



Modélisation numérique de l'effet de la rugosité du sol sur la direction du ruissellement de l'eau

INRA d'Orléans - Université de Nantes - Université d'Orléans

Pierre-Antoine KSINANT

Master Professionnel Ingénierie Mathématique
Spécialité Calcul Scientifique
Année universitaire 2009/2010

Encadrants :

Frédéric DARBOUX (responsable de stage) - UR SOLS, INRA d'Orléans
Christophe BERTHON (responsable pédagogique) - Université de Nantes
Christian LAGUERRE (co-responsable de stage) - MAPMO, CNRS

Table des matières

1	Remerciements	5
2	Contexte du stage	6
2.1	Présentation des structures d'accueil	6
2.2	Le projet ANR METHODE	8
2.3	Objectifs du stage	9
3	Méthodologie	11
3.1	Généralités	11
3.2	GARS, générateur de surfaces rugueuses écrit en C	12
3.2.1	Présentation du programme	12
3.2.2	Modifications apportées	13
3.2.3	Exemples de surfaces générées	16
3.3	FullSWOF_2D, logiciel de simulation du ruissellement en C++	16
3.3.1	Présentation du logiciel	16
3.3.2	Les équations de Saint-Venant (ou « <i>shallow water equations</i> »)	22
3.3.3	Travail collaboratif sur un code de calcul	23
3.3.4	Adaptations apportées	25
3.4	Ga2Fu, script shell pour la modélisation paramétrique	27
3.4.1	Modèle structurel utilisé	27
3.4.2	Moyens de calcul	32
3.4.3	Exemple d'utilisation	33
4	Résultats	40
4.1	Démarche suivie	40
4.1.1	Première étape : Réduction des temps de calcul numérique	40
4.1.2	Seconde étape : Calcul paramétrique	42
4.2	Premiers résultats	43
4.2.1	Première étape : Réduction des temps de calcul numérique	44
4.2.2	Seconde étape : Calcul paramétrique	46
5	Conclusion et perspectives	50
	Bibliographie	51
A	Adhémar Barré de Saint-Venant	i

Partie 1

Remerciements

Ce stage de fin d'études de master, entrant dans le cadre du projet ANR METHODE - pour Modélisation de l'Écoulement sur une Topographie avec des Hétérogénéités Orientées et des Différences d'Echelle - regroupant plusieurs partenaires académiques et instituts de recherche (INRA d'Orléans, MAPMO, BRGM, INRIA, ...), m'a permis de cotoyer au quotidien plusieurs participants venant d'horizons différents et issus de parcours distincts, ce qui s'est avéré très enrichissant, me faisant naviguer dans un univers pluridisciplinaire des plus plaisants et formateurs.

En prélude à ce rapport de stage, je tiens ainsi à adresser mes sincères remerciements à :

- Frédéric Darboux, mon responsable de stage principal, pour son accueil chaleureux, sa pédagogie, sa compréhension et sa rigueur, contribuant à faire de moi jour après jour un professionnel plus efficace et opérationnel, me poussant sans cesse à m'améliorer et à adopter une posture introspective quant à mon travail et à sa qualité,
- Christian Laguerre, mon second responsable de stage, pour sa patience, sa pédagogie, son sens de la méthode, sa générosité et sa disponibilité,
- Carine Lucas, membre du MAPMO, pour son aide, sa disponibilité et ses recommandations éclairantes et constructives,
- François James et Stéphane Cordier, membres du MAPMO, pour leurs points de vue érudits et leurs conseils avisés,
- Olivier Delestre, ancien doctorant du MAPMO, pour son aide spontanée et ses suggestions réfléchies,
- l'ensemble de mes collègues de bureau à l'INRA d'Orléans (Lei Yan, Alain Couturier, Hongtao Hao, ...) qui par leur joie de vivre et leur bonne humeur contribuent à faire de ce stage un agréable moment,
- et enfin à Christophe Berthon, de l'Université de Nantes, sans qui je n'aurais sans doute pas pu faire ce stage.

Il m'aurait été bien difficile de pouvoir trouver meilleur cadre de stage et disposer d'interlocuteurs aussi compétents et professionnels.

Partie 2

Contexte du stage

Sommaire

2.1	Présentation des structures d'accueil	6
2.2	Le projet ANR METHODE	8
2.3	Objectifs du stage	9

2.1 Présentation des structures d'accueil

Placé sous le statut d'établissement public à caractère scientifique et technologique, l'INRA¹ - pour Institut National de la Recherche Agronomique - est un organisme, fondé en 1946, actuellement sous les tutelles conjointes du Ministère chargé de la Recherche et de celui chargé de l'Agriculture.

L'établissement, dont le siège est à Paris, est dirigé par Marion Guillou et emploie 1820 chercheurs, 2462 ingénieurs et 4108 techniciens et administratifs, qui, complétés par 1833 doctorants, contribuent à faire de l'INRA le premier centre de recherche agronomique en Europe et le deuxième dans le monde au regard du nombre de publications en sciences agricoles et en sciences de la plante et de l'animal.

Les missions de l'institut consistent à :

- œuvrer au service de l'intérêt public tout en maintenant l'équilibre entre les exigences de la recherche et les demandes de la société,
- produire et diffuser des connaissances scientifiques et des innovations, principalement dans les domaines de l'agriculture, de l'alimentation et de l'environnement,
- contribuer à l'expertise, à la formation, à la promotion de la culture scientifique et technique et au débat de la science au sein de la société.

Plus globalement, les ambitions des recherches menées à l'INRA sont en ce début de 21^{ème} siècle sous-jacentes aux grands défis que constituent pour l'humanité les questions de

¹Pour des informations complémentaires à celles dispensées par cette présentation, le lecteur est invité à consulter le site www.inra.fr.

l'alimentation, de l'environnement et de la valorisation des territoires. En effet, notre période contemporaine se voit confrontée à un grand nombre d'enjeux, tels le changement climatique, la nutrition humaine, l'équilibre entre les cultures alimentaires et non alimentaires ou encore l'épuisement des ressources fossiles, qui placent indéniablement l'agronomie au cœur de la réflexion à mener pour un développement harmonieux sur les plans sociaux, économiques et environnementaux.

Structuré autour de 20 centres régionaux comprenant 381 unités de recherche, d'expérimentation, d'appui à la recherche ou de service, l'INRA est présent dans toutes les régions françaises mais également dans certains pays étrangers - la Chine, le Brésil et l'Inde -, où elle dispose de délégations.

Le centre INRA d'Orléans², dans lequel j'effectue mon stage, a été inauguré en 1977 sur le site d'Ardon. Articulé autour de 6 unités, dont l'unité de recherche de science du sol qui m'accueille, il compte environ 200 agents titulaires répartis de la manière suivante :

- 38% de chercheurs et d'ingénieurs,
- 49% de techniciens,
- 13% d'administratifs.

Créée en 2000 et constituée d'une unique équipe de recherche composée de 26 membres permanents, dont 7 chercheurs et 6 ingénieurs, l'unité de recherche de science du sol concentre ses activités autour de deux axes :

- l'évolution et le fonctionnement spatialisé des sols,
- la valorisation et la protection des sols.

Pour mener à bien ses projets, l'unité couple des approches d'observations et de mesures sur le terrain, sur des sites choisis comme représentatifs de grands systèmes pédologiques, des expérimentations en laboratoire sous conditions contrôlées ainsi que de la modélisation. Elle dispose notamment d'un simulateur de pluie équipé de bacs de grande taille (10 mètres carrés) uniques en France et met en place, sur un terrain en grandes cultures (10 kilomètres carrés), un site-atelier de suivi spatialisé de l'évolution et du fonctionnement des sols (hydro-thermie et émissions de gaz à effet de serre) selon les pratiques agricoles. De plus, elle emploie et bâtit des modèles de fonctionnement spatialisé des sols, afin aussi bien de formaliser que d'éprouver de nouvelles hypothèses. Il convient ainsi de souligner ses importantes connaissances et compétences relativement à la géostatistique et aux systèmes d'informations géographiques.

Le MAPMO³ - pour **M**athématiques et **A**pplications, **P**hysique **M**athématique d'**O**rléans - est la seconde structure m'accueillant dans le cadre de mon stage. Appartenant à la Fédération Denis Poisson et rattaché à l'Université d'Orléans, ce laboratoire de recherche et d'enseignement compte pour personnel permanent une quarantaine de personnes (enseignants-chercheurs, ingénieurs, techniciens et administratifs) et pour personnel temporaire un effectif regroupant une vingtaine d'individus (doctorants, post-doctorants, ATER).

²Un lecteur intéressé trouvera à l'adresse www.orleans.inra.fr l'ensemble des renseignements complémentaires qui n'auraient pas été portés dans cette présentation.

³Pour un descriptif plus conséquent à celui proposé ici, le lecteur est invité à consulter le site www.univ-orleans.fr/mapmo.



FIG. 2.1 – Vue aérienne du centre INRA d’Orléans, Janvier 2007 - Photo : ©INRA J.-C. Bastien & O. Bertel

Dirigé par Stéphane Cordier, le MAPMO se structure autour de l’analyse mathématique, des probabilités ainsi que de la physique mathématique en se reposant sur quatre pôles de recherche hautement interactifs :

1. Équations aux Dérivées Partielles, Physique, Modélisation (équipe EPM) ;
2. Probabilités, Statistiques et Modélisation (équipe PSM) ;
3. Analyse, Systèmes Dynamiques, Géométrie (équipe ADG) ;
4. Algèbres d’Opérateurs et Applications (équipe AOA).

On peut également noter que ces dernières années, ce laboratoire a su lier des liens aussi bien avec d’autres partenaires académiques, aux niveaux national et international, qu’avec certains groupes industriels, par le jeu de projets communs.

2.2 Le projet ANR METHODE

Démarré en 2007, le projet ANR METHODE a pour objectif d’étudier les effets des hétérogénéités de surface sur le ruissellement, en d’autres termes, il s’agit de déterminer l’impact des sillons agricoles sur l’écoulement des eaux de pluie à la surface du sol.

En effet, le ruissellement de l’eau sur les sols cultivés engendre des problèmes quant à la conservation des ressources environnementales : affaiblissement des épaisseurs du sol par érosion, pertes en nutriments et dégradation de la qualité des eaux. Ainsi, les périodes ruisselantes, de nature discontinue dans le temps et dans l’espace, sont également à l’origine de coulées boueuses pouvant affecter les biens et les personnes. Pour améliorer l’aménagement des bassins versants⁴, il est donc nécessaire de prédire correctement la localisation des écou-

⁴On appelle bassin versant une zone géographique sur laquelle l’écoulement de l’eau nourrit un exutoire commun.

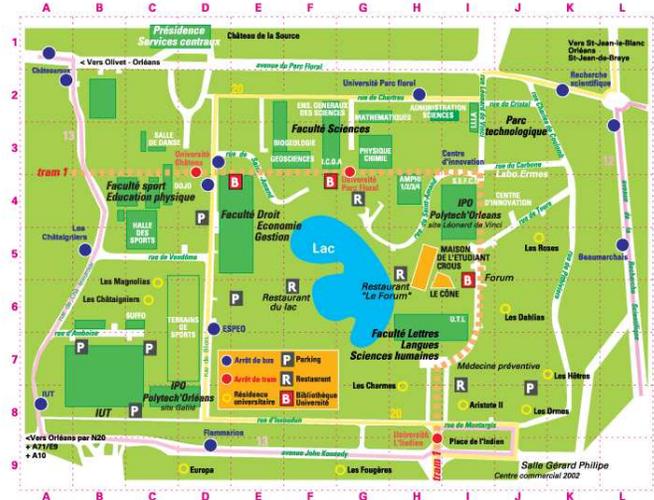


FIG. 2.2 – Plan du Campus d'Orléans

lements de surface, ce que les modèles hydrologiques actuellement utilisés sont en grande partie incapables de faire.

En contexte agricole, des travaux empiriques ont permis de démontrer que l'interaction sillons-topographie⁵ était déterminante sur la géométrie du réseau d'écoulement. Il a ainsi pu être établi que pour de faibles écoulements, le ruissellement suit la direction des sillons tandis que pour de forts écoulements, le ruissellement suit la direction de la pente topographique. Cependant, les modèles hydrologiques actuellement en cours ne prennent en compte cette interaction que de manière binaire, en se basant sur des modèles heuristiques du type « loi du tout ou rien »⁶.

Dans le cadre du projet, des études sont menées pour modéliser ce type d'écoulement pour comprendre et de prédire l'effet de la topographie sur l'écoulement de l'eau, afin d'inclure l'effet de l'interaction topographie-sillons dans les modèles de ruissellement utilisés en hydrologie.

2.3 Objectifs du stage

Comme cela a été précisé précédemment, le ruissellement d'eau à la surface du sol cause des redistributions de matières. Afin de contrôler ces processus, il convient donc prédire la direction des écoulements. Cette direction résulte de l'interaction entre :

1. la pente générale du sol,
2. les sillons agricoles,
3. la rugosité aléatoire due aux mottes de terre.

⁵On désigne par topographie les caractéristiques (relief, forme, configuration) d'un lieu géométrique.

⁶Voir [4], [5], [6] et [7].

Dans ce travail, ce troisième caractère ne sera pas pris en considération.

Ainsi, l'objectif du stage est de proposer une méthodologie qui permettra d'établir une loi de comportement prévoyant la direction de l'écoulement à partir des paramètres décrivant chacune de ces deux composantes. De cette manière, à partir d'une description exhaustive de la surface, le protocole devra formaliser un processus de calcul paramétrique qui conduira à une prédiction de la direction du ruissellement de l'eau n'utilisant qu'une représentation implicite de la topographie de la surface du sol. Il sera donc alors possible de paramétrer les modèles hydrologiques sans avoir à y inclure la représentation détaillée du domaine considéré.

Partie 3

Méthodologie

Sommaire

3.1	Généralités	11
3.2	GARS, générateur de surfaces rugueuses écrit en C	12
3.2.1	Présentation du programme	12
3.2.2	Modifications apportées	13
3.2.3	Exemples de surfaces générées	16
3.3	FullSWOF_2D, logiciel de simulation du ruissellement en C++	16
3.3.1	Présentation du logiciel	16
3.3.2	Les équations de Saint-Venant (ou « <i>shallow water equations</i> »)	22
3.3.3	Travail collaboratif sur un code de calcul	23
3.3.4	Adaptations apportées	25
3.4	Ga2Fu, script shell pour la modélisation paramétrique	27
3.4.1	Modèle structurel utilisé	27
3.4.2	Moyens de calcul	32
3.4.3	Exemple d'utilisation	33

3.1 Généralités

Afin de mener à bien le projet de ce stage, deux étapes de travail ont été définies.

Dans une première étape, il s'agit de mettre en place l'ensemble des conditions adéquates inhérentes au lancement des simulations numériques qui permettront de modéliser la loi de comportement à définir. En ce sens, deux logiciels développés au MAPMO - un générateur de surfaces rugueuses et un logiciel de simulation de ruissellement sur des surfaces données - devront être modifiés ou adaptés selon les cas, puis ensuite couplés par le biais d'un script. Ce dernier devra notamment permettre d'exploiter pleinement l'ensemble des ressources d'une plateforme de calcul à disposition, tout en automatisant le lancement de simulations numériques dont les paramètres devront pouvoir être définis par l'utilisateur de manière commode. Enfin, il devra également être en mesure de récupérer et d'ordonner les résultats obtenus de telle sorte que ces derniers soient facilement exploitables dans le cadre d'une étude statistique.

Dans une seconde étape, il conviendra d'établir le protocole de travail qui permettra de :

1. Valider la méthodologie mise en œuvre,
2. Restreindre les champs et temps de calcul,
3. Fixer un ensemble de cas tests caractéristiques de la réalité du terrain et dont les résultats des simulations numériques permettront d'extraire une loi de comportement à formaliser.

Dans la suite de ce rapport, c'est cette démarche qui sera suivie et développée.

3.2 GARS, générateur de surfaces rugueuses écrit en C

3.2.1 Présentation du programme

Écrit en langage C, le logiciel GARS - pour **Generate A Rough Surface** - est, comme son nom l'indique, un générateur de surfaces rugueuses. Il a été à l'origine développé par Frédéric Darboux, Christian Laguerre et Qiao Wang sous une licence CeCILL¹ - pour **Cea Cnrs Inria Logiciel Libre** - avec pour objectif de pouvoir simuler une portion d'une parcelle agricole aux caractéristiques connues.

Les surfaces rugueuses, qu'on dénomme ainsi en opposition aux surfaces lisses, sont généralement classées selon trois critères :

1. caractère périodique ou non périodique,
2. présence d'un support borné ou non borné - en d'autres termes, présence d'une déformation finie ou infinie -,
3. caractère déterministe ou aléatoire.

Ce sont les surfaces rugueuses aléatoires à support borné que cherche à modéliser GARS.

Afin d'atteindre ce dessein, le programme repose sur trois grands piliers permettant de caractériser de telles topographies :

1. la donnée d'une pente générale caractérisée par son pourcentage de dénivellation ainsi que par sa direction,
2. la définition de sillons agricoles modélisés par des sinusoïdes, elles-mêmes précisées par leur amplitude, leur longueur d'onde, leur direction ainsi que la distance entre l'origine du repère et le premier maximum de la sinusoïde la plus proche,
3. la conception d'un bruit particularisé par son spectre de puissance, dont l'écart-type, les longueurs d'onde des seuils de coupure inférieur et supérieur ainsi que le coefficient directeur de la droite liant ces deux seuils, tout comme deux paramètres permettant d'initialiser les générateurs aléatoires, seront fournis.

¹Plus précisément, il s'agit de la licence CeCILL-B (les termes réglementant cette dernière se trouvent à l'adresse www.cecill.info/licences/Licence_CeCILL-B_V1-fr.html).

S'utilisant en ligne de commande, on peut noter qu'en tapant « `./gars` » dans un terminal, l'utilisateur verra s'afficher à l'écran une aide lui rappelant la syntaxe intrinsèque à un bon emploi du programme, ainsi que la spécification des différents paramètres d'entrée.

Par défaut, on remarquera aussi que GARS affiche à l'écran son unique sortie, qui correspond aux différentes coordonnées des points, relatifs aux nœuds du maillage, de la surface rugueuse générée. Bien entendu, l'utilisateur peut rediriger cette sortie dans un fichier texte, s'il le souhaite, par le biais du mécanisme de redirection « `>` ».

3.2.2 Modifications apportées

Afin de pouvoir coupler GARS avec le logiciel FullSWOF_2D, permettant de simuler le ruissellement de l'eau sur des surfaces données, un certain nombre de modifications et d'adaptations ont été opérées.

Maillage du domaine considéré

Le premier changement effectué dans le logiciel GARS concerne le maillage utilisé. Illustrons ce dernier à l'aide du dessin répertorié sur la figure 3.1, page 13, où figureront également l'ensemble des paramètres concernés.

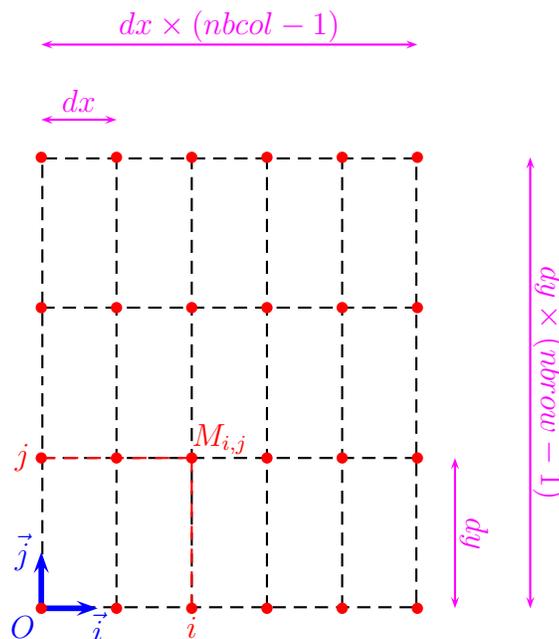


FIG. 3.1 – Maillage utilisé dans le programme GARS

Sur ce dessin, on a noté :

- (O, \vec{i}, \vec{j}) le repère d'espace considéré,
- dx , le pas d'espace horizontal,

- dy , le pas d'espace vertical,
- $nbcol$, le nombre de colonnes,
- $nbrow$, le nombre de lignes.

et où chaque point du maillage $M_{i,j}$, avec $i \in \{1, \dots, nbcol\}$ et $j \in \{1, \dots, nbrow\}$, est défini par :

$$\overrightarrow{OM_{i,j}} = [dx \times (i - 1)] \vec{i} + [dy \times (j - 1)] \vec{j}$$

Remarque. Initialement, suivant une convention normative courante en imagerie, domaine dont GARS fournit des applications, le repère d'espace considéré avait son origine située dans le coin Nord-Ouest tandis que les vecteurs de base \vec{i} et \vec{j} étaient respectivement égal et opposé aux nouveaux vecteurs de base ici définis. Notons également que précédemment, les nœuds du maillage utilisé par GARS ne se trouvaient non pas aux extrémités des éléments définissant la « grille », mais au centre de ces derniers. Remarquons enfin qu'avant ces modifications, il n'existait qu'un unique pas d'espace, valable aussi bien dans la définition de l'épaisseur des lignes que celle des colonnes.

Gestion de la pente générale

Le deuxième type d'adaptations effectuées, conditionnées par les changements de maillage et de repère ainsi que par l'ajout de nouveaux paramètres, relève de la gestion de la pente générale. Comme ceci a déjà été indiqué dans la section 3.2.1, page 12, cette dernière se caractérise par sa direction, subordonnée à un angle noté α , exprimé en degrés et tel que $\alpha \in [0; 360[$, ainsi que par un pourcentage de dénivellation, noté p , tel que $p \geq 0$ et étant entendu que les altitudes diminuent suivant la direction choisie.

Illustrons une nouvelle fois l'incidence des paramètres ici décrits par le biais de deux dessins représentés sur la figure 3.2, page 15.

Il est alors aisé de démontrer que pour tout $i \in \{1, \dots, nbcol\}$ et $j \in \{1, \dots, nbrow\}$, si on note $z_{i,j}$ l'altitude au point $M_{i,j}$ tel que défini précédemment, alors, après la prise en compte de l'influence de la pente générale, on établit les relations suivantes :

$$\left\{ \begin{array}{l} \text{si } p = 0, \text{ alors : } z_{i,j} = z_{i,j} + 0 \\ \text{sinon : } \left\{ \begin{array}{l} \text{si } \cos \left[(90 + \alpha) \frac{\pi}{180} \right] (j - 1)dy = \sin \left[(90 + \alpha) \frac{\pi}{180} \right] (i - 1)dx, \\ \text{alors : } z_{i,j} = z_{i,j} + 0 \\ \text{sinon : } z_{i,j} = z_{i,j} - \frac{p}{100} \cos \left(\alpha \frac{\pi}{180} - \arctan \left[\frac{(j-1)dy}{(i-1)dx} \right] \right) \sqrt{[(i-1)dx]^2 + [(j-1)dy]^2} \end{array} \right. \end{array} \right.$$

Gestion des sillons agricoles

Le troisième type d'adaptations apportées est lui relatif à la gestion des sillons agricoles. Ces derniers, modélisés par des sinusoides, sont caractérisés dans le programme par une amplitude a , telle que $a \in \mathbb{R}$ et $a \geq 0$, une longueur d'onde T , telle que $T \in \mathbb{R}$ et $T > 0$, une direction sous-jacente à un angle noté β , exprimé en degrés et tel que $0 \leq \beta < 180$, et par la distance d , telle que $d \in \mathbb{R}$ et $0 \leq d < T$, séparant l'origine du repère au premier maximum de la sinusoides la plus proche.

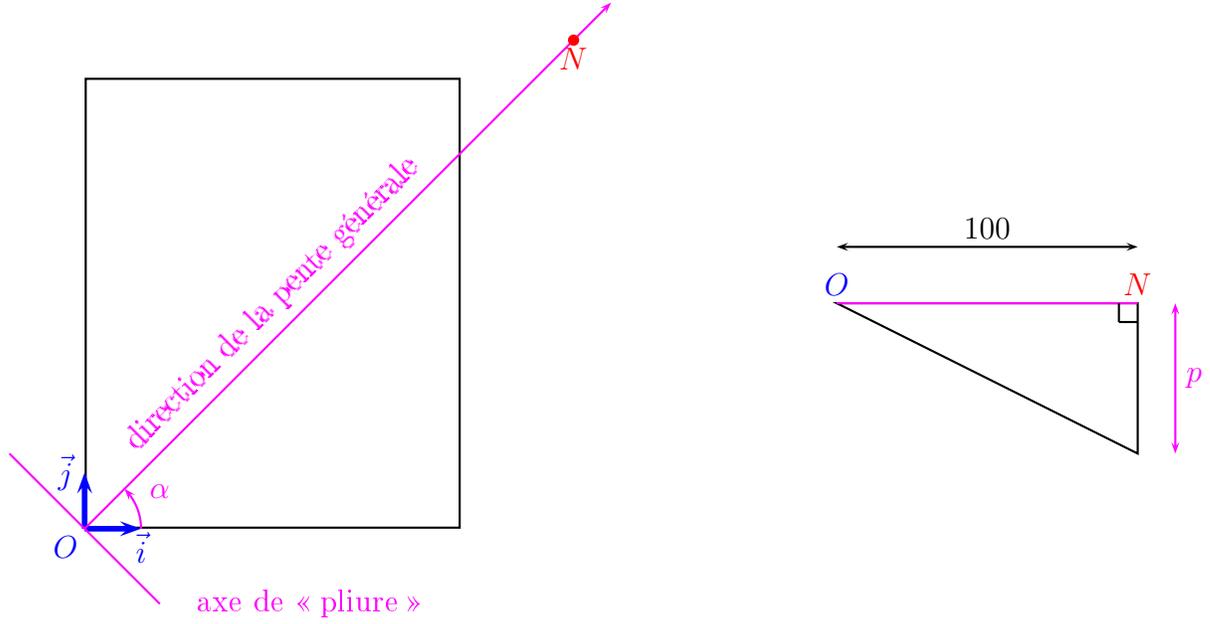


FIG. 3.2 – Illustrations concernant la gestion de la pente générale de la surface par GARS

Illustrons cette fois encore l'influence des paramètres définis ci-dessus grâce aux deux dessins représentés sur la figure 3.3, page 16.

On démontre alors pour tout $i \in \{1, \dots, nbcol\}$ et $j \in \{1, \dots, nbrow\}$, suite à la prise en compte de l'influence des sillons agricoles, on obtient les relations suivantes :

$$\left\{ \begin{array}{l} \text{si } a = 0, \text{ alors : } z_{i,j} = z_{i,j} + 0 \\ \text{sinon : } \left\{ \begin{array}{l} \text{si } \cos \left[(90 + \alpha) \frac{\pi}{180} \right] (j - 1)dy = \sin \left[(90 + \alpha) \frac{\pi}{180} \right] (i - 1)dx, \\ \text{alors : } z_{i,j} = z_{i,j} + a \sin \left[\frac{2\pi}{T} \left(\frac{T}{4} - d \right) \right] \\ \text{sinon : } z_{i,j} = z_{i,j} + \\ a \sin \left[\frac{2\pi}{T} \left(\cos \left(\beta \frac{\pi}{180} - \arctan \left[\frac{(j-1)dy}{(i-1)dx} \right] \right) \sqrt{[(i-1)dx]^2 + [(j-1)dy]^2} + \frac{T}{4} - d \right) \right] \end{array} \right. \end{array} \right.$$

Gestion du bruit

Comme ceci a été précisé lors de l'annonce des objectifs de ce stage dans la section 2.3, page 9, aucune adaptation ou modification n'a été effectuée concernant la gestion du bruit dans GARS. En effet, la composante des parcelles agricoles relative à la rugosité aléatoire due aux mottes de terre n'est pas prise en compte dans le cadre de ce travail.

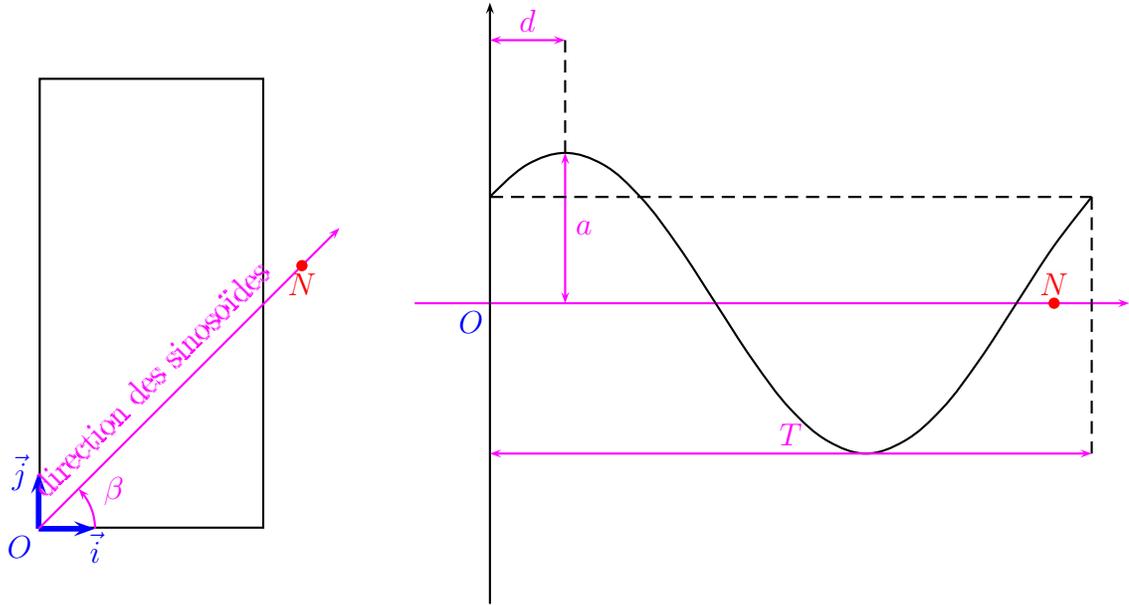


FIG. 3.3 – Illustrations concernant la gestion des sillons agricoles de la surface par GARS

3.2.3 Exemples de surfaces générées

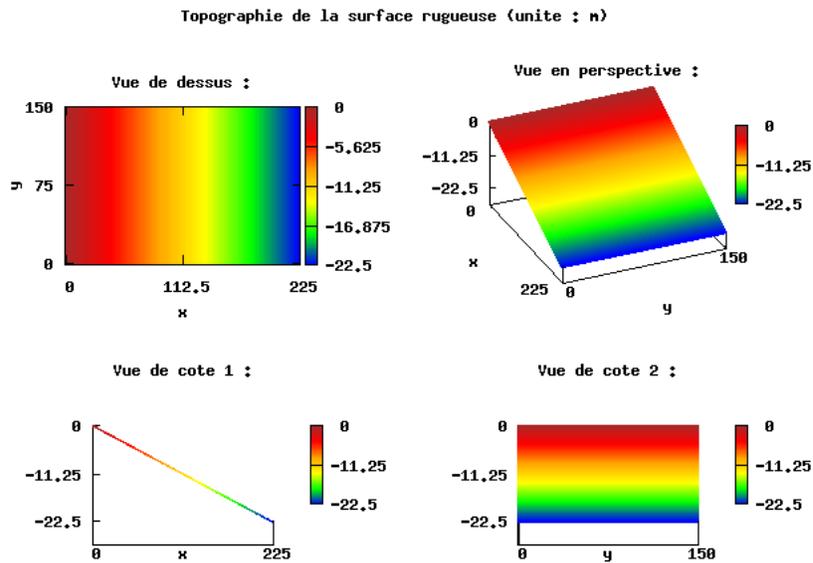
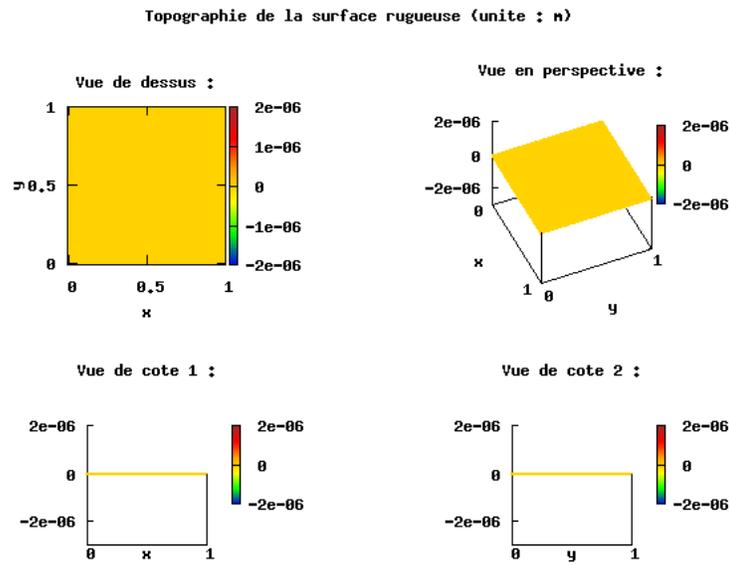
Afin d'apprécier l'influence des différents paramètres décrits dans la section précédente, le lecteur peut observer le jeu de graphiques qui est proposé² (voir les figures 3.4, page 17, 3.5, page 17, 3.6, page 18 et 3.7, page 18).

3.3 FullSWOF_2D, logiciel de simulation du ruissellement en C++

3.3.1 Présentation du logiciel

Écrit en langage C++, FullSWOF_2D - pour **F**ull **S**hallow **W**ater equations for **O**verland **F**low - est un logiciel permettant de simuler le ruissellement de l'eau en deux dimensions sur une surface donnée. Son auteur principal, Olivier Delestre, a débuté son développement dans le cadre d'un stage de master 2 d'informatique effectué à l'INRA d'Orléans en 2008, avant de poursuivre ce travail durant sa thèse de mathématiques appliquées, intitulée « Simulation numérique du ruissellement d'eau de pluie sur des surfaces agricoles » et réalisée au sein du MAPMO, qu'il a soutenue le 13 juillet 2010 à l'Université d'Orléans. FullSWOF_2D a été

²Celui-ci a été obtenu par le biais du petit utilitaire GarsPlot, que j'ai écrit en C, qui s'utilise dans un terminal en ligne de commande et exporte au format png les sorties du programme GARS. Notons que celui-ci utilise le logiciel Gnuplot, dont un lecteur intéressé trouvera une présentation complète à l'adresse www.gnuplot.info et dont les termes réglementant la licence sont exposés à l'adresse gnuplot.cvs.sourceforge.net/gnuplot/gnuplot/Copyright?view=markup.



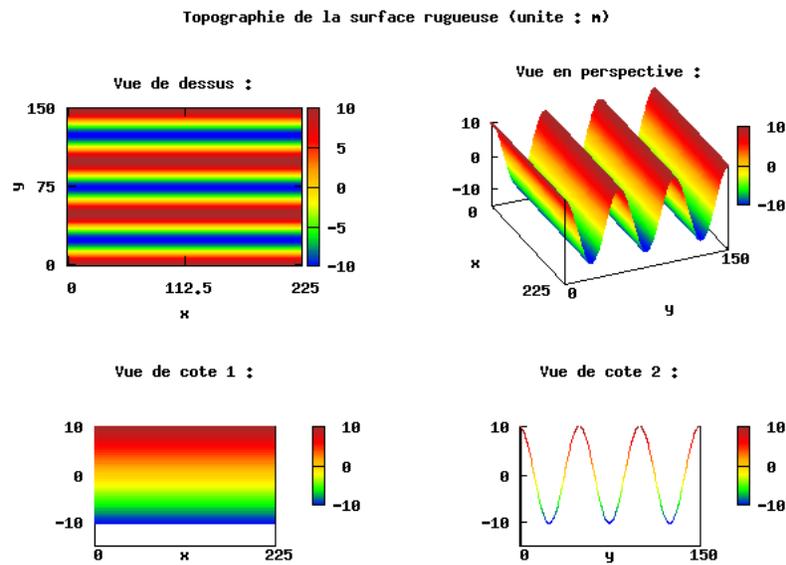


FIG. 3.6 – Exemple de surface définie par des sillons agricoles ($a = 10$ m, $T = 50$ m, $\beta = 90^\circ$ et $d = 0$ m)

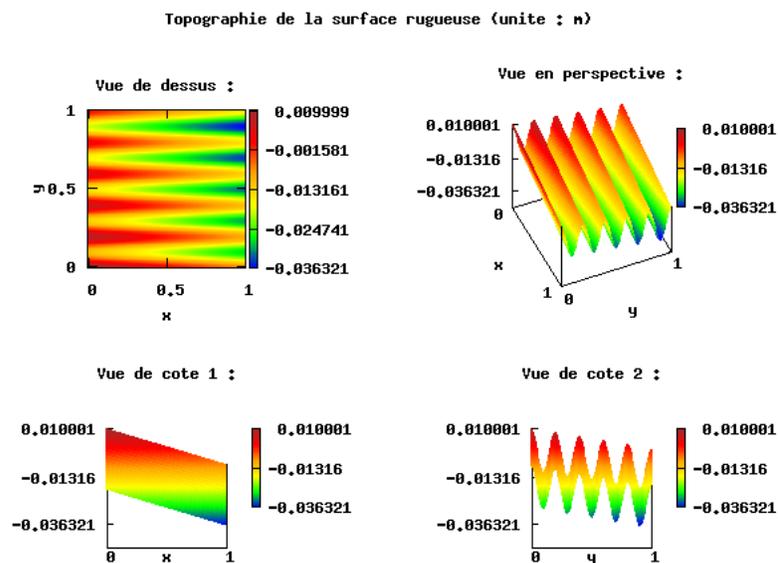


FIG. 3.7 – Exemple de surface définie par une pente générale ($p = 2\%$ et $\alpha = 30^\circ$) et des sillons agricoles ($a = 0,01$ m, $T = 0,2$ m, $\beta = 90^\circ$ et $d = 0$ m)

déposé à l'Agence pour la Protection des Programmes avec le numéro de déclaration *DL* 03434 - 01 et a été placé sous une licence CeCILL³.

Afin de répondre à la problématique posée par sa thèse, Olivier Delestre, après avoir nourri sa réflexion par une étude approfondie du sujet et consulté les rapports et conclusions de différents experts, a pris la décision d'utiliser le système de Saint-Venant⁴ bidimensionnel pour modéliser l'écoulement des eaux de pluie sur des domaines agricoles. En effet, ce système d'équations aux dérivées partielles est bien adapté à la modélisation des écoulements en eaux peu profondes. En vue de le résoudre numériquement, en raison de l'hyperbolicité et du caractère conservatif du système, son choix s'est alors porté sur un schéma aux volumes finis « équilibré » (ou encore « *well-balanced* ») avec reconstruction hydrostatique. Contrairement aux schémas numériques largement répandus et utilisés par les hydrologues⁵, « communauté » avec laquelle il a collaboré durant ses recherches, le choix judicieux d'Olivier Delestre présente l'avantage d'assurer la positivité des hauteurs d'eau ainsi que de permettre de calculer correctement les états d'équilibre⁶.

Le fonctionnement global de FullSWOF_2D est illustré à l'aide de la figure 3.8, page 20 Précisons maintenant chacune des deux étapes présentes sur ce graphique :

1. Initialement, FullSWOF_2D doit être lancé dans un répertoire contenant deux sous-répertoires nommés respectivement Inputs et Outputs, qui, comme leurs noms l'indiquent, contiennent les entrées et les sorties du logiciel. Intéressons-nous à Inputs. Ce sous-répertoire doit, ou peut selon les cas, contenir les quatre fichiers suivants :
 - *parameters.txt*⁷. Fichier texte regroupant l'ensemble des paramètres et instructions desquels FullSWOF_2D a besoin afin de fonctionner. Ces derniers sont relatifs :
 - au domaine considéré (pas d'espace et nombres de nœuds horizontaux et verticaux),
 - au temps de simulation souhaité (le temps lui-même ainsi que le pas de temps),
 - aux conditions de bord désirées (de type liquide, mur et Neumann),
 - à la loi de frottement à inclure (coefficient de frottement ainsi que loi de type Manning ou Darcy-Weisbach),
 - aux choix relatifs au schéma volumes finis à employer (flux numérique - Rusanov ou HLL -, ordre - 1, 2 ou relaxation -, reconstruction - MUSCL, ENO ou ENO modifié -, limiteur - MINMOD ou Van Albada - et CFL),
 - à divers paramètres physiques,
 - à trois choix relatifs à la topographie, aux conditions initiales de l'eau reposant sur la topographie (hauteur, intensités des composantes de la vitesse) ainsi qu'aux conditions de pluie.

Ce fichier est indispensable au bon déroulement du programme.

³Plus précisément, il s'agit de la licence CeCILL-V2 (les termes réglementant cette dernière se trouvent à l'adresse www.cecill.info/licences/Licence_CeCILL_V2-fr.html).

⁴Une présentation plus complète de ce modèle se trouve à la section 3.3.2, page 22.

⁵Nous pouvons par exemple ici citer le schéma de Mac Cormack, datant de 1969 et constituant un schéma d'ordre 2 basé sur la résolution du système de Saint-Venant avec une méthode de type différences finies.

⁶Voir [2].

⁷Dans cette présentation, les noms des fichiers qui seront écrits en *italique* sont des noms imposés par le programme.

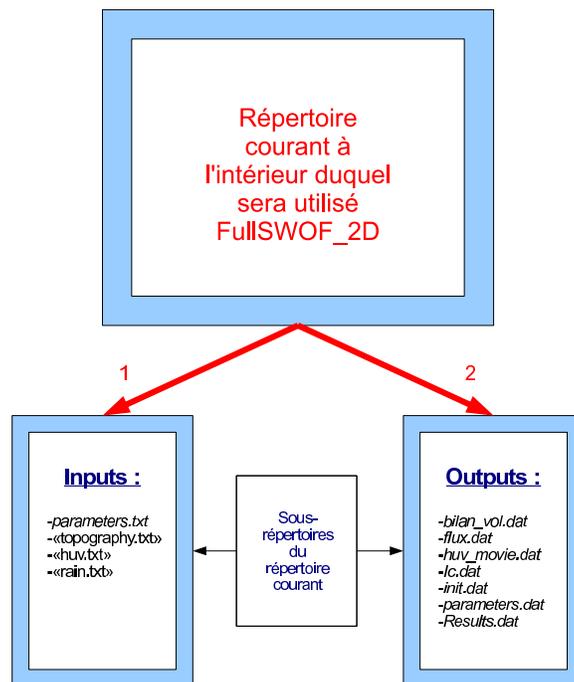


FIG. 3.8 – Modèle structurel de fonctionnement de FullSWOF_2D

- « topography.txt »⁸. Selon le choix qu'il a effectué dans le fichier « parameters.txt », l'utilisateur dispose de la possibilité d'indiquer la topographie qu'il souhaite utiliser à travers un fichier texte, devant respecter certaines normes quant à son format, et dans lequel doivent figurer les altitudes aux différents nœuds du maillage.
 - « huv.txt ». Comme précédemment, ce fichier est subordonné au choix de l'utilisateur, qui peut y faire figurer, selon les mêmes normes de format en cours dans le fichier précédent, les conditions initiales de hauteurs d'eau ainsi que de vitesses ce cette dernière (intensités des composantes) aux nœuds du maillage.
 - « rain.txt ». Ici encore, ce fichier est dépendant du choix de l'utilisateur et peut lui permettre d'imposer les conditions de pluie souhaitées pendant la simulation.
2. Une fois FullSWOF_2D lancé, le programme va écrire dans divers fichiers du sous-répertoire Outputs les résultats qu'il obtient. En fin de simulation, les fichiers obtenus sont les suivants :
- *parameters.dat*. Ce fichier comporte l'ensemble des paramètres qui ont été lus dans le fichier « parameters.txt » par le logiciel, et constitue de fait un outil de contrôle pour l'utilisateur ainsi qu'une sauvegarde des données avec les résultats.
 - *init.dat*. Ce fichier comporte dans une norme de format exploitable par FullSWOF_2D les informations relatives à la topographie ainsi qu'au comportement initial de l'eau (hauteur, intensités des composantes de la vitesse) sur le domaine considéré. Ce fichier constitue également un levier de contrôle pour l'utilisateur.
 - *Ic.dat*. Ce fichier contient la liste des différents pas de temps employés aux différentes itérations effectuées par le programme pendant la simulation⁹.
 - *huv_movie.dat*. Ce fichier contient une liste temporelle des différentes conditions relatives à l'eau (hauteur, intensités des composantes de la vitesse) en chaque nœud du maillage.
 - *bilan_vol.dat*, *flux.dat* et *Results.dat*, divers fichiers annexes relatifs à la gestion de la pluie au cours de la simulation.

Notons qu'initialement, Olivier Delestre avait entamé la programmation de FullSWOF_2D en Fortran 77. Néanmoins, après s'être rendu compte de la redondance des phases de tests quant aux choix faits par l'utilisateur, parties du code coûteuses et peu lisibles, ainsi que de la possibilité de regrouper ces mêmes choix en un certain nombre de catégories (conditions de bord, frottement, flux numérique, ordre du schéma, méthode de reconstruction, ...), sa réflexion l'a naturellement poussé à finalement adopter un langage de programmation orientée objet, en l'occurrence le C++. Comblant les lacunes inhérentes à l'emploi d'un langage de programmation impérative, ce choix permet également une plus grande modularité quant à la structure du code, ce qui lui assure une compréhension plus aisée, une maintenance plus efficace ainsi que la possibilité de l'enrichir de manière plus commode à l'avenir (ajouts de nouvelles conditions de bord, de nouveaux flux numériques, ...) ¹⁰.

⁸Dans cette présentation, les noms des fichiers qui seront écrits entre guillemets : « ... » sont des noms que l'utilisateur a la possibilité de choisir par le biais du fichier *parameters.txt*.

⁹L'algorithme étant basé sur le principe de la CFL fixe, le pas de temps est adaptatif.

¹⁰Voir [1].

3.3.2 Les équations de Saint-Venant (ou « *shallow water equations* »)

Une présentation complète et exhaustive des équations de Saint-Venant, ainsi que des choix ayant été opérés dans FullSWOF_2D afin de les traiter numériquement, dépasserait le cadre de ce stage et de ce rapport. Dans cette section, seul sera formulé un bref résumé tentant de fournir les points fondamentaux de cette étude. Afin d'obtenir de plus amples informations, des explications complémentaires et des justifications, un lecteur intéressé est vivement invité à consulter [2].

Présentation théorique du modèle

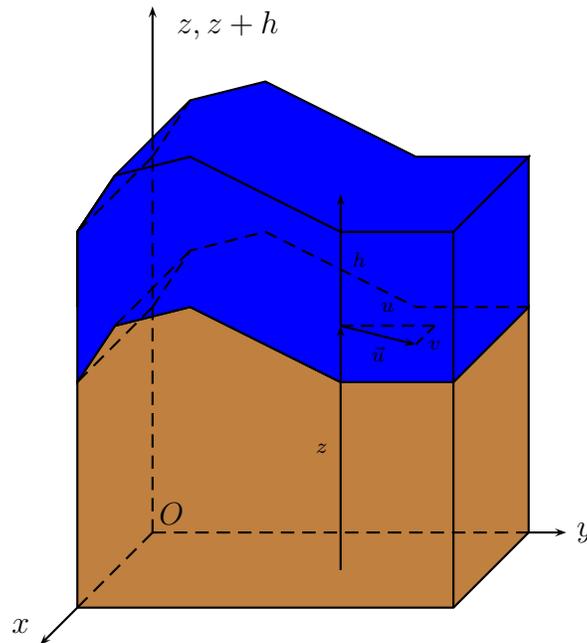


FIG. 3.9 – Illustration du système de Saint-Venant bidimensionnel et de ses variables

Le système de Saint-Venant bidimensionnel est un système d'équations aux dérivées partielles modélisant les écoulements en eaux peu profondes par l'intermédiaire de la hauteur d'eau $h(t, x, y) \geq 0$ (en m) et du vecteur vitesse moyen $\vec{u}(t, x, y) = (u, v) \in \mathbb{R}^2$ (en m.s^{-1}), avec x (en m) et y (en m) les coordonnées spatiales et t (en s) le temps (voir la figure 3.9, page 22). Il se présente sous la forme suivante :

$$\begin{cases} \partial_t h + \partial_x(hu) + \partial_y(hv) = P - I \\ \partial_t(hu) + \partial_x(hu^2 + gh^2/2) + \partial_y(huv) = gh(S_{0x} - S_{fx}) \\ \partial_t(hv) + \partial_x(huv) + \partial_y(hv^2 + gh^2/2) = gh(S_{0y} - S_{fy}) \end{cases} \quad (3.1)$$

avec $S_{0x} = -\partial_x z(x, y)$ et $S_{0y} = -\partial_y z(x, y)$,

où :

- $g = 9,81 \text{ m.s}^{-2}$ est la constante de gravité,
- $P(t, x, y)$ l'intensité de la pluie (en m.s^{-1}),
- $I(t, x, y)$ le taux d'infiltration de l'eau dans le sol (en m.s^{-1}),
- $\vec{S}_f = (S_{f_x}, S_{f_y}) \in \mathbb{R}^2$ le terme de frottement qui dépend de la loi de frottement choisie,
- $z(x, y)$ la topographie (en m),
- l'opposé de S_{0_x} (respectivement de S_{0_y}) est un nombre sans dimension qui représente la variation de la topographie selon x (respectivement selon y) et est plus communément appelé pente selon x (respectivement selon y).

Deux propriétés fondamentales de ce système d'équations aux dérivées partielles sont :

1. Le système de Saint-Venant est un système de lois de conservation. La première équation de (3.1) est l'équation de conservation de la masse tandis que les deux suivantes sont les équations de conservation de la quantité de mouvement.
2. Pour des hauteurs d'eau non nulles (c'est à dire en dehors des zones sèches), le système de Saint-Venant est strictement hyperbolique.

Traitement numérique du modèle

Bien que plusieurs méthodes numériques existent afin de traiter le système de Saint-Venant (méthodes des différences finies, des éléments finis, des volumes finis, ...), dans le cadre de la programmation de FullSWOF_2D, au regard des propriétés mathématiques énoncées lors du paragraphe précédent, ainsi que de la nécessité de conserver certains équilibres fondamentaux relatifs au problème traité (les flaques d'eau par exemple), le choix s'est porté sur un schéma aux volumes finis « équilibré » avec reconstruction hydrostatique.

En se basant sur ce choix, Olivier Delestre a notamment montré au cours de sa thèse qu'afin de répondre à la problématique qui lui était soumise, l'utilisation du flux HLL à l'ordre 2 avec une reconstruction MUSCL exploitant le limiteur de pente MINMOD constituait la combinaison la plus judicieuse.

C'est donc ces paramètres qui seront utilisés dans la suite de ce travail.

3.3.3 Travail collaboratif sur un code de calcul

La partie de ce stage relative au travail que j'ai effectuée sur le code de FullSWOF_2D est sans conteste la plus professionnalisante. En effet, au terme de ce master professionnel d'ingénierie mathématique, il ne fait aucun doute que les métiers auxquels je vais pouvoir postuler et dans lesquels je pourrais m'épanouir auront une forte connotation informatique, métiers dans lesquels je serais très certainement amené à m'inscrire dans des projets collaboratifs de programmation. En ce sens, ce stage aura été très épanouissant.

Dans le cadre de ma participation au développement de FullSWOF_2D, la problématique du stage exigeait que je modifie le code existant et que j'y implémente notamment de nouvelles

conditions de bord¹¹. j'ai donc naturellement intégré l'équipe¹² chargée du bon déroulement de son évolution.

Au contact des autres membres de cette cellule de travail, j'ai ainsi pu acquérir la façon de procéder qu'exige cet exercice ainsi que découvrir des outils informatiques communément utilisés :

- Tout d'abord, en tant qu'individu m'inscrivant dans un collectif, mon travail doit être propre, clair et lisible par autrui, ce qui se traduit notamment par le fait que les parties du programme que je code doivent être commentées, et que je dois me conformer aux impératives normes de codage qui auraient été décidées (conventions quant aux noms des variables, langue à utiliser pour les commentaires, ...). Ma contribution, bien que tout un chacun ait ses habitudes, doit pouvoir se greffer de manière uniforme et homogène à la production globale.
- Dans le cadre du développement de FullSWOF_2D, Olivier Delestre a pris soin d'utiliser le logiciel Doxygen¹³ afin, en plus des commentaires classiques se trouvant dans le code source, de générer de la documentation à partir de celui-ci même, par le biais de l'ajout d'instructions interprétables par le logiciel. Gérant un grand nombre de langages (en plus du C++, citons par exemple le C, le Java ou encore le Python), Doxygen est capable de générer de la documentation annexe en divers formats, comme le html, le Rich Text Format (RTF), le XML ou encore le \LaTeX ¹⁴.
- FullSWOF_2D étant un logiciel à caractère évolutif, la nécessité d'utiliser un système de gestion de versions - en l'occurrence ici Subversion¹⁵ (SVN en abrégé) - s'avère indispensable. J'ai ainsi eu l'occasion de découvrir un logiciel client de SVN, TortoiseSVN¹⁶. En plus de sa fonction première d'assistance visant à simplifier la gestion de versions, j'ai pu apprécier quelques-unes de ses autres fonctions, comme par exemple son interface graphique permettant de comparer deux codes sources afin de suivre les évolutions effectuées (module permettant une visualisation plus agréable que celle résultant de l'emploi de la commande UNIX « diff »).
- Dans une équipe de travail collaborative, il est important que certains rôles soient attribués, ainsi que certains outils internes soient mis en place. Il est ainsi indispensable qu'une ou des personnes soient en charge de valider, ou non, chaque nouvelle version

¹¹Une description exhaustive et explicite de ma contribution au code FullSWOF_2D ainsi que de ses motivations se trouvent à la section 3.3.4, page 25.

¹²Mes principaux interlocuteurs ont été Christian Laguerre, Frédéric Darboux, Carine Lucas, Olivier Delestre, François James et Stéphane Cordier.

¹³En grande partie réalisé par Dimitri Van Heesch, ce logiciel entre dans le cadre de la licence GNU General Public License (les termes de cette licence sont spécifiés à l'adresse www.gnu.org/licenses/old-licenses/gpl-2.0.html). Pour des informations complémentaires concernant Doxygen, le lecteur est invité à consulter l'adresse www.doxygen.org.

¹⁴Ce qui se révèle pratique pour l'écriture d'un logiciel à portée scientifique.

¹⁵Un lecteur intéressé est invité à consulter l'adresse subversion.apache.org pour obtenir de plus amples informations sur cet utilitaire, placé sous licence Apache, version 2.0 (les termes régulant cette licence se trouvent à l'adresse www.apache.org/licenses/LICENSE-2.0).

¹⁶Pour des renseignements complémentaires, un lecteur intéressé est invité à consulter l'adresse tortoisesvn.net, où il trouvera une présentation complète de ce logiciel placé sous licence GNU General Public License (voir www.gnu.org/licenses/gpl.html pour les termes et conditions régulant cette licence).

proposée d'un code, après avoir soumis cette dernière à une batterie de tests conséquente discutée et établie de manière collégiale, ou non, en fonction des protocoles de travail choisis. Avec pour objectif l'efficacité, il est aussi nécessaire que des réseaux de communication interconnectent les différents membres de l'équipe (pour faire remonter des bogues par exemple) et que chacun sache parfaitement quels sont sa place et son rôle dans l'organigramme de la cellule de travail, ainsi qu'à quel interlocuteur s'adresser en fonction de la situation à laquelle il est confronté. Afin de suivre une politique de *management* percutante et performante, la tenue de réunions régulières¹⁷, en assemblée plénière ou restreinte selon les thèmes discutés, doit être décidée, et comporter systématiquement un « ordre du jour » clair, concret, réalisable et facilement intégrable par l'ensemble des participants.

- Enfin, si le projet en cours, comme c'est le cas avec FullSWOF_2D, a pour aspiration d'être diffusé à une échelle plus conséquente que celle d'une petite équipe de travail (en terme d'effectif étant entendu), une réflexion doit être menée pour décider de la stratégie à adopter, stratégie devant marier harmonieusement visibilité du projet, confort de gestion pour les personnes en charge d'administrer ainsi que qualité et robustesse des outils à créer et des réseaux de communication à déployer, par souci, encore une fois, d'efficacité.

Cette phase de mon travail a dès le départ piqué ma curiosité et - ne le cachons pas - suscité mon vif intérêt. En effet, en plus de son caractère fortement professionnalisant et pédagogique, le travail en équipe constitue de nos jours un pilier incontournable du monde du travail, tout en étant indéniablement un exercice qui permet de développer son identité professionnelle dans le respect de la diversité constituée des autres membres du groupe. En ce sens, comme je l'évoquais plus haut, ce stage aura été pleinement satisfaisant, car en plus de me permettre de confronter les notions théoriques acquises sur les bancs de l'Université à un travail reposant sur la réalité de la pratique, il m'aura également donné l'opportunité de mûrir professionnellement et de nourrir ma réflexion quant à la place et au rôle de l'individu dans un groupe social et au sein d'un modèle organisationnel dans le contexte du monde professionnel, objectifs indissociables d'un stage de fin d'études, dont il serait, à mes yeux, réducteur de n'attribuer comme uniques vertus la consolidation et l'acquisition de connaissances et compétences d'ordre technique et théorique.

3.3.4 Adaptations apportées

Dans le cadre de ce stage, la problématique exigeait que j'effectue des modifications dans le code de FullSWOF_2D et que j'y implémente notamment de nouvelles conditions de bord : des conditions de bord périodiques. Afin de bien comprendre le bien-fondé de ce choix, il convient à présent d'explicitier clairement la démarche qui a été suivie au sein du projet ANR METHODE.

¹⁷Dans le cadre de ma collaboration avec cette équipe de travail, des réunions étaient tenues de manière hebdomadaire.

Démarche

Comme ceci a déjà été évoqué, un des objectifs du projet ANR METHODE est de formaliser une loi de comportement permettant de modéliser le ruissellement sur des surfaces agricoles. S'appuyant sur la richesse des points de vue des différents experts de ce pôle de travail, des réflexions ont été menées afin de proposer une réponse pratique à cette problématique.

Au cours des réunions s'étant tenues à ce propos, un consensus s'est dégagé entre les différents participants quant au *modus operandi* des simulations à mettre en œuvre : déposer à un temps $t = 0$ une nappe d'eau, de hauteur variable mais de vitesse nulle, sur une topographie, aux caractéristiques également variables, puis, une fois parvenu à un état stationnaire, observer la répartition du flux entre l'influence inhérente aux sillons agricoles et celle caractéristique de la pente générale.

Deux questions se sont alors posées :

1. Quel domaine de calcul considéré relativement aux expériences numériques à pratiquer ?
2. Comment décider si un état est stationnaire ou non ?

Conditions de bord périodiques

La question épineuse du domaine de calcul à considérer a longuement été débattue. En effet, dans un premier temps, l'idée envisagée fut de considérer un grand domaine de calcul et de ne prendre en compte à l'intérieur de celui-ci qu'une petite parcelle. Cependant, les limites de cette première approche furent rapidement pointées du doigt : elle se révélerait beaucoup trop « gourmande » en terme de temps de calcul pour être raisonnablement exploitable et envisageable à l'échelle de plusieurs milliers de simulations.

Il fut finalement décidé d'employer des conditions de bord périodiques, qui réinjecteraient les informations - hauteur ainsi que les deux composantes de la vitesse - relatives à l'eau sortant par le bord droit du maillage en son bord gauche, ainsi que celles sortant de son bord haut en son bord bas.

Une illustration de ces propos se trouve sur la figure 3.10, page 27.

La « condition- ϵ »

Afin de répondre à la seconde question posée, nous avons défini une « condition- ϵ », condition permettant d'évaluer l'amplitude des variations des grandeurs caractéristiques de l'eau (hauteurs, intensités des deux composantes de la vitesse) entre deux temps de calcul.

Pour cela, des calculs déterminent les normes L^2 des différences entre respectivement les hauteurs d'eau ainsi que les intensités des deux composantes de la vitesse entre deux temps successifs en les nœuds du maillage appartenant respectivement à la ligne et à la colonne médiane.

Si chacune des six normes calculées est inférieure à un certain réel $\epsilon > 0$ - fixé par l'utilisateur du programme - alors nous avons considéré que l'état était stationnaire. Dans ce

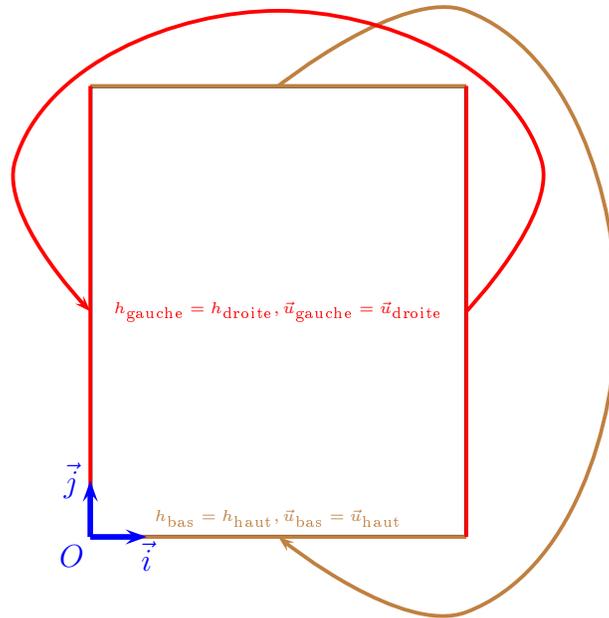


FIG. 3.10 – Les conditions de bord périodiques

cas-là, le programme s'arrête. Dans la cas contraire, il continue.

Une illustration de ces propos se trouve sur la figure 3.11, page 28.

3.4 Ga2Fu, script shell pour la modélisation paramétrique

3.4.1 Modèle structurel utilisé

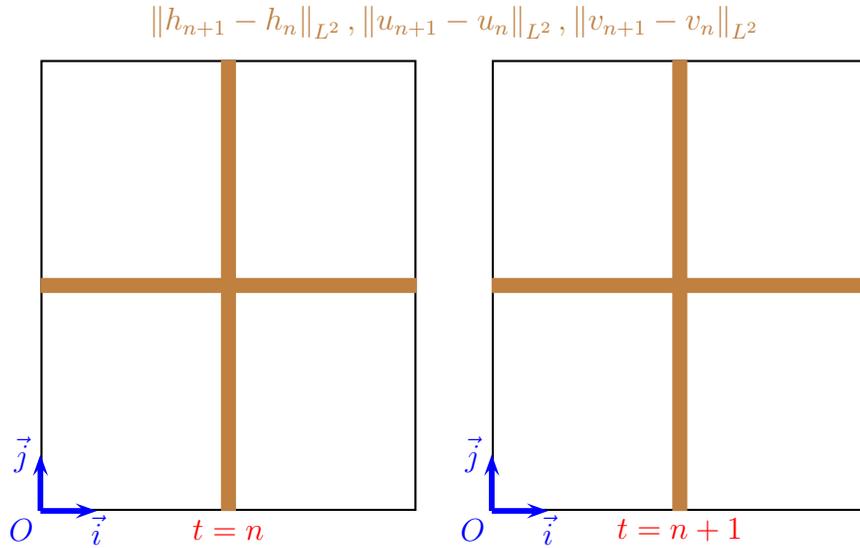
Pour répondre à la problématique posée dans le cadre de ce stage, il s'avère opportun d'écrire un script shell. En effet, afin de pouvoir mettre en place une méthodologie de calcul paramétrique qui permettra de déterminer la loi de comportement recherchée, un très grand nombre de simulations, correspondant aux différentes situations physiques que l'on rencontre dans la nature, sont à effectuer. Les paramètres d'entrée de ces dernières doivent ainsi pouvoir être choisis de manière commode et leurs résultats présentent l'impératif de devoir être facilement exploitables dans le cadre d'une étude statistique.

Le modèle structurel de ce script, écrit en bash et baptisé Ga2Fu, est décrit sur la figure 3.12, page 29.

Détaillons à présent les différentes étapes de son exécution :

Initialisation de la simulation (1) Cette étape est relative à la lecture dans un fichier de données - *simulations.txt*¹⁸ -, correspondant à une liste de simulations classées ligne par ligne, des variables inhérentes à une simulation donnée et nécessaires à l'utilisation des différents programmes qui vont être utilisés. Après cette lecture, le script crée

¹⁸Comme précédemment, les noms des fichiers écrits en *italique* indiquent qu'ils sont imposés.

FIG. 3.11 – La condition- ϵ

un sous-répertoire portant le nom que l'utilisateur a souhaité donner à l'expérience numérique qui va être effectuée, se place à l'intérieur de celui-ci et crée à nouveau la structure de sous-répertoires - Inputs et Outputs - impérative au bon fonctionnement du logiciel FullSWOF_2D. Notons qu'au lancement de l'exécution de Ga2Fu, seuls se trouvaient dans le répertoire de travail alloué par l'utilisateur à cette série de simulations le script ainsi que le fichier de données. Notons également que le script écrit un fichier - *configuration.txt* -, à l'intérieur du sous-répertoire de simulation, reprenant l'ensemble des paramètres qu'il a lu relativement à la simulation qu'il traite et dans lequel il commente explicitement à quoi ces derniers correspondent, permettant de la sorte un contrôle ultérieur de la part de l'utilisateur.

Topographie et conditions initiales relatives à l'eau (2) Après s'être placé à l'intérieur du sous-répertoire Inputs, à partir de la liste de paramètres lus, le script lance GARS et récupère sa sortie, caractérisant la surface souhaitée, dans un fichier (*topography.txt*). Dans un second temps, il lance le programme GenHUV. Ce petit utilitaire que j'ai écrit en C a pour vocation de créer un fichier contenant aux différents nœuds du maillage les hauteurs et intensités des vitesses d'une nappe d'eau recouvrant initialement la surface exploitée dans l'expérience numérique. La sortie de GenHUV est collectée au sein du fichier *huv.txt*.

Initialisation et lancement de FullSWOF_2D (3) Dans le sous-répertoire Inputs, le script crée cette fois-ci le fichier de paramètres impératif à l'utilisation du logiciel FullSWOF_2D, puis, quittant le sous-répertoire Inputs, il se place directement sous le répertoire portant le nom de la simulation en cours. Une fois ces opérations réussies, il lance FullSWOF_2D.

Post-traitement des résultats (4) Les calculs de FullSWOF_2D terminés, il convient à présent de traiter les résultats obtenus. Pour cela, deux programmes que j'ai écrits en C,

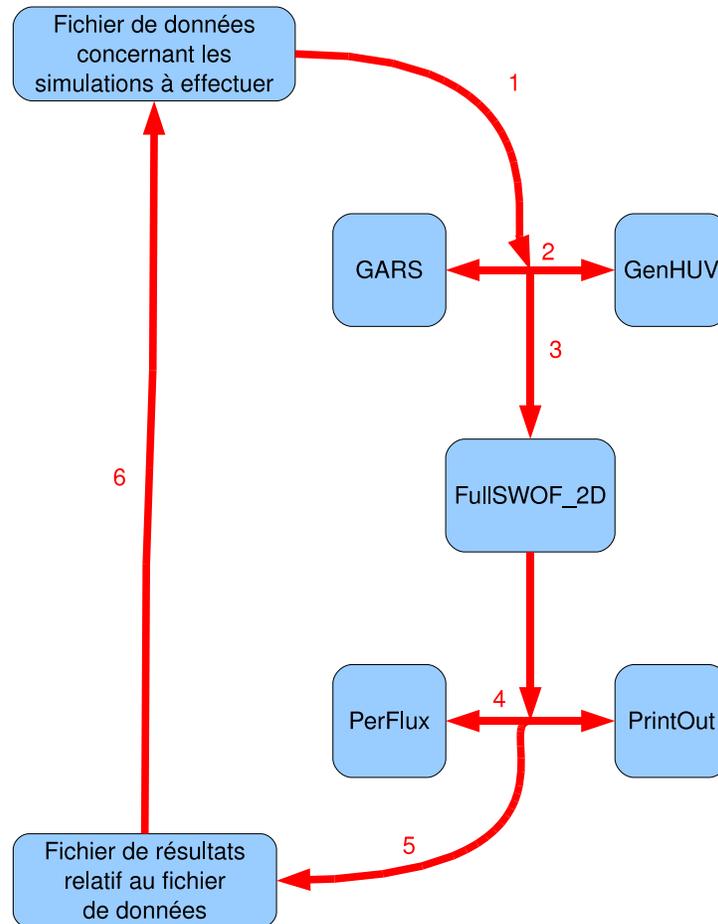


FIG. 3.12 – Modèle structurel du script Ga2Fu

PerFlux et PrintOut, entrent en scène. Après s'être placé dans le sous-répertoire Outputs, contenant les résultats de FullSWOF_2D, PerFlux crée un fichier - *perflux.dat* - contenant l'ensemble des informations en fin de simulation relatives au vecteur flux moyen (intensités relatives à ses deux composantes, intensité globale, direction, pourcentage d'influence dû à la pente générale, pourcentage d'influence dû aux sillons agricoles). Ensuite, le script lance PrintOut¹⁹, qui crée des « cartes » d'altitudes de surface, de hauteurs d'eau, d'altitudes globales, de nombres de Froude ainsi que d'intensités de vitesses aux nœuds du maillage du domaine considéré (*z.png*, *h.png*, *hz.png*, *fr.png* et *s.png*).

Bilan de la simulation (5) Suite à celà, le script récupère les informations se trouvant dans la sortie principale - *outcome.dat* - de FullSWOF_2D (temps de calcul, nombre d'itérations de l'algorithme, utilisation de la condition- ϵ ou non, nombre de Froude moyen²⁰ et pourcentage d'erreur relative concernant la conservation de la masse d'eau) ainsi que celles résultant de l'utilisation de PerFlux (décrites lors de l'étape précédente), se place dans le répertoire de travail, puis inscrit dans un fichier de résultats - *results.txt* - l'ensemble des paramètres et sorties de la simulation venant d'être traitée.

Fin de cycle et test de réinitialisation (6) Enfin, le script lit dans le fichier de données si une nouvelle simulation doit être effectuée. Si c'est le cas, il redémarre pour un cycle. Dans le cas contraire, il s'arrête.

L'arborescence de répertoires et fichiers créés par Ga2Fu se trouve schématisé sur la figure 3.13, page 31.

Remarque. Rappelons que le nombre de Froude, nombre adimensionnel baptisé ainsi en hommage à l'hydrodynamicien anglais William Froude, caractérise l'importance comparée des forces liées à la vitesse et à la force de pesanteur dans un fluide. En dimension 2, il est défini en chaque point du maillage $M_{i,j}$, avec $i \in \{1, \dots, nbc\}$ et $j \in \{1, \dots, nbrow\}$, par la relation suivante :

$$Fr_{i,j} = \frac{\sqrt{u_{i,j}^2 + v_{i,j}^2}}{\sqrt{gh_{i,j}}}$$

où on a noté :

- $u_{i,j}$ et $v_{i,j}$, exprimées en m.s^{-1} , les intensités des composantes du vecteur vitesse de l'eau en $M_{i,j}$,
- g , exprimée en m.s^{-2} , l'accélération de la pesanteur ($g \simeq 9,81\text{m.s}^{-2}$),
- $h_{i,j}$, exprimée en m, la hauteur d'eau en $M_{i,j}$.

Notons que respectivement à l'écoulement de l'eau si :

- $Fr < 1$, celui-ci est dit fluvial, caractérisé par le fait que le mouvement est déterminé par le comportement à l'aval et à l'amont,

¹⁹Ce programme repose sur le logiciel Gnuplot, dont un lecteur intéressé trouvera une présentation complète à l'adresse www.gnuplot.info et dont les termes réglementant la licence sont exposés à l'adresse gnuplot.cvs.sourceforge.net/gnuplot/gnuplot/Copyright?view=markup.

²⁰Le nombre de Froude calculé est celui obtenu si on considère le domaine ramené à un point

```
Répertoire_de_travail\  
results.txt  
Répertoire_Simulation_1\  
configuration.txt  
Inputs\  
  parameters.txt  
  topography.txt  
  huv.txt  
Outputs\  
  parameters.dat  
  init.dat  
  Ic.dat  
  huv_movie.dat  
  bilan_vol.dat  
  flux.dat  
Results.dat  
outcome.dat  
perflux.dat  
z.png  
h.png  
hz.png  
fr.png  
s.png  
Répertoire_Simulation_2\  
...
```

FIG. 3.13 – Arborescence générée par Ga2Fu

- $Fr > 1$, celui-ci est dit torrentiel, regime qui est lui caractérisé par le fait que relativement à son mouvement, la masse du fluide devient négligeable au regard des forces qui le meuvent (la plupart du temps la gravité), seul le comportement à l'amont ayant une incidence,
- $Fr = 1$, celui-ci est dit critique,
- $Fr > 2$, les conditions sont optimales pour la formation du phénomène physique appelé *roll-waves*, qui sont des ondes de propagation de l'eau périodiques et discontinues.

Remarque. Concernant le fonctionnement du programme PerFlux, remarquons qu'après avoir déterminé le vecteur flux moyen ainsi que son intensité et celle de ses composantes de manière classique, il calcule quant à celui-ci les pourcentages d'influence dus respectivement à la pente générale et aux sillons agricoles. Pour cela, le programme part du principe que la direction des sillons agricoles a été fixée par un angle β tel que $\beta = 90^\circ$, ce qui sera le cas pour chaque simulation effectuée dans le cadre de ce stage. L'illustration de cette situation est décrite par le dessin de la figure 3.14, page 32.

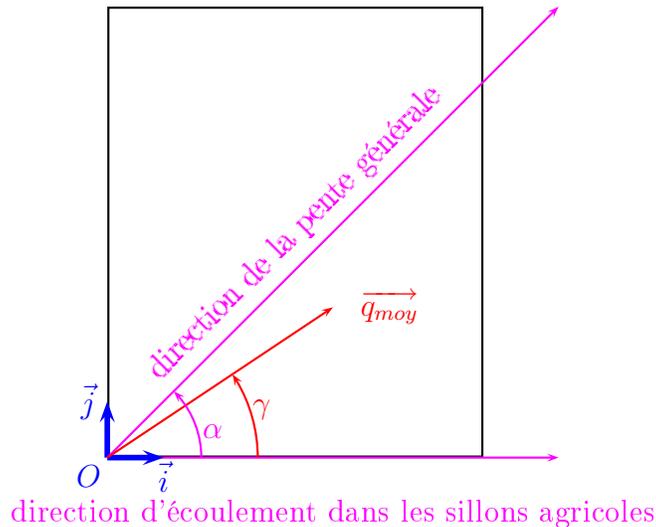


FIG. 3.14 – Contexte situationnel de travail du programme PerFlux

Nous avons alors défini le pourcentage dû à la pente générale comme étant égal à $100 \times \frac{\gamma}{\alpha}$ tandis que celui dû aux sillons agricoles comme égal, par conséquent, à $100 \times \frac{1-\gamma}{\alpha}$ (lorsque $\alpha = 0^\circ$, ces deux pourcentages sont arbitrairement fixés à 100%).

3.4.2 Moyens de calcul

Les simulations réalisées dans le cadre de ce stage sont le produit de deux moyens de calcul différents.

Tout d'abord, en ses locaux du centre INRA d'Orléans, l'unité de recherche de science du sol a mis à ma disposition un ordinateur portable possédant les caractéristiques suivantes :

- fabricant : Dell ;
- système d'exploitation : Microsoft Windows XP Professionnel, Version 2002, Service Pack 3 ;
- processeur : Intel[®] Core[™] 2 Duo CPU, P8700 2,53 GHz ;
- mémoire : 3,48 Go de RAM, DDR2 400 MHz, Dual Channels.

Ce premier moyen de calcul, en raison de sa quasi-impossibilité à gérer une série de simulations autrement que séquentiellement (l'utilisation des deux cœurs permet seulement de scinder une liste en deux), a essentiellement servi pour les phases de test (mise au point des divers programmes utilisés ainsi que du script).

En complément, le MAPMO m'a permis d'utiliser l'un de ses pôles de calcul. Ce dernier, regroupant 22 ordinateurs de caractéristiques distinctes tout en ayant néanmoins en commun le fait de posséder un système d'exploitation de type Linux ainsi que d'être munis d'un processeur Intel, est exploitable par un utilisateur à travers le logiciel Condor²¹. Développé au sein de l'université américaine du Wisconsin-Madison depuis maintenant près de quinze ans, il est placé sous licence Apache (Version 2.0)²² et est de nos jours déployé sur plusieurs centaines d'entreprises, instituts gouvernementaux et universités à travers le monde. Se pilotant en ligne de commande de manière conviviale et caractérisé par des règles d'utilisation aisément appropriables, Condor, face à une liste de tâches à effectuer, dispatche ces dernières sur autant d'ordinateurs faisant partis du pôle de calcul, exploitant de cette manière les ressources à sa disposition au maximum de ses possibilités.

Ce second moyen de calcul, réduisant les temps de calcul, est celui servant pour les phases de calcul intensif, lorsqu'un très grand nombre de simulations doivent être effectuées. Notons qu'il est de plus bien adapté à notre demande de calcul paramétrique, les simulations étant indépendantes les unes des autres.

3.4.3 Exemple d'utilisation

Afin d'illustrer le fonctionnement du script Ga2Fu, considérons une simulation portant sur la surface présentée sur la figure 3.7, page 18, sur laquelle on dépose une nappe d'eau de 5 cm, initialement à vitesse nulle. Le temps maximal de simulation est fixé à 60 s et les conditions de bord choisies sont les conditions de bord périodiques. Pour cette expérience numérique, on utilise concernant le frottement la loi de Darcy-Weisbach avec pour coefficient 0,1, ainsi qu'une condition d'arrêt basé sur $\epsilon = 10^{-5}$ (les autres paramètres sont choisis de manière « standard »).

Les différents fichiers et graphiques exploités et produits par le script dans le cadre de cette simulation sont présentés sur les figures 3.15, page 35, 3.16, page 35, 3.17, page 36,

²¹Pour des informations complémentaires à celles dispensées dans ce paragraphe, le lecteur est invité à consulter le site www.cs.wisc.edu/condor.

²²Plus précisément, il s'agit de la version 2.0 de cette licence (les termes régulant cette dernière se trouvent à l'adresse <http://www.apache.org/licenses/LICENSE-2.0.html>).

3.18, page 36, 3.19, page 37, 3.20, page 37, 3.21, page 38, 3.22, page 38 et 3.23, page 39. On remarquera que la condition- ϵ n'a pas été utilisée comme condition d'arrêt au programme.

```

#####
# SIMULATIONS PARAMETERS LIST #
#####
#
# name nbc col nbrow dx dy slope slope_azimut wave_amplitude wave_length wave_azimut
wave_origin offc h u v time dt lbc rbc bbc tbc friction_law friction_coef numerical_flux
order reconstruction limiter amortrec modifENO infiltration ks dteta psi zcrust kssoil
cfl epsilon topo_choice topo_name huv_choice huv_name rain_choice rain_name
#
simulation 201 201 0.005 0.005 2. 30. 0.01 0.2 90. 0. 1 0.05 0. 0. 60. 0.01 4 4 4 4 2 0.1
2 2 1 1 0.25 0.9 1 0. 0.139 1.3795 0.05 2.15e-5 0.4 1.e-5 1 topography.txt 1 huv.txt 2
Nothing

```

FIG. 3.15 – Fichier de données fourni par l'utilisateur

```

#####
# RESULTS OF THE SIMULATIONS LIST #
#####
#
# name nbc col nbrow dx dy slope slope_azimut wave_amplitude wave_length wave_azimut
wave_origin offc h u v time dt lbc rbc bbc tbc friction_law friction_coef numerical_flux
order reconstruction limiter amortrec modifENO infiltration ks dteta psi zcrust kssoil
cfl epsilon topo_choice topo_name huv_choice huv_name rain_choice rain_name time_seconds
time_minutes time_hours nb_iterations epsilon_cond froude_number percentage_water_error
mean_horizontal_flux mean_vertical_flux mean_flux mean_flux_direction percentage_furrows
percentage_slope
#
simulation 201 201 0.005 0.005 2. 30. 0.01 0.2 90. 0. 1 0.05 0. 0. 60. 0.01 4 4 4 4 2 0.1
2 2 1 1 0.25 0.9 1 0. 0.139 1.3795 0.05 2.15e-5 0.4 1.e-5 1 topography.txt 1 huv.txt 2
Nothing 8337 138.95 2.31583 49051 0 1.21575 0.0865175 0.037059 0.019466 0.041861
27.710950 7.630168 92.369832

```

FIG. 3.16 – Fichier de résultats écrit par le script

```

SUMMARY :
-----
parameter J=201
parameter K=201
parameter dimt=60
parameter dx=0.005
parameter dy=0.005
parameter dt=0.01
parameter tx=2
parameter ty=2
parameter Ks=0
parameter dteta=0.139
parameter Psi=1.3795
parameter zcrust=0.05
parameter c=0.1
parameter Kssoil=2.15e-05
parameter cflfix=0.4
parameter amortec=0.25
parameter modifeno=0.9
parameter epsilon=1e-05
suffix for the 'Outputs' directory=
infiltration model = green-ampt
left condition = periodic
right condition = periodic
down condition = periodic
up condition = periodic
flux condition = HLL
parameter order=2
friction condition = Darcy_Weisbach
reconstruction condition = MUSCL
limitation condition = Minmod
huv initial condition = file reading
rain initial condition = function generated
topography initial condition = file reading
entries ok!
-----
Schema construction is successful!
Plan chosen!
Calculation time : 2 h 18 m 57 s
- In seconds : 8337
- In minutes : 138.95
- In hours : 2.31583
Number of iterations in the algorithm : 49051
Use of the epsilon-condition to stop (0:No 1:Yes) : 0
Froude number : 1.21575
Relative error relatively to the water quantity conservation : 0.0865175
Computation ok!
Destructor ok!

```

FIG. 3.17 – Fichier de sortie principale de FullSWOF_2D

```

Outcome
=====
Mean horizontal flux (in mÅ²/s) : 0.037059
Mean vertical flux (in mÅ²/s) : 0.019466
Mean flux (in mÅ²/s) : 0.041861
Mean flux direction (in Å°) : 27.710950
Percentage of the mean flux which follows the furrows direction : 7.630168
Percentage of the mean flux which follows the general slope direction : 92.369832

```

FIG. 3.18 – Fichier de sortie de PerFlux

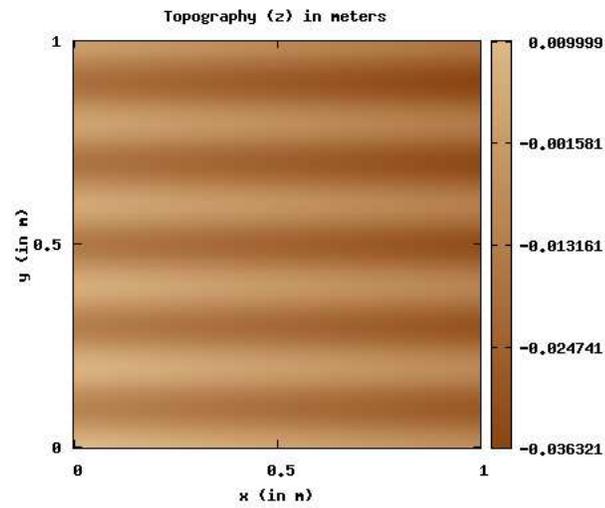
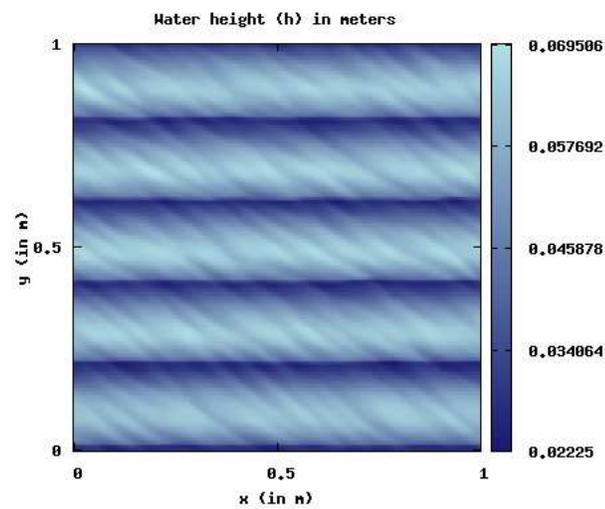
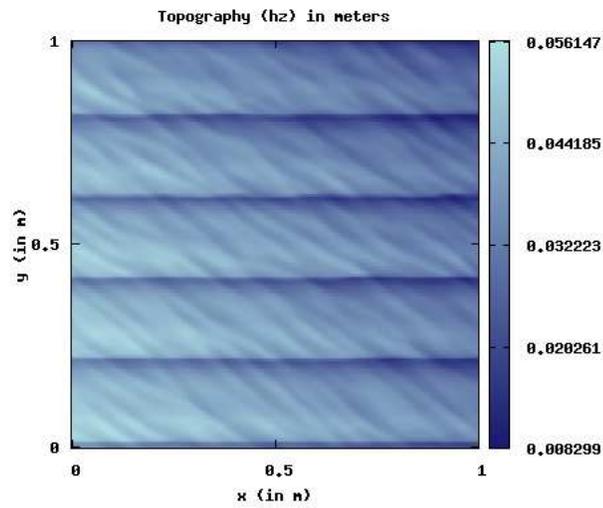
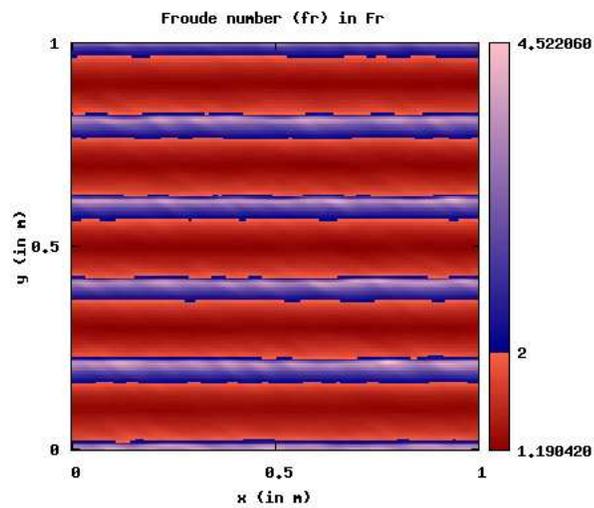


FIG. 3.19 – Carte des altitudes de la surface du sol

FIG. 3.20 – Carte des hauteurs d'eau au temps $t = 60s$

FIG. 3.21 – Carte des altitudes globales au temps $t = 60s$ FIG. 3.22 – Carte des nombres de Froude au temps $t = 60s$

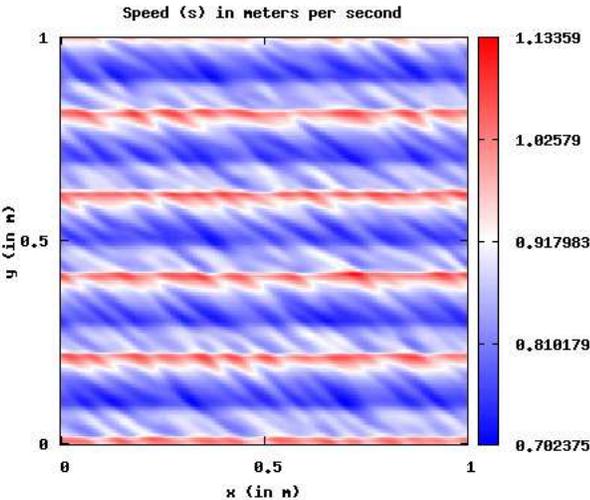


FIG. 3.23 – Carte des intensités des vitesses au temps $t = 60s$

Partie 4

Résultats

Sommaire

4.1 Démarche suivie	40
4.1.1 Première étape : Réduction des temps de calcul numérique	40
4.1.2 Seconde étape : Calcul paramétrique	42
4.2 Premiers résultats	43
4.2.1 Première étape : Réduction des temps de calcul numérique	44
4.2.2 Seconde étape : Calcul paramétrique	46

4.1 Démarche suivie

Afin de répondre à la problématique de ce stage, nous allons déterminer une méthodologie de calcul paramétrique devant conduire ultérieurement à la formalisation d'une loi de comportement modélisant le comportement du ruissellement sur des parcelles agricoles, en fonction des composantes caractérisant de telles surfaces, et ce, en comparaison avec les conditions initiales régissant la situation de l'eau sur le domaine donné.

A cet effet, deux étapes ont été définies.

4.1.1 Première étape : Réduction des temps de calcul numérique

Afin de répondre à l'objectif de travail fixé, un grand nombre de simulations numériques, conformes à des situations physiques réelles, vont être effectuées.

Au cours d'une première étape, il va donc falloir déterminer les conditions limites permettant de réduire les calculs numériques au maximum. Pour cela, trois cas-tests de surfaces données ont été mis en place : un premier cas-test correspondant à des conditions intermédiaires relativement à la réalité du terrain, un deuxième cas-test correspondant à des conditions extrêmes favorisant l'influence de la pente générale quant à la direction du ruissellement, et enfin un troisième cas-test correspondant à des conditions extrêmes favorisant l'influence des sillons agricoles quant à la direction de l'écoulement de l'eau.

Taille du domaine

Dans un premier temps, il s'agit de minimiser la taille du domaine, choisi arbitrairement de forme carrée et considéré en terme du nombre de sillons agricoles qu'il contient. Pour cela, les protocoles numériques suivis sont les suivants :

Cas intermédiaire			
Sillons	Longueur d'onde		20 cm
	Amplitude		1 cm
Pente	Amplitude		2%
	Direction		30°
Frottement	Darcy-Weisbach		0,1
Hauteur d'eau	Amplitude	pourcentages de 50% pour la taille maximale	
Domaine	Longueur	Bornes	1 à 20 sillons
		Pas	1 sillon
		Résolution	0,5 cm
Cas extrême favorisant la pente générale			
Sillons	Longueur d'onde		80 cm
	Amplitude		0,25 cm
Pente	Amplitude		22%
	Direction		80°
Frottement	Darcy-Weisbach		0,01
Hauteur d'eau	Amplitude	pourcentages de 50% pour la taille maximale	
Domaine	Longueur	Bornes	1 à 20 sillons
		Pas	1 sillon
		Résolution	0,5 cm
Cas extrême favorisant les sillons agricoles			
Sillons	Longueur d'onde		10 cm
	Amplitude		10 cm
Pente	Amplitude		1%
	Direction		10°
Frottement	Darcy-Weisbach		100
Hauteur d'eau	Amplitude	pourcentages de 50% pour la taille maximale	
Domaine	Longueur	Bornes	1 à 20 sillons
		Pas	1 sillon
		Résolution	0,5 cm

Pour chacun des trois cas de figure envisagés, on déterminera le nombre de sillons minimal permettant la conservation du comportement global du ruissellement envisagé sur le domaine de taille maximale. Une fois ceci fait, on fixera comme taille du domaine en terme de sillons agricoles le maximum des trois minimums obtenus.

Résolution des sillons agricoles

Dans un second temps, il s'agit de minimiser la résolution des sillons agricoles, en considérant donc comme taille de domaine celle déterminée au cours du temps précédent. Pour cela, les protocoles numériques suivis sont les suivants :

Cas intermédiaire		
Sillons	Longueur d'onde	20 cm
	Amplitude	1 cm
Pente	Amplitude	2%
	Direction	30°
Frottement	Darcy-Weisbach	0,1
Hauteur d'eau	Amplitude	pourcentages de 50% pour la résolution maximale
Domaine	Longueur	fixée à l'étape « Taille du domaine »
	Résolution	5 à 50 mailles par longueur d'onde de sillon
Cas extrême favorisant la pente générale		
Sillons	Longueur d'onde	80 cm
	Amplitude	0,25 cm
Pente	Amplitude	22%
	Direction	80°
Frottement	Darcy-Weisbach	0,01
Hauteur d'eau	Amplitude	pourcentages de 50% pour la résolution maximale
Domaine	Longueur	fixée à l'étape « Taille du domaine »
	Résolution	5 à 50 mailles par longueur d'onde de sillon
Cas extrême favorisant les sillons agricoles		
Sillons	Longueur d'onde	10 cm
	Amplitude	10 cm
Pente	Amplitude	1%
	Direction	10°
Frottement	Darcy-Weisbach	100
Hauteur d'eau	Amplitude	pourcentages de 50% pour la résolution maximale
Domaine	Longueur	fixée à l'étape « Taille du domaine »
	Résolution	5 à 50 mailles par longueur d'onde de sillon

Pour chacun des trois cas de figure envisagés, on déterminera la résolution minimale des sillons permettant la conservation du comportement global de l'écoulement de l'eau envisagé pour la résolution maximale. Une fois ceci fait, on fixera comme résolution des sillons agricoles le maximum des trois minimums obtenus.

4.1.2 Seconde étape : Calcul paramétrique

Une fois la première étape terminée, il va falloir au cours d'une seconde étape définir une batterie de simulations à effectuer avec pour objectif de conduire à la loi de comportement à formaliser.

Encore une fois, cette étape va être structurée en deux temps.

Forme générale

Dans un premier temps, il convient d'établir une gamme d'expériences numériques caractéristiques et pertinentes de la réalité du terrain. Afin de répondre à cette attente, la série de simulations suivantes, visant à échantillonner le domaine de façon à identifier les zones sensibles, a été établie :

Sillons	Longueur d'onde	Bornes	10 cm à 80 cm
		Pas	+ 5 cm
	Amplitude	Bornes	0,25 cm à 10,25 cm
		Pas	+ 2 cm
Pente	Amplitude	Bornes	1 à 22%
		Pas	+ 3%
	Direction	Bornes	0 à 90°
		Pas	+ 10°
Frottement	Darcy-Weisbach	Bornes	0,01 à 100
		Pas	x 10
Hauteur d'eau	Amplitude	Bornes	0,1 cm à 10,1 cm
		Pas	+ 1 cm

Une fois les résultats des différentes simulations présentes dans cette série obtenus, ultérieurement, une étude statistique devra déterminer les influences respectives des différents paramètres en jeu quant à la répartition des pourcentages du débit respectivement pour la pente générale et pour les sillons agricoles.

Raffinement

Enfin, dans un second temps, afin de compléter l'étude statistique précédente, on augmentera la densité d'échantillonnage dans les zones où la forme générale change (courbures) tout comme les bornes, afin de couvrir les cas extrêmes (notamment concernant la hauteur d'eau).

Une fois ce travail effectué, on affinera la loi de comportement trouvée au cours du premier temps.

4.2 Premiers résultats

Dans cette section, nous allons exposer les premiers résultats obtenus. Nous ne considérerons ici que les simulations relatives au cas intermédiaire que nous avons défini lors de la section précédente.

4.2.1 Première étape : Réduction des temps de calcul numérique

Taille du domaine

Comme ceci a déjà été précisé précédemment, cette phase a pour vocation de réduire la taille du domaine considéré, lorsque ce dernier est considéré en terme de nombre de sillons agricoles.

En suivant le protocole établi, nous déterminons que pour une taille de domaine maximale (c'est à dire définie par 20 sillons), une nappe d'eau de hauteur 0,02 m permet d'obtenir un pourcentage de répartition du flux en terme d'influence dûe respectivement à la pente générale et aux sillons agricoles proche de 50% dans chacun des deux cas (tout du moins compris dans une gamme allant de 40 à 60%).

L'évolution de ces pourcentages en fonction du nombre de sillons agricoles se trouve représentée sur la figure 4.1, page 45.

L'objectif est ici de réduire au maximum la taille du domaine tout en conservant un comportement globalement identique à celui obtenu sur un domaine de grande taille. Au regard de ce graphique, un choix raisonnable peut être de choisir comme taille minimale celle sous-jacente à l'emploi de 10 sillons agricoles. En effet, d'une part, on peut observer sur la figure que pour une taille de domaine définie approximativement entre 10 et 20 sillons agricoles, le comportement du ruissellement semble être stabilisé, alors qu'il paraît varier de manière plus importante si on réduit la taille du domaine en dessous de 10 sillons agricoles. D'autre part, le pourcentage d'erreur relative dans ce cas-là respectivement aux pourcentages de répartition du flux est de 1,99%, ce qui est inférieur à la barre de 2% que nous nous étions fixés au préalable.

Il est également intéressant d'observer les temps de calcul nécessaires pour chacune des 20 simulations qui ont été effectuées. Ces derniers sont représentés sur la figure 4.2, page 45.

Ce graphique a deux principales vertus. D'une part, il nous permet d'observer les disparités de performance parmi les ordinateurs appartenant à la plateforme de calcul utilisée. D'autre part, il nous donne la possibilité d'estimer les temps de calcul nécessaires concernant la palette de paramètres utilisés, donnée indispensable à connaître avant de lancer des simulations numériques à « grande échelle ». Ainsi, on peut observer que pour les paramètres utilisés ainsi que le choix d'une taille de domaine définie par 10 sillons agricoles, le temps de simulation est approximativement de cinq heures.

Résolution des sillons agricoles

Lors de cette seconde phase, l'objectif est de réduire la résolution des sillons agricoles, en considérant donc comme taille de domaine celle caractérisée par 10 sillons agricoles. Pour la gamme de résolutions définie par le protocole, nous allons donc nous intéresser à l'évolution des pourcentages de répartition du flux en fonction de des différentes résolutions de cette gamme.

Cette information est visualisable sur la figure 4.3, page 47.

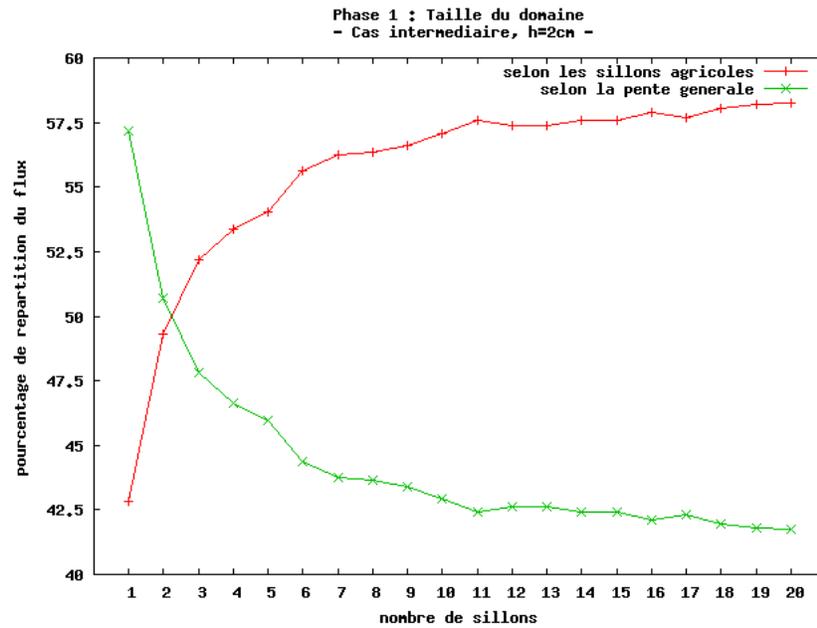


FIG. 4.1 – Réduction de la taille du domaine

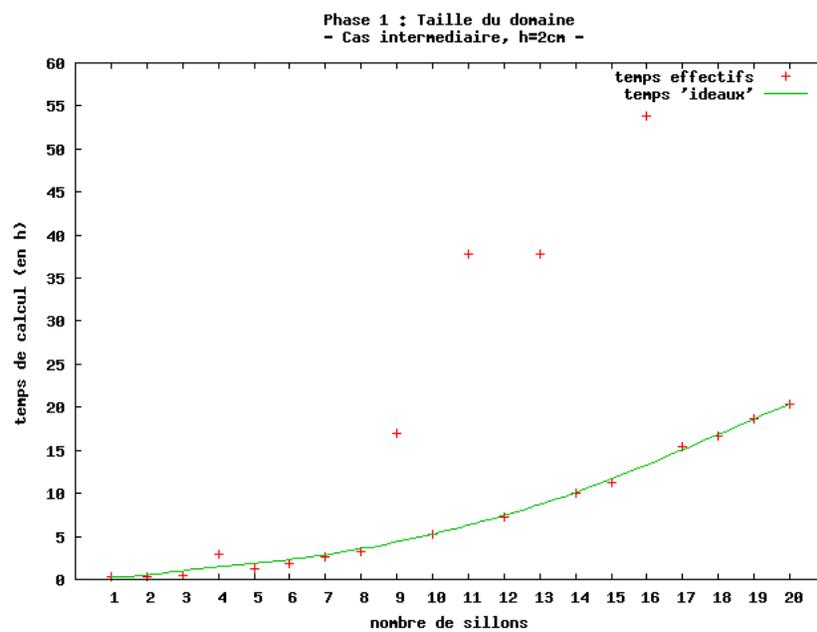


FIG. 4.2 – Temps de calcul correspondant

Sur cette figure, on peut observer que le comportement de l'écoulement semble être stabilisé jusqu'à une réduction de la résolution relative à la description d'un sillon agricole par 41 nœuds, alors que ses variations sont plus conséquentes si la réduction opérée est plus importante. De plus, ce choix est conforme à notre exigence d'une barre de 2% quant à l'erreur relative respective aux pourcentages de répartition du flux puisque dans ce cas-là, l'erreur relative considérée est de 0,58%.

Pour les mêmes raisons que celles évoquées lors du traitement de la phase précédente, il peut être intéressant de visualiser les temps de calcul obtenus au cours des simulations numériques effectuées. Ces derniers sont représentés sur la figure 4.4, page 47. Cette figure nous permet de prendre note que pour les paramètres de simulation considérés ainsi que les conditions de réduction des temps de calcul numérique définies, le temps de simulation numérique à prévoir est approximativement de cinq heures.

4.2.2 Seconde étape : Calcul paramétrique

À présent que l'étape correspondant à la réduction des temps de calcul a été effectuée, nous allons lancer une série de simulations basée sur les conditions de réduction que nous venons de déterminer.

Le jeu de 552 expériences numériques considéré distingue chaque simulation deux à deux par le biais de quatre paramètres :

1. le pourcentage de dénivellation de la pente générale : $p = 1, 2$ ou 5% ;
2. la direction de la pente générale : $\alpha = 0, 10, 20, 30, 40, 50, 60, 70, 80$ ou 90° ;
3. l'amplitude des sillons agricoles : $a = 0,0025; 0,005; 0,0065; 0,008; 0,01$ ou $0,02$ m ;
4. la longueur d'onde des sillons agricoles : $T = 0,15; 0,20$ ou $0,25$ m.

Notons également que la hauteur de la nappe d'eau recouvrant initialement la topographie est fixée à 0,01 m.

Avec pour objectif de déterminer un ordre d'influence de ces quatre paramètres au regard de l'écoulement de l'eau et pour cette gamme de paramètres de simulation, nous allons représenter deux figures.

Considérons la première d'entre elles, répertoriée figure 4.5, page 48.

Pour cette gamme de paramètres de simulation, cette première figure permet de visualiser l'influence comparée de l'amplitude des sillons agricoles et de la direction de la pente générale quant au comportement de l'écoulement de l'eau. Deux remarques peuvent être tirées de son analyse :

1. Tout d'abord, il semblerait que l'amplitude des sillons agricoles soit plus influente que la direction de la pente générale. En effet, on peut observer que pour une direction donnée, la variation de l'amplitude permet de couvrir l'ensemble des pourcentages témoignant du ruissellement selon la direction des sillons agricoles, ce qui n'est pas vrai dans le cas contraire : Pour une amplitude donnée, la variation de la direction ne permet pas de couvrir l'ensemble de ces pourcentages.

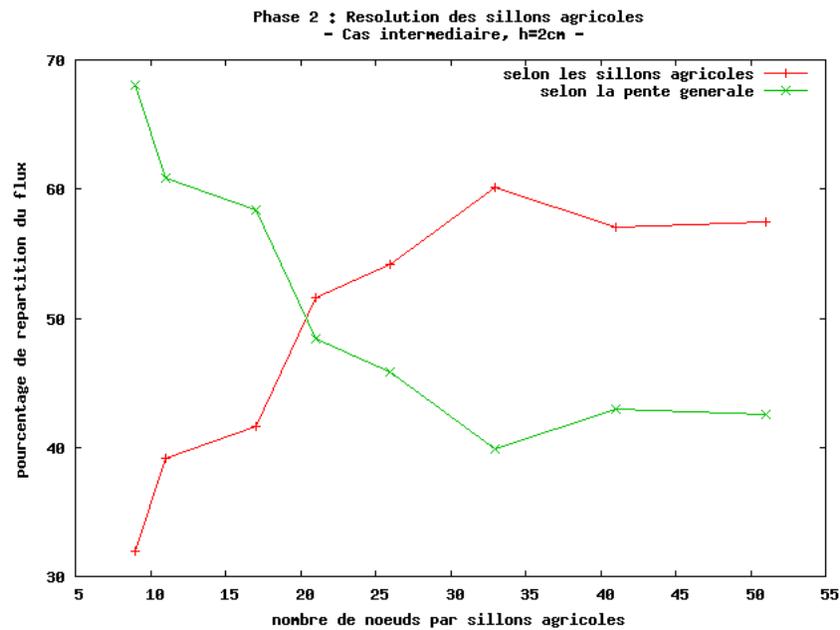


FIG. 4.3 – Réduction de la résolution des sillons agricoles

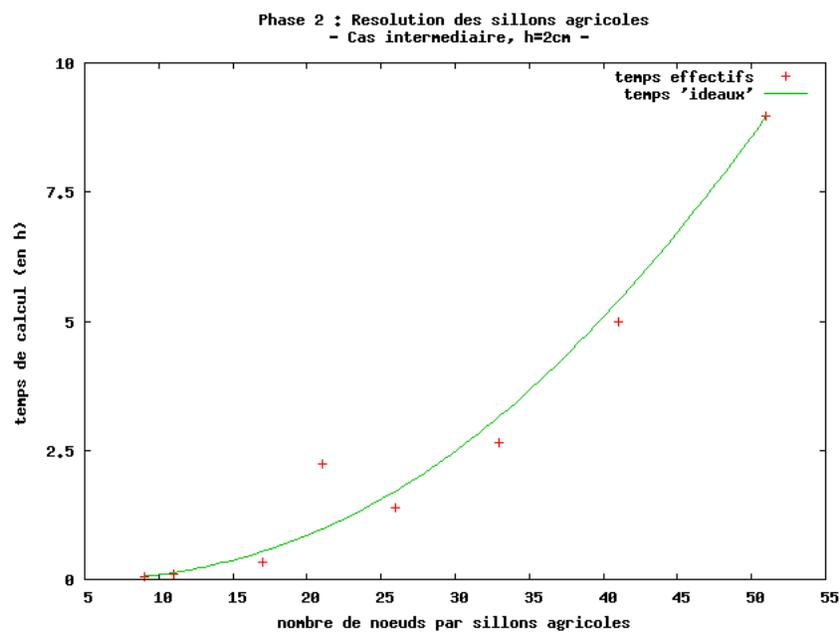


FIG. 4.4 – Temps de calcul correspondant

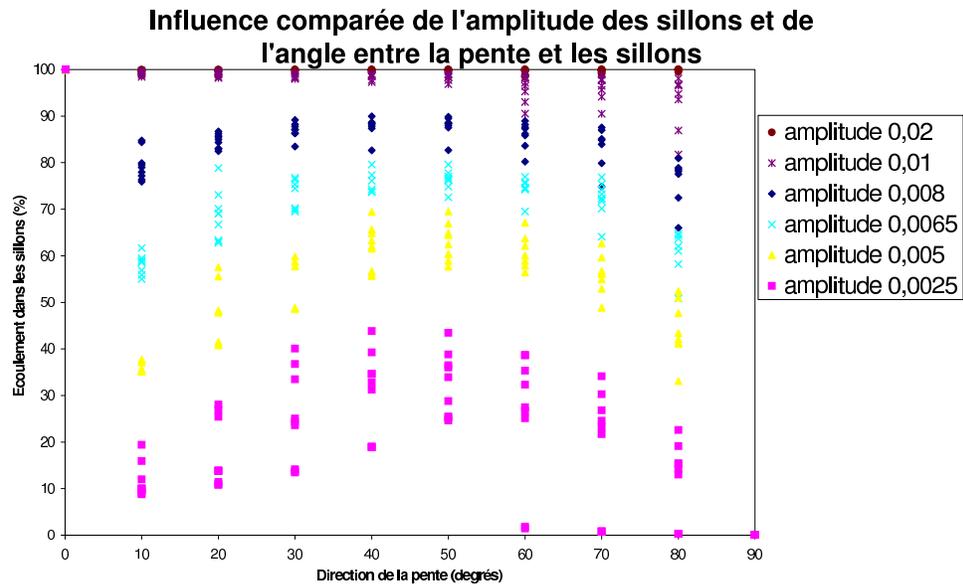


FIG. 4.5 – Premier tri des données

- Si on fixe une amplitude de sillons agricoles et si on fait varier la direction de pente générale, l'étendue des pourcentages d'écoulement de l'eau selon les sillons agricoles semble être pratiquement constante. De même, ce phénomène semble se reproduire de nouveau si on fixe une direction de pente générale et si on fait varier l'amplitude des sillons agricoles (tout du moins pour des amplitudes qui ne sont pas suffisamment élevées afin d'orienter l'ensemble du ruissellement selon la direction des sillons agricoles). Cet état de fait laisse supposer que les influences du pourcentage de dénivellation de la pente générale ainsi que de la longueur d'onde des sillons agricoles sont négligeables au regard de celles de l'amplitude des sillons agricoles ainsi que de la direction de la pente générale.

Après l'analyse de cette première figure, pour la gamme de paramètres de simulation considérée, il semble donc que l'ordre d'influence sur l'écoulement de l'eau des quatre paramètres testés place en première position l'amplitude des sillons agricoles, en deuxième position la direction de la pente générale, et en troisième et quatrième positions le pourcentage de dénivellation de la pente générale et la longueur d'onde des sillons agricoles, ces deux derniers paramètres n'ayant pas encore été classés l'un par rapport à l'autre.

La figure 4.6, page 49, se propose de nous permettre d'affiner notre analyse.

Pour la gamme de paramètres de simulation employée, cette seconde figure, pour une direction de pente générale fixée à 50° , représente l'influence comparée du pourcentage de dénivellation de la pente générale ainsi que de la longueur d'onde et de l'amplitude des sillons agricoles quant au comportement du ruissellement. Afin de classer les deux derniers paramètres testés par cette étude, observons ce qui se produit lorsque l'on fixe l'amplitude des sillons agricoles. Encore une fois, deux remarques peuvent être tirées de l'analyse qui est

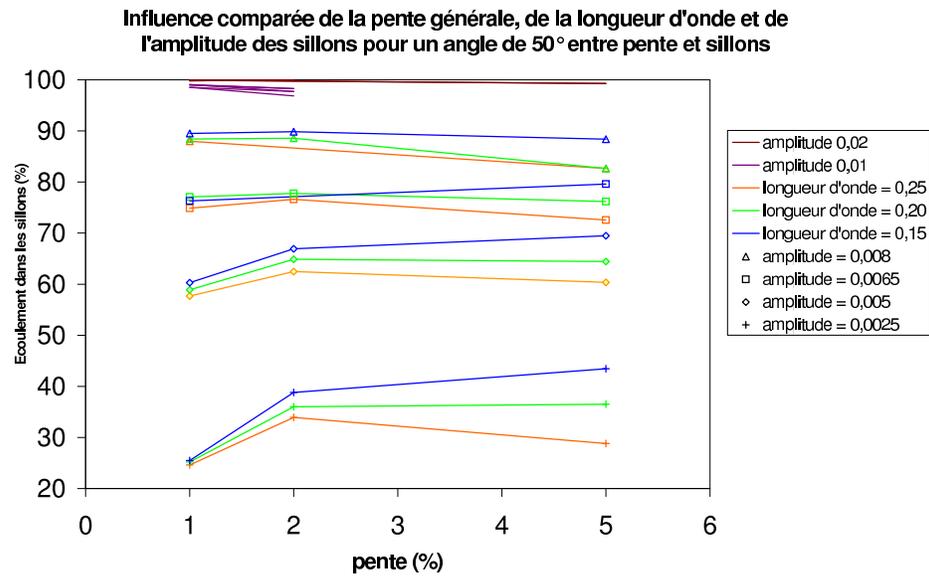


FIG. 4.6 – Second tri des données

faite :

1. La première observation qui peut être formulée est que pour une amplitude de sillons agricoles fixée, plus le pourcentage de dénivellation de la pente générale augmente, plus l'influence de la longueur d'onde des sillons agricoles se fait sentir quant au comportement de l'écoulement de l'eau. Il existe donc très certainement une corrélation entre ces deux paramètres.
2. La seconde observation concerne le fait que plus l'amplitude des sillons agricoles augmente, moins l'influence du pourcentage de dénivellation de la pente générale se fait ressentir, ce qui est également vrai concernant l'influence de la longueur d'onde des sillons agricoles, mais dans une moindre mesure. Cette remarque nous permet encore une fois de conjecturer que les paramètres considérés sont corrélés entre eux et que l'influence de la longueur d'onde des sillons agricoles semble être prépondérante au regard de celle du pourcentage de dénivellation de la pente générale.

Après l'analyse de cette seconde figure, pour la gamme de paramètres de simulation considérée, on a donc mis en lumière qu'il semblerait que l'ordre d'influence des quatre paramètres testés quant au comportement du ruissellement soit le suivant :

1. l'amplitude des sillons agricoles,
2. la direction de la pente générale,
3. la longueur d'onde des sillons agricoles,
4. le pourcentage de dénivellation de la pente générale.

Cette analyse nous a également permis de conjecturer que ces paramètres semblent être corrélés entre eux.

Partie 5

Conclusion et perspectives

Dans le cadre de ce stage, une méthodologie permettant d'établir une loi de comportement prévoyant la direction de l'écoulement de l'eau sur une surface agricole a été proposée. Cette dernière prend en considération de manière exhaustive l'ensemble des paramètres décrivant la topographie, à l'exception de la rugosité aléatoire due aux mottes de terre, et préconise l'emploi du calcul paramétrique sur « toute » la palette des situations physiques que l'on rencontre en pratique. Suivi par une étude statistique approfondie, ce dernier devrait pouvoir conduire à la loi que l'on désire formaliser.

Afin de parvenir à cet objectif, un important travail de programmation a été effectué et a notamment permis de coupler par la biais d'un script deux logiciels développés au MAPMO : GARS et FullSWOF_2D. Ce script, Ga2Fu, a été écrit dans le but de s'accorder au mieux à la méthodologie énoncée. Il permet ainsi une saisie commode de l'ensemble des paramètres exhaustifs décrivant les topographies considérées. Il génère également un fichier de résultats facilement exploitable dans le cadre d'une étude statistique. Enfin, il a été conçu dans l'optique du calcul paramétrique et a donc été adapté à la plateforme de calcul sur laquelle il a vocation d'être utilisé.

Après avoir décrit explicitement le protocole concernant les simulations numériques à effectuer, nous avons lancé en cette fin de stage une série de simulations répondant à certains critères, choisis de manière arbitraire, et ne testant que certains paramètres, choisis de manière tout aussi arbitraire. En effet, cette série avait comme unique objectif de nous donner un « avant-goût » de la complexité de la loi de comportement que l'on souhaite établir. C'est ainsi qu'une brève analyse exploratoire, ayant confirmé nos espérances, a été menée lors l'ultime section de ce rapport.

Pour conclure ce texte, nous pouvons évoquer les perspectives d'avenir de ce projet. D'ores et déjà, sa suite naturelle consiste à mettre en œuvre la méthodologie qui a été proposée. En ce sens, il convient dès à présent de constituer un pôle de réflexion au sujet des méthodes statistiques qu'il serait judicieux d'appliquer au jeu de données qui résultera des nombreuses simulations numériques qui seront effectuées. Enfin, une fois la loi de comportement déterminée, il sera intéressant de la comparer à celles exposées dans [4] et dans [7].

Bibliographie

- [1] DELESTRE, Olivier. *Écriture d'un code C++ pour la simulation en hydrologie*. Thèse de master, Université d'Orléans (France), 2008.
- [2] DELESTRE, Olivier. *Simulation du ruissellement d'eau de pluie sur des surfaces agricoles*. Thèse de doctorat, Université d'Orléans (France), 2010.
- [3] RAZAFISON, Ulrich, et CORDIER, Stéphane, et DELESTRE, Olivier, et DARBOUX, Frédéric, et LUCAS, Carine, et JAMES, François. *A shallow water model for the numerical simulation of overland flow on surfaces with ridges and furrows*. Soumis, 2009.
- [4] SOUCHERE, V., et KING, D., et DAROUSSIN, J., et PAPY, F., et CAPILLON, A.. *Effects of tillage on runoff directions : consequences on runoff contributing area within agricultural catchment*. Journal of Hydrology, Elsevier, 1998.
- [5] TAKKEN, Ingrid, et GOVERS, Gerard, et JETTEN, Victor, et NACHTERGAELE, Jeroen, et STEEGEN, An, et POESEN, Jean. *Effects on tillage on runoff and erosion patterns*. Soil & Tillage Research, Elsevier, 2001.
- [6] TAKKEN, Ingrid, et JETTEN, Victor, et GOVERS, Gerard, et NACHTERGAELE, Jeroen, et STEEGEN, An. *The effect of tillage-induced roughness on runoff and erosion patterns*. Geomorphology, Elsevier, 2001.
- [7] TAKKEN, Ingrid, et GOVERS, Gerard, et STEEGEN, An, et NACHTERGAELE, Jeroen, et GUÉRIF, Jérôme. *The prediction on runoff flow directions on tilled fields*. Journal of Hydrology, Elsevier, 2001.

Annexe A

Adh mar Barr  de Saint-Venant



FIG. A.1 – Un portrait du comte Adh mar Barr  de Saint-Venant

Saint-Venant (Adh mar Barr , comte de)¹, ing nieur fran ais (Villiers-en-Bi re, Seine-et-Marne, 1797 - Saint-Ouen, pr s de Vend me, 1886). Avec Wautzel, il effectua, en 1839, les premi res exp riences sur l' coulement des gaz   grande vitesse.

¹Cette biographie est extraite du *Larousse encyclop dique en couleurs* de 1993.

Annexe B

Le code du script Ga2Fu

À titre d'illustration du travail de programmation qui a été mené durant ce stage, le code du script Ga2Fu va être ici exposé. Ce dernier correspond à la version utilisée dans le cadre de l'utilisation du logiciel Condor, qui pour des raisons d'adaptabilité, a été scindée en deux scripts distincts : Ga2Fu1 et Ga2Fu2.

Voici le code de script Ga2Fu1 :

```
#!/bin/bash

#####
# Script : Ga2Fu1.sh                #
#-----#
# Author : Pierre-Antoine Ksinant #
#-----#
# Date : August 2010                #
#####

# Executables location definition
EXECPATH=~/.condor/bin

# Variables declaration list
DIEZE=#
UN=1

# Pre-print of the condor configuration file
echo "#####" >> simulations.submit
echo "# CONDOR PARAMETERS #" >> simulations.submit
echo "#####" >> simulations.submit
echo "Executable = FullSWOF_2D.exe" >> simulations.submit
echo "Output = outcome.dat" >> simulations.submit
echo "Error = outcome.err" >> simulations.submit
echo "Log = outcome.log" >> simulations.submit
```

```
# Loop over the simulations parameters list file
cat simulations.txt | while read LIGNE
do
    echo $LIGNE > FICHIERTMP
    PREMIER='cut -c 1 FICHIERTMP'
    if [ "$PREMIER" != "$DIEZE" ]
    then
        LIGNE=( ${LIGNE} )
        mkdir ${LIGNE[0]}
        cd ${LIGNE[0]}

# Print of the configuration (to make sure that the parameters have been\
read right)

        echo "#=====#" >> configuration.txt
        echo "# SIMULATION PARAMETERS #" >> configuration.txt
        echo "#=====#" >> configuration.txt
        echo "# Name #" >> configuration.txt
        echo "#-----#" >> configuration.txt
        echo ${LIGNE[0]} >> configuration.txt
        echo "#-----#" >> configuration.txt
        echo "# Shared parameters #" >> configuration.txt
        echo "#-----#" >> configuration.txt
        echo "# nbcol :" >> configuration.txt
        echo ${LIGNE[1]} >> configuration.txt
        echo "# nbrow :" >> configuration.txt
        echo ${LIGNE[2]} >> configuration.txt
        echo "# dx :" >> configuration.txt
        echo ${LIGNE[3]} >> configuration.txt
        echo "# dy :" >> configuration.txt
        echo ${LIGNE[4]} >> configuration.txt
        echo "#-----#" >> configuration.txt
        echo "# GARS #" >> configuration.txt
        echo "#-----#" >> configuration.txt
        echo "# slope :" >> configuration.txt
        echo ${LIGNE[5]} >> configuration.txt
        echo "# slope_azimut :" >> configuration.txt
        echo ${LIGNE[6]} >> configuration.txt
        echo "# wave_amplitude :" >> configuration.txt
        echo ${LIGNE[7]} >> configuration.txt
        echo "# wave_length :" >> configuration.txt
        echo ${LIGNE[8]} >> configuration.txt
        echo "# wave_azimut :" >> configuration.txt
        echo ${LIGNE[9]} >> configuration.txt
        echo "# wave_origin :" >> configuration.txt
        echo ${LIGNE[10]} >> configuration.txt
```

```
echo "# offc :" >> configuration.txt
echo ${LIGNE[11]} >> configuration.txt
echo "#-----#" >> configuration.txt
echo "# GenHUV #" >> configuration.txt
echo "#-----#" >> configuration.txt
echo "# water height :" >> configuration.txt
echo ${LIGNE[12]} >> configuration.txt
echo "# speed horizontal component :" >> configuration.txt
echo ${LIGNE[13]} >> configuration.txt
echo "# speed vertical component :" >> configuration.txt
echo ${LIGNE[14]} >> configuration.txt
echo "#-----#" >> configuration.txt
echo "# FullSWOF_2D #" >> configuration.txt
echo "#-----#" >> configuration.txt
echo "# number of meshes (x-axis) :" >> configuration.txt
echo ${LIGNE[1]} >> configuration.txt
echo "# number of meshes (y-axis) :" >> configuration.txt
echo ${LIGNE[2]} >> configuration.txt
echo "# time of simulation :" >> configuration.txt
echo ${LIGNE[15]} >> configuration.txt
echo "# space step (x) :" >> configuration.txt
echo ${LIGNE[3]} >> configuration.txt
echo "# space step (y) :" >> configuration.txt
echo ${LIGNE[4]} >> configuration.txt
echo "# time step initialization :" >> configuration.txt
echo ${LIGNE[16]} >> configuration.txt
echo "# left boundary condition (x=0) (1: liquid, 2: wall,\
3: Neumann, 4: periodic) :" >> configuration.txt
echo ${LIGNE[17]} >> configuration.txt
echo "# right boundary condition (x=xmax) :" >> configuration.txt
echo ${LIGNE[18]} >> configuration.txt
echo "# bottom boundary condition (y=0) :" >> configuration.txt
echo ${LIGNE[19]} >> configuration.txt
echo "# top boundary condition (y=ymax) :" >> configuration.txt
echo ${LIGNE[20]} >> configuration.txt
echo "# friction law (1: Manning, 2: Darcy-Weisbach) :" >>\
configuration.txt
echo ${LIGNE[21]} >> configuration.txt
echo "# friction coefficient :" >> configuration.txt
echo ${LIGNE[22]} >> configuration.txt
echo "# numerical flux choice (1: Rusanov, 2: HLL) :" >>\
configuration.txt
echo ${LIGNE[23]} >> configuration.txt
echo "# order of the scheme choice (1: ordre 1, 2: ordre 2,\
3: relaxation) :" >> configuration.txt
```

```
echo ${LIGNE[24]} >> configuration.txt
echo "# reconstruction choice (1: MUSCL, 2: ENO, 3: ENO\
modifié) :" >> configuration.txt
echo ${LIGNE[25]} >> configuration.txt
echo "# limiter choice (1: Minmod, : Vanalbada) :" >>\
configuration.txt
echo ${LIGNE[26]} >> configuration.txt
echo "# amortec :" >> configuration.txt
echo ${LIGNE[27]} >> configuration.txt
echo "# modifENO (between 0. and 1.) :" >> configuration.txt
echo ${LIGNE[28]} >> configuration.txt
echo "# infiltration model (1: Green-Ampt) :" >> configuration.txt
echo ${LIGNE[29]} >> configuration.txt
echo "# ks, hydraulic conductivity (saturation) of the surface\
(1.7e-8) :" >> configuration.txt
echo ${LIGNE[30]} >> configuration.txt
echo "# dteta, water content :" >> configuration.txt
echo ${LIGNE[31]} >> configuration.txt
echo "# psi, load pressure :" >> configuration.txt
echo ${LIGNE[32]} >> configuration.txt
echo "# zcrust, crust layer :" >> configuration.txt
echo ${LIGNE[33]} >> configuration.txt
echo "# kssoil, hydraulic conductivity (saturation) of the soil\
:" >> configuration.txt
echo ${LIGNE[34]} >> configuration.txt
echo "# cfl :" >> configuration.txt
echo ${LIGNE[35]} >> configuration.txt
echo "# epsilon :" >> configuration.txt
echo ${LIGNE[36]} >> configuration.txt
echo "# topography choice (1: file, 2: function, 3: thacker,\
4: radial_dam_dry, 5: radial_dam_wet) :" >> configuration.txt
echo ${LIGNE[37]} >> configuration.txt
echo "# topography file name :" >> configuration.txt
echo ${LIGNE[38]} >> configuration.txt
echo "# huv initialization choice :" >> configuration.txt
echo ${LIGNE[39]} >> configuration.txt
echo "# huv initialization file name :" >> configuration.txt
echo ${LIGNE[40]} >> configuration.txt
echo "# rain choice (1: file, 2: function) :" >> configuration.txt
echo ${LIGNE[41]} >> configuration.txt
echo "# rain file name :" >> configuration.txt
echo ${LIGNE[42]} >> configuration.txt
echo "# suffix for the 'Outputs' directory :" >> configuration.txt
mkdir Inputs
mkdir Outputs
```

```

# Generation of the topography file
cd Inputs
$EXECPATH/gars ${LIGNE[1]} ${LIGNE[2]} ${LIGNE[3]} ${LIGNE[4]}\
${LIGNE[5]} ${LIGNE[6]} ${LIGNE[7]} ${LIGNE[8]} ${LIGNE[9]}\
${LIGNE[10]} ${LIGNE[11]} > ${LIGNE[38]}

# Generation of the huv initialization file
if [ "${LIGNE[39]}" == "$UN" ]
then
    $EXECPATH/GenHUV ${LIGNE[1]} ${LIGNE[2]} ${LIGNE[3]}\
    ${LIGNE[4]} ${LIGNE[12]} ${LIGNE[13]} ${LIGNE[14]} >\
    ${LIGNE[40]}
fi

# Generation of the input parameters FullSWOF_2D file
echo "#####" >> parameters.txt
echo " FullSWOF_2D Parameters " >> parameters.txt
echo "#####" >> parameters.txt
echo >> parameters.txt
echo "Number of meshes (x-axis) :" ${LIGNE[1]} >>\
parameters.txt
echo "Number of meshes (y-axis) :" ${LIGNE[2]} >>\
parameters.txt
echo >> parameters.txt
echo "Time of simulation :" ${LIGNE[15]} >> parameters.txt
echo >> parameters.txt
echo "Space step (x) :" ${LIGNE[3]} >> parameters.txt
echo "Space step (y) :" ${LIGNE[4]} >> parameters.txt
echo >> parameters.txt
echo "Time step :" ${LIGNE[16]} >> parameters.txt
echo >> parameters.txt
echo "Left boundary condition (x=0) (1-liquid/2-wall\
/3-Neumann/4-periodic) :" ${LIGNE[17]} >> parameters.txt
echo "Right boundary condition (x=xmax) :" ${LIGNE[18]} >>\
parameters.txt
echo "Bottom boundary condition (y=0) :" ${LIGNE[19]} >>\
parameters.txt
echo "Top boundary condition (y=ymax) :" ${LIGNE[20]} >>\
parameters.txt
echo >> parameters.txt
echo "Friction law (1-Manning/2-Darcy-Weisbach) :"\
${LIGNE[21]} >> parameters.txt
echo "Friction coefficient :" ${LIGNE[22]} >> parameters.txt
echo >> parameters.txt

```

```
echo "Numerical flux choice (1-Rusanov/2-HLL) :"\
${LIGNE[23]} >> parameters.txt
echo >> parameters.txt
echo "Order of the scheme choice (1-ordre 1/2-ordre 2\
/3-relaxation) :" ${LIGNE[24]} >> parameters.txt
echo >> parameters.txt
echo "Reconstruction choice (1-MUSCL/2-ENO/3-ENO modifié) :"\
${LIGNE[25]} >> parameters.txt
echo "Limiter choice (1-Minmod/2-Vanalbada) :" ${LIGNE[26]}\
>> parameters.txt
echo "Armortrec :" ${LIGNE[27]} >> parameters.txt
echo "ModifENO (Between 0 and 1.) :" ${LIGNE[28]} >>\
parameters.txt
echo >> parameters.txt
echo "Infiltration model (1-Green-Ampt) :" ${LIGNE[29]}\
>> parameters.txt
echo "Ks, hydraulic conductivity (saturation) of the surface\
(1.7e-8) :" ${LIGNE[30]} >> parameters.txt
echo "Dteta, water content :" ${LIGNE[31]} >> parameters.txt
echo "Psi, load pressure :" ${LIGNE[32]} >> parameters.txt
echo "Zcrust, crust layer :" ${LIGNE[33]} >> parameters.txt
echo "Kssoil, hydraulic conductivity (saturation) of the soil\
:" ${LIGNE[34]} >> parameters.txt
echo >> parameters.txt
echo "Cfl :" ${LIGNE[35]} >> parameters.txt
echo >> parameters.txt
echo "Epsilon :" ${LIGNE[36]} >> parameters.txt
echo >> parameters.txt
echo "Topography choice (1-file/2-function/3-thacker\
/4-radial_dam_dry/5-radial_dam_wet) :" ${LIGNE[37]} >>\
parameters.txt
echo "Topography file name :" ${LIGNE[38]} >> parameters.txt
echo >> parameters.txt
echo "Huv initialization choice :" ${LIGNE[39]} >>\
parameters.txt
echo "Huv initialization file name :" ${LIGNE[40]} >>\
parameters.txt
echo >> parameters.txt
echo "Rain choice (1-file/2-function) :" ${LIGNE[41]}\
>> parameters.txt
echo "Rain file name :" ${LIGNE[42]} >> parameters.txt
echo >> parameters.txt
echo "Suffix for the 'Outputs' directory :" >>\
parameters.txt
cd ../../
```

```

# To add a job to the Condor list
    echo "Initialdir =" ${LIGNE[0]} >> simulations.submit
    echo "Queue" >> simulations.submit
    fi
done
rm FICHIERTMP

# To run Condor
condor_submit simulations.submit

```

Et voici le code relatif à Ga2Fu2 :

```

#!/bin/bash

#####
# Script : Ga2Fu2.sh                #
#-----#
# Author : Pierre-Antoine Ksinant #
#-----#
# Date : August 2010                #
#####

# Executables location definition
EXECPATH=~ /condor/bin

# Variables declaration list
DIEZE=#
UN=1

# Pre-print of the results file
echo "#=====#" >> results.txt
echo "# RESULTS OF THE SIMULATIONS LIST #" >> results.txt
echo "#=====#" >> results.txt
echo "# " >> results.txt
echo "# name nbc col nbrow dx dy slope slope_azimut wave_amplitude wave_length\
wave_azimut wave_origin offc h u v time dt lbc rbc bbc tbc friction_law\
friction_coef numerical_flux order reconstruction limiter amortrec modifENO\
infiltration ks dteta psi zcrust kssoil cfl epsilon topo_choice topo_name\
huv_choice huv_name rain_choice rain_name time_seconds time_minutes time_hours\
nb_iterations epsilon_cond froude_number percentage_water_error\
mean_horizontal_flux mean_vertical_flux mean_flux mean_flux_direction\
percentage_furrows percentage_slope" >> results.txt
echo "# " >> results.txt

```

```
# Loop over the simulations parameters list file
cat simulations.txt | while read LIGNE
do
    echo $LIGNE > FICHIERTMP
    PREMIER='cut -c 1 FICHIERTMP'
    if [ "$PREMIER" != "$DIEZE" ]
    then
        LIGNE=( ${LIGNE} )
        cd ${LIGNE[0]}

# To acquire the output parameters that FullSWOF_2D has generated
        TMPSEC=$(sed -n '40p' outcome.dat | awk -F":" '{ print $2 }')
        let $[ LNG = ${#TMPSEC} - $UN ]
        TMPSEC=${TMPSEC:1:$LNG}
        TMPMIN=$(sed -n '41p' outcome.dat | awk -F":" '{ print $2 }')
        let $[ LNG = ${#TMPMIN} - $UN ]
        TMPMIN=${TMPMIN:1:$LNG}
        TMPHRS=$(sed -n '42p' outcome.dat | awk -F":" '{ print $2 }')
        let $[ LNG = ${#TMPHRS} - $UN ]
        TMPHRS=${TMPHRS:1:$LNG}
        NBITRS=$(sed -n '43p' outcome.dat | awk -F":" '{ print $2 }')
        let $[ LNG = ${#NBITRS} - $UN ]
        NBITRS=${NBITRS:1:$LNG}
        CNDEPS=$(sed -n '44p' outcome.dat | awk -F":" '{ print $4 }')
        let $[ LNG = ${#CNDEPS} - $UN ]
        CNDEPS=${CNDEPS:1:$LNG}
        FROUDE=$(sed -n '45p' outcome.dat | awk -F":" '{ print $2 }')
        let $[ LNG = ${#FROUDE} - $UN ]
        FROUDE=${FROUDE:1:$LNG}
        WATERR=$(sed -n '46p' outcome.dat | awk -F":" '{ print $2 }')
        let $[ LNG = ${#WATERR} - $UN ]
        WATERR=${WATERR:1:$LNG}
        mv outcome.dat Outputs/
        mv outcome.err Outputs/
        mv outcome.log Outputs/
        cd Outputs

# To run PerFlux
        $EXECPATH/PerFlux ${LIGNE[1]} ${LIGNE[2]} ${LIGNE[3]}\
        ${LIGNE[4]} 2 ${LIGNE[6]} huv_movie.dat > perflux.dat

# To acquire the output parameters that PerFlux has generated
        HUMEAN=$(sed -n '3p' perflux.dat | awk -F":" '{ print $2 }')
        let $[ LNG = ${#HUMEAN} - $UN ]
        HUMEAN=${HUMEAN:1:$LNG}
```

```

HVMEAN=$(sed -n '4p' perflux.dat | awk -F":" '{ print $2 }')
let $[ LNG = ${#HVMEAN} - $UN ]
HVMEAN=${HVMEAN:1:$LNG}
QMEAN=$(sed -n '5p' perflux.dat | awk -F":" '{ print $2 }')
let $[ LNG = ${#QMEAN} - $UN ]
QMEAN=${QMEAN:1:$LNG}
QDIR=$(sed -n '6p' perflux.dat | awk -F":" '{ print $2 }')
let $[ LNG = ${#QDIR} - $UN ]
QDIR=${QDIR:1:$LNG}
PERFUR=$(sed -n '7p' perflux.dat | awk -F":" '{ print $2 }')
let $[ LNG = ${#PERFUR} - $UN ]
PERFUR=${PERFUR:1:$LNG}
PERSLOP=$(sed -n '8p' perflux.dat | awk -F":" '{ print $2 }')
let $[ LNG = ${#PERSLOP} - $UN ]
PERSLOP=${PERSLOP:1:$LNG}

# To run PrintOut
$EXECPATH/PrintOut ${LIGNE[1]} ${LIGNE[2]} ${LIGNE[3]}\
${LIGNE[4]} 2 huv_movie.dat
cd ../..

# To save the simulation results in the results file
echo "${LIGNE[0]} ${LIGNE[1]} ${LIGNE[2]} ${LIGNE[3]} ${LIGNE[4]}\
${LIGNE[5]} ${LIGNE[6]} ${LIGNE[7]} ${LIGNE[8]} ${LIGNE[9]}\
${LIGNE[10]} ${LIGNE[11]} ${LIGNE[12]} ${LIGNE[13]} ${LIGNE[14]}\
${LIGNE[15]} ${LIGNE[16]} ${LIGNE[17]} ${LIGNE[18]} ${LIGNE[19]}\
${LIGNE[20]} ${LIGNE[21]} ${LIGNE[22]} ${LIGNE[23]} ${LIGNE[24]}\
${LIGNE[25]} ${LIGNE[26]} ${LIGNE[27]} ${LIGNE[28]} ${LIGNE[29]}\
${LIGNE[30]} ${LIGNE[31]} ${LIGNE[32]} ${LIGNE[33]} ${LIGNE[34]}\
${LIGNE[35]} ${LIGNE[36]} ${LIGNE[37]} ${LIGNE[38]} ${LIGNE[39]}\
${LIGNE[40]} ${LIGNE[41]} ${LIGNE[42]} $TMPSEC $TMPMIN $TMPHRS\
$NBITRS $CNDEPS $FROUDE $WATERR $HUMEAN $HVMEAN $QMEAN $QDIR\
$PERFUR $PERSLOP" >> results.txt

fi
done
rm FICHIERTMP

```