 > 0$.

The above leads to the following result.

Theorem 6.9. • *For any positive gains λ_1 and λ_2 , there exist a Lyapunov function R_2 for the variables (H, V, S) and a time-varying full state feedback $u_2 : \mathbb{C}^1([0, L])^2 \times \mathbb{R} \times [0, +\infty) \rightarrow \mathbb{R}$, defined, for all $(H, V, S) \in \mathbb{C}^1([0, L])^2 \times \mathbb{R}$, by*

$$u_2(H, V, S, t) = \left(\lambda_1 \int_0^L HV dx - \lambda_2(S - \bar{S} - \bar{A}t) \right) + \bar{A} \quad (6.30)$$

such that R_2 has a non-positive time-derivative.

- *For any positive gains λ_1 , λ_2 , $\tilde{\lambda}_2$, there exist a Lyapunov function $\tilde{R}_2$ for the variables (H, V, S, u) and an output feedback $\tilde{u}_2 : \mathbb{C}^1([0, +\infty), \mathbb{R}^5) \times [0, +\infty) \rightarrow \mathbb{R}$, defined, for all $Y : [0, +\infty) \rightarrow \mathbb{R}^5$, $Y(t) = (H(0, t), H(L, t), S(t), D(t), t)$, by $\tilde{u}_2(t = 0) \in \mathbb{R}$ and*

$$\widetilde{\dot{u}_2}(Y) = -\left(\frac{1}{\tilde{\lambda}_2} + \lambda_1 \text{Vol} + \lambda_2\right) \tilde{u}_2(Y) - \lambda_1 \frac{g}{2}((H(L, t)^2 - (H(0, t)^2) + \lambda_2 \bar{A}) \quad (6.31)$$

such that $\tilde{R}_2$ has a non-positive time-derivative.

Remark 6.10. Note that the relative degree of the dynamical system for the variable (H, V) and S is 1 (indeed the zero time-derivative of the output are independent of the control, while the first time-derivative of the output can be

expressed from the control, see e.g. [50] for a precise definition of the relative degree in a finite dimensional context).

In the following section, the system relative degree is 2, which can not be stabilized by an output feedback. See Section 6.3.3 for more details.

6.3.3 Stabilization of the fluid-tank state

In this section, the control problem is augmented with the entire function D and not only its first derivative. To stabilize also the tank position D to a prescribed trajectory $\bar{D} + \bar{S}t + \frac{1}{2}\bar{A}t^2$, one can require to stabilize the integrated trajectory $D = \int \dot{D}dt$ around the reference trajectory $\bar{D}(t) = \int(\bar{S} + \bar{A}t)dt$. To do this a forward approach should be used (see [52] or [57, Chapter 6]), i.e. the solvability of the following equation has to be studied:

$$\dot{\mathcal{M}} = S - \bar{S} - \bar{A}t \quad (6.32)$$

where $\mathcal{M} : \mathbb{X} \rightarrow \mathbb{R}$ is a Frechet-differentiable function and $\dot{\cdot}$ in (6.32) is to be understood as the time derivative along the solutions of (6.1)-(6.3). If (6.32) is solvable, then $R_3 : \mathbb{X} \rightarrow \mathbb{R}$ is defined as the following modification of R_2 , for all (h, v, s, d) in $\mathbb{X}$,

$$R_3(H, V, S, D) = R_2(H, V, S) + \frac{\lambda_3}{2}(D - \bar{D} - \bar{S}t - \frac{1}{2}\bar{A}t^2 - \mathcal{M})^2$$

where $\lambda_3 > 0$. The PDE (6.32) is too difficult to solve (except for the trivial solution $\mathcal{M} = D - \bar{D} - \bar{S}t - \frac{1}{2}\bar{A}t^2$ which gives $R_3 = R_2$). Thus a modification of the forward approach, the *Forwarding modulo $L_g V$* approach (see [58]), which allows us more flexibility, should be used.

More precisely instead of looking a $\mathcal{M} : \mathbb{X} \rightarrow \mathbb{R}$ which solves (6.32), one has to find two functions $\mathcal{M} : \mathbb{X} \rightarrow \mathbb{R}$ Frechet-differentiable and $m : \mathbb{X} \rightarrow \mathbb{R}$ continuous such that (see (6.28))

$$\dot{\mathcal{M}} = S - \bar{S} - \bar{A}t + m \frac{\dot{R}_2}{u - \bar{A}} = (S - \bar{S} - \bar{A}t)(1 + m\lambda_2) - m\lambda_1 \int_0^L HV dx .$$

This PDE is satisfied by

$$\mathcal{M} = (D - \bar{D} - \bar{S}t - \frac{1}{2}\bar{A}t^2)(1 + m\lambda_2) + m\lambda_1 \int_0^L (\int_0^x (H - \bar{H})(\xi)d\xi)dx ,$$

and $m \in \mathbb{R}$. Indeed, note that, due to (6.1),

$$\frac{d}{dt} \left(\int_0^L (\int_0^x (H - \bar{H}))dx \right) = - \int_0^L \left(\int_0^x \frac{\partial HV}{\partial x} \right) dx = - \int_0^L HV . \quad (6.33)$$

This allows us to introduce the Frechet-differentiable function $R_3 : \mathbb{X} \rightarrow \mathbb{R}$ defined by, for all $(H, V, S, D) \in \mathbb{X}$,

$$R_3(H, V, S, D) = R_2(H, V, S) + \frac{\lambda_3}{2} \left(\lambda_2(D - \bar{D} - \bar{S}t - \frac{1}{2}\bar{A}t^2) + \lambda_1 \int_0^L \left(\int_0^x (H - \bar{H})(\xi) d\xi \right) dx \right)^2 \quad (6.34)$$

where λ_1 , λ_2 and λ_3 are three positive constants and R_2 is defined by (6.27). Note that R_3 is positive and is zero only at the point $(H, V, S, D) = (\bar{H}, \bar{V}, \bar{S}, \bar{D})$. Due to (6.28) and (6.33), it follows that

$$\dot{R}_3 = (-\lambda_1 \int_0^L HV dx + \lambda_2(S - \bar{S} - \bar{A}t))(u - \bar{A} + \lambda_2\lambda_3(D - \bar{D} - \bar{S}t - \frac{1}{2}\bar{A}t^2) + \lambda_1\lambda_3 \int \int (H - \bar{H})) .$$

Thus, a natural expression for u_3 is

$$u_3(t) = \lambda_1 \int_0^L H(x, t) V(x, t) dx - \lambda_2(S(t) - \bar{S} - \bar{A}t) - \lambda_2\lambda_3(D(t) - \bar{D} - \bar{S}t - \frac{1}{2}\bar{A}t^2) - \lambda_1\lambda_3 \int_0^L \left(\int_0^x (H(\xi, t) - \bar{H}(\xi)) d\xi \right) dx + \bar{A} .$$

This is a full state feedback. In order to find an output feedback, R_3 has to be rewritten as

$$\dot{R}_3 = (u + \lambda_3 k(X, t)) l(X, t) , \quad \overbrace{k(X, t)}^\cdot = l(X, t) , \quad \overbrace{\tilde{l}(X, t)}^\cdot = \tilde{l}(Y, u) ,$$

where $k: \mathcal{X} \times [0, +\infty) \rightarrow \mathbb{R}$, $l: \mathcal{X} \times [0, +\infty) \rightarrow \mathbb{R}$ and $\tilde{l}: \mathbb{R}^5 \times \mathbb{R} \rightarrow \mathbb{R}$ are three Frechet-differentiable functions defined, for all $(X, Y, u) \in \mathcal{X} \times \mathbb{R}^4 \times \mathbb{R}$, by

$$k(X, t) = \lambda_1 \int_0^L \left(\int_0^x (H - \bar{H}) dx + \lambda_2(D - \bar{D} - \bar{S}t - \frac{1}{2}\bar{A}t^2) \right) ,$$

$$l(X, t) = -\lambda_1 \int_0^L HV dx + \lambda_2(S - \bar{S} - \bar{A}t) ,$$

and

$$\tilde{l}(Y, u) = \lambda_1 \frac{g}{2} (H(L, t)^2 - H(0, t)^2) + \lambda_1 \text{Vol } u + \lambda_2(u - \bar{A}) .$$

Let us study equations (6.1)-(6.4) and also the variables u , $\dot{u}$ as new states and $\ddot{u}$ as new control law. Note that the relative degree of this system is 2 and, for any Lyapunov function R_1 whose relative degree is 1, by using the forward approach, we need to consider Lyapunov function with relative degree 2. Therefore (see [11, Th. 4.3] and [6, Th. 4] in a finite dimensional context) there does not exist an output feedback stabilizing the fluid-tank state and such that a Lyapunov function has a non-positive time-derivative.

The above leads to the following result.

Theorem 6.11. • *For any positive gains λ_1 , λ_2 and λ_3 , there exist a Lyapunov function R_3 for the variables (H, V, S, D) and a time-varying full state feedback $u_3 : \mathbb{C}^1([0, L])^2 \times \mathbb{R}^2 \rightarrow \mathbb{R}$, defined, for all $(H, V, S, D) \in \mathbb{C}^1([0, L])^2 \times \mathbb{R}^2$, by*

$$\begin{aligned}
u_3(H, V, S, D) = & \lambda_1 \int_0^L HV dx - \lambda_2(S - \bar{S} - \bar{A}t) - \lambda_2\lambda_3(D - \bar{D} - \bar{S}t - \frac{1}{2}\bar{A}t^2) \\
& - \lambda_1\lambda_3 \int_0^L (\int_0^x (H(\xi, t) - \bar{H}(x))d\xi)dx + \bar{A} . \quad (6.36)
\end{aligned}$$

such that R_3 has a non-positive time-derivative.

- From the Lyapunov function R_1 , there is neither a Lyapunov function $\tilde{R}_3$ for the fluid-tank state nor a dynamical output feedback $\tilde{u}_3: \mathbb{R}^5 \rightarrow \mathbb{R}$, such that $\tilde{R}_3$ has a non-positive time-derivative.

We are now in position to check with numerical simulations that we have a stabilization property.

6.4 Numerical results

In this section we study three numerical simulations and check that the different problems of stabilization are achieved with our control law.

6.4.1 Observing stabilization in simulation

The shallow water equations are discretized with the 4 points semi-implicit Preissman scheme (see [51] or [33]), which is a classic finite difference scheme for this type of equations.

It is well known that the discretization of the PDEs introduces artificial damping of the solution. This numerical damping is necessary to have a stable integration of the equation, otherwise the numerical errors would not be damped and the simulation would finally blow up. The downside of this is that, in the simulations, the stabilization is due to both the control law and the numerical scheme. Therefore, it is important to take into account this numerical damping when studying the control law stabilizing performance.

The conservative character of the PDEs can be used to answer the question whether the stabilization is due to the control law and not only to the numerical scheme. In fact, if the control action is zero, $u = 0$, and if $\bar{A} = 0$, the function R_1 must be conserved for all t , according to (6.15). In simulation, one can observe that the entropy is actually decreasing due to the numerical damping. Hence, if the performance of a stabilizing control law is significantly better than for the zero control action, it can be deduced that the control law actually stabilizes the system (in simulation).

In the case of the Preissman scheme, this damping can be tuned by the Preissman coefficient θ and Courant number C_r . Moreover, it is possible to choose Preissman coefficient θ and Courant number C_r (namely $\theta = 0.5$ and $C_r = 1$) such that the discretization does not introduce numerical damping for the linear equations.

Other integration schemes were also investigated. For instance, the Godunov scheme from [32] leads to analogous results but introduces more numerical damping effect in the system.

For all the simulations, the following parameters have been chosen:

- The spatial and time steps of the scheme are respectively $\Delta x = 0.2\text{m}$ and $\Delta t = 0.1\text{s}$.
- We choose the following equilibrium of the fluid-tank system: $\bar{V} = \bar{A} = \bar{S} = \bar{D} = 0$ and $\bar{H} = 1$.
- The controller gains are chosen as follows, $\lambda_1 = 0.1$, $\lambda_2 = 0.3$, $\tilde{\lambda}_2 = 2$, and $\lambda_3 = 0.15$.

Note that for these parameters, the tank length L is 10m and the Courant number is slightly superior to 1, $C_r = 1.56$.

6.4.2 Stabilization with output feedback and full-state feedback

The system initial conditions are, for all $x \in [0, L]$,

$$\tilde{D} = 0, \tilde{S} = 0, \tilde{H}(x) = 1 - 2\frac{(x - L/2)}{gL}, \tilde{V}(x) = 0.$$

Three control laws are compared: zero control, $\tilde{u}_2$ defined by (6.31), and u_3 defined by (6.36).

Simulation results, with $\theta = 0.53$, are presented in Figures 6.2a, 6.2b, 6.3a and 6.3b where the following observations are made:

- Figures 6.3a and 6.3b show that $\tilde{u}_2$ and u_3 control laws succeed to stabilize the fluid state in contrast with of the system without control, where oscillations of the fluid stay at the end of the simulation.
- Figure 6.2a shows that $\tilde{u}_2$ stabilizes the tank velocity, while u_3 stabilizes also the tank position.
- A slight and constant damping of the wave in the zero control simulation can be observed in Figures 6.2b, 6.3a and 6.3b. As discussed in Section 6.4.1, it is due to the numerical scheme since θ is set to 0.53. However the control law u_3 makes the Lyapunov function R_3 quite more decreasing than the numerical damping. See Figure 6.2b.
- Figures 6.3a and 6.3b show that waves remain after 25 seconds of simulation for the null control law. Due to the conservative nature of the Shallow Water Equation, these waves should not damp (they do because of the numerical damping) and continue to oscillate, while $\tilde{u}_2$ and u_3 succeed to stabilize in less than 10 seconds which is very quick in comparison with the simulation of Section 6.4.3.
- The controller gains were tuned in a trial and error fashion. Gains were chosen to make the fluid stabilization (λ_1) dominant over the tank acceleration, velocity (λ_2) and position (λ_3). Increasing λ_1 will not significantly improve the fluid stabilization, but will introduce a greater deviation on

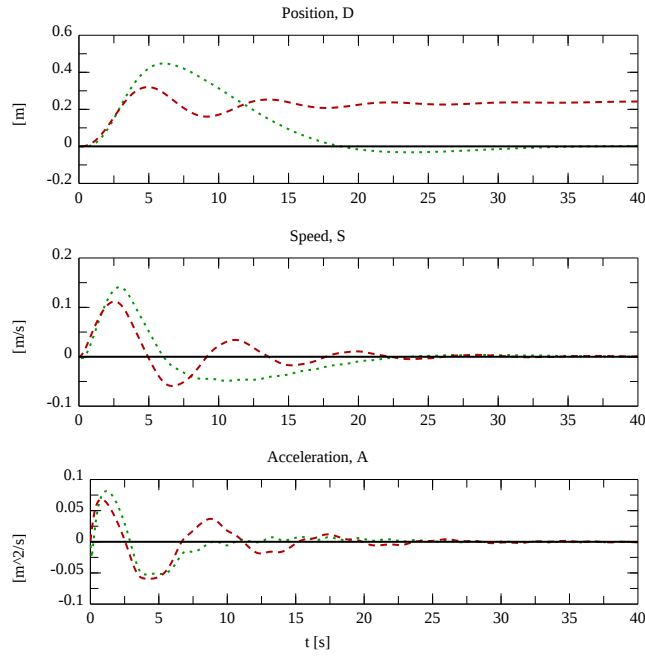


Fig. 6.2a. Tank position D , velocity S and acceleration A . Legend: null control - black solid line, $\bar{u}_2$ - red dashed line, u_3 - green dotted line.

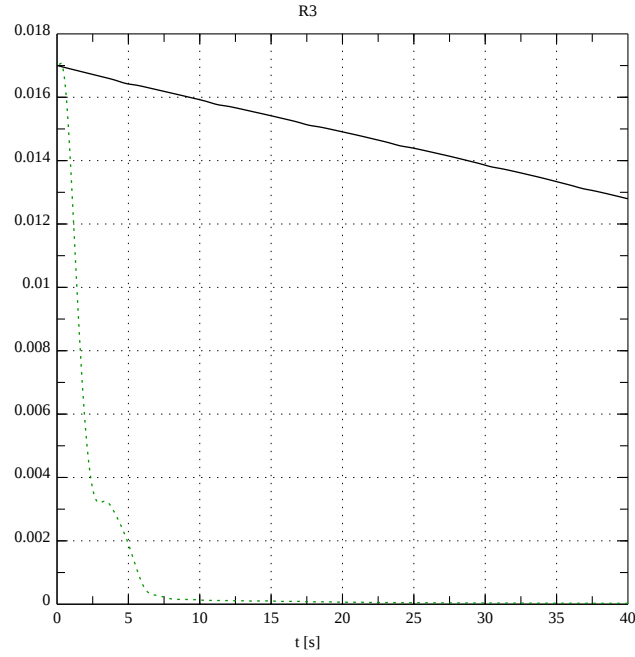


Fig. 6.2b. Lyapunov function R_3 . Legend: null control - black solid line, u_3 - green dotted line.

D and S . The λ_2 parameters behaves as one could expect from a Proportional controller on the velocity, it lowers the overshoot but introduces oscillations for high gains. The simulation is quite sensitive to the λ_3 gain which introduces oscillations in the positions and velocity.

6.4.3 Importance of the nonlinear terms of the shallow water equations

Note that the shallow water equations linearized around this equilibrium are uncontrollable, even locally (see [28] and also the introduction section). Indeed the functions $H, V : [0, L] \times [0, +\infty) \rightarrow \mathbb{R}$ and $D : [0, +\infty) \rightarrow \mathbb{R}$ defined by, for all $t \geq 0$ and for all x in $[0, L]$,

$$\begin{aligned} D(t) &= 0 \\ H(x, t) &= 0.5 + \sin^2\left(\frac{\pi x}{L}\right) \\ V(x, t) &= -2\frac{\pi}{gL}t \cos\left(\frac{\pi x}{L}\right) \sin\left(\frac{\pi x}{L}\right) \end{aligned} \tag{6.37}$$

are solutions of the linearized equations around $((\bar{H}, \bar{V}, \bar{S}, \bar{D}), \bar{A})$ with $u = 0$. However the nonlinear shallow water equations are locally controllable (see [17]), we illustrate numerically that the nonlinear equations are stabilizable.

To do this, let us consider as initial condition the value of the functions (6.37) at $t = 0$ and the feedback (6.36). In this simulation, θ is chosen as 0.5001, which is very close to the critical value (namely 0.5), in order to minimize the numerical damping. Therefore non-smooth numerical solutions (see Figure 6.5a) are obtained. Figure 6.4a shows that the tank stays very close to the initial position but succeed in stabilizing the fluid speed (see Figure 6.5b) and the fluid level (see Figure 6.5a). In Figure 6.4b, one can see that the numerical damping of the Lyapunov function R_3 which is measured by the simulation with the null control, may be neglected in the decreasing rate of the R_3 in the simulation with the feedback (6.36).

Note finally that the stabilization is very slow in comparison with the simulation of Section 4.2 and some waves remain after 70 seconds. This is due to the fact that only the nonlinear part of the shallow water equations stabilized the system (since the linear approximation is uncontrollable).

6.5 Conclusion

In this chapter we study the problems of the stabilization of a tank containing a fluid by an full state feedback and by an output feedback. We use a Lyapunov approach to do this. We check numerically this stabilization problems are achieved with our control laws.

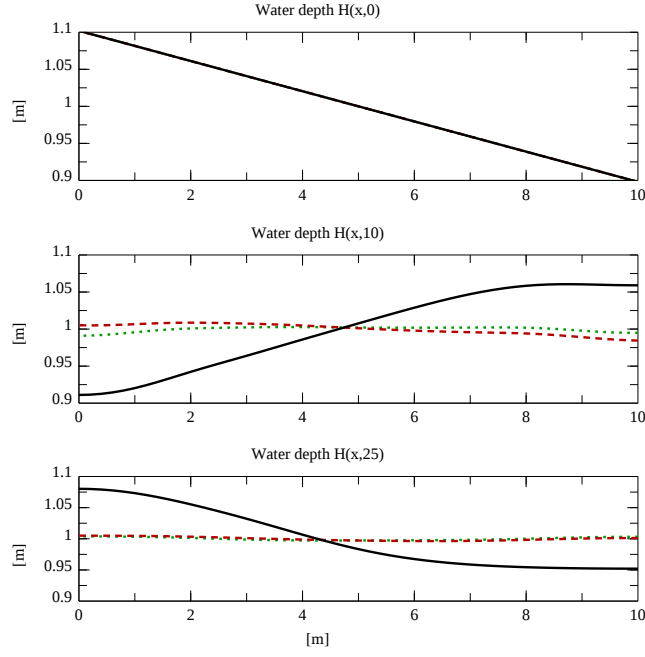


Fig. 6.3a. Fluid level in the tank at time $t = 0, 10, 25$ s. Legend: null control - black solid line, $\tilde{u}_2$ - red dashed line, u_3 - green dotted line.

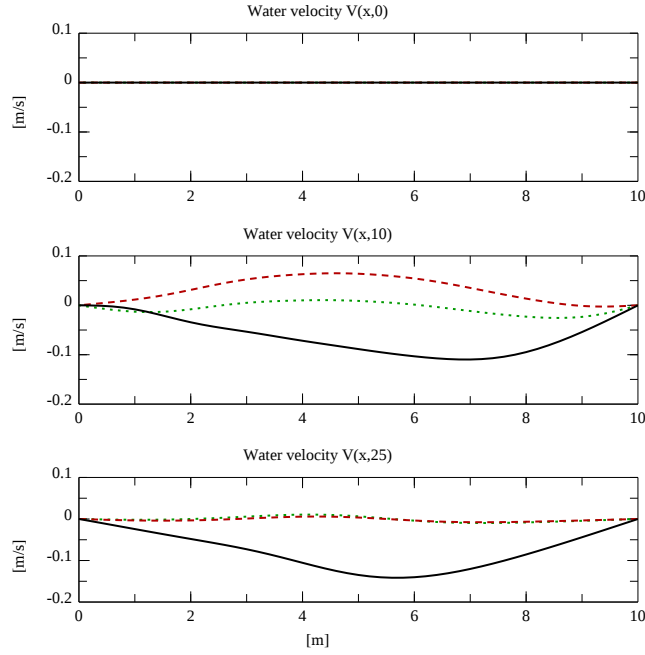


Fig. 6.3b. Fluid velocity in the tank at time $t = 0, 10, 25$ s. Legend: null control - black solid line, $\tilde{u}_2$ - red dashed line, u_3 - green dotted line.

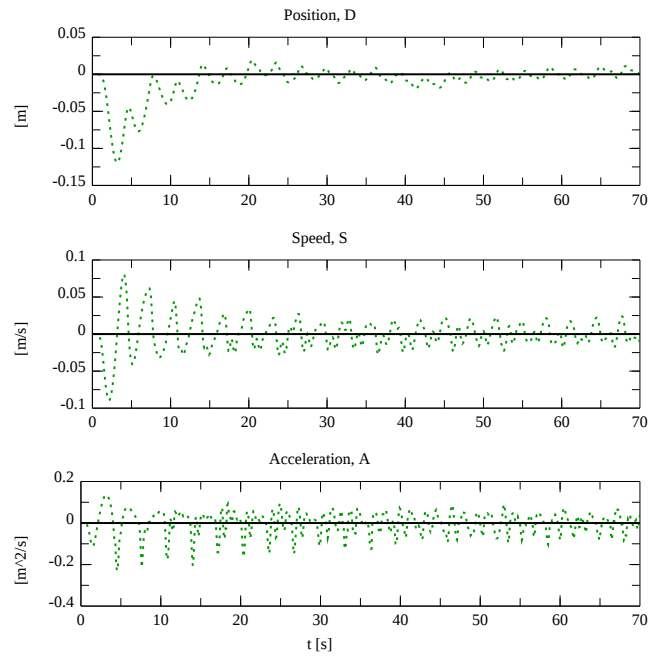


Fig. 6.4a. Tank trajectory. Legend: null control - black solid line, u_3 - green dashed line.

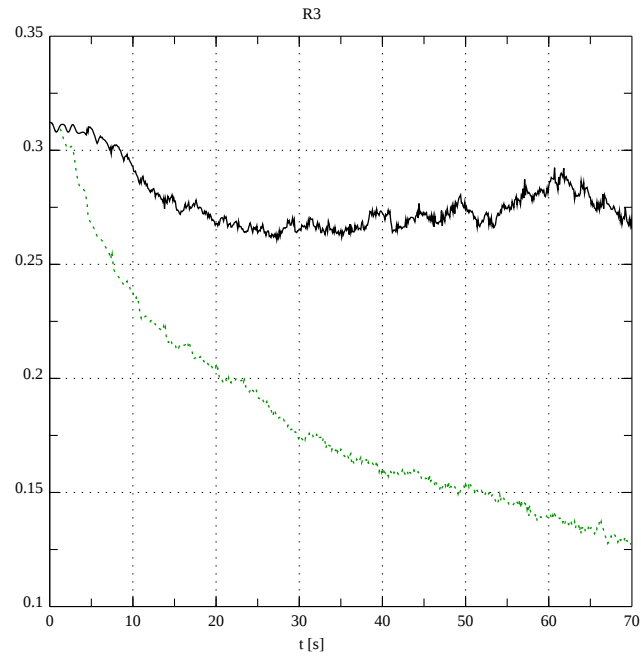


Fig. 6.4b. Lyapunov function R_3 . Legend: null control - black solid line, u_3 - green dashed line.

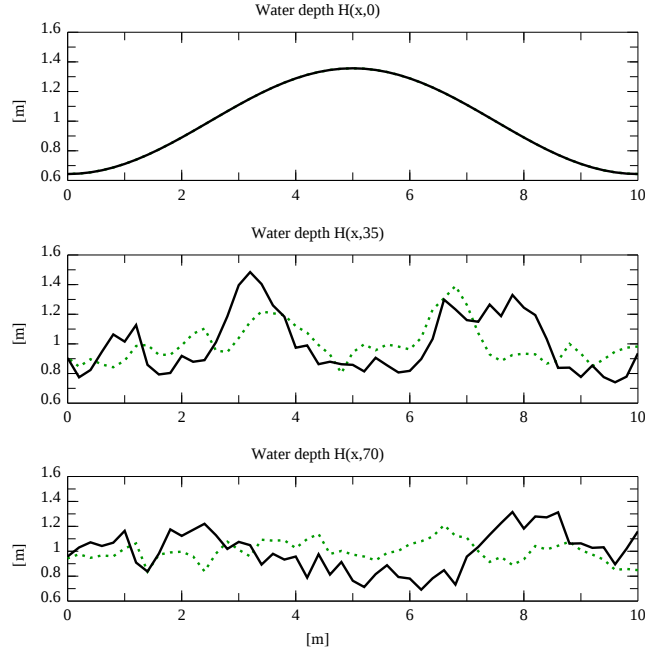


Fig. 6.5a. Fluid depth at 0, 30, 70s. Legend: null control - black solid line, u_3 - green dashed line.

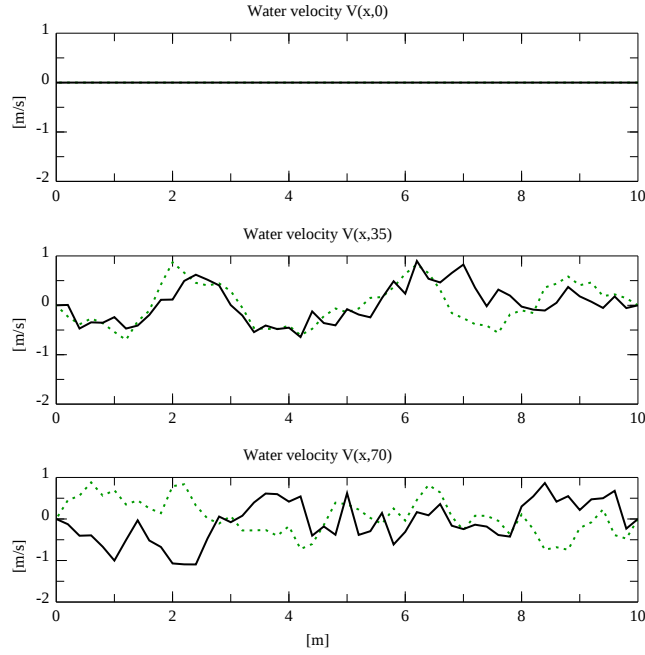


Fig. 6.5b. Fluid velocity at 0, 30, 30, 70s. Legend: null control - black solid line, u_3 - green dashed line.

Numerical Simulation of the Shallow Water Equation

This chapter gives the technical details of the numerical scheme for simulating the Saint-Venant equations.

7.1 Model

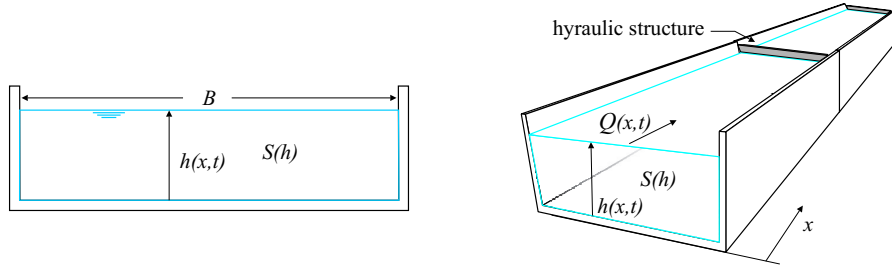


Fig. 7.1. Canal description

7.1.1 Saint-Venant equations

The so called Saint-Venant equations are (see e.g. [33, Chapter 4] and [51]):

$$\partial_t h + \partial_x \left(\frac{Q}{B} \right) = \frac{q}{B} \quad (7.1)$$

$$\partial_t Q + \partial_x \left(\frac{Q^2}{S} \right) + gS\partial_x h - gSI + gSJ = \phi(q)qV \quad (7.2)$$

where

- t is the time variable (s),
- x is the space variable (x), oriented in the direction of flow,
- $Q(x, t)$ is the flow (m^3/s),
- $S(x, t)$ is the wetted surface (m^2),
- $V(x, t)$ is the wetted surface (m/s),
- $q(x, t)$ is the lateral in-flow or out-flow per meter (m^2/s) :

$$\begin{cases} q > 0 & : \text{in-flow,} \\ q < 0 & : \text{out-flow,} \end{cases}$$

- $\phi(q) = \begin{cases} 0 & \text{if } q > 0, \\ 1 & \text{if } q < 0, \end{cases}$, since we suppose that lateral in-flows are perpendicular and don't contribute to the quantity of movement, and that out-flows are parallel to the flow and lower the quantity of movement,
- $h(x, t)$ is the water depth (m),
- $I(x)$ is the bed slope oriented such that I is positive for downhill slope. The slope has to be small enough so that $I \simeq \tan(I)$,
- $J(x, t)$ is the friction slope ,
- B is the width of the canal,
- g is the gravity (m^2/s).

These variable are depicted in Figure 7.1.

The flow, surface and water depth are related by the following relation :

$$Q(x, t) = S(h)V(x, t) \quad (7.3)$$

where $V(x, t)$ is the water mean speed. In the case of a rectangular cross section, S rewrites

$$S(h) = h(x, t)B \quad (7.4)$$

The friction slope denotes the loss of charge per meter. This loss is due to friction on the channel bed and walls. The simplest model is *Manning* formula depending on a rugosity parameter K_s (*Strickler coefficient*) for channel walls:

$$J = \frac{Q|Q|}{K_s^2 S^2 R_h^{4/3}} = \frac{Q|Q|}{K_h^2} \quad (7.5)$$

where R is the hydraulic radius, the ratio between the wetted surface S and the wetted perimeter P ,

$$R_h(h) = \frac{Bh}{B + 2h}, \quad (7.6)$$

and K_h is the conveyance :

$$K_h = K_s S R_h^{2/3} \quad (7.7)$$

The conveyance characterizes the channel; it represents a measure of the capacity of water transport through the cross section.

Remark 7.1. The coefficient K for the Sambre river has been estimated in [69, pg. 3.12], however it is very difficult to identify its exact value due to the lack of measurements.

7.1.2 Hydraulic characteristics

The following quantities are used by hydraulicians to characterize the type of flow in channels:

- positive and negative velocities $c_{\pm} = V \pm \sqrt{gH}$ denote the velocity at which a wave is travelling upstream and downstream.
The *Froude number*, $F_r = \frac{V}{\sqrt{gH}}$, (which corresponds to the famous *Mach number* in the air) measures the ratio between effective fluid velocity and the characteristic speed of this fluid. Similarly to the Mach number, we call *sub-critical* or *fluvial* a flow with $F_r < 1$ and *super-critical* a flow with $F_r > 1$. In navigation channels, flow are usually strongly sub-critical because of high water depth and slow velocity.
- The *Courant number*, $C_r = \frac{dt(\bar{V} + \sqrt{gH})}{dx}$, is a measurement of the distortion of the frequencies of the model by the finite difference numerical scheme [51].
 - $C_r < 1$, the model is not correct
 - $C_r = 1$, all the frequencies are simulated without numerical damping. Therefore, numerical noise is likely to destabilize the simulation,
 - $C_r > 1$, high frequencies are progressively trimmed off. In terms of water waves, one can see this effect as truncating the "little" waves
 A Courant number between 5 and 10 is a reasonable choice in terms of numerical stability and model accuracy.
- θ is the parameter that controls the amount of "implicit behavior" in the numerical scheme. This value must be between 0.5 and 1 (fully implicit). For $\theta = 0.5$ and $C_r = 1$, the schema has no numerical damping and therefore, becomes unstable. Indeed, numerical errors integrate with damping and make the simulation blow up.
A choice of θ between 0.57 and 0.6 is reasonable.

7.1.3 Boundary conditions

Boundary conditions at the gates are given by the charge conservation and the mass conservation at the nodes.

Mass conservation

Since we consider hydraulic structures such as weir gates or jumps as punctual, the sum of flow entering and leaving these nodes must be zero.

$$Q^i(L, t) = Q^{i+1}(0, t). \quad (7.8)$$

Weir gate

The charge conservation for a weir gate is given by [33, Section 4.4.1]:

$$Q(L, t) = cB\sqrt{2gh_d(t)}^{\frac{3}{2}} \quad (7.9)$$

where $h_d = h(L, t) - u$ is the upstream lip and u is the height of the gate and B is the width of the upstream reach. As one can see, the downstream reach has no influence on the equation.

Sluice gate

The charge conservation for a sluice gate is given by :

$$Q(t) = \sqrt{u(B_{up}h_{up}(t) - B_{do}(h_{do} - f))} \quad (7.10)$$

where $h_{up} = h_i(L, t)$ and $h_{do} = h_{i+1}(0, t)$ are respectively the upstream and downstream water depth, f is the fall and u represent the gate opening and B_{up} and B_{do} are respectively the upstream and downstream width of the channel.

Strong condition

An hydrogram can also be set at a gate :

$$Q(t) = f(t) \quad (7.11)$$

where $f(t)$ is a user defined function.

7.1.4 Preissman scheme

Consider the spatial discretisation points $x_i = i\Delta x$ where Δx is the *spatial step* and the time discretisation points $t_i = i\Delta t$ where Δt is the *time step* as depicted in Figure 7.2. For a function $f(x, t)$, the Preissman discretisation scheme is :

$$\begin{aligned} f(x, t) &\cong \frac{1}{2}(f_{i+1} + f_i) + \frac{\theta}{2}(\Delta f_{i+1} + \Delta f_i) \\ \partial_x f(x, t) &\cong \frac{1}{\Delta x} ((f_{i+1} - f_i) + \theta(\Delta f_{i+1} - \Delta f_i)) \\ \partial_t f(x, t) &\cong \frac{1}{2\Delta t} (\Delta f_{i+1} + \Delta f_i) \end{aligned} \quad (7.12)$$

where $0.5 \leq \theta < 1$.

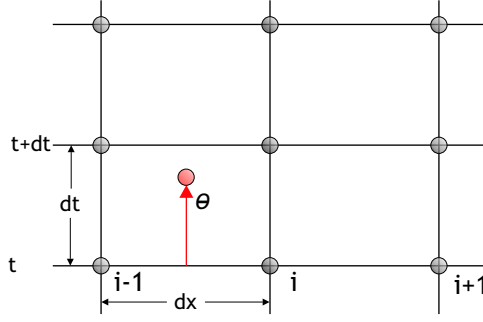


Fig. 7.2. Preissman scheme

7.2 General idea for the resolution

We present here a classical method to solve Saint-Venant equations using Preissman discretisation scheme.

Suppose that we know the water state (Q, h) at time t . We want to compute $(Q + \Delta Q, h + \Delta h)$ at time $t + \Delta t$. If we discretize the shallow water equations, we can separate the linear part from the non-linear part in 2 separate matrices $\mathbf{S}$, $\mathbf{Q}$:

$$\mathbf{S}(Q, h, \Delta Q, \Delta h) \begin{bmatrix} \Delta Q \\ \Delta h \end{bmatrix} = \mathbf{Q}(Q, h, \Delta Q, \Delta h) \quad (7.13)$$

and we can easily and efficiently the solution of this system since $\mathbf{S}$ is a banded matrix ($o(2n_x)$). Of course, this is still a non-linear implicit equations since $\mathbf{S}$ depends of ΔQ and Δh . Anyway, we can iterate on $\Delta Q, \Delta h$ until we converge to a solution (it is equivalent of solving the problem $x = f(x)$): Let $\Delta Q_n, \Delta h_n$ the previous estimation of $\Delta Q, \Delta h$, we find the next estimate $\Delta Q_{n+1}, \Delta h_{n+1}$ using the following:

$$\mathbf{S}(Q, h, \Delta Q_n, \Delta h_n) \begin{bmatrix} \Delta Q_{n+1} \\ \Delta h_{n+1} \end{bmatrix} = \mathbf{Q}(Q, h, \Delta Q_n, \Delta h_n) \quad (7.14)$$

The successive iterates will converge to the desired solution (usually after 2 to 4 model estimate).

7.3 Model validation

The numerical model validation was done by comparing the hydrograms measured in 3 reaches of the Sambre against a numerical model. The model has been implemented as a Windows application described in Appendix C.

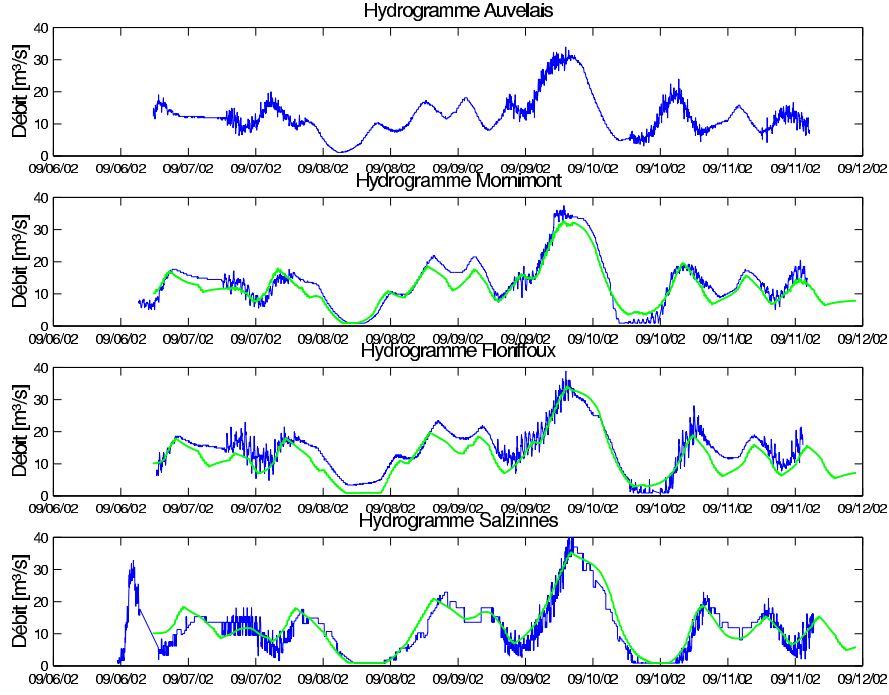


Fig. 7.3. Hydrograms at the different gates. Blue: measured flow, Yellow: simulated flow

7.4 Linearized Model

7.4.1 Saint-Venant equations

Let us consider the elementary reach $[x_i x_{i+1}]$, the associated quantities are $Q_i, h_i, Q_{i+1}, h_{i+1}$ and $\Delta Q_i, \Delta h_i, \Delta Q_{i+1}, \Delta h_{i+1}$.

The Saint-Venant equations are discretized using the Preissman scheme (7.12) and linearized for $\Delta Q_i, \Delta S_i, \Delta Q_{i+1}, \Delta S_{i+1}$ (see [51]):

$$\mathbf{M}_i \begin{bmatrix} \Delta Q_{i+1} \\ \Delta h_{i+1} \end{bmatrix} = \mathbf{N}_i \begin{bmatrix} \Delta Q_i \\ \Delta h_i \end{bmatrix} + \mathbf{P}_i \quad (7.15)$$

where

$$\mathbf{M}_i = \begin{bmatrix} m_i^{11} & m_i^{12} \\ m_i^{21} & m_i^{22} \end{bmatrix}, \quad \mathbf{N}_i = \begin{bmatrix} n_i^{11} & n_i^{12} \\ n_i^{21} & n_i^{22} \end{bmatrix} \quad \text{and} \quad \mathbf{P}_i = \begin{bmatrix} p_i^1 \\ p_i^2 \end{bmatrix} \quad (7.16)$$

Continuity equation

The continuity equation (7.1) is discretized using (7.12) :

$$\alpha_0 = \frac{\theta}{2}(q_{i+1}^+ + q_i^+) + \frac{1-\theta}{2}(q_{i+1} + q_i), \quad (7.17)$$

$$\frac{1}{2\Delta t}(\Delta h_{i+1} + \Delta h_i) + \left[\frac{\theta(\Delta Q_{i+1} - \Delta Q_i) + (Q_{i+1} - Q_i)}{\Delta x B} \right] = \frac{\alpha_0}{B} \quad (7.18)$$

Hence, using (7.4) and since we consider constant rectangular section:

$$\begin{aligned} m_i^{11} &= n_i^{11} = \frac{\theta}{B\Delta x}, \\ m_i^{12} &= -n_i^{12} = \frac{1}{2\Delta t}, \\ p_i^1 &= -\frac{1}{\Delta x} \frac{Q_{i+1} - Q_i}{B} + \frac{\alpha_0}{B}. \end{aligned} \quad (7.19)$$

Dynamic equation.

The dynamic equation (7.2) is rewritten as :

$$\partial_t Q + \frac{2Q}{S} \partial_x Q + \left(gS - B \frac{Q^2}{S^2} \right) \partial_x h + gS(J - I) = \phi(q)q \frac{Q}{S} \quad (7.20)$$

Each term of (7.20) can be discretized using (7.12) :

1. $\partial_t Q$:

$$\frac{1}{2\Delta t}(\Delta Q_{i+1} + \Delta Q_i). \quad (7.21)$$

2. $\frac{2Q}{S} \partial_x Q$:
Let

$$\alpha_1 = \frac{\theta}{2} \left(\frac{Q_{i+1}^+}{S_{i+1}^+} + \frac{Q_i^+}{S_i^+} \right) + \frac{1-\theta}{2} \left(\frac{Q_{i+1}}{S_{i+1}} + \frac{Q_i}{S_i} \right), \quad (7.22)$$

then,

$$2\alpha_1 \left(\frac{1}{\Delta x} (Q_{i+1} - Q_i) + \frac{\theta}{\Delta x} (\Delta Q_{i+1} - \Delta Q_i) \right). \quad (7.23)$$

3. $\left(gS - B \frac{Q^2}{S^2} \right) \partial_x h$: Let

$$\alpha_2 = \frac{\theta}{2} \left[\left(\frac{Q_{i+1}^+}{S_{i+1}^+} \right)^2 + \left(\frac{Q_i^+}{S_i^+} \right)^2 \right] + \frac{1-\theta}{2} \left[\left(\frac{Q_{i+1}}{S_{i+1}} \right)^2 + \left(\frac{Q_i}{S_i} \right)^2 \right], \quad (7.24)$$

$$\alpha_3 = \frac{\theta}{2} (S_{i+1}^+ + S_i^+) + \frac{1-\theta}{2} (S_{i+1} + S_i), \quad (7.25)$$

then,

$$(g\alpha_3 - B\alpha_2) \left(\frac{1}{\Delta x} (h_{i+1} - h_i) + \frac{\theta}{\Delta x} (\Delta h_{i+1} - \Delta h_i) \right). \quad (7.26)$$

4. $gS(I - J)$:
Let

$$\alpha_4 = \frac{\theta}{2} (Q_{i+1}^+ |Q_{i+1}^+| + Q_i^+ |Q_i^+|) + \frac{1-\theta}{2} (Q_{i+1} |Q_{i+1}| + Q_i |Q_i|), \quad (7.27)$$

$$\alpha_5 = \frac{\theta}{2}(R_h(h_{i+1}^+))^{2/3} + R_h(h_i^+))^{2/3} + \frac{1-\theta}{2}(R_h(h_{i+1}))^{2/3} + R_h(h_i))^{2/3}, \quad (7.28)$$

$$\alpha_6 = K\alpha_3\alpha_5, \quad (7.29)$$

hence,

$$gS(J-I) = g \left(\frac{\alpha_4}{\alpha_6^2} - I \right) \left(\frac{S+S_1}{2} + \frac{B\theta}{2}(\Delta h_i + \Delta h_{i+1}) \right) \quad (7.30)$$

5. $\phi(q)q \frac{Q}{S} :$

$$\phi(\alpha_0) \frac{\alpha_0}{\alpha_3} \left(\frac{Q+Q_1}{2} + \frac{\theta}{2}(\Delta Q_i + \Delta Q_{i+1}) \right). \quad (7.31)$$

Hence the discretized dynamic function rewrite in terms of m , n , p using (7.21)-(7.30):

$$\begin{aligned} m_i^{21} &= \frac{1}{2\Delta t} + 2\alpha_1 \frac{\theta}{\Delta x}, \\ n_i^{21} &= \frac{-1}{2\Delta t} + 2\alpha_1 \frac{\theta}{\Delta x}, \\ m_i^{22} &= \theta \left[(g\alpha_3 - B\alpha_2) \frac{1}{\Delta x} + g \left(\frac{\alpha_4}{\alpha_6^2} - I \right) \frac{B}{2} - \phi(\alpha_0) \frac{\alpha_0}{\alpha_3} \frac{1}{2} \right], \\ n_i^{22} &= \theta \left[(g\alpha_3 - B\alpha_2) \frac{1}{\Delta x} - g \left(\frac{\alpha_4}{\alpha_6^2} - I \right) \frac{B}{2} + \phi(\alpha_0) \frac{\alpha_0}{\alpha_3} \frac{1}{2} \right] \\ p_i^2 &= -2\alpha_1 \frac{1}{\Delta x} (Q_{i+1} - Q_i) \\ &\quad - (g\alpha_3 - B\alpha_2) \frac{1}{\Delta x} (h_{i+1} - h_i) \\ &\quad - g \frac{S+S_1}{2} \left(\frac{\alpha_4}{\alpha_6^2} - I \right) \\ &\quad + \phi(\alpha_0) \frac{\alpha_0}{\alpha_3} \frac{Q+Q_1}{2}. \end{aligned} \quad (7.32)$$

Linearized model

Let us rewrite (7.15) as

$$\mathbf{S}_i \begin{bmatrix} \Delta Q_i \\ \Delta h_i \\ \Delta Q_{i+1} \\ \Delta h_{i+1} \end{bmatrix} = \mathbf{Q}_i \quad (7.33)$$

where

$$\mathbf{S}_i = \text{diag}(\beta_i^1, \beta_i^2)^{-1} [-\mathbf{M}_i^{-1} \mathbf{N}_i \quad I_{2 \times 2}] \quad \text{and} \quad \mathbf{Q}_i = \text{diag}(\beta_i^1, \beta_i^2)^{-1} \mathbf{M}_i^{-1} \mathbf{P}_i, \quad (7.34)$$

$$\beta_i^1 = \|S_i^{1*}\|_\infty \quad \text{and} \quad \beta_i^2 = \|S_i^{2*}\|_\infty. \quad (7.35)$$

For numerical reasons (lower the condition number the global matrix), each line of (7.33) have been normalized by their infinite norm.

7.4.2 Boundary functions

Let us introduce the following notation :

$$Q_L^i = Q_{n_{x_i}-1}^i \quad \text{and} \quad h_L^i = h_{n_{x_i}-1}^i \quad (7.36)$$

Mass conservation

The mass conservation rewrites for the j node :

$$\Delta Q_L^j - \Delta Q_0^{j+1} = -(Q_L^j - Q_0^{j+1}) \quad (7.37)$$

or in a matrix-vector form,

$$[1 \ 0 \ 1 \ 0] \begin{bmatrix} \Delta Q_L^j \\ \Delta h_L^j \\ \Delta Q_0^{j+1} \\ \Delta h_0^{j+1} \end{bmatrix} = -(Q_{n_{x_j}-1}^j - Q_0^{j+1}) \quad (7.38)$$

or defining $\mathbf{S}_j^{MC}$ and $\mathbf{Q}_j^{MC}$:

$$\mathbf{S}_j^{MC} \begin{bmatrix} \Delta Q_L^j \\ \Delta h_L^j \\ \Delta Q_0^{j+1} \\ \Delta h_0^{j+1} \end{bmatrix} = -\mathbf{Q}_j^{MC} \quad (7.39)$$

Weir gate condition

The boundary conditions write for the j -th node, defining $\mathbf{S}_j^{WG}$ and $\mathbf{Q}_j^{WG}$:

$$\mathbf{S}_j^{WG} \begin{bmatrix} \Delta Q_L^j \\ \Delta h_L^j \\ \Delta Q_0^{j+1} \\ \Delta h_0^{j+1} \end{bmatrix} = -\mathbf{Q}_j^{WG} \quad (7.40)$$

The linearized form of (7.9) is

$$\Delta Q_L^j - cB \frac{3}{2} \sqrt{2g} \sqrt{h_d} \Delta h_L^j = -(Q - cB \sqrt{2g} h_d^{\frac{3}{2}}) \quad (7.41)$$

or, for $\gamma_{WG} = \max\{1, -cB \frac{3}{2} \sqrt{2g} \sqrt{h_d}\}$,

$$\frac{1}{\gamma_{WG}} [1 \ -cB \frac{3}{2} \sqrt{2g} \sqrt{h_d} \ 0 \ 0] \begin{bmatrix} \Delta Q_L^j \\ \Delta h_L^j \\ \Delta Q_0^j \\ \Delta h_0^j \end{bmatrix} = -\frac{1}{\gamma_{WG}} (Q_L^j - cB \sqrt{2g} h_d^{\frac{3}{2}}) \quad (7.42)$$

Sluice gate condition

The boundary conditions write for the j -th node, defining $\mathbf{S}_j^{SG}$ and $\mathbf{Q}_j^{SG}$:

$$\mathbf{S}_j^{SG} \begin{bmatrix} \Delta Q_L^j \\ \Delta h_L^j \\ \Delta Q_0^{j+1} \\ \Delta h_0^{j+1} \end{bmatrix} = -\mathbf{Q}_j^{HC} \quad (7.43)$$

The linearized form of (7.10) is

$$\begin{aligned} \Delta Q_L^j - \frac{\sqrt{u}}{2\sqrt{B_{up}h_{up} - B_{do}(h_{do} - f)}} (B_{up}\Delta h_{up} - B_{do}\Delta h_{do}) \\ = - \left(Q - \sqrt{u(B_{up}h_{up} - B_{do}(h_{do} - f))} \right) \end{aligned} \quad (7.44)$$

or, for

$$\begin{aligned} \alpha_6 &= \sqrt{u} \sqrt{B_{up}h_{up} - B_{do}(h_{do} - f)}, \\ \alpha_7 &= \frac{u}{2\sqrt{\alpha_6}}, \\ \gamma_{SG} &= \max\{1, \alpha_6\}, \\ \frac{1}{\gamma_{SG}} [1 \quad -B_{up}\alpha_7 \quad 0 \quad B_{do}\alpha_7] \begin{bmatrix} \Delta Q_L^j \\ \Delta h_L^j \\ \Delta Q_0^j \\ \Delta h_0^j \end{bmatrix} &= -\frac{1}{\gamma_{SG}} (Q - \alpha_6) \end{aligned} \quad (7.45)$$

Strong condition

The hydrogram condition (7.11) rewrite :

$$\Delta Q_0^{j+1} = f(t) - Q_0^{j+1} \quad (7.46)$$

or

$$[0 \ 0 \ 1 \ 0] \begin{bmatrix} \Delta Q_L^j \\ \Delta h_L^j \\ \Delta Q_0^j \\ \Delta h_0^j \end{bmatrix} = f(t) - Q_0^{j+1} \quad (7.47)$$

This equation is normalized to have (7.39).

7.5 Global system

All the contribution of the boundaries and the saint-Venant equations are the piled together in a global matrix.

$$\mathbf{A}\mathbf{y} = \mathbf{b} \quad (7.48)$$

where (for a 2 reach channel)

$$\mathbf{A} = \begin{bmatrix} \mathbf{S}_0^{HC} & & & & \\ \mathbf{S}_0^0 & & & & \\ & \ddots & & & \\ & & \mathbf{S}_{n_{x_0}-1}^0 & & \\ & & \mathbf{S}_1^{HC} & & \\ & & \mathbf{S}_1^{MC} & & \\ & & \mathbf{S}_0^1 & & \\ & & & \ddots & \\ & & & & \mathbf{S}_{n_{x_1}-1}^1 \\ & & & & \mathbf{S}_2^{HC} \end{bmatrix} \quad (7.49)$$

$$\mathbf{in} = \begin{bmatrix} Q_0^0 \\ h_1^0 \\ \vdots \\ Q_{n_{x_0}-1}^0 \\ h_{n_{x_0}-1}^0 \\ Q_0^1 \\ h_1^1 \\ \vdots \\ Q_{n_{x_1}-1}^1 \\ h_{n_{x_1}-1}^1 \end{bmatrix} \quad \text{and} \quad \mathbf{b} = \begin{bmatrix} Q_0^{HC} \\ Q_0^0 \\ \vdots \\ Q_{n_{x_0}-1}^0 \\ Q_1^{HC} \\ Q_1^{MC} \\ Q_0^1 \\ \vdots \\ Q_{n_{x_1}-1}^1 \\ Q_1^{HC} \end{bmatrix} \quad (7.50)$$

The matrix $\mathbf{A}$ is banded with 2 sub-diagonals and 1 super-diagonal when we have cascade reaches.

For each time step, the following algorithm is applied :

```
// Initialize  $\mathbf{in}_{prec}$  :
 $\mathbf{in}_{prec}$ =0;  $e$ =1.0;
while(  $e \geq \text{tol}$  )
{
  // update  $\mathbf{in}_{prec}$ 
   $\mathbf{in}_{prec} \leftarrow \mathbf{in}$ ;
  // Compute  $\mathbf{A}$ ,  $\mathbf{b}$ 
  [ $\mathbf{A}$ ,  $\mathbf{b}$ ]=ComputeAb( $state, \mathbf{in}_{prec}$ );
  // Compute  $\mathbf{in}$ 
   $y = \mathbf{A}^{-1}\mathbf{b}$ ;
  // compute convergence test
   $e = \|\mathbf{in}_{prec} - y\|_2$ ;
}
```

7.6 Computing steady-states

7.6.1 Flow Q_i

Using continuity equations, we can directly compute the Q_i :

$$Q_{i+1} = Q_i + \Delta x \frac{q_i + q_{i+1}}{2} \quad (7.51)$$

with Q_0 given as an initial condition. If there are no inflow/outflow, the Q_i are constant.

7.6.2 Depth h_i

Suppose that h_i , Q_i are known. We use the following scheme to compute iteratively h_{i+1} :

```
// Initialize  $h_{i+1}^n$ 
 $h_{i+1}^n = h_i$ ;
e=1;
// Start iterating
while( e > tol)
{
// Compute mean values of  $\alpha$ 
 $\bar{\alpha}_k = \alpha_k(Q_i, h_i, Q_{i+1}, h_{i+1}^n)_{\theta=0}$ , for  $k = 0, \dots, 6$ 
// Compute new iterate for  $h_{i+1}$ 


$$h_{i+1}^{n+1} = \frac{\Delta x}{g\bar{\alpha}_3 - B\bar{\alpha}_2} \left( 2\bar{\alpha}_1(Q_{i+1} - Q_i) - \frac{g\bar{\alpha}_3 - B\bar{\alpha}_2}{\Delta x} h_i - g\bar{\alpha}_3 \left( I - \frac{\bar{\alpha}_4}{\bar{\alpha}_5} \right) - \phi(\bar{\alpha}_0)\bar{\alpha}_0\bar{\alpha}_1 \right)$$


// Update error and new value for  $h_{i+1}$ 
 $e = |h_{i+1}^{n+1} - h_{i+1}^n|$ 
 $h_{i+1}^n = h_{i+1}^{n+1}$ 
}
```

7.7 Details on S_i , Q_i

Let $\det \mathbf{M}_i = m_i^{11}m_i^{22} - m_i^{12}m_i^{21}$,

$$\mathbf{M}_i^{-1} = \frac{1}{\det \mathbf{M}_i} \begin{bmatrix} m_i^{22} & -m_i^{12} \\ -m_i^{21} & m_i^{11} \end{bmatrix} \quad (7.52)$$

Then $\mathbf{S}_i$ and $\mathbf{Q}_i$ rewrite

$$\mathbf{S}_i = \mathbf{M}_i^{-1} \mathbf{N}_i = \frac{1}{\det \mathbf{M}_i} \begin{bmatrix} m_i^{22}n_i^{11} - m_i^{12}n_i^{21} & m_i^{22}n_i^{12} - m_i^{12}n_i^{22} \\ -m_i^{21}n_i^{11} + m_i^{11}n_i^{21} & -m_i^{21}n_i^{12} + m_i^{11}n_i^{22} \end{bmatrix} \quad (7.53)$$

$$\mathbf{Q}_i = \mathbf{M}_i^{-1} \mathbf{P}_i = \frac{1}{\det \mathbf{M}_i} \begin{bmatrix} m_i^{22} p_i^{11} - m_i^{12} p_i^{21} \\ -m_i^{21} p_i^{11} + m_i^{11} p_i^{21} \end{bmatrix} \quad (7.54)$$

Conclusions and perspective

8.1 Conclusions

Conservations laws such as (2.1) are an important family of PDEs that are encountered in a broad range of engineering domains. In many cases, the behavior of such systems can have an impact on the performance of the entire system. Therefore, there is a practical need for stabilization tools for such systems. The tools should be, as far as possible, easily understandable, tunable and practically implementable in real life. Unfortunately, PDEs have a mathematical complexity that requires a significant amount of mathematical analysis in order to prove the existence of the solutions and to compute them. Therefore, the understanding of such system can easily be obfuscated from a reader by the Mathematical complexity.

In this thesis, we have studied new tools for designing boundary control laws that try to satisfy both sides, Mathematical and Engineer. A rigorous Mathematical approach was taken in order to define and prove a new stability condition in Theorem 2.28. Using this result, two new design approaches for stabilizing control laws were developed in Chapter 3 and 5. The first approach relies on the tracking of Riemann invariants solely while the second uses a Lyapunov based design methodology to derive control laws, and afterwards, use Theorem 2.28 to prove the stability result. In order to have valuable results from a practical point of view, a new representation of the Riemann invariants, the invariant graph in Chapter 3, was developed. This representation was successfully used to compute an analytic solution of the stability condition of Theorem 2.28 sufficiently simple to be able to tune it "by hand".

The Lyapunov approach was also applied to the problem of stabilizing a moving tank. By studying problems with increasing complexity, two families of stabilizing boundary laws were designed: full state feedback that required the knowledge of the entire state and output feedback that required the knowledge of a limited number of parameters such as the water depth at both boundaries and the tank acceleration. The full state feedback is a pure mathematical result that has no practical interest since the full state of the system cannot

be usually measured. However, the output feedback is implementable in a real plant and therefore, fulfill the engineer need.

8.2 Perspectives

In Remark 2.4, we stated that the thesis was not handling the case where the second member of (2.1) is non zero (2.11):

$$\partial_t u + A(u)\partial_x u = h(x, t, u) .$$

Usually, in physical systems, the function h has a significant importance and cannot be neglected. For example, in the case of channels and Saint-Venant equations, the second hand term represents the slope and the friction walls as described in Section 7.1.1. The control design techniques presented in the thesis have been applied to such system. Numerical simulations have shown that the approach scaled very well to such systems. For sure, extending the Theorem 2.28 to systems of the form (2.11) would open the design approach to other interesting problems.

Throughout the stability analysis, the assumption of a constant steady state $\bar{u}$ is made. While this assumption is comfortable from a mathematical point of view, many engineering problems do not have a static steady state. For example, the input flow in channels is generally time varying (because of external environmental perturbations) and the steady state flow tracked by the controller should be adapted in order to take into account this modification of the inflow. This problem opens the door to a wider subject which is the "coordination" of the local controllers in order to achieve a "better" stability of the system. Currently, the approach that we have taken considers that each controller is separated from the others without any communication. In a real life scenario, the operator has a central controller that can modify the gains and steady state tracked by each local controller. Therefore, this brings a new control problem at the "network" level:

Given a network constituted of n systems, design a controller that choose the steady states of each system in order to minimize the perturbation of the system.

References

1. Berman A. and Plemmons R.J. *Nonnegative Matrices in the Mathematical Sciences*. SIAM, 1994.
2. Gibbons A. *Algorithmic Graph Theory*. Cambridge University Press, 1999.
3. H.N. Abramson. Slosh suppression. (NASA SP-106 1966), 1966 (Corrected and reprinted by Southwest Research Institute, 1982).
4. H.N. Abramson. Slosh suppression. Technical Report NASA SP-8031, National Aeronautics and Space Administration, 1969.
5. Ali H. Nayfeh Ahmed F. Abdel Gawad, Saad A. Ragab and Dean T. Mook. Roll stabilization by anti-roll passive tanks. *Ocean Engineering*, 28(5):457–469, 2001.
6. A. Astolfi, R. Ortega, and R. Sepulchre. Stabilization and disturbance attenuation of nonlinear systems using dissipativity theory. *European Journal of Control*, 8:408–431, 2002.
7. S. Bubhash Babu and S.K. Bhattacharya. Finite element analysis of fluid-structure interaction effect on liquid retaining structures due to sloshing. *Computers & Structures*, 59(6):1165–1171, 1994.
8. 2000) Bernhard Riemann (transcription D.R. Wilkins. Über die fortpflanzung ebener luftwellen von endlicher schwingungsweite. *Gott. Abh. Math. Cl.*, 8:43–65, 1860.
9. W. Boothby. *An Introduction to Differentiable Manifolds and Riemannian Geometry*. Academic Press, San Diego, 2nd edition edition, 1986.
10. Alberto Bressan. *Hyperbolic Systems of Conservation Laws*. Oxford University Press, 2000.
11. C.I. Byrnes, A. Isidori, and J.C. Willems. Passivity, feedback equivalence, and the global stabilization of minimum phase nonlinear systems. *IEEE Trans. Autom. Control*, 36:1228–1240, 1991.
12. Alain Cariou and Guido Casella. Liquid sloshing in ship tanks: a comparative study of numerical simulation. *Marine Structures*, 12:183–198, 1999.
13. H. Cartan. *Calcul différentiel*. Hermann, 1967.
14. Jr. Charles C. Smith. The effect of fuel sloshing on the lateral stability of a free-flying airplane model. Technical report, NACA, Langley Memorial Aeronautical Laboratory, 1948.

15. Gui-Qiang Chen and Hermano Frid. Decay of entropy solutions of nonlinear conservation laws. *Archive for Rational Mechanics and Analysis*, 1999.
16. V.T. Chow. *Open-Channel Hydraulics, International Student Edition*. Mc Graw-Hill Civil Engineering Series, 1954.
17. J.-M. Coron. Local controllability of a 1-d tank containing a fluid modeled by the shallow water equations. *ESAIM: COCV*, 8:513–554, 2002.
18. J.-M. Coron, B. d’Andréa-Novel, and G. Bastin. A Lyapunov approach to control irrigation canals modeled by Saint-Venant equations. In *CD-Rom Proceedings, Paper F1008-5*, Karlsruhe, Germany, September 1999. ECC 99.
19. J.-M. Coron, J. de Halleux, G. Bastin, and B. d’Andréa Novel. On boundary control design for quasilinear hyperbolic systems with entropies as lyapunov functions. In *41st IEEE Conference on Decision and Control (CDC02), Las Vegas, USA, December, 2002.*, 2002.
20. R. Courant and D. Hilbert. *Methods of Mathematical Physics*, volume II. Interscience Publishers, 1962.
21. R.F. Curtain and H. Zwart. *An introduction to infinite-dimensional linear systems theory*. Springer-Verlag, New York, 1995.
22. Clint Dawson and Jennifer Proft. Discontinuous and coupled continuous/discontinuous galerkin methods for the shallow water equations. *Comput. Methods Appl. Mech. Engrg.*, 191:47214746, 2002.
23. J. de Halleux and G. Bastin. Stabilization of Saint-Venant equations using Riemann invariants: Application to waterways with mobile spillways. In *CD-Rom proceedings*, Barcelona, Spain, July 2002.
24. J. de Halleux, B. d’Andréa-Novel, J.-M. Coron, and G. Bastin. A Lyapunov approach for the control of multi reach channels modelled by Saint-Venant equations. In *CD-Rom Proceedings, pg. 1515-1520*, St-Petersburg, Russia, June 2001. NOLCOS’01.
25. J. de Halleux, Ch. Prieur, and G. Bastin. Boundary control design for cascades of hyperbolic 2×2 pde systems via graph theory. In *43rd IEEE Conference on Decision and Control*, 2004.
26. L. Debnath. *Nonlinear water waves*. Academic Press, Boston, 1944.
27. Jean Dieudonné. *Foundations Of Modern Analysis*. Academic Press, New York & London, 1960.
28. F. Dubois, N. Petit, and P. Rouchon. Motion planning and nonlinear simulations for a tank containing a fluid. In *CD-Rom Proceedings*, Karlsruhe, Germany, September 1999. ECC 99.
29. J.T. Feddema, C.R. Dohrmann, G.G. Parker, R.D. Robinett, V.J. Romero, and D.J. Schmitt. Control for slosh-free motion of an open container. *IEEE Control Systems*, 17(1):29–36, 1977.
30. J.-F. Gerbeau and B. Perthame. Derivation of viscous Saint-Venant system for laminar shallow water; numerical validation. *Research report RR_4084, INRIA Rocquencourt*, 2000.
31. Jean-Michel Ghidaglia and Frédéric Pascal. On boundary conditions for multi-dimensional hyperbolic systems of conservation laws in the finite volume framework. Technical report, Report CMLA, Ens de Cachan, 2003.
32. S.K. Godunov. Finite difference method for numerical computations of discontinuous solutions of the equations of fluid dynamics. *Mat. Sbornik*, 47:271–306, 1956.
33. W. Graf. *Fluvial Hydraulics*. John Wiley & Sons, 1998.

34. J.-M. Greenberg and Ta-tsien Li. The effect of boundary damping for the quasilinear wave equations. *J. Differential equations*, 52:66–75, 1984.
35. M. Grundelius. *Motion Control of Open Container With Slosh Constraints*. PhD thesis, Department of Automatic Control, Lund Institute of Technology, Lund, November 1998.
36. M. Grundelius. Iterative optimal control of liquid slosh in an industrial packaging machine. In *Conf. on Dec. and Contr. Sydney, Australy*, 2000.
37. M. Grundelius and B. Bernhardsson. Control of liquid slosh in an industrial packaging machine. In *Proc. IEEE Int. Conf. Cont. Applications*, 1999.
38. A. Haraux. *Systèmes dynamiques dissipatifs et applications*. Masson, Paris, 1991.
39. Ami Harten, Peter D. Lax, C. David Levermore, and William J. Morokoff. Convex entropies and hyperbolicity for general euler equations. *SIAM J. NUMER. ANAL.*, 35(6):2117–2127, December 1998.
40. L. Hormander. *Lectures on nonlinear hyperbolic differential equations*, volume 26. 1997.
41. de Halleux J., Prieur C., Coron J.-M., d’Andréa-Novel B., and Bastin G. Boundary feedback control in networks of open channels. *Automatica*, 39(8):1365–1376, August 2003.
42. Donald E. Knuth. *Literate Programming*. Stanford, California: Center for the Study of Language and Information, 1992.
43. Olga Koroleva, M. Krstic, and Y. Hu. Nonlinear control of mine ventilation networks. In *Proc. 41th IEEE CDC, Las Vegas*, 2002.
44. Vladimir Kozlov and Nikolay Kuznetsov. On the fundamental sloshing frequency in the ice-fishing problem. *C.R.Mécanique*, 330:723–728, 2002.
45. M. Krstic, I. Kanellakopoulos, and P. Kokotovic. *Nonlinear and adaptive control design*. John Wiley & Sons, New-York, 1995.
46. P. Lax. *Contributions to Nonlinear Functional Analysis*, chapter Shock waves and entropy, pages 603–634. Academic Press, New York, 1971.
47. Peter. Lax. *Hyperbolic systems of Conservation Laws and The Mathematical Theory of Shock Waves*, volume 11 of *Conference Board of the Mathematical Sciences Regional Conference Series in Applied Mathematics*. SIAM, Philadelphia, 1973.
48. Ta-tsien Li. *Global classical solutions for quasilinear hyperbolic systems*. Research in Applied Mathematics, Masson and Wiley, Paris Milan Barcelona, 1994.
49. Ta-tsien Li and Wen-Ci Yu. *Boundary value problems for quasilinear hyperbolic systems*. Duke university mathematical series - DUM. Duke university press (Durkam), 1985.
50. D. Liberzon, A. S. Morse, and E. Sontag. Output-input stability and minimum-phase nonlinear systems. *IEEE Trans. Autom. Control*, 47:422–436, 2002.
51. P.O. Malaterre. *Modelisation, Analysis and LQR Optimal Control of an Irrigation Canal*. PhD thesis, LAAS-CNRS-ENGREF-Cemagref, 1994.
52. F. Mazenc and L. Praly. Adding integrations, saturated controls and global asymptotic stabilization for feedforward systems. *IEEE Transactions on Automatic Control*, 41(11), November 1996.
53. M.V. Modi and M.L. Seto. Suppression of flow-induced oscillations using sloshing liquid dampers: analysis and experiments. *Journal of Wind Engineering and Industrial Aerodynamics*, 67&68:611–625, 1997.

54. V.J. Modi and A. Akinturk. An efficient liquid sloshing damper for control of wind-induced instabilities. *Journal of Wind Engineering and Industrial Aerodynamics*, 90:1907-1918, 2002.
55. S. Mottelet. Controllability and stabilizability of liquid vibration in a container during transportation. In *Conf. on Dec. and Contr., Sydney, Australia*, 2000.
56. N. Petit and P. Rouchon. Dynamics and solutions to some control problems for water-tank systems. *CIT-CDC*, 0(004), 2000.
57. L. Praly. *Fonctions de Lyapunov et Stabilisation de point d'équilibre (preprint)*. Ecole des Mines de Paris, France, 2002.
58. L. Praly, R. Ortega, and G. Kaliora. Stabilization of nonlinear systems via forwarding modulo lgv. *IEEE Trans. Autom. Control*, 46(9):1461-1466, 2001.
59. C. Prieur and J. de Halleux. Complete stabilization of a tank. In *Journées Doctorales d'Automatique 2001*, 2001.
60. C. Prieur and J. de Halleux. Stabilization of a tank via output feedback. In *Conf. on Dec. and Cont.*, 2002.
61. Ch. Prieur and de Halleux J. Complete stabilization of a tank. In *CD-Rom proceedings*, Barcelona, Spain, July 2002.
62. Christophe Prieur and Jonathan de Halleux. Stabilization of a 1-d tank containing a fluid modeled by the shallow water equations. *Systems & Control Letters*, 52(3-4):167-178, July 2004.
63. B. de Saint-Venant. Théorie du mouvement non-permanent des eaux avec applications aux crues des rivières et à l'introduction des marées dans leur lit. *Comptes-rendus de l'Académie des Sciences*, 73:148-154, 237-240, 1871.
64. D. Serre. *Systèmes de lois de conservations I and II*. Diderot éditeur, Art et Sciences, 1996.
65. Aliabadi Shahrouz, Johnson Andrew, and Abedi Jalal. Comparison of finite element and pendulum models for simulation of sloshing. *Computers & Fluids*, 32:535-545, 2002.
66. L. Sine, S. Gaspar, and F. Corlier. Méthode de calcul du débit de la sambre namur. Technical report, Faculté des Sciences Agronomiques, Hydraulique, Gembloux, 1965.
67. R. Venugopal and D.S. Bernstein. State space modeling and active control of slosh. In *Proc. IEEE Int. Conf. Cont. Applications, Dearborn, MI*, pages 1072-1077, 1996.
68. P. Warnitchai and T. Pinkaew. Modelling of liquid sloshing in rectangular tanks with flow-dampening devices. *Engineering Structures*, 20(7):593-600, 1998.
69. Hervé Wathelet. Les barrages modiles: Généralités et régulation. Travail de fin d'étude, Faculté des Sciences Appliquées, Université de Liège., 1999-2000.
70. K. Yano, T. Yoshida, M. Hamaguchi, and K. Terashima. Liquid container transfer considering the suppression of sloshing for the change of liquid level. In *Proc. 13th IFAC World Congress, San Francisco, CA*, 1996.

A

Mathematical preliminaries

A.1 Implicit Function Theorem

In the following, let $m > 0$ positive and $n = 2m$. Let us define B as the boundary law method defined in (2.51).

Theorem A.1 (Implicit Function Theorem). *Let $B : \mathbb{R}^m \times \mathbb{R}^m \mapsto \mathbb{R}^n$ be k times continuously differentiable, with $k \geq 1$. If $B(0, 0) = 0$ and the Jacobian matrix $\nabla_{[\xi_-(1,t), \xi_+(0,t)]} B(0, 0)$ is invertible, then there exists a neighborhood $\mathbb{N}$ of 0 and a $\mathbb{C}^k$ mapping $\mathbf{G} : \mathbb{N} \mapsto \mathbb{R}^m$ such that*

$$\mathbf{G} \begin{pmatrix} \xi_-(0, t) \\ \xi_+(1, t) \end{pmatrix} = 0, \quad (\text{A.1})$$

$$B \left(\begin{pmatrix} \xi_-(0, t) \\ \xi_+(1, t) \end{pmatrix}, \mathbf{G} \begin{pmatrix} \xi_-(0, t) \\ \xi_+(1, t) \end{pmatrix} \right) = 0 \quad (\text{A.2})$$

If the k -th derivatives of B are Lipschitz continuous, then the same is true of the k -th derivatives of $\mathbf{G}$. The derivative of $\mathbf{G}$ at the point 0 is the $m \times m$ Jacobian matrix

$$\nabla \mathbf{G}(0) = -[\nabla_{[\xi_-(1,t), \xi_+(0,t)]} B(0, 0)]^{-1} \nabla_{[\xi_-(0,t), \xi_+(1,t)]} B(0, 0) . \quad (\text{A.3})$$

Proof. See e.g. [27, 13, 9].

B

List of notations

Symbol	Description
MIBVP	Mixed Initial Boundary Value Problem
SWE	Shallow Water Equations
n	dimension of the state
m	number of 2×2 systems
T	a positive time instant
U	open non-empty convex set of $\mathbb{R}^n$
x	space coordinate
t	time coordinate
$u(x, t)$	Distributed state
$u^{\natural}$	Value of u at $t = 0$
$A(u)$	Characteristic matrix
K	linear Characteristic matrix
$B_{\dots}(u)$	Boundary condition
$\xi_i(x, t)$	Riemann invariant in the i -th system
$\boldsymbol{\xi}(x, t)$	Vector of Riemann invariants
$\boldsymbol{\xi}_{-}(x, t)$	Vector of backward Riemann invariants
$\boldsymbol{\xi}_{+}(x, t)$	Vector of forward Riemann invariants
$\lambda_i(x, t)$	Characteristic velocity in the i -th system
$\Lambda(x, t)$	Diagonal matrix of characteristic velocities
∂_t, ∂_x	Partial derivative w.r.t. the time and space dimensions
∇_u	Jacobian w.r.t. the vector u
$l_i(u)$	i -th left eigenvector of $A(u)$
$r_i(u)$	i -th right eigenvector of $A(u)$
$\mathbf{G}_{\dots}(\dots)$	boundary law expressed in Riemann invariants
$\rho(A)$	spectral radius of the matrix A
$\xi(t, \cdot) \ _{\mathbb{C}^1}$	$\xi(t, \cdot) \ _{\mathbb{C}^0} + \ \partial_x \xi(t, \cdot)\ _{\mathbb{C}^0}$

Table B.1. List of notations

C

Saint-Venant equation simulator

In parallel to the theoretical study of the channels, a simulator was developed in order to illustrate the result using numerical simulation. The simulator uses the numerical model described in Chapter 7. It can be used to simulate an arbitrary number of reaches in cascade with different gates and controllers. The simulator comes with a built-in support for visualisation, data analysis and report generation.

The simulator has been developed in C++ (using Visual Studio 6.0).

C.1 Minimum requirements

- PC with Windows NT4, 2000, Me or XP.
- PII minimum,
- The RAM, the better.

C.2 Simulation configuration

This section describes the different step that are necessary to set up and execute a simulation using **WinChannel**.

C.2.1 General parameters - Figure C.1

1. (optional) Give a name and a description to the current simulation. The name is used in the visualization and in the automatic report generation.
2. Choose the time step length (dt) and the total time (T). Automatically,
 - the number of the number of time steps n_t is updated using

$$n_t = \left\lfloor \frac{T}{dt} \right\rfloor,$$

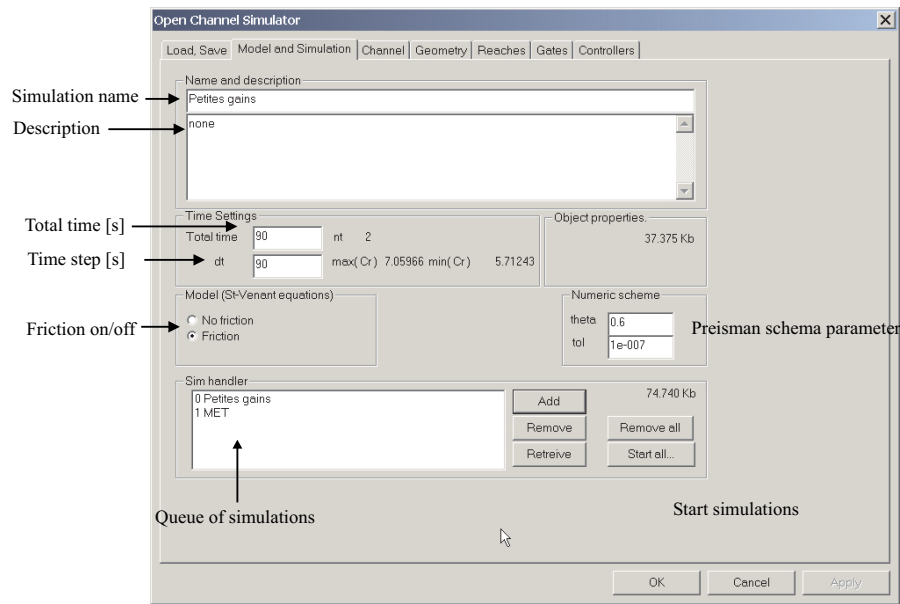


Fig. C.1. Simulation manager page

- minimum and maximum Courant numbers C_r are computed,
 - an estimation of the memory requirements for the simulation is displayed.
3. Select to add friction or not in the model (Friction, No Friction)
 4. Choose θ
 5. Once the rest of the simulation is set up, add the simulation in the simulation execution queue.
 6. When all simulation are ready, launch the computation using *Start All*.

C.2.2 Number of reaches - Figure C.2

This page is used to set the number of reaches and the upstream and downstream water level.

1. Set the number of reaches,
2. Set the upstream and downstream water level. Those water levels are used in the case of sluice gates (see Section C.2.5).
3. Set the initial and steady state inflow, Q_0 et $\bar{Q}$. If lateral inflows or outflows are added to the reaches, **WinChannel** updates the flow curves.

C.2.3 Description of the geometry - Figure C.3

This page is used to describe the dimensions of the reaches and the reference levels are the gates.

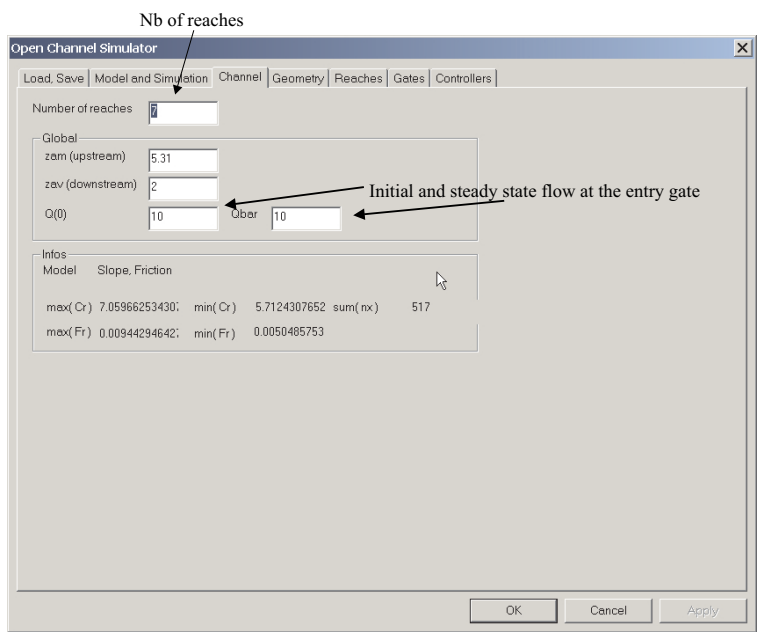


Fig. C.2. Configuration of the channel properties

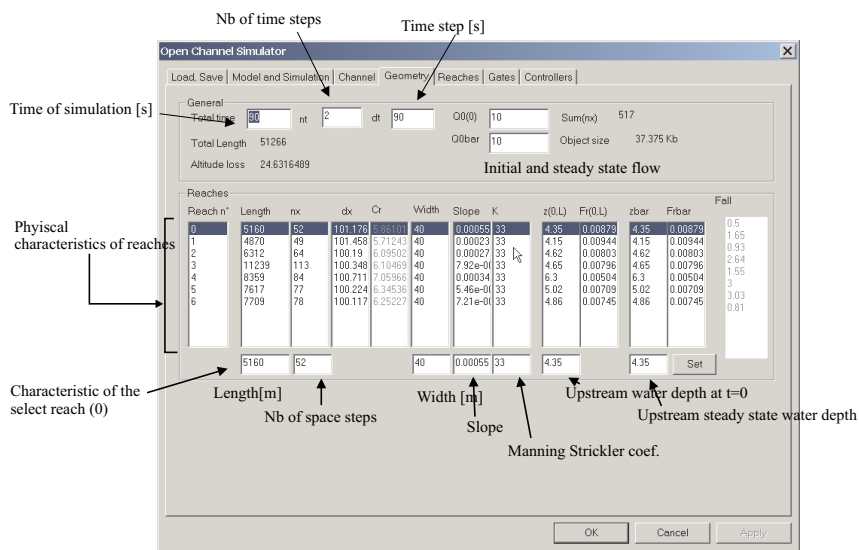


Fig. C.3. Configuration of the reaches physical properties

1. (optional) Modify the total time, time step or number of time steps (see Section C.2.1).
2. (optional) Modify the initial and steady state inflow (see Section C.2.2).
3. For each reach, modify the following parameters:
 - $Length$, reach length in meters [m],
 - n_x , number of space discretisation steps in the reach,
 - d_x et C_r are automatically updated with respect to the length and n_x ,
 - $Width$, the width of the reach in [m]. The section is considered rectangular with constant width,
 - $slope$, the reach slope expressed in $\tan(\gamma)$, where γ is the slope in radians,
 - K , Manning-Strickler coefficient,
 - $z(0, L)$, water depth in the upstream reach at time 0 and $x = L$,
 - $\bar{z}(L)$, steady state water depth in the upstream reach at $x = L$,
4. Once the parameters are set, hit the 'Set' button,
5. Do this procedure for each reach of the channel

C.2.4 Detailed configuration of each reach

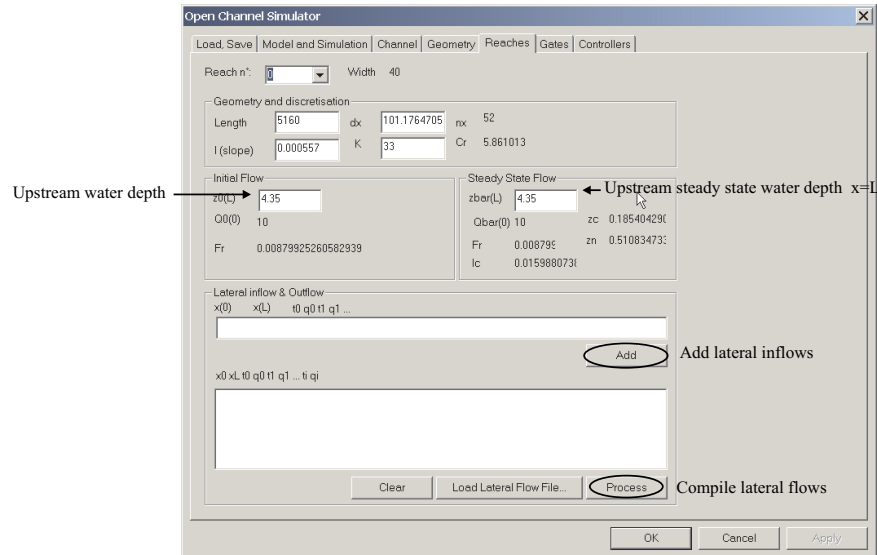


Fig. C.4. Detailed configuration of a reach

This page is used to set up detailed information on each reach.

Choose a reach with the dropdown listbox. Each reach is numbered from 0 at the upstream and increasingly along the downstream reaches. The param-

eters of each reaches can be modified: length, discretisation step, slope, etc... The following hydraulic quantities are computed:

- F_r , Froude number,
- z_c , Critical water depth,
- z_n , Normal water depth. For a normal water depth, the water line is parallel to the bottom,
- I_c , Critical slope. If the reach slope is greater than this value, the flow will be super-critical and the simulation will fail.

Lateral inflows and outflows

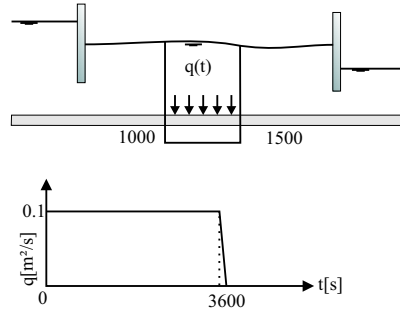


Fig. C.5. Schematic view of a lateral flow

WinChannel supports multiple, time varying, lateral inflow and outflow per reaches. In each lateral flow zone, the user provides:

- the portion of the reach where the lateral flow is active,
- a list of (t_i, q_i) pairs who describe the evolution of the flows with respect to the time. Positive value of q_i correspond to inflows and conversely negative values of q_i correspond to outflows. Lateral flows are expressed in m^3/s . Hence, for a total lateral flow Q_i distributed on L_i meters, the value of q_i is computed as follows:

$$q_i = \frac{Q_i}{L_i} .$$

WinChannel handles the interpolation (linear) between the value pairs.

If the simulation time is greater than the last value of t_i , **WinChannel** assumes that the lateral flows remain constant at the last value of q_i .

- Click 'Add' and add the appropriate list of values,
- Once all values added, click process to update the simulation.

Example C.1. For a lateral flow of 10m^3 between 1000m and 1500m during one hour as depicted in Figure C.5, the following values must be used:

$$\underbrace{1000 \ 1500}_{\text{flowdomain}} \quad \underbrace{0 \ 0.1 \ 3600 \ 0.1}_{\text{onehour}} \quad \underbrace{3601 \ 0}_{\text{nomoreflow}}$$

C.2.5 Detailed configuration of each gate

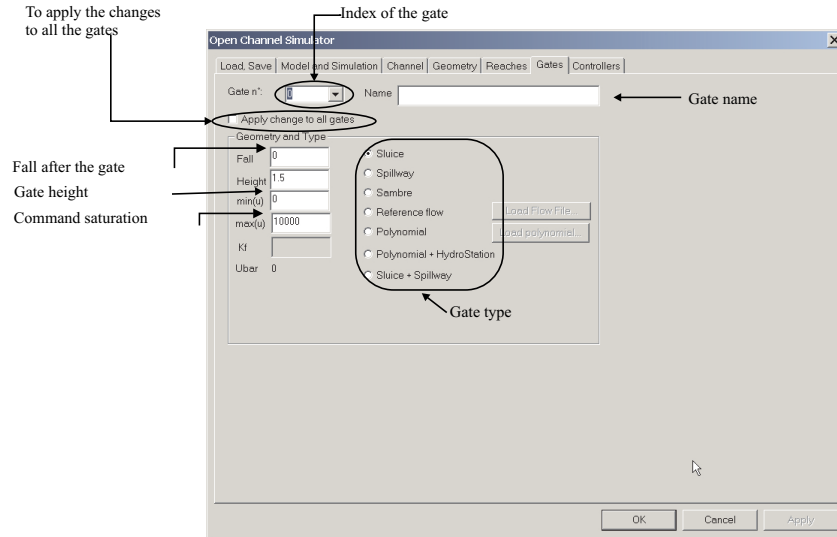


Fig. C.6. Gate configuration

WinChannel supports different types of classic hydraulic gates which can be customized easily. Gates are numbered from 0 at the upstream, and increasingly along the downstream reaches.

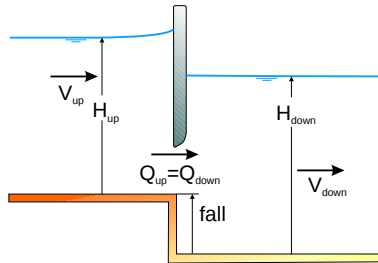


Fig. C.7. Sluice gate.

In general, to set up a gate, the following information must be provided:

1. (optional) A name. This name is used in the result visualization and the report.

2. Physical parameters $Fall$, $Height$, $min(u)$, $max(u)$

Sluice gate

In this particular gate, the water flow passes under the sluice. The discharge at the gate is modelled as follows:

$$Q = uk\sqrt{h_{\text{up}} - h_{\text{down}}} .$$

The parameter k is updated for the simulation in order to match the steady state and has no physical meaning.

Spillway

In this particular gate, the flow passes over the nappe:

$$Q = k(h_{\text{up}} - u)^{3/2} .$$

The parameter k is fixed for the simulation in order to match the steady states. The action u represents the height of the spillway in meters.

Sambre Gates

This model, similar to a Spillway, can be found in [66]. The discharge at the gate is modelled by:

$$Q = c(u)l\sqrt{2g}(h_{\text{up}} - u)^{3/2} .$$

The flow coefficient c depends on the height of the nappe and is computed by interpolating a value from a table.

Reference flow

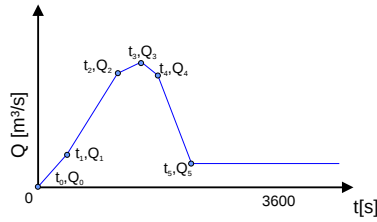


Fig. C.8. Fixed inflow.

The flow going through a gate can be bound to a function of the time:

$$Q = f(t) .$$

The function $f(t)$ is usually loaded from a data file containing (t_i, Q_i) pairs:

$$t_0 \ Q_0 \ t_1 \ Q_1 \ \cdots t_n \ Q_n$$

for n arbitrary.

This flow is useful to set an inflow at the first gate of the channel.

Polynomial

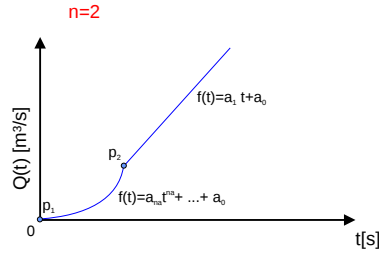


Fig. C.9. Polynomial discharge law

The flow going through the gate can be set as a user given function of the height of the slip.

$$Q = f_{\text{user}}(h_{\text{up}} - u) .$$

The user must provide a file containing the description of the piecewise continuous function.

$$\begin{array}{c} n \\ p_0 \ p_1 \ \cdots \ p_{n-2} \\ n_a \ a_0 \ a_1 \ \cdots \ a_{n_a-1} \} \ n \ \text{fois} \end{array}$$

where n is the number of polynomial, kp are the coordinates of the polynomial junctions, a_i are the coefficient of the polynomials in decreasing orders:

$$f(x) = a_0 x^{n_a} + a_1 x^{n_a-1} \cdots + a_{n_a-1}$$

C.2.6 Detailed configuration of each controller

WinChannel allows the user to chose a controller type for each individual gate.

For each controller, the following characteristics can be set:

1. the *gain* is generally used for the tuning of the controller,
2. the *delay* is the time in seconds between two actions,

If the gate has a fixed flow, the controller has no effect.

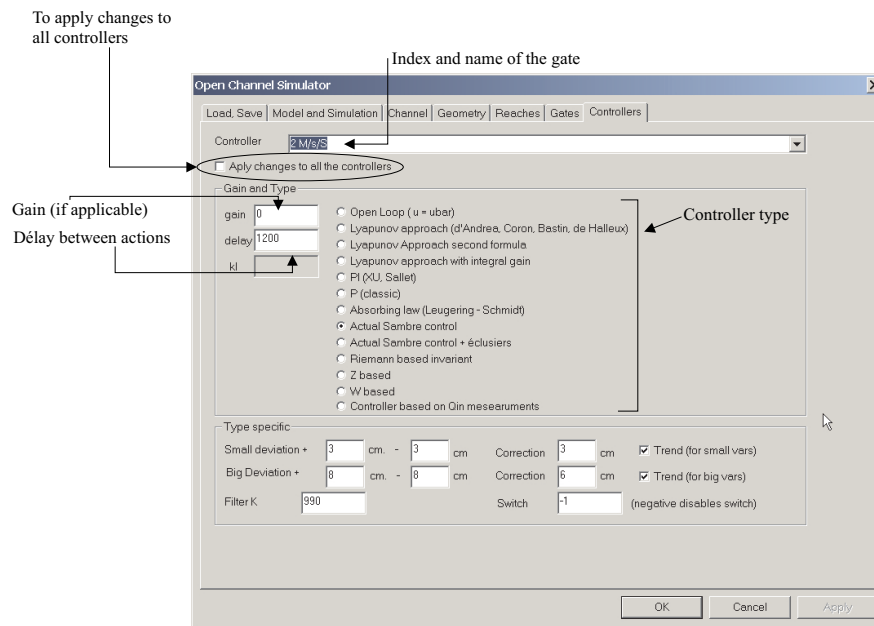


Fig. C.10. Controller configuration

C.3 Visualisation of the results

This section describes the different possibilities of data analysis provided by **WinChannel**.

C.3.1 Animations

C.3.2 Flow and water depth.

The Figure C.11 represents the window for visualizing the flow and the water depth. This display can be animated using the animation toolbar (see Figure C.3.5).

C.3.3 3D display

C.3.4 Toolbars

Loading and saving

This toolbar can be used to load and save simulations.

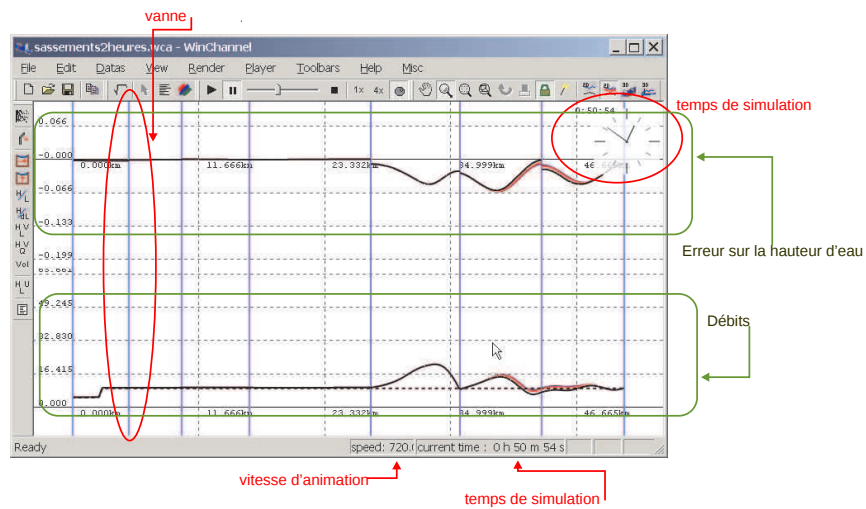


Fig. C.11. 2D view

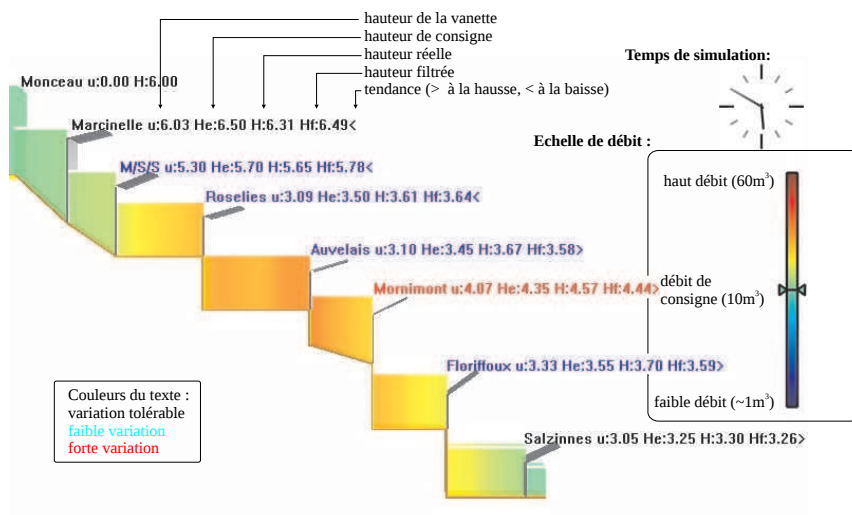


Fig. C.12. 3D view

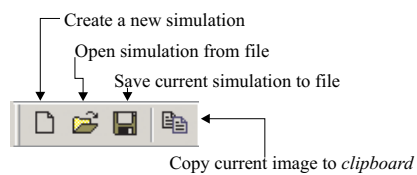


Fig. C.13. IO toolbar

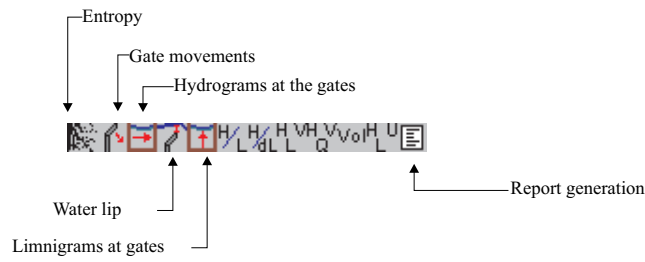


Fig. C.14. Result analysis toolbar

C.3.5 Result analysis

The toolbar provides access to result analysis and visualization of the results. The following quantities can be displayed:

- The entropy,
- Gate movements,
- Flow variation at the gates,
- Water depth variation at the gates
- Water volume in each reach,
- Profile of the water, flow and velocity at different time instants. In the

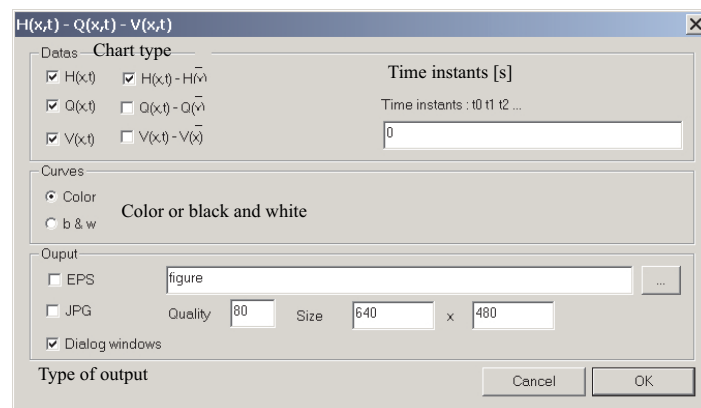


Fig. C.15. Profiles

C.15 dialog window,

1. Chose the type of profile:
 - water depth, $H(x, t)$,
 - deviation of the water depth with respect to the steady state, $H(x, t) - \bar{H}(x)$,
 - flow, $Q(x, t)$,

- deviation of the flow with respect to the steady state flow, $Q(x, t) - \bar{Q}(x)$,
- velocity, $V(x, t)$,
- deviation of the velocity with respect to the steady state velocity, $V(x, t) - \bar{V}(x)$,
- 2. Choose the time instant where the profile must be outputted,
- 3. Choose color or black and white figures,
- Flow, water depth or gate movement can be compared using the dialog depicted in Figure C.16

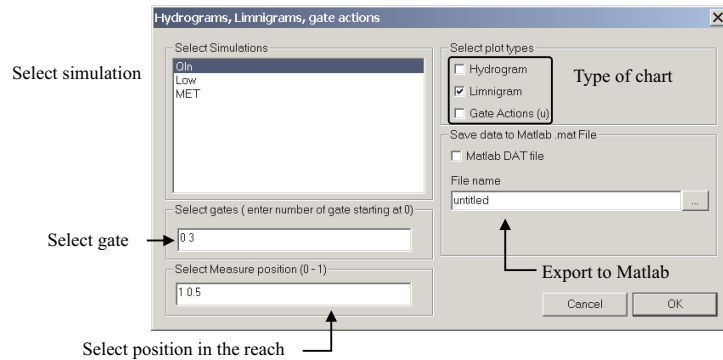


Fig. C.16. Dialog window to compare flows, water depth and gate movement

- Report generation: **WinChannel** can generate reports in *Postscript*. **This functionality requires MikTeX**

Real time Visualisation

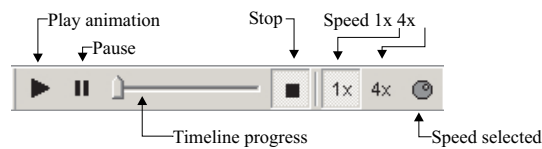


Fig. C.17. Animation toolbar

Simulation

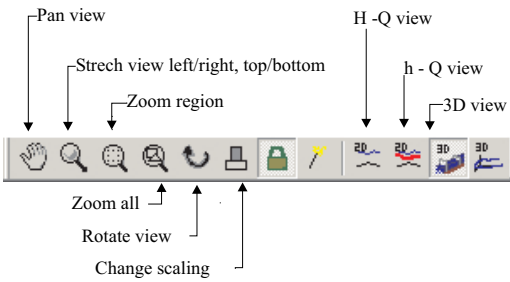


Fig. C.18. Visualization toolbar

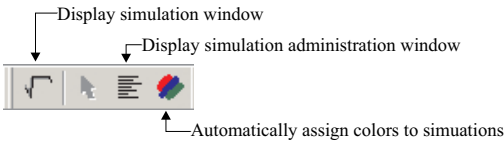


Fig. C.19. Tools toolbar

D

Data structure and algorithms for Flux graph and Invariant graphs

This chapter describes the software used to generate the network diagrams showed in the above section. At first, the necessary data structures are described, then the algorithm that builds an invariant graph from a flux graph is given.

The algorithms below are given in a *pseudo-code* similar to popular languages like Java or C# (the original implementation is done in C#). The style used to describe the algorithms is similar to the literate programming introduced by D. Knuth in [42] where the algorithms are separated into several pieces in order to make them more understandable.

D.1 Data structures

In the following, we shall consider that a data structure that models a bidirectional directed graph is available. Generally, this structure is expressed in terms of two abstract parameters: **V** for the vertex type and **E** for the edge type.

Listing D.1. 'Graph'

```
class Graph<V,E>
{
    ICollection<V> Vertices; // returns the collection of vertices
    ICollection<E> Edges; // returns the collection of edges

    AddVertex(V v); // add a vertex
    AddEdge(E e); // adds an edge
}
```

where *ICollection* < *T* > usually denotes a collection of *T* objects.

This structure is further specialized into a **FluxGraph** and a **InvariantGraph** by selecting the appropriate type for **V** and **E**. The **FluxGraph** structure is defined by the **BoundaryVertex** class for the vertices, and the **DistributedEdge** for the edges:

Listing D.2. 'FluxGraph'

```

class BoundaryVertex
{
    // the index of the node
    int Index;
}
class DistributedEdge;

class FluxGraph : Graph<BoundaryVertex , DistributedEdge >;

```

The **InvariantGraph** is defined by the following:

Listing D.3. 'InvariantGraph'

```

class InvariantVertex
{
    // edge from which the invariant is coming from
    DistributedEdge FromEdge;
    BoundaryNode Node;
}

class InvariantEdge
{
    // edge to which the invariant is associated
    DistributedEdge FromEdge;
    // true if the edge is associated to a
    // positive Riemann invariant
    bool Forward;
}

class InvariantGraph : Graph<InvariantVertex , InvariantEdge >;

```

For the sake of simplicity, we shall also define the following variables:

Listing D.4. 'Variables'

```

// the flow graph
FluxGraph flux;
// the invariant graph
InvariantGraph inv;
//the number of edges in the flow graph
int edgesCount=flux.EdgesCount;

```

D.2 Algorithm

The goal of the algorithm is, given a **FluxGraph** instance, to build the corresponding **InvariantGraph**. The method **CreateInvariantGraph** defines the two mains steps to achieve the following: first, create the vertices of the invariant graph, then link them as appropriate.

Listing D.5. 'CreateInvariantGraph'

```

CreateInvariantGraph ()
{
    AddVerticesToInvariantGraph ();
    AddEdgesToInvariantGraph ();
}

```

Each distributed edge, i.e. distributed system, generates two vertices associated to each of its extremity node.

The method `AddVerticesToInvariantGraph` iterates over the edges of the flux graph and adds two invariant vertex per distributed edge:

Listing D.6. 'AddVerticesToInvariantGraph'

```
AddVerticesToInvariantGraph()
{
    // for each distributed edge,
    // add invariants vertices for each boundary
    foreach(DistributedEdge e in flux.Edges)
    {
        inv.AddVertex(e, e.Source);
        inv.AddVertex(e, e.Target);
    }
}
```

Each new invariant edge stores the distributed edge that generated him and the corresponding boundary node:

Listing D.7. 'InvariantGraph.AddVertex'

```
InvariantGraph.AddVertex(
    DistributedEdge fromEdge,
    BoundaryNode node
)
{
    InvariantVertex v=this.graph.AddVertex();
    v.FromEdge = fromEdge;
    v.Node = node;
}
```

The second step, building the invariant edges, is done as follows: we shall create an edge between each pair `source,target` of invariant vertices if `source` and `target` are located at the extremity of the same boundary node:

Listing D.8. 'AddEdgesToInvariantGraph'

```
AddEdgesToInvariantGraph()
{
    foreach(InvariantVertex source, target in inv.ValidPairsOfVertices)
    {
        // add edge from source to target
        inv.AddEdge(source, target);
    }
}
```