

# Animation de l'eau en synthèse d'images : exemple de cours d'eau

Sébastien THON et Djamchid GHAZANFARPOUR

Laboratoire MSI, équipe Visualisation et Rendu Réaliste  
ENSIL, Technopole, 87068 LIMOGES Cedex, France  
thon@ensil.unilim.fr, ghazanfa@ensil.unilim.fr

## Résumé

L'eau est certainement l'un des phénomènes naturels les plus complexes à représenter en synthèse d'images. Outre la difficulté posée par le rendu de l'eau, la difficulté première est celle de la modélisation et de l'animation de sa surface, car l'eau n'a pas de forme propre dans son état liquide. Afin de modéliser et animer cette forme, deux approches sont possibles. La première, la plus simple mais la plus limitée, consiste à modéliser les conséquences du phénomène, en cherchant à reproduire l'apparence du comportement de l'eau. La seconde approche, plus complexe, consiste à modéliser les causes du phénomène, en s'intéressant aux forces agissant sur l'eau. Dans le cas de cette seconde approche se pose alors la question de savoir jusqu'à quel point il est nécessaire de descendre à un niveau physique pour obtenir un résultat visuellement satisfaisant. Dans le cadre de la représentation de cours d'eau, nous proposons ici deux modèles faisant appel à l'une et à l'autre de ces approches. Très peu de modèles de cours d'eau ont été proposés en synthèse d'images du fait de la complexité du mouvement de l'eau. Le premier modèle que nous proposons vise une grande simplicité d'utilisation et offre un réalisme visuel satisfaisant dans des cas simples. Le second modèle, plus réaliste, permet de palier les défauts du modèle précédent et des autres modèles existants.

## 1. Introduction

La représentation réaliste de phénomènes naturels est l'un des objectifs majeurs de la synthèse d'images. De nombreux phénomènes tels que l'eau, le feu, la fumée, les nuages ou encore la végétation ont fait et font toujours l'objet de recherches actives. Parmi ces phénomènes naturels, l'eau est certainement l'un des plus importants du fait de son omniprésence dans le monde qui nous entoure.

Mais si la représentation de l'eau est toujours l'objet de recherches, c'est qu'il s'agit probablement d'un des phénomènes naturels les plus complexes à représenter. Outre la difficulté posée par le rendu de l'eau, liée à la complexité de ses interactions avec la lumière, la difficulté première est celle de la modélisation de sa surface, car l'eau n'a pas de forme propre dans son état liquide. Elle s'adapte à son environnement, sa forme change selon les situations et selon les forces auxquelles elle est soumise. La modélisation de la forme de l'eau est donc indissociable de son animation, car cette forme évolue au cours du temps selon un mouvement qui peut être très complexe. Pour modéliser et animer la forme de l'eau, il existe deux approches :

- La modélisation des **conséquences** du phénomène, en utilisant un modèle différent pour chaque situation. On cherche alors à reproduire l'apparence du comportement de l'eau dans une situation bien précise (des vagues en pleine mer, des vagues s'échouant sur une plage, une chute d'eau, etc.) en donnant une description explicite de la forme de la surface de l'eau dans cette situation. Ces modèles sont empiriques et ont principalement été décrits dans les premiers travaux présentés en synthèse d'images. On peut par exemple citer les travaux de [MAX 81] [PEAC 86] [FOUR 86] [TSO 87] [GONZ 97] pour la représentation de vagues d'océans, ceux de [SIMS 90] [MALL 95] pour la représentation de chutes d'eau. L'avantage principal de cette approche est de permettre une animation visuellement réaliste, nécessitant généralement peu de temps de calcul et de ressources mémoire. Cependant, étant donné que ces modèles décrivent explicitement l'allure du mouvement de l'eau pour certaines situations bien précises, ils ne peuvent être utilisés dans des situations plus générales. Par exemple, un modèle conçu pour représenter explicitement les vagues d'un océan ne pourra pas être utilisé pour simuler une chute d'eau. De plus, ces modèles empiriques ne peuvent réagir dynamiquement à toutes sortes de situations qui n'auraient pas été explicitement prévues au préalable dans le modèle, comme l'application de nouvelles forces telles que le vent ou la chute d'un objet dans l'eau ;

- La modélisation des **causes** du phénomène, en s'intéressant aux forces à l'origine de la forme prise par l'eau, sans chercher à définir explicitement cette forme. Ces modèles, qui constituent la nouvelle tendance en synthèse d'images, sont physiques. Ils sont basés sur des résultats de la mécanique des fluides, un domaine de la physique dont le sujet d'étude est le comportement de fluides (les gaz et les liquides). Les équations fondamentales utilisées en mécanique des fluides sont celles de Navier-Stokes, abondamment décrites dans la littérature (lire par exemple [CAND 95]). Ces équations ont été utilisées en synthèse d'images soit sous leur formulation complète [FOST 96] [FOST 97] [STAM 99] [FOST 01], soit de manière plus ou moins simplifiée [KASS 90] [CHEN 95] [OBRI 95] [MOUL 97]. L'utilisation de modèles physiques permet d'obtenir un plus grand réalisme qu'avec des modèles empiriques. Ces modèles présentent aussi l'avantage d'être très généraux et de pouvoir représenter toutes sortes de situations faisant intervenir des fluides, pour peu que le domaine étudié soit correctement discrétisé et que les forces entrant en jeu soient correctement prises en compte. Cependant, les équations de la mécanique des fluides sont très coûteuses en temps de calculs et en place mémoire, complexes à mettre en œuvre et posent des problèmes de résolution. Par conséquent, il n'est pas envisageable de les utiliser pour des scènes faisant intervenir de grandes surfaces d'eau telles que des océans, des mers ou des grands lacs.

Dans le domaine de l'animation de l'eau, nous nous sommes intéressés à la représentation de cours d'eau tels que des rivières, des torrents ou des ruisseaux. La représentation de cours d'eau a été très peu étudiée en synthèse d'images, car il est difficile de prendre en compte les interactions complexes de l'eau avec les obstacles et les berges, et de représenter les différents phénomènes turbulents qui en découlent (remous, tourbillons...). Chiba et al [CHIB 95] représentent un cours d'eau au moyen d'un système de particules interagissant entre elles et avec les obstacles, mais les équations décrivant leurs mouvements ne permettent pas de représenter des phénomènes turbulents tels que des remous ou des tourbillons. Xu et al [XUSU 97] simulent l'animation de l'eau au moyen d'équations hydrauliques, mais ce modèle ne gère pas les interactions de l'eau avec son environnement.

La solution qui permettrait d'obtenir les meilleurs résultats d'un point de vue physique serait d'utiliser des modèles complets provenant de la mécanique des fluides, mais ces modèles seraient très coûteux en temps de calcul et en mémoire pour représenter d'aussi grandes étendues d'eau.

Nous présentons ici deux modèles permettant de représenter des cours d'eau. Le premier de ces modèles est purement empirique et vise à reproduire l'apparence du phénomène. L'eau y est représentée explicitement au moyen d'une texture 3D procédurale animée. Nécessitant très peu de mémoire et de temps de calcul, ce modèle permet d'obtenir des résultats visuellement réalistes dans des cas simples, sans toutefois prendre en compte les interactions eau/obstacles [THON 00]. Le second modèle que nous proposons est semi-physique et repose sur l'étude des forces à l'origine du mouvement de l'eau. Partiellement basé sur les équations de Navier-Stokes, ce modèle prend en compte les interactions entre l'eau et les obstacles, et permet de représenter les phénomènes turbulents qui en découlent. Il s'agit d'un modèle physique simplifié offrant un compromis entre réalisme physique et besoins en temps de calcul et en mémoire [THON 01].

Dans la section 2, nous décrivons la création du lit du cours d'eau. Le modèle empirique de cours d'eau est présenté dans la section 3 et le modèle semi-physique dans la section 4. Nous concluons dans la section 5 et exposons des idées de travaux futurs dans la section 6.

## 2. Lit de cours d'eau

### 2.1. Principe

Idéalement, afin d'obtenir un terrain dans lequel s'intègre de façon réaliste un cours d'eau, il faudrait définir un sol ayant un ensemble de propriétés géologiques, y placer des sources d'eau et définir des conditions climatiques, puis faire évoluer le système en laissant les processus d'érosion tracer les lits des cours d'eau [MUSG 89] [ROUD 93]. Mais bien que cette démarche donne de bons résultats, elle est plutôt appropriée à une simulation scientifique et n'est pas facilement utilisable en synthèse d'images. En effet, cette démarche offre peu de contrôle car il est difficile de trouver les bonnes propriétés du sol et du climat qui permettront d'obtenir un certain résultat voulu.

C'est pourquoi nous proposons une méthode de génération de terrain plus directe qui peut être utilisée très intuitivement. Plutôt que de partir d'un terrain pour y tracer des lits de cours d'eau par une simulation scientifique, nous utilisons une démarche inverse en laissant l'utilisateur définir directement le lit du cours d'eau autour duquel le terrain est ensuite construit. Il s'agit donc là d'un processus de modélisation de paysage par érosion inverse. Une telle démarche a déjà été utilisée entre autres dans [ARQU 91]. Une restriction cependant : les cours d'eau que nous allons modéliser seront plans, il ne sera donc pas possible d'obtenir des torrents dévalant une pente ou des cascades.

## 2.2. Trajectoire du cours d'eau

Dans la réalité, la trajectoire d'un cours d'eau peut être très tortueuse (figure 1). Pour permettre à un utilisateur de représenter facilement de tels méandres, nous définissons la trajectoire du lit par une courbe de Bézier ayant un nombre quelconque de points de contrôle (figure 2).



Figure 1. Photographie d'un cours d'eau réel

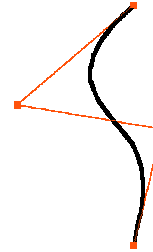


Figure 2. Courbe de Bézier définissant la trajectoire d'un cours d'eau

## 2.3. Coupe du cours d'eau

La forme finale du lit du cours d'eau est obtenue en translatant une courbe décrivant la coupe du lit le long de la trajectoire qui vient d'être définie. L'allure de cette coupe est définie par une seconde courbe de Bézier à 6 points de contrôle dont les positions dépendent d'un ensemble de paramètres donnés par l'utilisateur : la largeur du lit, sa profondeur et la largeur des berges (figure 3).

Pour éviter d'obtenir un lit beaucoup trop régulier en translatant cette coupe le long de la trajectoire du cours d'eau, on perturbe la position des 6 points de contrôle de la coupe au cours de sa translation au moyen de fonctions pseudo-aléatoires dont les paramètres sont donnés par l'utilisateur (figure 4). Ces fonctions pseudo-aléatoires sont de simples fonctions de turbulence de Perlin en 1D [PERL 85].

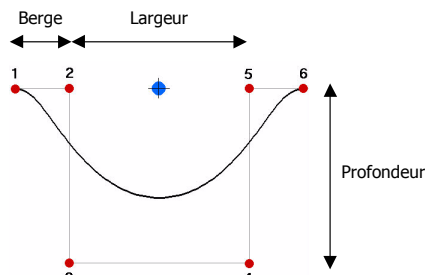


Figure 3. Coupe du lit déterminée par une courbe de Bézier à 6 points de contrôle.

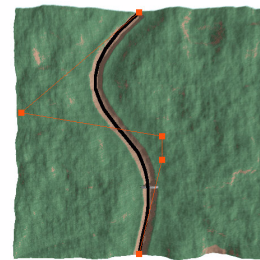


Figure 4. Création du lit du cours d'eau par translation de la coupe de la figure 3 le long de la trajectoire de la figure 2.

## 2.4. Génération du terrain

Une fois le lit du cours d'eau modélisé, le reste du terrain est créé autour du lit au moyen d'une fonction de turbulence 2D. En jouant sur l'amplitude de cette fonction on peut obtenir des hauteurs plus ou moins élevées. Plus le nombre d'octaves utilisées pour la fonction de turbulence est élevé et plus le sol sera accidenté. La hauteur du sol est modifiée à proximité du cours d'eau pour qu'elle tende vers celle de la berge (figure 5).

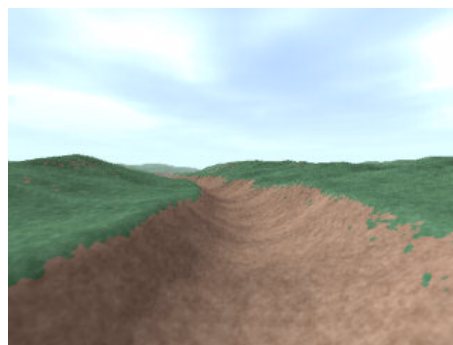


Figure 5. Lit de cours d'eau à sec obtenu selon notre modèle et rendu par lancer de rayons.

La carte de hauteurs constituant le terrain est affichée sous la forme d'un ensemble de triangles. Le calcul de l'intersection d'un rayon avec cette carte de hauteurs lors du lancer de rayons est accéléré en utilisant un système de boîtes englobantes hiérarchiques organisées en arbre quaternaire.

Maintenant que le lit est créé, on peut y simuler l'écoulement de l'eau. Nous proposons deux modèles de cours d'eau. Le premier est empirique et ne cherche qu'à reproduire l'apparence du phénomène. Bien que les résultats soient visuellement réalistes, on verra que ce modèle est limité car il ne permet pas de reproduire certains effets très complexes du comportement de l'eau. C'est pourquoi nous proposons ensuite un second modèle partiellement basé sur des équations physiques de la mécanique des fluides qui permet de pallier les défauts du modèle précédent.

### 3. Modèle empirique de cours d'eau

#### 3.1. Introduction

Comme on peut le constater sur la photographie de la figure 8a, la surface d'un cours d'eau réel est plutôt chaotique. Il n'y a pas de vagues bien formées comme on peut en observer à la surface d'un océan. La raison principale est due au fait que la surface d'une rivière n'est pas suffisamment grande, contrairement à celle d'une mer ou d'un océan, pour que le vent puisse y générer des vagues bien marquées, d'autant plus que de nombreux obstacles situés autour du cours d'eau, tels que des arbres, des taillis ou encore le terrain lui-même, bloquent l'influence du vent.

#### 3.2. Texture d'eau

On propose de représenter l'apparence de la surface de l'eau par un modèle visuellement équivalent. Notre but avec ce modèle est d'obtenir un résultat visuellement réaliste, et non pas une simulation physiquement réaliste. Nous avons choisi de représenter la surface de l'eau au moyen d'une texture 3D procédurale animée, basée sur la fonction de turbulence de Perlin [PERL 85]. Comme on le verra dans la section 3.3, la fonction de turbulence de Perlin présente des caractéristiques fréquentielles proches de celles de la surface d'un cours d'eau réel.

La formulation procédurale de notre texture  $h$  est:

$$h(x,y,t) = A.Turb(n, x.x_s, (y+v_l(x).t).y_s, v_2.t.z_s) \quad (1)$$

où  $Turb$  est la fonction de turbulence 3D de Perlin,  $A$  est son amplitude,  $n$  est le nombre d'octaves de la turbulence,  $(x,y)$  sont les coordonnées d'un point du courant dans un repère défini par la trajectoire du lit (figure 6),  $x_s, y_s, z_s$  sont des coefficients de mise à l'échelle le long des trois axes,  $t$  est le temps,  $v_l$  est la vitesse de propagation de l'eau le long de la trajectoire du lit et  $v_2$  est la vitesse de déplacement vertical dans la fonction de turbulence 3D le long de l'axe  $z$ . Ce déplacement vertical dans la texture 3D permet un mouvement vertical de l'eau et donc une modification de la forme des « vagues ».

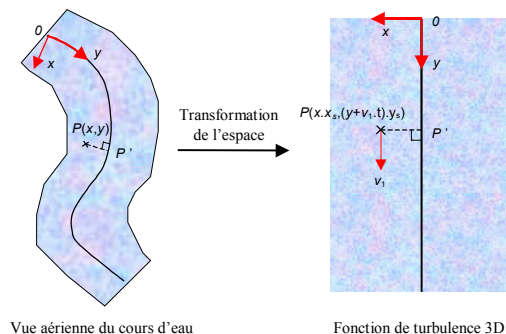


Figure 6. Calcul de la hauteur de l'eau en un point  $P$  du cours d'eau

Cette fonction  $h$  permet de calculer une hauteur d'eau en n'importe quel point  $(x,y)$  du courant à un instant  $t$ .

En pratique, pour calculer la hauteur de l'eau en un point  $P$  du courant, nous procédons comme suit :

1. Nous calculons les coordonnées  $(x,y)$  du point  $P$  dans le repère défini par la trajectoire du cours d'eau :

- a) Nous projetons le point  $P$  sur la courbe de Bézier définissant la trajectoire centrale du cours d'eau. Pour cela, nous subdivisons cette courbe de Bézier en segments, nous projetons  $P$  sur tous ces segments et nous retenons le point projeté  $P'$  correspondant à la distance  $PP'$  la plus courte.
- b) La coordonnée  $x$  du point  $P$  est donnée par la distance  $PP'$  que nous venons de calculer à l'étape a), correspondant à la distance du point  $P$  à la trajectoire centrale du cours d'eau.
- c) La coordonnée  $y$  du point  $P$  est la distance le long de la trajectoire centrale entre le premier point de contrôle de la courbe de Bézier et le point  $P'$  projeté sur cette courbe. Cette distance est la somme des longueurs des segments obtenus en subdivisant la courbe de Bézier à l'étape a), entre le premier point de contrôle et le point  $P'$ .

2. Ces coordonnées  $(x,y)$  sont utilisés dans l'équation (1) pour calculer une valeur dans la fonction de turbulence 3D, cette valeur étant ensuite utilisée comme hauteur de l'eau au point  $P$ .

Avec ce modèle de cours d'eau, la fonction de turbulence est contrainte à suivre la trajectoire du cours. En jouant sur l'amplitude de la turbulence  $A$  et sur les coefficients de mise à l'échelle  $x_s, y_s, z_s$ , il est possible d'obtenir différentes tailles de « vagues ». De plus, en jouant sur les vitesses de déplacement horizontale  $v_l$  et verticale  $v_z$  de l'eau, on peut reproduire l'apparence de cours d'eau plus ou moins agités, depuis les eaux calmes d'une rivière jusqu'au flot tumultueux d'un torrent (figure 7). Le rendu final est obtenu par lancer de rayons, ce qui permet de prendre en compte facilement les effets de réflexion et de réfraction de la lumière par l'eau. La surface 3D de l'eau est rendue comme une surface implicite au moyen d'une technique dite de « ray-marching » [TUY 84].

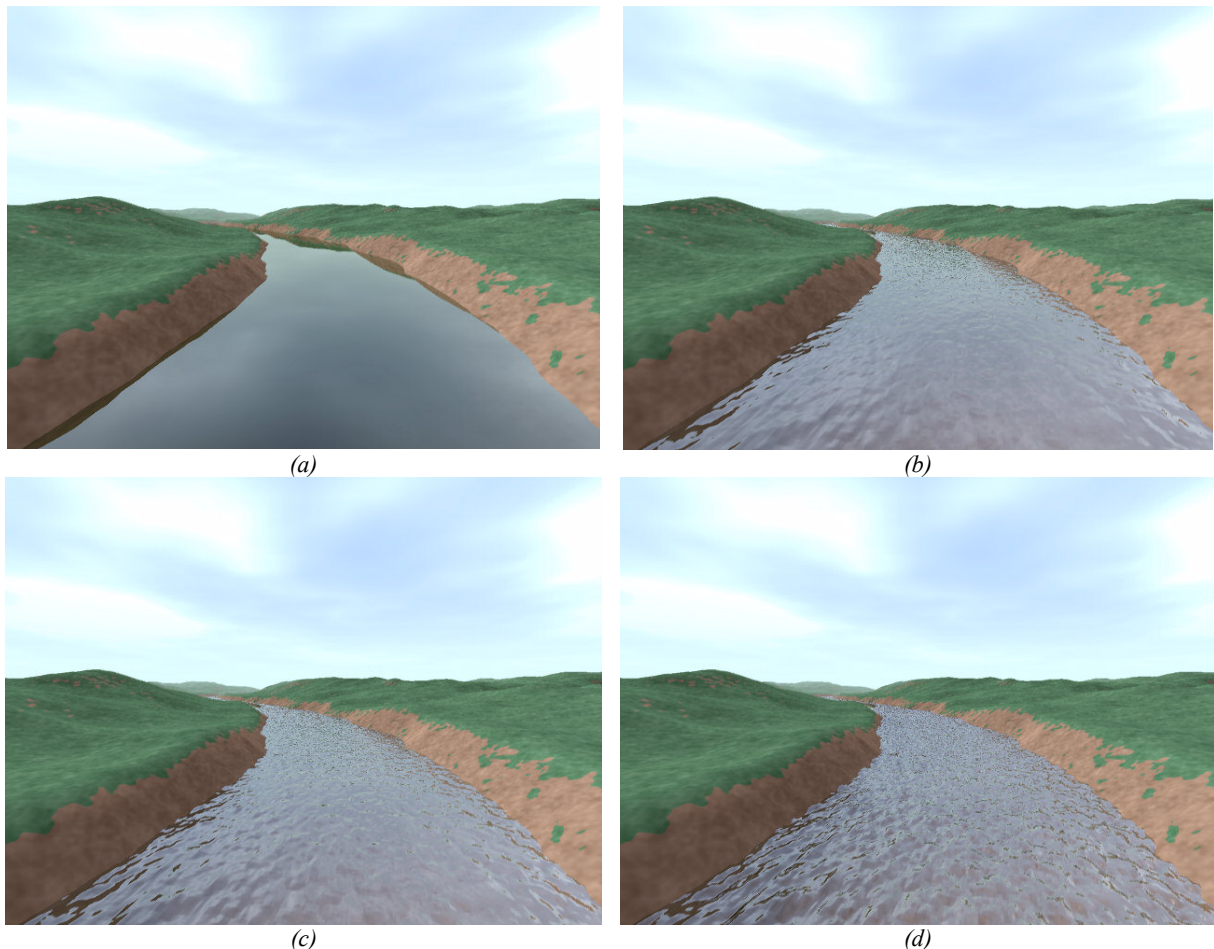


Figure 7. Un cours d'eau de plus en plus agité obtenu au moyen de notre modèle empirique



### 3.3. Etude fréquentielle

L'utilisation de la fonction de turbulence de Perlin pour modéliser la surface d'une rivière permet de reproduire l'aspect pseudo chaotique de la surface d'une rivière réelle, ce que l'on peut vérifier dans le domaine fréquentiel. A partir d'une prise de vue réelle numérisée (figure 8a) et d'une image de synthèse utilisant notre modèle de cours d'eau (figure 8b), toutes deux prises approximativement sous le même angle de vue, nous avons extrait deux vignettes ne présentant que de l'eau et ayant à peu près le même nombre de « périodes » et le même éclairage (figure 8c et 8e). Nous avons ensuite calculé le spectre fréquentiel de chacune de ces deux vignettes transformées en niveaux de gris (figure 8d et 8f) au moyen d'une transformée rapide de Fourier (FFT) 2D. On peut constater que les deux spectres sont très similaires. Notre modèle permet donc de synthétiser une surface de cours d'eau ayant des caractéristiques fréquentielles très semblables à celles de cours d'eau réels, ce qui justifie l'utilisation de la fonction de turbulence de Perlin.

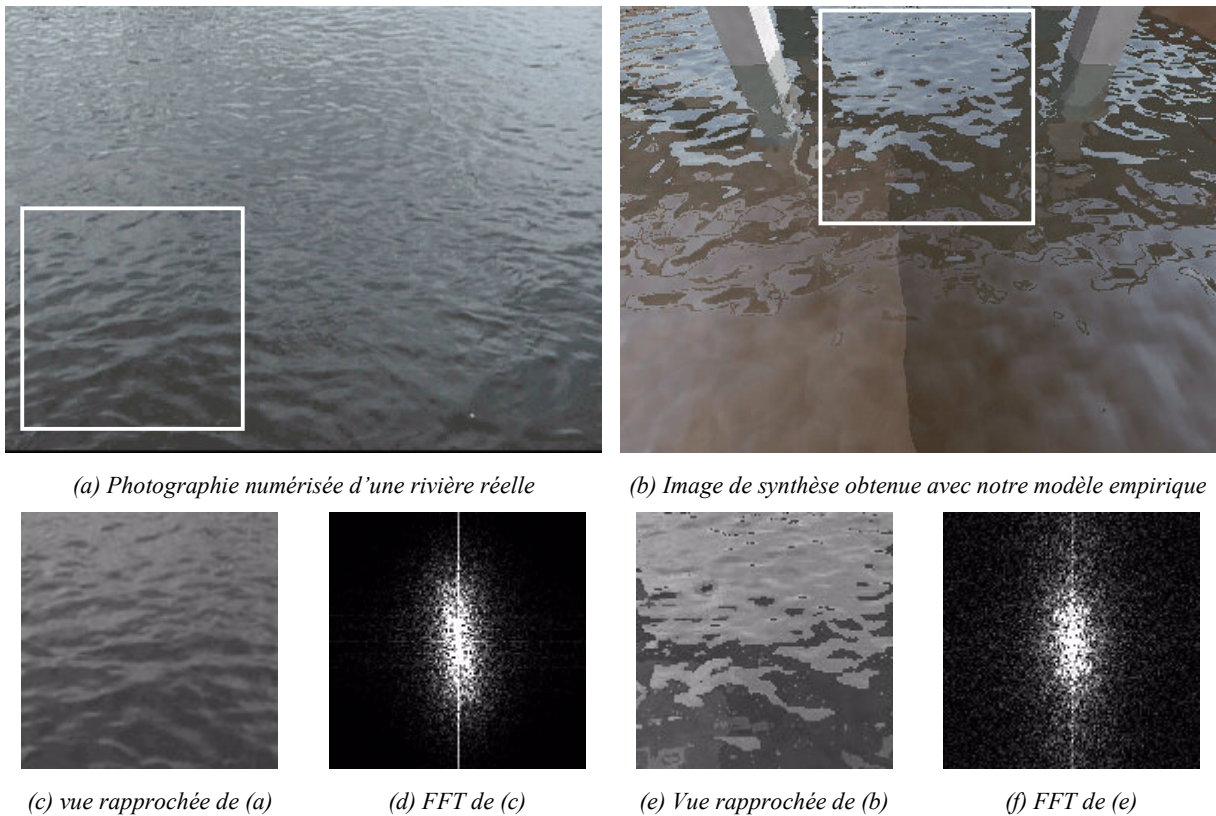


Figure 8. Comparaison dans le domaine fréquentiel entre une prise de vue réelle numérisée et notre modèle.

### 3.4. Conclusion

Nous avons proposé un modèle empirique visant à reproduire explicitement l'apparence de la surface d'un cours d'eau. L'aspect chaotique de la surface d'un cours d'eau réel est reproduit au moyen d'une turbulence de Perlin présentant des caractéristiques fréquentielles similaires, guidée le long d'une courbe de Bézier définissant la trajectoire du lit du cours d'eau. Ce modèle, nécessite des temps de calcul très faibles ainsi qu'une occupation mémoire insignifiante due à sa définition procédurale. Très simple d'utilisation du fait du faible nombre et de la simplicité de ses paramètres, il permet d'obtenir des résultats visuellement réalistes.

Cependant, étant donné que ce modèle est complètement empirique, il ne permet pas de reproduire avec exactitude le comportement complexe de l'eau dans un cours d'eau réel, tel que ses interactions avec des obstacles ou les phénomènes turbulents qui en découlent tels que des remous ou des tourbillons.

## 4. Modèle semi-physique de cours d'eau

### 4.1. Introduction

Un modèle complètement empirique ne permet donc pas de représenter de façon suffisamment réaliste le comportement très complexe de l'eau. Une solution serait d'utiliser des modèles physiques complets de la mécanique des fluides, tels que les équations de Navier-Stokes sous leur forme 3D. Cependant, ces équations nécessitent des temps de calcul très élevés et d'importantes ressources en mémoire pour représenter de façon suffisamment détaillée d'aussi grands volumes d'eau que des cours d'eau. Il faut donc trouver des solutions intermédiaires, utilisant de façon simplifiée des équations physiques.

Nous proposons un modèle semi-physique où l'animation de l'eau est décomposée en deux étapes distinctes. Lors de la première étape, le mouvement horizontal de l'eau est simulé physiquement au moyen des équations de Navier-Stokes sous leur forme 2D, bien moins coûteuses en temps de calcul et en mémoire que leur forme 3D complète, ce qui permet de prendre en compte les obstacles de manière réaliste. Lors de la seconde étape, le mouvement vertical de l'eau est simulé au moyen d'une fonction pseudo-aléatoire très peu coûteuse en temps de calcul.

### 4.2. Equations de Navier-Stokes

Les équations de Navier-Stokes, qui dérivent de la seconde loi de Newton, sont les équations fondamentales de la mécanique des fluides. Ces équations décrivent le comportement de fluides (gaz et liquides) au moyen d'un système d'équations différentielles partielles formé par l'équation des moments (équation 2a) et l'équation de continuité (équation 2b). Dans notre situation, le fluide étudié est de l'eau, donc un liquide incompressible. Dans le cas de fluides incompressibles, la notation sans dimensions des équations de Navier-Stokes est la suivante :

$$\frac{\partial}{\partial t} \vec{u} + (\vec{u} \cdot \text{grad}) \vec{u} + \text{grad } p = \frac{1}{Re} \Delta \vec{u} + \vec{g} \quad (2a)$$

$$\text{div } \vec{u} = 0 \quad (2b)$$

où  $t$  représente le temps,  $\vec{u}$  est un vecteur vitesse,  $p$  est la pression,  $Re$  est le nombre de Reynolds et  $\vec{g}$  représente les forces agissant sur le fluide, telle que la gravité. Le nombre de Reynolds permet de définir la viscosité du fluide. Si ce nombre est petit, l'écoulement est *régulier* ou *laminaire*. S'il est grand, l'écoulement est *turbulent*. En jouant sur ce paramètre, on peut donc reproduire le comportement de fluides plus ou moins visqueux aussi différents que de l'eau, une coulée de boue ou de la lave en fusion.

Pour utiliser ces équations, il faut tout d'abord donner une description discrète du domaine contenant le fluide étudié. Lors de leur résolution numérique, les équations de Navier-Stokes permettent de simuler le transfert de fluide entre les cellules données par la discrétisation du domaine, en calculant des valeurs de vitesse du fluide et de pression pour chacune des cellules.

La forme complète des équations de Navier-Stokes est tridimensionnelle. Elles doivent donc être résolues sur un maillage 3D résultant de la discrétisation volumique du domaine étudié. Cependant, sous cette forme, ces équations sont difficilement utilisables pour simuler le comportement de vastes étendues d'eau. En effet, il faudrait pour cela utiliser un maillage 3D suffisamment vaste et suffisamment fin pour pouvoir représenter des petits détails tels que des vaguelettes. Par conséquent, le maillage 3D requis serait très important et nécessiterait des capacités mémoire très importantes, ainsi que des temps de calcul très élevés lors de la résolution des équations de Navier-Stokes.

C'est pourquoi nous proposons un modèle de cours d'eau qui n'est pas un modèle purement physique utilisant la forme complète 3D des équations de Navier-Stokes, mais qui est un modèle semi-physique utilisant la forme 2D de ces équations.

Dans le cas 2D, les équations de Navier-Stokes se notent :

équation des moments :

$$\frac{\partial u}{\partial t} + \frac{\partial p}{\partial x} = \frac{1}{Re} \left( \frac{\partial^2 u}{\partial x^2} + \frac{\partial^2 u}{\partial y^2} \right) - \frac{\partial(u^2)}{\partial x} - \frac{\partial(uv)}{\partial y} + g_x \quad (3a)$$

$$\frac{\partial v}{\partial t} + \frac{\partial p}{\partial y} = \frac{1}{Re} \left( \frac{\partial^2 v}{\partial x^2} + \frac{\partial^2 v}{\partial y^2} \right) - \frac{\partial(uv)}{\partial x} - \frac{\partial(v^2)}{\partial y} + g_y \quad (3b)$$

équation de continuité :

$$\frac{\partial u}{\partial x} + \frac{\partial v}{\partial y} = 0 \quad (3c)$$

où  $u$  et  $v$  sont les composantes horizontales de la vitesse selon les axes  $x$  et  $y$ .

#### 4.3. Domaine du fluide

Afin de résoudre ces équations 2D de Navier-Stokes, il faut donc discrétiser le domaine du fluide sous la forme d'une grille 2D. Comme nous résolvons ces équations par une méthode de différences finies, nous discrétisons le domaine du fluide sous la forme d'une grille cartésienne uniforme. Ces équations nous permettront de calculer une valeur de pression et de vitesse pour chacune des cellules de la grille.

Afin de ne résoudre les équations de Navier-Stokes que dans les zones correspondant au lit du cours d'eau, nous affectons également à chacune des cellules de la grille une valeur booléenne indiquant si cette cellule est une cellule d'obstacle ou une cellule de fluide. Nous générons automatiquement ces valeurs à partir de la carte de hauteurs du terrain (figure 9), en prenant une coupe horizontale du terrain au niveau de l'eau au repos. Les dimensions de la grille sont les mêmes que celles de la coupe, chaque hauteur de la coupe correspondant à une cellule de la grille. Les cellules de la grille correspondant au lit du cours d'eau sont marquées comme étant des cellules de fluide, et celles correspondant au reste du terrain ou aux obstacles sont marquées comme étant des cellules d'obstacle (cellules respectivement blanches et noires sur la figure 10).

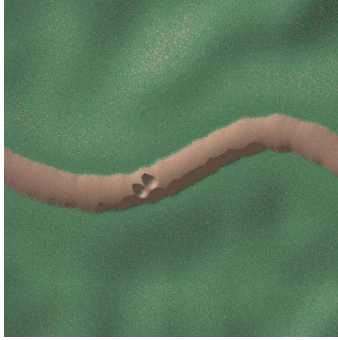


Figure 9. Vue aérienne d'un lit de cours d'eau avec deux rochers

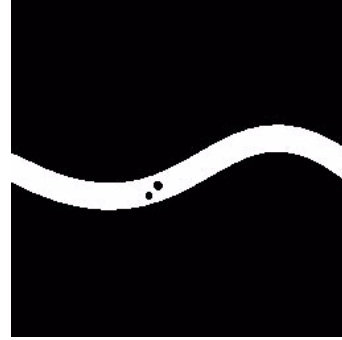


Figure 10. Coupe 2D du lit de la figure 9 au niveau de l'eau au repos (1024x1024 cellules)

#### 4.4. Mouvement horizontal de l'eau

Dans notre modèle, le mouvement horizontal de l'eau est simulé physiquement au moyen des équations 2D de Navier-Stokes. En résolvant ces équations au cours du temps, nous obtenons à chaque pas temporel un vecteur vitesse 2D  $(u, v)$  ainsi qu'une valeur de pression  $p$  pour chaque cellule de la grille d'échantillonnage, définissant ainsi un champ 2D de vitesses (figure 11a) et un champ 2D de pressions.

Afin d'étudier le mouvement horizontal de l'eau, nous suivons le mouvement de particules élémentaires de fluide injectées dans le champ 2D de vitesses. Pour cela, nous injectons des particules sans masse dans le champ de vitesses à intervalles de temps régulier le long d'une ligne à la source du cours d'eau. Le nombre de ces particules injectées au même moment est donné par l'utilisateur. Ensuite, le calcul de la position d'une particule au temps  $t_{n+1} = t_n + \delta t$ , étant donné sa position  $(x^{(n)}, y^{(n)})$  au temps  $t_n$ , est effectué en deux étapes :

1. Tout d'abord, nous déterminons la vitesse  $(u^{(n)}, v^{(n)})$  à la position  $(x^{(n)}, y^{(n)})$  de la particule. Etant donné que les vitesses  $(u, v)$  ne sont calculées par les équations de Navier-Stokes que pour des positions discrètes sur la grille, nous calculons la vitesse par interpolation bilinéaire des vitesses discrètes entourant la position de la particule.
2. Une fois que l'on connaît la vitesse de la particule, nous calculons sa nouvelle position au temps  $t_{n+1}$  avec une méthode d'Euler avant :

$$x^{(n+1)} = x^{(n)} + \delta t \cdot u^{(n)}, \quad y^{(n+1)} = y^{(n)} + \delta t \cdot v^{(n)} \quad (4)$$

Si la nouvelle position de la particule se trouve en dehors des limites de la grille ou dans une cellule d'obstacle, alors on la supprime de la liste des particules.



Les particules injectées au même endroit dans le courant suivent une ligne appelée ligne de courant. Grâce à l'utilisation des équations physiques de Navier-Stokes, nous pouvons voir sur la figure 11a que le champ de vecteurs vitesse s'adapte automatiquement à la géométrie du domaine et évite les obstacles. Ces équations permettent également de représenter des phénomènes turbulents, comme on peut le voir sur la figure 11b, où des tourbillons et des remous se forment derrière les obstacles.

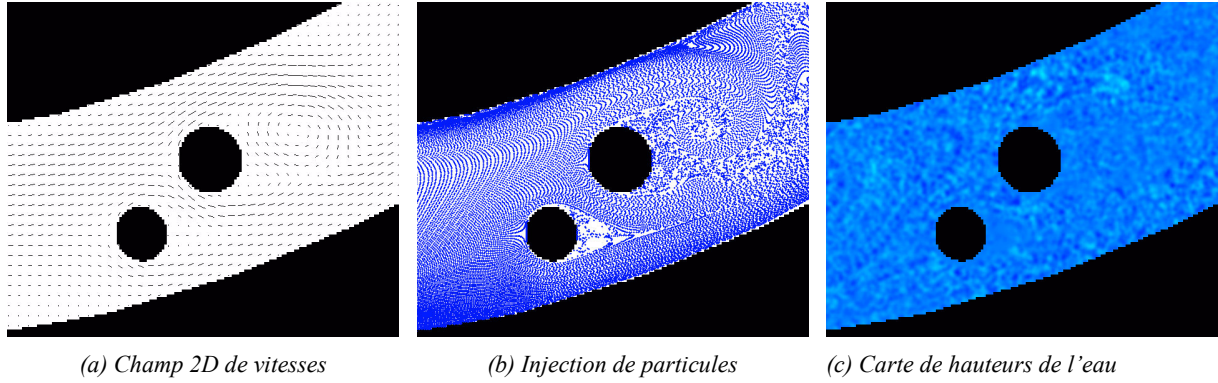


Figure 11. Simulation du mouvement de l'eau sur une vue rapprochée de la figure 10, l'eau provenant de la gauche

#### 4.5. Mouvement vertical de l'eau

Comme nous l'avons vu dans la section 3.1, il n'y a pas de vagues véritablement bien formées à la surface d'un cours d'eau réel, le mouvement vertical de l'eau étant visuellement chaotique. C'est pourquoi nous proposons de simuler la hauteur de l'eau dans notre modèle semi-physique au moyen d'une fonction produisant des valeurs pseudo-aléatoires. Pour cela, nous utilisons une fonction de turbulence de Perlin 2D.

Une fonction de turbulence différente est associée à chaque ligne de courant. La hauteur de l'eau pour une position donnée dans le cours d'eau est modulée par la valeur de la vitesse en ce point. Par conséquent, si la vitesse est nulle en un point, l'élévation de la surface de l'eau sera nulle en ce point. En résumé, à un instant  $t$ , la hauteur de l'eau associée à une particule injectée au temps  $t_i$  et ayant pour position  $(x,y)$  est donnée par :

$$h(x, y, t) = U_{(x,y)} \cdot \text{Turb}(n, v_1 \cdot t_i, v_2 \cdot t) \quad (5)$$

où  $U$  est la valeur absolue de la vitesse horizontale au point  $(x,y)$  donnée par les équations de Navier-Stokes,  $t$  est le temps,  $n$  est le nombre d'octaves de la fonction de turbulence,  $v_1$  et  $v_2$  sont respectivement les vitesses de déplacement horizontale et verticale dans la fonction de turbulence de Perlin 2D. Lorsqu'une particule a été injectée dans le courant, sa position horizontale dans la fonction de turbulence 2D reste la même, alors que sa position verticale dans la turbulence change au cours du temps  $t$  en fonction de sa vitesse verticale  $v_2$ .

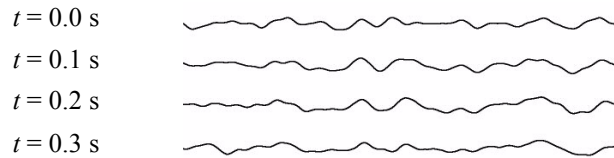


Figure 12. Hauteur de l'eau donnée par une fonction de turbulence 2D au temps  $t$  le long d'une ligne de courant, pour une vitesse verticale  $v_2 = 0.1 \text{ m.s}^{-1}$

En modifiant la position verticale des particules dans la fonction de turbulence, la hauteur de l'eau le long d'une ligne de courant évolue au cours du temps (figure 12), donnant un mouvement plus dynamique que dans le cas où la hauteur de l'eau associée à une particule resterait toujours la même. La hauteur de l'eau en tout autre point du courant est calculée par interpolation des hauteurs associées aux particules voisines de ce point (figure 11c).

Tout comme pour le modèle précédent, le rendu est obtenu par lancer de rayons (figure 13).

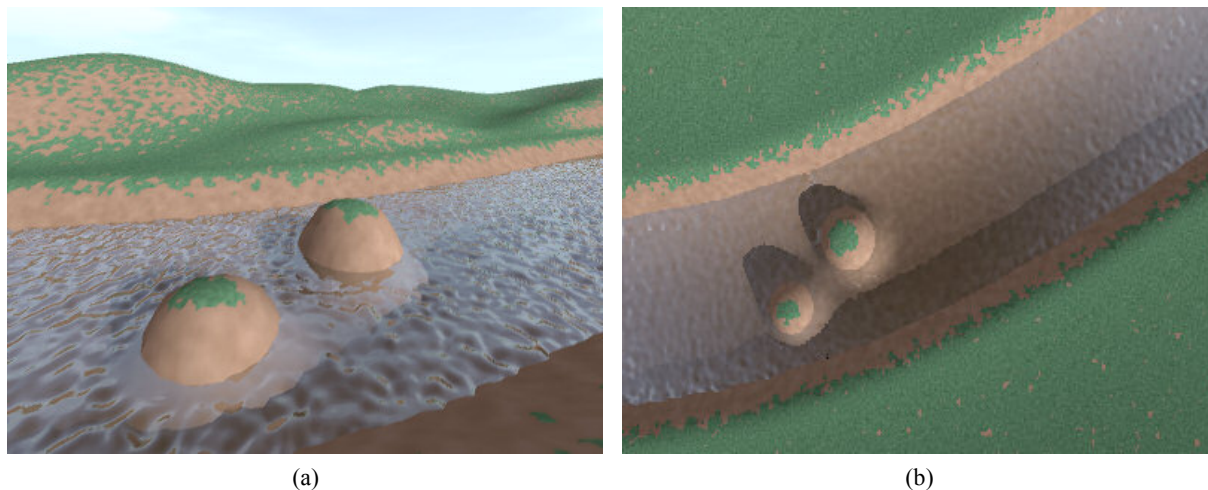


Figure 13. Vue perspective et aérienne d'un cours d'eau généré par notre modèle semi-physique, résultant de la simulation de la figure 11

#### 4.6. Conclusion

Grâce aux équations de Navier-Stokes 2D utilisées pour simuler le mouvement horizontal de l'eau, le comportement de l'eau obtenu avec notre modèle est visuellement réaliste, en dépit de la détermination de la hauteur de l'eau par une simple fonction pseudo-aléatoire.

Par conséquent, pour un réalisme visuel suffisant, on pourrait s'en arrêter à cette solution hybride, sans faire appel à un modèle physiquement plus poussé. Toutefois, si on recherche un réalisme encore plus grand, par exemple en cherchant à reproduire des effets complexes d'éclaboussures dans le cas de cours d'eau très agités, il faudra utiliser un modèle plus physique encore.

### 5. Conclusion

Le mouvement de l'eau est un phénomène très complexe. Essayer de le reproduire en synthèse d'images peut se faire selon deux approches différentes, selon que l'on s'intéresse aux conséquences du phénomène en cherchant à donner une description explicite de l'apparence de l'eau, ou que l'on s'intéresse aux causes du phénomène en cherchant à modéliser au moyen d'équations physiques les forces auxquelles l'eau est soumise. Nous avons proposé deux modèles permettant de représenter des cours d'eau, en utilisant l'une et l'autre de ces approches. Nous avons pu constater les faiblesses d'un modèle purement empirique, en dépit d'avantages en termes de vitesse de calcul et de besoins en mémoire. Bien que ce modèle empirique puisse donner des résultats visuellement acceptables dans certains cas simples, il montre ses limites quand il s'agit de modéliser des situations complexes où l'eau interagit avec des obstacles. Modéliser un comportement aussi complexe de l'eau nécessite alors de se tourner vers des modèles physiques tels que ceux qui ont été décrits en mécanique des fluides. Malheureusement, de tels modèles sont difficilement utilisables directement en synthèse d'images, pour des problèmes de besoins importants en temps de calcul et en mémoire. Il devient alors indispensable de trouver des compromis, en cherchant à simplifier ces équations. La question qui se pose alors est de savoir jusqu'à quel point ces équations peuvent être simplifiées, en trouvant un compromis entre un résultat visuellement acceptable et une simulation physiquement correcte, tout en gardant en tête que l'évolution sans cesse grandissante du matériel en terme de puissance de calcul et de capacité mémoire repousse sans cesse cette limite.

### 6. Travaux futurs

La représentation de phénomènes très turbulents tels que des tourbillons pose problème, car les équations de Navier-Stokes doivent être résolues sur une grille suffisamment fine pour pouvoir prendre en compte les effets turbulents. Afin de ne pas être confrontés à des problèmes d'occupation mémoire et de temps de calcul élevés liés à l'augmentation de la finesse de la grille, nous étudions actuellement l'utilisation de grilles adaptatives, raffinées dans les zones turbulentes du courant. De plus nous cherchons à étendre notre modèle afin qu'il puisse représenter les éclaboussures qui apparaissent dans le cas de cours d'eau très agités, ce qui n'est pas possible actuellement étant donné que nous traitons la surface de l'eau comme une simple carte de hauteurs. Enfin, pour plus de réalisme, spécialement dans le cas de cours d'eau très agités, il est nécessaire de représenter de l'écume, due au mélange de l'eau avec l'air.

## Bibliographie

- [ARQU 91] D. Arquès, N. Janey, "Cartes planaires pour la synthèse d'image d'un relief à partir de son bassin fluvial", *Actes des Journées Graphiques GROS PLAN 91*, Université des Sciences et Technologies de Lille, pp 141-148.
- [CAND 95] S. Candel, "Mécanique des fluides", *Editions Dunod*, 1995.
- [CHEN 95] J. Chen, N. Lobo, "Toward interactive-rate simulation of fluids with moving obstacles using Navier-Stokes equations", *Graphical Models and Image Processing*, 1995, pp 107-116.
- [CHIB 95] N. Chiba, S. Sanakanishi, K. Yokoyama, I. Ootawara, K. Muraoka, N. Saito, "Visual simulation of water currents using a particle-based behavioural model", *Journal of Visualization and Computer Animation*, 1995, **6**, pp 155-171.
- [FOST 96] N. Foster, D. Metaxas, "Realistic animation of liquids", *Proceedings of Graphics Interface '96*, 1996, pp 204-212.
- [FOST 97] N. Foster, D. Metaxas, "Controlling fluid animation", *Proceedings of Graphics Interface '97*, 1997, pp 178-188.
- [FOST 01] N. Foster, R. Fedkiw, "Practical animation of liquids", *Computer Graphics (Siggraph 2001 proceedings)*, 2001.
- [FOUR 86] A. Fournier, W.T. Reeves, "A simple model of ocean waves", *Computer Graphics (Siggraph '86 proceedings)*, **20**(4), 1986, pp 75-84.
- [GONZ 97] J.C Gonzato, B. Le Saëc, "A phenomenological model of coastal scenes based on physical considerations", *Computer Animation and Simulation '97*, 1997, pp 137-148.
- [KASS 90] M. Kass, G. Miller, "Rapid, stable fluid dynamics for computer graphics", *Computer Graphics (Siggraph '90 proceedings)*, **24**(4), 1990, pp 49-57.
- [MALL 95] H. Mallinder, "The modelling of large waterfalls using string texture", *The Journal of Visualisation and Computer Animation*, **6**(1), 1995, pp 3-10.
- [MAX 81] N. Max, "Vectorized procedural models for natural terrain: waves and islands in the sunset", *Computer Graphics (Siggraph '81 proceedings)*, **15**(3), 1981, pp 317-324.
- [MOUL 97] D. Mould, Y.H. Yang, "Modeling water for computer graphics", *Computers and Graphics*, **21**(6), 1997, pp 801-814.
- [MUSG 89] F.K. Musgrave, C.R. Kolb, R.S. Mace, "The synthesis and rendering of eroded fractal terrains", *Computer Graphics*, 1989, **23**(3), pp 41-50.
- [OBRI 95] J.F. O'Brien, J.K. Hodgins, "Dynamic simulation of splashing fluids", *Proceedings of Computer Animation '95*, 1995, pp 198-205.
- [PEAC 86] D.R. Peachey, "Modeling waves and surf", *Computer Graphics (Siggraph '86 proceedings)*, **20**(4), 1986, pp 65-74.
- [PERL 85] K. Perlin, "An image synthesizer", *Computer Graphics (Siggraph '85 proceedings)*, **19**(3), 1985, pp 287-296.
- [ROUD 93] P. Roudier, B. Peroche, M. Perrin, "Landscapes synthesis achieved through erosion and deposition process simulation", *Computer Graphics Forum*, 1993, **12**(3), pp 375-383.
- [SIMS 90] K. Sims, "Particle animation and rendering using data parallel computation", *Computer Graphics (Siggraph '90 proceedings)*, **24**(4), 1990, pp 405-413.

- [STAM 99] J. Stam, "Stable fluids", *Computer Graphics (Siggraph '99 proceedings)*, 1999, pp 121-128.
- [THON 00] S. Thon, J.M. Dischler, D. Ghazanfarpour, "A simple model for visually realistic running waters", *Proceedings of Eurographics UK 2000*, 2000, pp 81-86.
- [THON 01] S. Thon, D. Ghazanfarpour, "A semi-physical model of running waters", *Proceeding of Eurographics UK 2001*, 2001, pp 53-59.
- [TSO 87] P.Y. Ts'o, B.A. Barsky, "Modeling and rendering waves: wave-tracing using beta-splines and reflective and refractive texture mapping", *ACM Transactions on Graphics*, **6**(3), 1987, pp 191-214.
- [TUY 84] H. Tuy, L. Tuy, "Direct 2D display of 3D objects", *Computer Graphics and Applications*, **4**(10), 1984, pp 29-33.
- [XUSU 97] Y. Xu, C. Su, D. Qi, H. Li, S. Liu, "Physically based simulation of water currents and waves", *Computers and Graphics*, **21**(3), 1997, pp 277-280.