



HAL
open science

Dassflow :a Direct and Adjoint model for 2D Shallow Water flows

Marc Honnorat, François-Xavier Le Dimet, Youssef Loukili, Jerome Monnier

► **To cite this version:**

Marc Honnorat, François-Xavier Le Dimet, Youssef Loukili, Jerome Monnier. Dassflow :a Direct and Adjoint model for 2D Shallow Water flows. RR-5756, INRIA. 2005, pp.21. inria-00070264

HAL Id: inria-00070264

<https://inria.hal.science/inria-00070264>

Submitted on 19 May 2006

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

***Dassflow : Un modèle direct et inverse
d'écoulements Saint-Venant 2D***

Marc Honnorat — François-Xavier Le Dimet — Youssef Loukili — Jérôme Monnier

N° 5756

Novembre 2005

Thème NUM



***rapport
de recherche***

Dassflow : Un modèle direct et inverse d'écoulements Saint-Venant 2D

Marc Honnorat^{*†}, François-Xavier Le Dimet^{‡*}, Youssef Loukili^{*},
Jérôme Monnier^{§*}

Thème NUM — Systèmes numériques
Projet IDOPT

Rapport de recherche n° 5756 — Novembre 2005 — 21 pages

Résumé : Le code de calcul Dassflow résout les équations de Saint-Venant bidimensionnelles utilisées pour la modélisation numérique des écoulements en hydraulique fluviale. Il a été écrit pour réaliser des expériences d'assimilation variationnelle de données. Basé sur une discrétisation en volumes finis des équations de Saint-Venant, le code est écrit en Fortran 90. Le code adjoint nécessaire pour calculer les dérivées partielles d'une fonction coût de la variable d'état par rapport aux variables de contrôle du modèle a été développé à l'aide de l'outil de différentiation automatique Tapenade.

Nous présentons les équations considérées, les fondements théoriques de l'assimilation variationnelle de données, le schéma numérique de résolution des équations ainsi que la mise en œuvre informatique du code adjoint. Enfin, nous présentons des expériences jumelles d'assimilation de données réalisées avec Dassflow.

Mots-clés : Assimilation de données, estimation de paramètres, hydraulique fluviale, différentiation automatique.

* LMC-IMAG, B.P. 53, 38041 Grenoble Cedex 9

† INPG, <mailto:marc.honnorat@imag.fr>

‡ UJF, <mailto:francois-xavier.le-dimet@imag.fr>

§ INPG, <mailto:jerome.monnier@imag.fr>

Dassflow : a Direct and Adjoint model for 2D Shallow Water flows

Abstract: The Dassflow software is designed to solve the bidimensional Shallow Water equations used for the numerical modeling of river hydraulics flows. It was written to carry out variational data assimilation experiments. Based on a finite volume discretization of the Shallow Water equations, the code is written in Fortran 90. The adjoint code necessary to compute the partial derivatives of a cost function of the state variables w.r.t. the control variables of the model was developed using the automatic differentiation tool Tapenade.

We present the considered equations, the theoretical bases of variational data assimilation, the numerical scheme used for solving the equations as well as the implementation of the adjoint code. Finally, we present twin experiments of data assimilation performed with Dassflow.

Key-words: Data assimilation, parameter identification, river hydraulics, automatic differentiation.

Table des matières

1	Introduction	4
2	Modèle mathématique	5
2.1	Équations	5
2.2	Conditions aux limites	5
3	Assimilation variationnelle de données	6
3.1	Fondements de l'assimilation variationnelle de données	6
3.1.1	Cadre fonctionnel	6
3.1.2	Modèle direct	7
3.1.3	Fonction coût	7
3.1.4	Modèle linéarisé	8
3.1.5	Modèle adjoint	9
3.1.6	Système d'optimalité	10
3.2	Adjoint du modèle Shallow Water	10
4	Mise en œuvre numérique	11
4.1	Discrétisation du modèle direct	11
4.1.1	Description volumes finis	11
4.1.2	Conditions aux limites	14
4.1.3	Mise en œuvre informatique	14
4.2	Mise en œuvre du modèle adjoint	15
4.2.1	Différentiation automatique	15
4.2.2	Utilisation du code adjoint	15
4.2.3	Validation du code adjoint	16
5	Résultats numériques	18
5.1	Exemple 1 : Identification de la topographie	18
5.2	Exemple 2 : Identification du débit amont	19

1 Introduction

Les équations de Saint-Venant, ou Shallow Water, sont utilisées pour décrire des écoulements géophysiques à surface libre en eaux peu profondes, pour la modélisation côtière et en hydraulique fluviale par exemple.

Pour réaliser une simulation d'un système hydraulique particulier, on doit pouvoir fournir aux modèles basés sur ces équations un certain nombre d'informations. D'une part, il faut définir des paramètres liés à la configuration physique du domaine, comme la topographie du fond de la rivière et des zones inondables. Certains phénomènes sont modélisés par des lois empiriques qui doivent être adaptées à la réalité du terrain. Ainsi, le frottement de l'eau sur le fond peut être décrit par une loi de Manning paramétrée par un coefficient qui dépend de la rugosité du sol et qui entre dans la liste des paramètres à déterminer. D'autre part, il faut fournir des informations liées à un événement particulier : ce sont les conditions initiales de l'écoulement ainsi que les conditions aux limites amont et aval de la rivière.

Ces variables de contrôle déterminent le résultat de la simulation. Malheureusement, elles ne sont souvent connues que partiellement ou avec une marge d'erreur importante, car coûteuses ou difficiles à mesurer. Le problème du calage des paramètres est donc un enjeu important pour la réalisation de simulations pertinentes. Dans cette optique, les méthodes d'assimilation de données [1] permettent de combiner de manière optimale l'information mathématique contenue dans le modèle et l'information physique provenant d'observations de l'écoulement, dans le but de trouver la valeur des variables de contrôle qui permet de faire correspondre aux mieux l'état du système à la réalité des observations. L'assimilation variationnelle de données [3] consiste à définir le jeu de paramètre optimal qui minimise une fonction coût mesurant l'écart entre les observations et la variable d'état du modèle numérique. On utilise une méthode de descente [5] qui demande le calcul du gradient de la fonction coût, obtenu à partir de l'état adjoint du système.

Le code de calcul **Dassflow** a été développé pour permettre la mise en œuvre de l'assimilation variationnelle de données sur des configurations hydrauliques simples dans un but de recherche. Il résout les équations Shallow Water par une méthode de volumes finis. Il a été codé en Fortran 90. Le code adjoint nécessaire au calcul du gradient de la fonction coût a été écrit à l'aide de l'outil de différentiation automatique TAPENADE [12], développé par l'équipe TROPICS de l'INRIA Sophia-Antipolis.

Dans la partie 2 de ce rapport, on présente les équations de Saint-Venant bidimensionnelles dont le code de calcul Dassflow fait une résolution numérique. Dans la partie 3, le principe de l'assimilation variationnelle de données est exposé d'un point de vue continu. La partie 4 est consacrée à la discrétisation des équations de Saint-Venant et la mise en œuvre informatique des modèles numériques. Enfin, des résultats numériques d'assimilation de données avec le code Dassflow sont présentés dans la partie 5.

2 Modèle mathématique

En hydraulique, les équations de Saint-Venant (ou Shallow Water en anglais) sont utilisées pour décrire les écoulements à surface libre en eaux peu profondes. Elles sont obtenues à partir des équations de Navier-Stokes incompressibles, en faisant les hypothèses de pression hydrostatique, de vitesses uniformes sur la verticale, d'un fond et d'une surface libre imperméables.

2.1 Équations

En chaque point du domaine de calcul Ω inclus dans le plan horizontal, on définit la hauteur d'eau h et le vecteur $\mathbf{u} = (u, v)^T$ des vitesses moyennées sur la verticale. Les variables d'état des équations de Saint-Venant bidimensionnelles sous forme conservative sont la hauteur d'eau h et le débit local de l'écoulement $\mathbf{q} = h\mathbf{u}$. Ces équations s'écrivent de la manière suivante :

$$\begin{cases} \partial_t h + \operatorname{div}(\mathbf{q}) = 0 & \text{dans } \Omega \times]0, T] \\ \partial_t \mathbf{q} + \operatorname{div}\left(\frac{1}{h} \mathbf{q} \otimes \mathbf{q}\right) + \frac{1}{2}g\nabla h^2 + gh\nabla z_f + g\frac{n^2\|\mathbf{q}\|_2}{h^{7/3}}\mathbf{q} = 0 & \text{dans } \Omega \times]0, T] \\ h(0) = h_0, \quad \mathbf{q}(0) = \mathbf{q}_0 \end{cases} \quad (1)$$

où g est l'intensité de la force de pesanteur, z_f l'élévation du fond, n le coefficient de Manning de frottement sur le fond, h_0 et \mathbf{q}_0 sont les conditions initiales sur la hauteur et le débit. On utilisera par la suite la grandeur $c = \sqrt{gh}$ qui correspond à la célérité des ondes.

Dans le cas d'un écoulement sur fond plat ($\nabla z_f \equiv 0$), sans frottement ($n \equiv 0$), le système d'équations (1) est hyperbolique. Si la hauteur d'eau est partout strictement positive ($h > 0$), le système est strictement hyperbolique.

2.2 Conditions aux limites

Il faut ajouter des conditions aux limites sur le bord Γ du domaine Ω , selon le type de frontière considéré. On décompose ce dernier de la manière suivante : $\Gamma = \Gamma_{in} \cup \Gamma_{out} \cup \Gamma_c$, où Γ_{in} est la partie du bord où le flux est entrant, Γ_c correspond à une frontière fermée et Γ_{out} à une frontière ouverte où le flux est sortant. Sur Γ_{in} , on impose un débit q^{in} et une condition de Neumann homogène sur la hauteur d'eau :

$$(\mathbf{q} \cdot \mathbf{n})|_{\Gamma_{in}} = -q^{in}(t), \quad \frac{\partial h}{\partial \mathbf{n}}|_{\Gamma_{in}}(t) = 0 \quad \forall t \in]0, T]. \quad (2)$$

Sur le bord Γ_c on impose une condition de glissement parfait sur la vitesse et une condition de Neumann homogène sur la hauteur d'eau :

$$\mathbf{u} \cdot \mathbf{n}|_{\Gamma_c}(t) = 0, \quad \frac{\partial h}{\partial \mathbf{n}}|_{\Gamma_c}(t) = 0 \quad \forall t \in]0, T]. \quad (3)$$

Sur le bord Γ_{out} , on peut soit imposer un niveau d'eau z_{out} , soit imposer une condition de Neumann homogène sur les variables d'état :

$$h|_{\Gamma_{out}}(t) = z_{out}(t) - z_f|_{\Gamma_{out}}, \quad \frac{\partial(\mathbf{u}\cdot\mathbf{n}-c)}{\partial\mathbf{n}}|_{\Gamma_{out}}(t) = 0 \quad \forall t \in]0, T], \quad (4)$$

$$\text{ou} \quad \frac{\partial h}{\partial\mathbf{n}}|_{\Gamma_{out}}(t) = 0, \quad \frac{\partial\mathbf{q}}{\partial\mathbf{n}}|_{\Gamma_{out}}(t) = 0 \quad \forall t \in]0, T]. \quad (5)$$

La première condition (4) n'est physiquement valide qu'en régime fluvial, c'est-à-dire si le nombre de Froude local $Fr = \frac{\|\mathbf{u}\|}{c}$ est strictement inférieur à 1. Dans le cas contraire, la condition (5) est imposée.

3 Assimilation variationnelle de données

Les méthodes d'assimilation de données permettent de combiner de manière optimale, dans un sens à préciser, des observations d'un système physique avec un modèle numérique afin d'améliorer les prédictions de ce dernier. On peut distinguer deux grandes familles de méthodes d'assimilation de données :

- les méthodes variationnelles sont basées sur la théorie du contrôle optimal [10] et consistent à minimiser une fonctionnelle mesurant l'écart entre les observations et la prévision à l'aide d'une méthode de descente. L'optimisation peut nécessiter un nombre important d'itérations, ce qui demande un temps de calcul élevé.
- les méthodes stochastiques sont fondées sur la théorie de l'estimation statistique optimale. La plus connue est le filtre de Kalman, qui requiert le stockage et la manipulation de matrices de très grande taille.

Dans le cadre de cette étude, nous utilisons l'assimilation variationnelle de données pour l'estimation de paramètres, des conditions initiales et conditions aux limites d'un modèle d'écoulement hydraulique basé sur les équations de Saint-Venant bidimensionnelles présentées dans la partie 2.

3.1 Fondements de l'assimilation variationnelle de données

Dans cette partie, nous présentons le principe de l'assimilation variationnelle de données dans le cadre théorique du contrôle optimal introduit par J.-L. Lions dans [10].

3.1.1 Cadre fonctionnel

Soient V et H deux espaces de Hilbert. On suppose que :

$$\begin{aligned} V &\subset H, \text{ l'injection de } V \text{ dans } H \text{ est continue,} \\ V &\text{ est dense dans } H. \end{aligned}$$

H est identifié avec son espace dual. On note V' le dual de V et l'on a :

$$V \subset H \subset V'.$$

On définit la variable dite d'état $y(t) \in V$. Cette variable décrit l'état du système étudié à un instant donné t . L'espace V est l'espace d'état. On définit également l'espace d'état auquel appartient y :

$$W(0, T) = \{f \mid f \in L^2(0, T; V), \partial_t f \in L^2(0, T; V')\} .$$

3.1.2 Modèle direct

Soit \mathcal{U} un espace de Hilbert que l'on appellera espace des paramètres. Ces paramètres pouvant être variables dans le temps, on introduit l'espace de Hilbert $\mathcal{U}_T := L^2(0, T, \mathcal{U})$. La variable d'état $y(t)$ du modèle est solution du problème direct :

$$(D) \quad \left\| \begin{array}{l} \text{Étant donnés } (y_0, v) \in H \times \mathcal{U}_T, \text{ trouver } y \in W(0, T) \text{ tel que :} \\ \left\{ \begin{array}{l} \partial_t y(t) + A(v(t); y(t)) = 0 \quad \forall t \in]0, T[\\ y(0) = y_0 \end{array} \right. \end{array} \right. \quad (6)$$

où $A : \mathcal{U} \times V \rightarrow V'$ est un opérateur aux dérivées partielles, $y_0 \in H$ est la condition initiale et $v(t) \in \mathcal{U}$ représente la valeur des paramètres du modèle à l'instant t . Les variables y_0 et v sont les variables de contrôle du système. Il est clair que la variable d'état y est fonction du temps mais dépend également des variables de contrôle y_0 et v . On la note alors $y(y_0, v; t)$. Cependant, par souci de clarté, on écrira selon le contexte $y(t)$ ou $y(y_0, v)$ ou bien plus simplement encore y .

On suppose par la suite que le problème (D) est bien posé :

Hypothèse 3.1

Étant donnés $(y_0, v) \in H \times \mathcal{U}_T$ et $T > 0$, on suppose qu'il existe une unique fonction $y \in W(0, T)$ solution du problème (D) et que cette solution dépend continûment de (y_0, v) , c'est-à-dire que l'application $H \times \mathcal{U}_T \rightarrow V : (y_0, v) \mapsto y(y_0, v; t)$ est continue pour tout $t \in]0, T[$.

On fera par la suite l'hypothèse supplémentaire :

Hypothèse 3.2

On suppose que l'application $H \times \mathcal{U}_T \rightarrow V : (y_0, v) \mapsto y(y_0, v; t)$ est différentiable pour tout $t \in]0, T[$.

3.1.3 Fonction coût

L'assimilation variationnelle de données consiste à identifier les variables de contrôle qui minimisent une fonctionnelle mesurant l'écart entre la variable d'état y , solution du modèle (6) et des observations du système physique, que l'on note $y^{obs} \in L^2(0, T, \mathcal{V})$.

L'espace de Hilbert \mathcal{V} est appelé espace des observations, il n'est pas forcément inclus dans l'espace d'état V . Un exemple de fonctionnelle est donné par :

$$J(y_0, v; y) = \frac{1}{2} \int_0^T \|C y(y_0, v; t) - y^{obs}(t)\|_{\mathcal{V}}^2 dt + \frac{1}{2} \langle N(y_0 - y_b), y_0 - y_b \rangle_H$$

où $C \in \mathcal{L}(V, \mathcal{V})$ est l'opérateur d'observation qui projette la variable d'état du modèle dans l'espace d'observation, $\|\cdot\|_{\mathcal{V}}$ est une norme sur l'espace \mathcal{V} , $\langle \cdot, \cdot \rangle_H$ le produit scalaire sur H et N est un endomorphisme symétrique défini positif sur H . La fonction $y_b \in H$ est une ébauche ou première estimation de la condition initiale.

Comme J est fonction de y_0 et v , mais aussi de la variable d'état y qui dépend elle-même implicitement de ces variables de contrôle, on définit la fonction coût :

$$j(y_0, v) := J(y_0, v; y(y_0, v)) . \quad (7)$$

La minimisation de cette fonction coût par une méthode de descente nécessite le calcul de son gradient, c'est-à-dire des dérivées partielles de j par rapport aux variables de contrôle y_0 et v . Une méthode particulièrement efficace consiste à introduire un modèle adjoint, ce qui permet de formuler une expression simple des dérivées partielles de la fonction coût en fonction de la variable d'état adjointe. Pour cela, il faut d'abord introduire le modèle linéarisé.

3.1.4 Modèle linéarisé

En différentiant l'équation d'état (6) par rapport aux variables de contrôle (y_0, v) dans une direction $(\delta y_0, \delta v)$, on peut introduire le problème linéaire suivant :

$$(LT) \quad \left\{ \begin{array}{l} \text{Étant donnés } (y_0, v) \in H \times \mathcal{U}_T, y \in W(0, T) \text{ solution du problème (D) et un} \\ \text{vecteur } (\delta y_0, \delta v), \text{ trouver } dy \in W(0, T) \text{ tel que :} \\ \\ \left\{ \begin{array}{l} \partial_t dy(t) + \frac{\partial A}{\partial y}(v(t), y(t)) \cdot dy(t) \\ \quad + \frac{\partial A}{\partial v}(v(t), y(t)) \cdot \delta v(t) = 0 \quad \forall t \in]0, T[\\ dy(0) = \delta y_0 \end{array} \right. \end{array} \right. \quad (8)$$

La variable $dy = \frac{\partial y}{\partial y_0} \cdot \delta y_0 + \frac{\partial y}{\partial v} \cdot \delta v$ correspond à la variation de l'état y quand le vecteur de contrôle (y_0, v) est perturbé dans la direction $(\delta y_0, \delta v)$. De la même façon, on différentie la fonction coût par rapport à y_0 et v dans les direction δy_0 et δv pour obtenir les dérivées directionnelles suivantes :

$$\frac{\partial j}{\partial y_0} \cdot \delta y_0 = \int_0^T \langle C^* \Lambda(C y - y^{obs}), \frac{\partial y}{\partial y_0} \cdot \delta y_0 \rangle_{V' \times V} dt + \langle N(y_0 - y_b), \delta y_0 \rangle_H \quad (9)$$

$$\frac{\partial j}{\partial v} \cdot \delta v = \int_0^T \langle C^* \Lambda(C y - y^{obs}), \frac{\partial y}{\partial v} \cdot \delta v \rangle_{V' \times V} dt , \quad (10)$$

où $\langle \cdot, \cdot \rangle_{V' \times V}$ désigne le produit de dualité, Λ est l'isomorphisme canonique de \mathcal{V} dans \mathcal{V}' , espace dual de \mathcal{V} , et $C^* \in \mathcal{L}(\mathcal{V}', V')$ est l'opérateur adjoint de C défini de la manière suivante :

$$\forall \eta \in \mathcal{V}', \quad \forall \xi \in V \quad \langle C^* \eta, \xi \rangle_{V' \times V} = \langle \eta, C\xi \rangle_{\mathcal{V}' \times \mathcal{V}}.$$

En notant $dj = \frac{\partial j}{\partial y_0} \cdot \delta y_0 + \frac{\partial j}{\partial v} \cdot \delta v$ la dérivée directionnelle de la fonction coût, on peut écrire :

$$dj = \int_0^T \langle C^* \Lambda (C y(t) - y^{obs}(t)), dy(t) \rangle_{V' \times V} dt + \langle N(y_0 - y_b), \delta y_0 \rangle_H. \quad (11)$$

3.1.5 Modèle adjoint

Pour obtenir une expression simple des dérivées partielles de la fonction coût, il suffit d'exhiber la linéarité de la dérivée directionnelle dj par rapport au vecteur de perturbation $(\delta y_0, \delta v)$. Pour cela, on introduit $\tilde{y} \in W(0, T)$ la variable adjointe, on prend le produit de dualité de l'équation (8) avec $\tilde{y}(t)$ puis on intègre entre 0 et T . Après intégration par parties, on obtient :

$$\int_0^T \langle \partial_t \tilde{y}(t) - [\frac{\partial A}{\partial y}]^* \tilde{y}(t), dy(t) \rangle_{V' \times V} dt = \langle \tilde{y}(T), dy(T) \rangle_V - \langle \tilde{y}(0), \delta y_0 \rangle_V + \int_0^T \langle [\frac{\partial A}{\partial v}]^* \tilde{y}(t), \delta v(t) \rangle_{\mathcal{U}' \times \mathcal{U}} dt \quad (12)$$

Introduisons alors maintenant le problème adjoint :

$$(A) \quad \left\{ \begin{array}{l} \text{Étant donnés } (y_0, v) \in H \times \mathcal{U}_T, T > 0 \text{ et } y \in W(0, T) \text{ solution du problème (D),} \\ \text{trouver } \tilde{y} \in W(0, T) \text{ tel que :} \\ \left\{ \begin{array}{l} \partial_t \tilde{y}(t) - [\frac{\partial A}{\partial y}]^* \tilde{y}(t) = C^* \Lambda (C y(t) - y^{obs}(t)) \quad \forall t \in]0, T[\\ \tilde{y}(T) = 0 \end{array} \right. \end{array} \right. \quad (13)$$

On fait l'hypothèse que le problème adjoint est bien posé :

Hypothèse 3.3

Étant donnés $(y_0, v) \in H \times \mathcal{U}_T, T > 0$ et $y \in W(0, T)$ solution du problème (D) on suppose qu'il existe une unique fonction $\tilde{y} \in W(0, T)$ solution du problème (A) et que cette solution dépend continûment de (y_0, v) , c'est-à-dire que l'application $H \times \mathcal{U}_T \rightarrow V : (y_0, v) \mapsto \tilde{y}(y_0, v; t)$ est continue pour tout $t \in]0, T[$.

En combinant les équations (11), (12) et (13), on obtient :

$$dj = \langle N(y_0 - y_b) - \tilde{y}(0), \delta y_0 \rangle_H + \int_0^T \langle [\frac{\partial A}{\partial v}]^* \tilde{y}(t), \delta v(t) \rangle_{\mathcal{U}' \times \mathcal{U}} dt$$

Ainsi, si \tilde{y} est solution du problème adjoint (A), on peut caractériser formellement les dérivées partielles de la fonction coût par :

$$\frac{\partial j}{\partial y_0}(y_0, v) = -\tilde{y}(0) + N(y_0 - y_b) \quad (14)$$

$$\frac{\partial j}{\partial v}(y_0, v) = \left[\frac{\partial A}{\partial v} \right]^* \tilde{y} \quad (15)$$

3.1.6 Système d'optimalité

Le principe de l'assimilation variationnelle de données est de trouver la valeur des variables de contrôle y_0 et v qui minimisent la fonction coût $j(y_0, v)$:

$$(OPT) \quad \left\{ \begin{array}{l} \text{Étant donné } T > 0, \text{ trouver } (y_0^*, v^*) \in H \times \mathcal{U}_T \text{ tel que :} \\ j(y_0^*, v^*) = \min_{y_0, v} j(y_0, v) \\ \text{sous la contrainte que } y \in W(0, T) \text{ soit solution du problème (D).} \end{array} \right. \quad (16)$$

Une condition nécessaire d'optimalité au premier ordre s'écrit :

$$\begin{cases} \frac{\partial j}{\partial y_0}(y_0, v) = 0 \\ \frac{\partial j}{\partial v}(y_0, v) = 0 \end{cases}$$

Ainsi, pour satisfaire cette condition d'optimalité, sous la contrainte que la variable y soit solution du problème direct (D), le système d'optimalité suivant doit être vérifié :

$$\left\{ \begin{array}{ll} \partial_t y(t) + A(v(t); y(t)) = 0 & \forall t \in]0, T[\\ \partial_t \tilde{y}(t) - \left[\frac{\partial A}{\partial y} \right]^* \tilde{y}(t) = C^* \Lambda (C y(t) - y^{obs}(t)) & \forall t \in]0, T[\\ y(0) = y_0 & \\ \tilde{y}(0) = N(y_0 - y_b) & \\ \tilde{y}(T) = 0 & \\ \left[\frac{\partial A}{\partial v} \right]^* \tilde{y}(t) = 0 & \forall t \in]0, T[\end{array} \right. \quad (17)$$

3.2 Adjoint du modèle Shallow Water

Nous allons écrire les équations adjointes du modèle Shallow Water (1) associé aux conditions aux limites décrites dans la partie 2. Pour cela, on considère une fonction coût mesurant l'écart entre des observations de hauteur et débit d'une part, et l'état du système d'autre part :

$$j(h_0, \mathbf{q}_0, n, z_f, q^{in}) = \int_0^T \left(\|C_h h(t) - h^{obs}(t)\|_\Omega^2 + \|C_q \mathbf{q}(t) - \mathbf{q}^{obs}(t)\|_\Omega^2 \right) dt$$

Les variables de contrôle sont les conditions initiales h_0 et \mathbf{q}_0 de l'écoulement, le coefficient de Manning n , la cote du fond z_f et la valeur du débit en amont q^{in} . Soient h et \mathbf{q} les variables d'état solution du système d'équations (1). Le modèle adjoint est obtenu par la procédure décrite dans la partie 3.1.5 :

$$\left\{ \begin{array}{l} \partial_t \tilde{h}(t) - \mathbf{u} \cdot (\mathbf{u} \cdot \nabla) \tilde{\mathbf{q}} + gh \operatorname{div}(\tilde{\mathbf{q}}) \\ \quad - g \tilde{\mathbf{q}} \cdot \nabla z_b + \frac{7}{3} g \frac{n^2 \|\mathbf{u}\|}{h^{4/3}} \mathbf{u} \cdot \tilde{\mathbf{q}} = C_h^T (C_h h(t) - h^{obs}(t)) \quad \forall t \in]0, T[\\ \partial_t \tilde{\mathbf{q}}(t) + (\mathbf{u} \cdot \nabla) \tilde{\mathbf{q}} + (\nabla \tilde{\mathbf{q}}) \mathbf{u} + \nabla \tilde{h} \\ \quad - g \frac{n^2 \|\mathbf{u}\|}{h^{4/3}} \tilde{\mathbf{q}} - g \frac{n^2}{h^{4/3} \|\mathbf{u}\|} (\mathbf{u} \otimes \mathbf{u}) \tilde{\mathbf{q}} = C_{\mathbf{q}}^T (C_{\mathbf{q}} \mathbf{q}(t) - \mathbf{q}^{obs}(t)) \quad \forall t \in]0, T[\\ \tilde{h}(T) = 0, \quad \tilde{\mathbf{q}}(T) = 0 \\ \tilde{\mathbf{q}}|_{\partial\Omega}(t) = 0, \quad \tilde{h}|_{\Gamma_{out}}(t) = 0, \quad \frac{\partial \tilde{h}}{\partial \mathbf{n}}|_{\Gamma_{in} \cup \Gamma_c}(t) = 0 \quad \forall t \in]0, T[\end{array} \right. \quad (18)$$

Les conditions aux limites du système adjoint sont choisies de manière à pouvoir exprimer les dérivées partielles de j en fonction des variables adjointes \tilde{h} et $\tilde{\mathbf{q}}$. Si ces dernières sont solution de (18), alors les dérivées partielles s'écrivent alors de la manière suivante :

$$\begin{aligned} \frac{\partial j}{\partial h_0} &= -\tilde{h}(0), & \frac{\partial j}{\partial z_f} &= -g \int_0^T h(t) \operatorname{div}(\tilde{\mathbf{q}}(t)) dt, \\ \frac{\partial j}{\partial \mathbf{q}_0} &= -\tilde{\mathbf{q}}(0), & \frac{\partial j}{\partial n} &= 2gn \int_0^T \|\mathbf{q}(t)\| \mathbf{q}(t) \cdot \tilde{\mathbf{q}}(t) h^{-\frac{7}{3}}(t) dt \\ \frac{\partial j}{\partial q^{in}} &= -\tilde{h}|_{\Gamma_{in}} \end{aligned}$$

4 Mise en œuvre numérique

Dans cette partie, nous allons étudier la mise en œuvre numérique et informatique du modèle Shallow Water et de son adjoint décrits dans les parties 2 et 3.2.

4.1 Discrétisation du modèle direct

Le modèle Shallow Water direct est discrétisé par la méthode des volumes finis, sur des maillages comportant des volumes de contrôle quadrangulaires. On présente dans cette partie la discrétisation du modèle direct par un schéma de Godunov décentré [6]. Une description détaillée est donnée dans [13].

4.1.1 Description volumes finis

D'une manière générale, les équations hyperboliques bidimensionnelles sous forme conservative avec terme source, comme le sont les équations (1), peuvent s'écrire de la manière

suivante :

$$\partial_t U + \operatorname{div} F(U) = S(U) ,$$

où U est le vecteur des variables d'état conservatives, F le vecteur de flux et S le terme source. Dans le cas des équations Shallow Water, on a $U = (h, \mathbf{q})^T$ et les opérateurs F et S sont définis par :

$$F(U) = \begin{pmatrix} G(U) \\ H(U) \end{pmatrix} \quad \text{et} \quad S(U) = \begin{pmatrix} 0 \\ -gh\nabla_{z_f} - g\frac{n^2\|\mathbf{q}\|_2}{h^{7/3}}\mathbf{q} \end{pmatrix} ,$$

avec

$$G(U) = \begin{pmatrix} q_x \\ \frac{1}{h}q_x^2 + \frac{1}{2}gh^2 \\ \frac{1}{h}q_xq_y \end{pmatrix} \quad \text{et} \quad H(U) = \begin{pmatrix} q_y \\ \frac{1}{h}q_xq_y \\ \frac{1}{h}q_y^2 + \frac{1}{2}gh^2 \end{pmatrix} .$$

Le domaine de calcul Ω est discrétisé par un maillage composé de volumes de contrôle quadrangulaires. Les équations sont intégrées sur chaque maille K_i , ce qui donne après intégration par parties :

$$\int_{K_i} \partial_t U \, d\Omega + \int_{\partial K_i} F(U) \cdot \mathbf{n}_i \, ds = \int_{K_i} S(U) \, d\Omega , \quad (19)$$

où $\mathbf{n}_i = [\cos \theta_i, \sin \theta_i]$ est le vecteur normal unitaire extérieur à K_i , formant un angle θ_i avec le bord de la maille (voir la figure 4.1). En décomposant l'intégrale sur les faces des mailles, le terme de flux peut s'écrire de la manière suivante :

$$\int_{\partial K_i} F(U) \cdot \mathbf{n}_i \, ds = \sum_{j=1}^4 \int_{E_{ij}} F(U) \cdot \mathbf{n}_{ij} \, ds = \sum_{j=1}^4 \int_{E_{ij}} T_{ij}^{-1} G(T_{ij} U) \, ds , \quad (20)$$

où E_{ij} est la $j^{\text{ème}}$ face du volume K_i , le vecteur \mathbf{n}_{ij} est le vecteur normal unitaire correspondant et T_{ij} la matrice de rotation :

$$T_{ij} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos \theta_{ij} & \sin \theta_{ij} \\ 0 & -\sin \theta_{ij} & \cos \theta_{ij} \end{bmatrix} .$$

Cette propriété d'invariance par rotation des équations Shallow Water permet de réduire le calcul du terme de flux à une somme de problèmes de Riemann que l'on peut résoudre numériquement par un solveur de Riemann. En faisant l'approximation que la variable d'état U est constante sur chaque cellule du maillage, on définit sa moyenne sur chaque cellule K_i par :

$$U_i = \frac{1}{|K_i|} \int_{K_i} U \, d\Omega .$$

Un solveur de Riemann approché calcule une approximation du flux (20) à travers la face E_{ij} à partir de U_i et de $U_{i(j)}$, valeur moyenne de U sur le volume adjacent à K_i par sa face

j , que l'on nomme $K_{i(j)}$. C'est un problème local, et pour plus de clarté, on note $R \equiv K_{i(j)}$, $U_L = T_{ij}U_i$ et $U_R = T_{ij}U_{i(j)}$. Ainsi, le solveur de Riemann approché calcule :

$$\tilde{G}(U_L, U_R) \approx \int_{E_{ij}} G(T_{ij}U) ds .$$

La relation intégrale (19) se réduit alors à :

$$\frac{d}{dt}U_i + \frac{1}{|K_i|} \sum_{j=1}^4 T_{ij}^{-1} \tilde{G}(U_L, U_R) = S_i \quad (21)$$

où $|K_i|$ est la surface du volume K_i et S_i une approximation de la moyenne de $S(U)$ sur K_i . Pour calculer le flux discret $\tilde{G}(U_L, U_R)$, on utilise le solveur de Riemann approché HLLC [14], extension du schéma HLL introduit par Harten, Lax et van Leer [8]. Ce schéma est d'ordre un et permet de traiter correctement la transition entre le régime sous-critique et le régime super-critique, contrairement à la plupart des autres schémas du premier ordre [15]. La description complète du schéma HLLC qui a servi à son implémentation dans Dassflow se trouve dans [11].

Finalement, l'équation (21) est discrétisée en temps par un schéma d'Euler explicite pour obtenir :

$$U_i^{n+1} = U_i^n - \frac{\Delta t}{|K_i|} \sum_{j=1}^4 T_{ij}^{-1} \tilde{G}(U_L^n, U_R^n) + \Delta t S_i^n , \quad (22)$$

où $U_i^n = (h_i^n, \mathbf{q}_i^n)^T$ est l'approximation au temps t^n de U_i . Ce choix d'un schéma explicite permet d'éviter la coûteuse résolution d'un système non-linéaire à chaque itération en temps mais impose une condition de stabilité sur le choix du pas de temps. Une analyse de stabilité

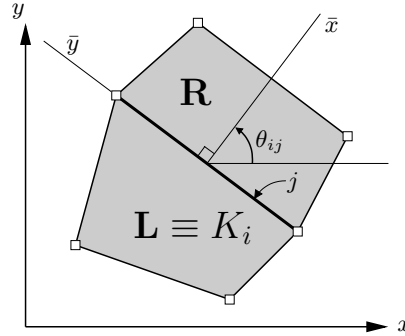


FIG. 4.1 – Deux volumes de contrôle adjacents

sur les équations linéarisées fournit la restriction CFL suivante [13] :

$$\text{CFL} = \Delta t \frac{\max(\|\mathbf{u}\| + c)}{\min(d_{L,R})} \leq 1, \quad (23)$$

où $d_{L,R}$ est la distance entre le centre du volume L et le centre de sa face qui le sépare du volume R.

4.1.2 Conditions aux limites

Les conditions aux limites sont prises en compte en calculant le flux à travers les faces du bord Γ entre les cellules intérieures et des cellules voisines “fantômes” extérieures au domaine. Si le volume K_i est adjacent à Γ par sa face j , alors on définit un volume “fantôme” $K_{i(j)}$ de même aire, avec une valeur particulière de la variable d’état U_R telle que le flux numérique $\tilde{G}(U_L^n, U_R^n)$ calculé corresponde à la condition aux limites désirée :

- sur Γ_{in} : on prescrit un débit scalaire $q^{in}(t)$ exprimé en $m^3 s^{-1}$. Cette valeur scalaire est distribuée sur toute la frontière à chaque pas de temps sous la forme un débit local q_L^n par unité de longueur exprimé en $m^2 s^{-1}$, choisi de manière à ce que la vitesse normale sur Γ_{in} soit constante sur la section mouillée. Cela donne : $q_L^n = \frac{1}{\mathcal{A}^n} h_L^n q^{in}(t^n)$, où \mathcal{A}^n est la surface de la section mouillée à l’instant t^n . Numériquement, on calcule le flux à travers chaque face de Γ_{in} en utilisant le solveur HLLC avec $U_R^n = (h_L^n, -q_L^n/h_L^n, v_L^n)^T$.
- sur Γ_c : on prescrit la condition de Neumann homogène sur la hauteur en imposant $h_R = h_L$ et la condition de glissement parfait sur la vitesse en imposant $u_R = -u_L$ et $v_R = v_L$. Ainsi, on calcule le flux à travers chaque face de Γ_c en utilisant le solveur HLLC avec $U_R^n = (h_L^n, -u_L^n, v_L^n)^T$.
- sur Γ_{out} : pour prescrire les conditions de Neumann homogènes (5) sur les variables d’état, on calcule le flux à travers chaque face de Γ_{out} en utilisant le solveur HLLC avec $U_R^n = U_L^n$. Pour prescrire les conditions (4), alors on impose la hauteur d’eau dans le volume fantôme adjacent $h_R = z_{out} - z_{f,R}$ et la vitesse de manière à ce que $u_L^n - \sqrt{gh_L^n} = u_R^n - \sqrt{gh_R^n}$. Ainsi, on calcule le flux à travers chaque face de Γ_{out} en utilisant le solveur HLLC avec $U_R^n = (z_{out} - z_{f,R}, u_L^n - \sqrt{gh_L^n} + \sqrt{gh_R^n}, v_L^n)^T$.

4.1.3 Mise en œuvre informatique

La formulation volumes finis du modèle direct a été codée en Fortran 90. Ce choix a été fait pour pouvoir utiliser l’outil de différentiation automatique TAPENADE [9] dans la mise en œuvre du modèle adjoint.

4.2 Mise en œuvre du modèle adjoint

Pour obtenir un code adjoint, il y a plusieurs possibilités :

- Discrétiser les équations continues adjointes et coder le schéma numérique correspondant ;
- Écrire l'adjoint du schéma numérique direct et le coder ;
- Écrire directement l'adjoint du code informatique direct.

Nous avons choisi la troisième méthode, qui permet, à l'aide d'un outil de différentiation automatique, d'obtenir rapidement un code adjoint qui est consistant avec le code direct. En effet, un tel code adjoint permet de calculer le gradient exact de l'approximation numérique de la fonction coût j et non une approximation numérique du gradient continu.

4.2.1 Différentiation automatique

TAPENADE [9] est un logiciel de différentiation automatique de codes écrits en Fortran. Il est développé à l'INRIA dans le cadre du projet TROPICS [12]. Ce logiciel fonctionne selon le principe de la transformation de sources : à partir du code source d'un programme direct écrit en Fortran, TAPENADE construit automatiquement le code source du programme linéaire tangent et/ou du programme adjoint correspondant. Pour une description détaillée du fonctionnement de la différentiation automatique, on peut se reporter à [7].

4.2.2 Utilisation du code adjoint

On peut représenter un code de calcul comme un opérateur \mathcal{M} qui agit sur un vecteur X de l'espace des contrôles \mathcal{X} contenant l'ensemble des variables de contrôle discrètes. On définit alors le vecteur $Y = \mathcal{M}(X)$ appartenant à l'espace \mathcal{Y} des résultats. Dans le cas du programme Dassflow, on a :

$$X = (y_0, n, z_f, q^{in})^T \quad \text{et} \quad Y = (y(T), j)^T$$

où $y_0 = (h_0, \mathbf{q}_0)^T$ et $y(T) = (h(T), \mathbf{q}(T))^T$ sont les valeurs du vecteur d'état aux pas de temps initial et final. Comme on le voit, le calcul de la fonction coût j est inclus dans le code direct et sa valeur fait partie des variables de sortie du programme. Le code linéaire tangent est alors représenté par l'opérateur $\frac{\partial \mathcal{M}}{\partial X}(X)$ qui agit sur une perturbation $dX \in \mathcal{X}$ du vecteur de contrôle, et on définit la variation $dY \in \mathcal{Y}$ sur le résultat :

$$dY = \frac{\partial \mathcal{M}}{\partial X}(X) \cdot dX .$$

Le code adjoint est à son tour représenté par l'opérateur adjoint $\left(\frac{\partial \mathcal{M}}{\partial X}(X)\right)^*$ qui agit sur une variable adjointe $dY^* \in \mathcal{Y}'$ et on définit la variable adjointe $dX^* \in \mathcal{X}'$ par :

$$dX^* = \left(\frac{\partial \mathcal{M}}{\partial X}(X)\right)^* \cdot dY^* .$$

On a par définition de l'adjoint :

$$\left\langle \left(\frac{\partial \mathcal{M}}{\partial X} \right)^* \cdot dY^*, dX \right\rangle_{\mathcal{X}' \times \mathcal{X}} = \left\langle dY^*, \left(\frac{\partial \mathcal{M}}{\partial X} \right) \cdot dX \right\rangle_{\mathcal{Y}' \times \mathcal{Y}}, \quad (24)$$

c'est-à-dire :

$$\langle dX^*, dX \rangle_{\mathcal{X}' \times \mathcal{X}} = \langle dY^*, dY \rangle_{\mathcal{Y}' \times \mathcal{Y}}. \quad (25)$$

Si on écrit le vecteur de perturbation dX sous la forme $(\delta y_0, \delta n, \delta z_f, \delta q^{in})^T$, alors en choisissant $dY^* = (0, 1)^T$, l'équation (25) se réécrit :

$$dj = \left\langle \begin{pmatrix} \delta y_0^* \\ \delta n^* \\ \delta z_f^* \\ \delta q^{in*} \end{pmatrix}, \begin{pmatrix} \delta y_0 \\ \delta n \\ \delta z_f \\ \delta q^{in} \end{pmatrix} \right\rangle_{\mathcal{X}' \times \mathcal{X}}$$

Or, on a par définition :

$$dj = \frac{\partial j}{\partial y_0}(X) \cdot \delta y_0 + \frac{\partial j}{\partial n}(X) \cdot \delta n + \frac{\partial j}{\partial z_f}(X) \cdot \delta z_f + \frac{\partial j}{\partial q^{in}}(X) \cdot \delta q^{in}$$

Par conséquent, la variable adjointe dX^* correspond au gradient de la fonction coût j et l'on a :

$$\begin{aligned} \frac{\partial j}{\partial y_0}(X) &= y_0^* & \frac{\partial j}{\partial n}(X) &= n^* \\ \frac{\partial j}{\partial z_f}(X) &= z_f^* & \frac{\partial j}{\partial q^{in}}(X) &= q^{in*} \end{aligned}$$

D'un point de vue informatique, si le code direct s'appelle `M`, le logiciel TAPENADE produit un code linéaire tangent `M_d` et un adjoint `M_b`. Si `y0`, `n`, `zf` et `qin` sont les variables de contrôle, c'est-à-dire les paramètres d'entrée du programme `M`, alors les variables adjointes correspondantes seront appelées `y0b`, `nb`, `zfb` et `qinb` dans le programme `M_b`.

4.2.3 Validation du code adjoint

Une fois le code adjoint obtenu, il s'agit de vérifier s'il est effectivement l'adjoint du code linéaire tangent et s'il calcule correctement le gradient de la fonction coût. Pour cela, on procède souvent à deux tests de validation.

Test du produit scalaire

L'objectif du test du produit scalaire est de vérifier si le code adjoint est bien l'adjoint du code linéaire tangent. Il consiste à vérifier que la relation (24) est bien établie quand on utilise les codes informatiques. En dimension finie, on identifie les espaces duaux \mathcal{X}' et \mathcal{Y}' à \mathcal{X} et \mathcal{Y} respectivement :

- Étant donné $dX \in \mathcal{X}$, on calcule avec le programme linéaire tangent :

$$dY = \left(\frac{\partial \mathcal{M}}{\partial X} \right) \cdot dX ;$$

- Étant donné $dY^* \in \mathcal{Y}$, on calcule avec le programme adjoint :

$$dX^* = \left(\frac{\partial \mathcal{M}}{\partial X} \right)^* \cdot dY^* ;$$

- On calcule les produits scalaires :
 - $ps_1 = \langle dY^*, dY \rangle_Y$
 - $ps_2 = \langle dX^*, dX \rangle_X$
- On vérifie que $ps_1 = ps_2$.

Le vecteur dY^* peut être quelconque. En pratique, on choisit souvent $dY^* = dY$.

Test du gradient

L'objectif du test du gradient est de vérifier si les variables adjointes composantes de dX^* calculées par le code adjoint correspondent bien aux dérivées partielles de la fonction coût. Soient $y_0 \in H$, $v \in \mathcal{U}_T$. Le développement de Taylor de la fonction coût j au point (y_0, v) pour une perturbation $\alpha \delta y_0 \in H$ (où $\alpha \in \mathbb{R}^+$) s'écrit :

$$j(y_0 + \alpha \delta y_0, v) = j(y_0, v) + \alpha \frac{\partial j}{\partial y_0}(y_0, v) \cdot \delta y_0 + o(\alpha \|\delta y_0\|) . \tag{26}$$

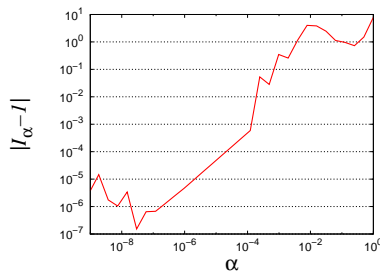
On définit I_α par :

$$I_\alpha = \frac{j(y_0 + \alpha \delta y_0, v) - j(y_0, v)}{\alpha \frac{\partial j}{\partial y_0}(y_0, v) \cdot \delta y_0} . \tag{27}$$

D'après la relation (26), on a $\lim_{\alpha \rightarrow 0} I_\alpha = 1$. Le test de Taylor consiste à vérifier cette propriété :

- Étant donnés $y_0 \in H$ et $v \in \mathcal{U}$, on calcule avec le code adjoint la dérivée partielle de j par rapport à y_0 au point (y_0, v) ; c'est la variable adjointe y_0^* .
- Avec le code direct, on calcule $j(y_0, v)$ la valeur de la fonction coût au point (y_0, v) .
- Pour $n = 0, \dots, N$:
 - On calcule $\alpha = 2^{-n}$;
 - Avec le code direct, on calcule $j(y_0 + \alpha \delta y_0)$;
 - On calcule I_α ;
- On vérifie que $\lim_{\alpha \rightarrow 0} I_\alpha = 1$.

De la même manière, on peut vérifier que le code adjoint calcule correctement les dérivées partielles de j par rapport à chaque variable de contrôle.



(a) Test du gradient

```
#####
## TEST DU PRODUIT SCALAIRE ##
#####

Appel du code linéaire tangent...
Appel du code adjoint...
<Xd, Xb> = -682.277083033688
<Yd, Yb> = -682.277082428555
relative error : -8.869324203484993E-010
```

(b) Test du produit scalaire

FIG. 4.2 – Validation du code adjoint

Sur le graphique de la figure 4.2 (a) est représenté le résultat du test du gradient pour une configuration particulière de simulation. La quantité $|I_\alpha - 1|$ est représentée en fonction de α en échelle logarithmique. On remarque que pour des valeurs de α suffisamment grandes ($\alpha > 10^{-7}$), il y a convergence de I_α vers 1 ; pour des valeurs de α trop petites, le test commence à se dégrader. Cela peut être expliqué par des erreurs de troncature dans l'approximation différences finies du gradient [4]. Sur la figure 4.2 (b), on peut voir un exemple de résultat du test du produit scalaire.

5 Résultats numériques

Dans cette partie, nous présentons des résultats numériques d'assimilation de données à travers des expériences jumelles. Lors d'expériences jumelles, les observations sont créées par le modèle à partir de la variable d'état correspondant à un jeu de paramètre de référence. L'objectif est d'identifier la valeur du jeu de paramètre qui a servi à la création des observations en partant d'une hypothèse *a priori* sur sa valeur.

5.1 Exemple 1 : Identification de la topographie

Le premier exemple d'expériences jumelles concerne l'identification de la topographie, c'est-à-dire du paramètre z_f dans les équations (1). On considère un domaine rectangulaire de dimensions 30×4 mètres, avec une topographie définie par :

$$\begin{aligned} z_f(x, y) &= 0.9 \exp\left(-\frac{1}{4}(x-10)^2\right) \exp\left(-(y-1)^2\right) \\ &+ 0.7 \exp\left(-\frac{1}{8}(x-20)^2\right) \exp\left(-2(y-3)^2\right) \end{aligned} \quad (28)$$

Le domaine est discrétisé par un maillage rectangulaire régulier de 90×20 volumes de contrôle. La limite amont se situe en $x = 0$, celle en aval en $x = 30$, les bords $y = 0$ et $y = 4$ sont des parois imperméables. Ainsi, l'eau circule dans le sens des x croissants. Le coefficient de Manning est fixé à $n = 0.025$, la hauteur d'eau en aval à 1.4 m et un débit constant $q^{in} = 8$ m³/s conduit l'écoulement à un état stationnaire au bout d'environ 80s de simulation. La figure 5.3 (a) montre la topographie $z_f(x, y)$ et l'élévation de la surface libre dans cette configuration.

Ce régime permanent sert de référence pour la création des observations de hauteur d'eau h^{obs} et de vitesse \mathbf{u}^{obs} : sur une période de $T = 20$ s de simulation, avec un pas de temps Δt de 0.02 s, on enregistre en chaque point de maillage la valeur de h et de \mathbf{u} à chaque pas de temps.

Le but de l'expérience est de retrouver la valeur de la topographie z_f définie en (28) en partant de l'hypothèse *a priori* d'un fond plat $z_f \equiv 0$. Pour cela, le processus d'assimilation de données décrit dans la partie 3 est mis en œuvre avec la fonction coût suivante à minimiser :

$$j_1(z_f) = \frac{1}{2} \int_0^T \left(\|h(t) - h^{obs}(t)\|_\Omega^2 + \|\mathbf{u}(t) - \mathbf{u}^{obs}(t)\|_\Omega^2 \right) dt, \quad (29)$$

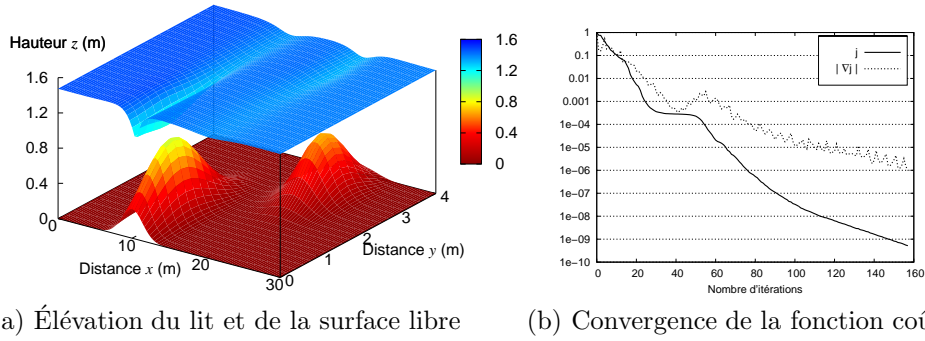


FIG. 5.3 – Identification de la topographie

où $\|\cdot\|_{\Omega}$ désigne la norme L^2 sur Ω . Le gradient de cette fonction coût utilisé par la routine de minimisation M1QN3 [5] est calculé avec le code adjoint. Sur la figure 5.3 (b), la valeur de la fonction coût et la norme de son gradient, toutes deux normalisées par leur valeur initiale, sont représentées en fonction des itérations du processus de minimisation. On peut voir que la convergence est atteinte, et que la forme de la topographie originale a été retrouvée.

5.2 Exemple 2 : Identification du débit amont

Le deuxième exemple concerne l'identification du débit amont q^{in} , qui est une condition limite des équations Shallow Water (1). On considère un domaine rectangulaire de dimensions 100×8 mètres, avec une topographie définie par :

$$z_f(x, y) = \frac{1}{2} - \frac{x}{200} + \frac{1}{10} \sin\left(\frac{x\pi}{20}\right) \cos\left(\frac{(y-2)\pi}{6}\right) \quad (30)$$

Le domaine est discrétisé par un maillage rectangulaire régulier de 100×10 volumes de contrôle. La limite amont correspond au bord $x = 0$. Le coefficient de Manning est fixé à $n = 0.025$, et un débit constant $q^{in} = 5 \text{ m}^3/\text{s}$ conduit l'écoulement à un régime permanent, qui sert de condition initiale à l'écoulement de référence qui servira à créer les observations.

On simule une crue en augmentant fortement le débit en amont sur une courte période de temps. À partir de l'état stationnaire mentionné ci-dessus, on effectue une simulation de 80 s, avec un pas de temps $\Delta t = \frac{5}{100} \text{ s}$ et un débit de référence défini par :

$$q_{ref}^{in}(t) = \begin{cases} 5 & \text{si } t < 10 \text{ s} \\ 5 + 2(t-10) \exp\left(-\frac{(t-15)^2}{100}\right) & \text{si } t \geq 10 \text{ s} . \end{cases} \quad (31)$$

L'hydrogramme correspondant est représenté sur la figure 5.4 (a). On peut voir la propagation de la vague sur la figure 5.4 (b), où une coupe de la surface libre dans le plan $y = 1.6 \text{ m}$ est représentée à plusieurs instants de la simulation. L'eau circule dans le sens des x croissants. Cet écoulement de référence dont on note le vecteur d'état $(h_{ref}, \mathbf{u}_{ref})^T$ est utilisé pour

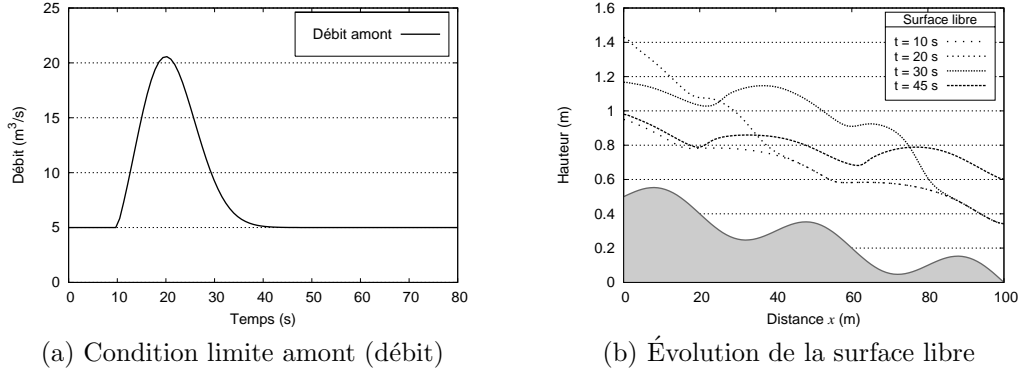


FIG. 5.4 – Identification du débit amont : configuration

la création des observations : la hauteur d'eau est mesurée à chaque pas de temps en un unique point d'observation situé en (x_m, y_m) . Cette mesure est utilisée comme une observation pendant l'événement de crue simulé. On la note $h_m^{obs}(t) = h_{ref}(x_m, y_m; t)$. L'expérience consiste à identifier le débit qui a servi à créer cette observation en partant de l'hypothèse *a priori* que le débit est constant à $5 \text{ m}^3/\text{s}$. Pour cela, on introduit la fonction coût suivante :

$$j_{2,m}(q_m^n) = \frac{1}{2} \int_0^T |h(x_m, y_m; t) - h_m^{obs}(t)|^2 dt, \quad (32)$$

où $h(x_m, y_m; t)$ est la hauteur d'eau au point (x_m, y_m) au temps t . On note q_m^n le débit qui minimise cette fonction coût. Pour évaluer la qualité du résultat, on introduit la fonction de coût globale qui mesure l'écart entre l'état de l'écoulement généré par le débit identifié q_m^n et l'état de l'écoulement de référence :

$$j_T(q_m^n) = \int_0^T \|h(t) - h_{ref}(t)\|_{\Omega}^2 + \|\mathbf{u}(t) - \mathbf{u}_{ref}(t)\|_{\Omega}^2 dt \quad (33)$$

Ainsi, on peut définir l'erreur globale absolue sur l'écoulement : $\varepsilon_m = \sqrt{j_T(q_m^n)}$ commise si l'on considère le débit identifié à la place du débit de référence. Dans le tableau 5.1, cette erreur globale est reportée pour différentes valeurs de l'abscisse du point d'observation. On

x_m	10 m	20 m	40 m	60 m	80 m	100 m
ε_m	0.21	0.51	1.57	3.47	6.99	12.22

TAB. 5.1 – Erreur globale absolue en fonction de l'abscisse du point d'observation.

observe que la qualité de l'identification du débit se dégrade lorsque le point d'observation est éloigné de l'entrée du domaine.

Références

- [1] F. Bouttier and P. Courtier. Data assimilation concepts and methods. Meteorological Training Course Lecture Series, ECMWF, 1999.
- [2] H. Brezis. *Analyse fonctionnelle : Théorie et applications*. Dunod, 1999.
- [3] F.-X. Le Dimet and O. Talagrand. Variational algorithms for analysis and assimilation of meteorological observations : theoretical aspects. *Tellus*, 38A :97–110, 1986.
- [4] D. Elizondo, C. Faure, and B. Cappelaere. Automatic- versus manual- differentiation for non-linear inverse modeling. Technical Report RR-3981, INRIA, 2000.
- [5] J.-C. Gilbert and C. Lemaréchal. Some numerical experiments with variable storage quasi-Newton algorithms. *Mathematical programming*, 45 :407–435, 1989.
- [6] S. K. Godunov. Finite difference methods for the computation of discontinuous solutions of the equations of fluid dynamics. *Mat. Sb.*, 47 :271–306, 1959.
- [7] A. Griewank. *Evaluating Derivatives : Principles and Techniques of Algorithmic Differentiation*. Number 19 in Frontiers in Appl. Math. SIAM, Philadelphia, PA, 2000.
- [8] A. Harten, P. Lax, and A. van Leer. On upstream differencing and Godunov-type schemes for hyperbolic conservation laws. *SIAM Review*, 25 :35–61, 1983.
- [9] L. Hascoët and V. Pascual. TAPENADE 2.1 user's guide. Technical Report RT-300, INRIA, 2004.
- [10] J.L. Lions. *Optimal Control of Systems Governed by Partial Differential Equations*. Springer-Verlag, 1971.
- [11] Y. Loukili and Soulaïmani A. Numerical tracking of shallow water waves by the unstructured finite volume WAF approximation. *Journal of Computational Methods in Sciences and Engineering*, to appear, 2005.
- [12] INRIA, projet TROPICS. TAPENADE 2.1.
<<http://www-sop.inria.fr/tropics/>>.
- [13] E.F. Toro. *Shock-Capturing Methods for Free-Surface Shallow Flows*. Wiley and Sons Ltd, 2001.
- [14] E.F. Toro, M. Spruce, and W. Speares. Restauration of the contact surface in the HLL-Riemann solver. *Shock Waves*, 4 :25–34, 1994.
- [15] C. Zoppou and S. Roberts. Explicit schemes for dam-break simulations. *Journal of Hydraulic Engineering*, 129(1) :11–34, January 2003.



Unité de recherche INRIA Rhône-Alpes
655, avenue de l'Europe - 38334 Montbonnot Saint-Ismier (France)

Unité de recherche INRIA Futurs : Parc Club Orsay Université - ZAC des Vignes
4, rue Jacques Monod - 91893 ORSAY Cedex (France)

Unité de recherche INRIA Lorraine : LORIA, Technopôle de Nancy-Brabois - Campus scientifique
615, rue du Jardin Botanique - BP 101 - 54602 Villers-lès-Nancy Cedex (France)

Unité de recherche INRIA Rennes : IRISA, Campus universitaire de Beaulieu - 35042 Rennes Cedex (France)

Unité de recherche INRIA Rocquencourt : Domaine de Voluceau - Rocquencourt - BP 105 - 78153 Le Chesnay Cedex (France)

Unité de recherche INRIA Sophia Antipolis : 2004, route des Lucioles - BP 93 - 06902 Sophia Antipolis Cedex (France)

Éditeur
INRIA - Domaine de Voluceau - Rocquencourt, BP 105 - 78153 Le Chesnay Cedex (France)
<http://www.inria.fr>
ISSN 0249-6399