Rapport de stage de fin d'étude

MACS 3 Sup Galilée - Paris 13

Calage statistique du coefficient de rugosité sur TELEMAC 2D

Ngoc Bao Tran LE
MACS-3
Sup Galilée - Université Paris 13

Encadrant : Mr Merlin KELLER
Encadrant : Mr Cedric GOEURY

EDF Chatou
04 avril - 30 septembre 2016

REMERCIEMENTS

Au terme de ce projet, je souhaite tout d'abord remercier mes deux encadrants Merlin KELLER et Cédric GOEURY, qui m'ont encadrée avec patience pendant toute la durée de la réalisation de ce stage. Leurs conseils m'ont été bien utiles, notamment pour la rédaction de ce rapport.

Je t iens à remercier vivement mes collègues et tous les gens des deux départements MRI et LNHE du site EDF Chatou, qui ont pris le temps d'écouter mes questionnements et m'ont aiguillée pendant la durée du stage.

Enfin nous tenons à remercier l'ensemble du corps enseignant de la formation Mathématiques Appliqués et Calculs Scientifiques.

Table des matières

1 Introduction

1.1 Objectif et résumé du stage

Au sein de la R&D d'EDF, le département LNHE (Laboratoire National d'Hydraulique et Environnement) participe à la réalisation d'études d'impact de ses installations industrielles sur l'environnement. Dans ce contexte, il développe des outils de simulation pour caractériser les écoulements aux abords des installations industrielles. Les études sont réalisées à l'aide de la plate - forme OPENTELEMAC - MASCARET (http ://www.opentelemac.org/), comprenant le code TELEMAC2D, qui permet de modéliser respectivement, les écoulements à surface libre. Ce système couvre un large champ d'études, de la propagation de crues et modélisation des champs d'inondation au calcul d'onde de submersion résultant de rupture de barrages en passant par le transport de sédiments ou la qualité de l'eau. Les résultats numériques fournis par le code de calcul doivent être comparés à des données de terrain afin de s'assurer de la fiabilité de l'outil. Ce processus appelé validation inclut la phase dite de calage du modèle. Le calage vise à reproduire des événements de référence aussi fidèlement que possible par un ajustement de paramètres à base physique. Cette étape déterminante est fastidieuse car généralement effectuée à la main. Ce stage porte ainsi sur le calage statistique des coefficients de frottement modélisant la nature du fond d'un cours d'eau dans les codes de calcul hydrauliques En effet, ces coefficients prennent en compte les frottements des parois sur le fluide ainsi que d'autres phénomènes non modélisés.

Le principe du calage statistique est, en utilisant des données, telles que les couples Débit / Hauteur d'eau, de retrouver la valeur vraisemblable du coefficient de rugosité. Cela signifie que l'on cherche à approcher le mieux possible les hauteurs d'eau observées en utilisant les débits mesurés et la valeur du coefficient de frottement en entrée du code TELEMAC2D. Plusieurs méthodes ont ainsi été testées afin d'en évaluer les avantages et inconvénients. Dans un premiers temps, les méthodes déterministes de moindres carrés et de maximum de vraisemblance ont été mises en place. Ces méthodes permettent de trouver la valeur la plus vraisemblable du coefficient de rugosité par rapport aux données observées. Dans un second temps, des techniques issues de l'approche statistique bayésienne (approximation de Laplace, acceptation – rejet et importance sampling), ont été développées et permettent ainsi le calcul de toute la densité de probabilité du coefficient de frottement. Cependant, l'inconvénient majeur de ce type de méthode est le nombre important d'appel au code de calcul. Ainsi, si le coût de calcul est important pour le modèle hydraulique à caler, les méthodes bayésiennes nécessitent un temps important de simulation. Ainsi, approximation peu coûteuse du modèle hydraulique, autrement dit un méta – modèle, a été développée. Celui-ci doit être assez proche en termes d'évaluation du comportement du modèle hydraulique évalué par TELEMAC-2D tout en prenant peu de temps d'exécution. Cette approche a été validée sur une étude de simulation basée sur un cas analytique. Enfin, pour valoriser ces méthodes de calage, nous avons réalisé le calage sur un cas d'application réel de TELEMAC2D.

1.2 Déroulement du stage

Au début du stage, nous avons commencé par l'étude de calage statistique sur un cas hydraulique simplifié (2.1). Nous avons récupéré les données Débits / Hauteurs d'eau de l'article [4] et effectué le calage du coefficient de Strickler par les méthodes de moindres carrés, maximum de vraisemblance, approximation de Laplace, acceptation - rejet et importance sampling sur ce modèle. Pour vérifier l'efficacité de ces méthodes, nous avons comparé les valeurs obtenues du coefficient de Strickler avec celle de référence indiquée dans [4].

Dans le deuxième temps, nous avons ré-appliqué toutes les méthodes ci-dessus dans dans le cas simple de TELEMAC2D, présenté dans le paragraphe (4.1.2), le cas *"estimation"*. Cette étape a été réalisée grâce à la plate forme SALOME hydro, présenté dans le paragraphe (4.2), qui nous permet de coupler les calculs du code TELEMAC2D avec un script PYTHON. Egalement dans cette partie, nous nous sommes rendus compte que les méthodes de calage précédentes fonctionnaient bien mais le temps de calculs sur TELEMAC2D était énorme. Par exemple, pour avoir un échantillon assez grand pour acceptation - rejet ou importance sampling, il nous faut environ deux jours de calculs. Pour résoudre ce problème, nous avons créé un modèle très proche du TELEMAC2D sur le domaine étudié ($[Ks_1, Ks_2] \times [Q_1, Q_2]$) mais assez facile à construire et prend beaucoup moins de temps à fonctionner, il s'appelle le méta - modèle.

Il existe plusieurs types de méta - modèle. Dans ce travail, nous avons choisi le krigeage, c'est un méta - modèle créé par des processus gaussiens, voir le paragraphe (2.7.3). Comme ce méta - modèle est construit

à partir d'un certain nombre de données, extraits du vrai code TELEMAC2D, il est nécessaire de vérifier l'efficacité du méta - modèle avant de l'utiliser. Pour valider un méta - modèle, nous avons calculé le coefficient de prédictivité et l'erreur du méta - modèle par rapport au vrai code dont les étapes de calcul seront détaillées dans le paragraphe (2.7.1). Une fois avoir validé ce méta - modèle, nous avons refait les calculs de calage du cas analytique et du cas estimation de TELEMAC2D. Cela nous permet de comparer les résultats obtenus par le vrai code et par le méta - modèle.

Enfin, nous avons appliqué toutes ces méthodes de calage avec le méta - modèle dans le cas réel de la Garonne sur les quatre ville : La Réole, Tonneins, Marmande et Le Mas d'Agenais.

1.3 La rugosité et le coefficient de Strickler

Effectivement, l'écoulement d'eau dépend de la nature du lit de la rivière. Plus le fond de la rivière est rugueux, plus l'eau est freinée et met du temps pour se déplacer. Plusieurs hydrauliciens ont tenté de donner une formule déterministe de ce coefficient en fonction de la nature du matériau constitutif de l'interface du fond, mais aucune n'a donné une entière satisfaction. D'après l'article [17], la difficulté de déterminer ce coefficient vient de la complexité des notions masquées derrière le terme de rugosité qui désigne, pour ce qui nous concerne, la somme des influences de la rugosité *"de peau"* des matériaux constitutifs du lit, de la rugosité *"de forme"* de ces éléments et de la rugosité *"de morphologie"* ou *"d'ensemble"* de l'agencement des matériaux. La combnaison de ces influences, évidemment très mal connu, détermine l'épaisseur de la couche de matériaux du lit, dont résulte plus ou moins directement Ks.

L'illustration ci-dessous (figure (1))montre comment la rugosité *"d'ensemble"* peut sensiblement varier en agençant de deux manières différentes les mêmes éléments.

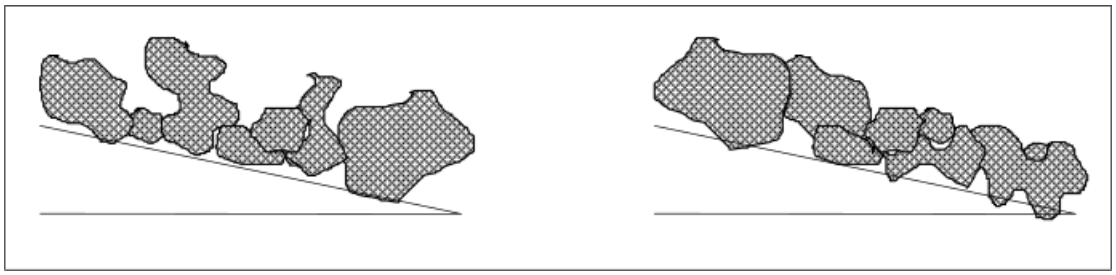


FIGURE 1: Illustration d'influence de la rugosité au fond de la rivière, source [17]

Voici quelques ordres de grandeur du coefficient de Strickler :

Nature des parois	Valeur de Ks $(m^{1/3}s^{-1})$
Béton lisse	75
Canal en terre, non enherbé	60
Canal en terre, enherbé	50
Rivière de plaine, sans végétation arbustive	35-40
Rivière de plaine, large, végétation peu dense	30
Rivière à berges étroites, très végétalisées	10-15
Lit majeur en prairie	20-30
Lit majeur en vigne ou taillis	10-15
Lit majeur urbanisé	10-15
Lit majeur en forêt	< 10

Il faut retenir que le coefficient de rugosité du lit d'une rivière varie en fonction du tirant d'eau, cela signifie qu'en fonction du débit. De plus, plus le lit de la rivière est rugueux, plus le coefficient de Strickler est petit et plus la hauteur est élevée.

2 Etude théorique du calage statistique et du méta-modèle

2.1 Modèle déterministe

D'après l'article *"Calibration d'un modèle Modelica"* [3], un modèle déterministe est décrit par une équation de la forme :

$$Y = G(X, \theta)$$

où X est le vecteur d'entrées, $G(\cdot, \theta)$ est le code déterministe (une fonction), qui dépend d'un certain paramètre incertain θ, et Y est la sortie du système. Par exemple, dans le premier cas test, nous avons utilisé le modèle hydraulique simplifié de débit/hauteur, abordé dans l'article *"Réflexions sur l'analyse d'incertitudes dans un contexte industriel : information disponible et enjeux décisionnels"* [4], obtenu par la résolution des équations de Saint-Venant 1D avec condition d'écoulement stationnaire et de section rectangulaire très large :

$$Z_c = Z_v + \left(\frac{Q}{BK_s \sqrt{\frac{Z_m - Z_v}{L}}} \right)^{3/5} \tag{2.1}$$

avec :

Z_c : la côte de la surface de la rivière en aval (en m)
Z_m et Z_v : les côtes du fond de la rivière en amont et en aval respectivement (en m)
Q : le débit (en m^3/s)
B : la largeur du cours d'eau (en m)
K_s : le coefficient de Strickler
L : la longueur du tronçon considéré (en m)

Ici, l'entrée du système est $X = Q$, nous voulons observer la hauteur d'eau en fonction du débit. Les paramètres (B, Z_m, Z_v, L) sont considérés comme fixés. Le coefficient de Strickler K_s est un paramètre incertain, $\theta = K_s$, que l'on cherche à déterminer par les méthodes de Moindres Carrés ordinaire (MCO), du maximum de vraisemblance (EMV) , et d'estimation bayésienne comme approximation de Laplacen acceptation - rejet ou importance sampling.

2.2 Incertitude épistémique

Les incertitudes épistémiques sont liées à un manque de connaissance sur un phénomène. Dans notre exemple, le paramètre θ n'est pas toujours parfaitement connu, à cause des raisons abordées dans la partie (1.3), mais nous pouvons estimer la valeur de θ à partir des couples de mesures débit/hauteur disponibles $(x_1, y_1), \cdots, (x_n, y_n)$, en supposant que les couples (x_i, y_i) vérifient la relation suivante :

$$\forall i \in \{1, \cdots, n\}, \quad y_i = G(x_i, \theta) + \epsilon_i,$$

où ϵ_i représente l'écart calcul/mesure. Nous avons supposé que l'erreur entre les calculs et les mesures suit une loi normale centrée : $\forall i \in \{1, \cdots, n\}, \quad \epsilon_i \sim \mathcal{N}(0, \sigma^2)$, où σ^2 est aussi un paramètre que nous voulons estimer.

$$f_{\epsilon_i}(t) = \frac{1}{\sqrt{2\pi\sigma^2}} e^{-\frac{t^2}{2\sigma^2}}$$

2.3 Méthode des moindres carrés

La méthode des moindres carrés permet de comparer des données expérimentales, généralement entachées d'erreurs de mesure, à un modèle mathématique censé décrire ces données. Ce modèle peut prendre diverses formes. Il peut s'agir de lois de conservation que les quantités mesurées doivent respecter. Cette méthode consiste à déterminer la valeur $\hat{\theta}$ qui minimise la somme quadratique des écarts calcul/mesure.

$$SC(\theta) \quad = \quad \sum_{i=1}^{n} (y_i - G(x_i, \theta))^2 \tag{2.2}$$

$SC(\theta)$ peut être considéré comme une mesure de la distance entre les données expérimentales et le modèle théorique qui prédit ces données. L'objectif de cette méthode est de minimiser l'écart entre les données observées et les résultats de calcul.

2.4 Calage par Maximum de vraisemblance

La méthode des moindres carrés est la méthode la plus populaire mais elle est applicable seulement aux modèles de régression, comme la relation (2.2). Mais en réalité, on ne peut pas écrire tous les modèles comme une égalité entre une fonction de régression plus un terme d'erreur. Si on connaît des informations sur les erreurs, le principe des moindres carrés ne les prend pas en compte, il s'agit de l'absence d'informations sur les erreurs des mesures. Dans ce cas, les moindres carrés ne sont pas tout simplement appropriés. Dans ce paragraphe, nous introduisons par conséquent une deuxième méthode qui est plus largement applicable que la précédente, il s'agit de l'estimation par le principe du *maximum de vraisemblance*.
L'idée fondamentale de l'estimation par maximum de vraisemblance est, comme le nom l'implique, de trouver un ensemble d'estimations de paramètres, appelé $\hat{\nu}$, telles que la vraisemblance d'avoir obtenu l'échantillon que nous utilisons soit maximisé. Nous signifions par là que la densité de probabilité jointe pour le modèle que l'on estime est évaluée aux valeurs observées de la variable dépendante, ici c'est le débit **Q**, et traitée comme une fonction de paramètres du modèle. Le vecteur $\hat{\nu}$ des estimations de vraisemblance donne alors le maximum de cette fonction.
En s'inspirant de l'article [3], nous avons supposé que les erreurs des mesures sont indépendantes entre elles et suivent une loi normale centrée $\mathcal{N}(0,\sigma^2)$. Autrement dit, elles sont $i.i.d$ de la loi $\mathcal{N}(0,\sigma^2)$. Le calage par maximum de vraisemblance consiste à déterminer $\hat{\nu}$ qui maximise la vraisemblance $\mathcal{L}(y|\nu)$ définie par :

$$\begin{aligned}
\mathcal{L}(y|\nu) &= \prod_{i=1}^{n} f_{\epsilon_i}\left(y_i - G(x_i,\theta)\right) \\
&= \prod_{i=1}^{n} \sqrt{\frac{\tau}{2\pi}} e^{-\frac{\tau(y_i - G(x_i,\theta))^2}{2}} \\
&= \left(\frac{\tau}{2\pi}\right)^{n/2} exp\left(-\frac{\tau}{2}\sum_{i=1}^{n}\left(y_i - G(x_i,\theta)\right)^2\right)
\end{aligned}$$

où $\nu = (\theta,\tau)$ et $\tau = \dfrac{1}{\sigma^2}$ est la précision des mesures. Donc on peut calculer le log de la vraisemblance :

$$log\,\mathcal{L}(y|\nu) = \frac{n}{2}log(\tau) - \frac{n}{2}log(2\pi) - \frac{\tau}{2}\sum_{i=0}^{n-1}\left(y_i - G(x_i,\theta)\right)^2 \tag{2.3}$$

Trouver $\hat{\nu}$ qui maximise (2.3) revient à minimiser l'équation (2.2) à τ fixé. Avec les deux méthodes des moindres carrés et du maximum de vraisemblance, nous obtenons toujours le même résultat pour le paramètre θ, qui est le coefficient de frottement dans le cas test.

Calcul de l'intervalle de confiance d'une quantité d'intérêt :
Comme l'estimateur du maximum de vraisemblance est asymptotiquement normal (voir l'annexe (6.2)), nous pouvons construire un intervalle de confiance pour n'importe quelle fonction ϕ, de niveau 95%, par la formule suivante, qui a été abordée dans le livre *"Asymptotic Statistics"* [8] :

$$IC_{95\%}\left(\phi_x(\nu)\right) = \left[\phi_x(\hat{\nu}) - 1.96\sqrt{\widehat{var}\left(\phi_x(\hat{\nu})\right)};\ \phi_x(\hat{\nu}) + 1.96\sqrt{\widehat{var}\left(\phi_x(\hat{\nu})\right)}\right]$$

avec $\hat{\nu}$ l'estimateur du maximum de la vraisemblance (EMV). $\widehat{var}\left(\phi_x(\hat{\nu})\right)$ est calculé par la formule suivante :

$$\widehat{var}\left(\phi_x(\hat{\nu})\right) = \nabla\phi_x(\hat{\nu})^T \frac{1}{n} I^{-1}(\hat{\nu}) \nabla\phi_x(\hat{\nu}),$$

où I est la matrice d'information de Fisher, dont l'expression générale est donnée par :

$$\begin{aligned}
I_{i,j}(\nu) &= -E_{y|\nu}\left[\frac{\partial log\,\mathcal{L}(y|\nu)}{\partial\nu_i} \times \frac{\partial log\,\mathcal{L}(y|\nu)}{\partial\nu_j}\right] \\
&= -E_{y|\nu}\left[\frac{\partial^2 log\,\mathcal{L}(y|\nu)}{\partial\nu_i\partial\nu_j}\right],
\end{aligned}$$

l'espérance étant calculée par rapport à la loi des observations y.
Dans notre cas, nous avons calculé les dérivées partielles de premier et second ordre de la fonction $log\ L$ par rapport à θ et à τ. On obtient les calculs ci-dessous :

$$\begin{aligned}
\frac{\partial log\,\mathcal{L}}{\partial\theta}(y|\nu) &= \tau\sum_{i=1}^{n}(y_i - G(x_i,\theta))\left(\frac{\partial G}{\partial\theta}(x_i,\theta)\right)\\
\frac{\partial^2 log\,\mathcal{L}}{\partial\theta^2}(y|\nu) &= \tau\sum_{i=1}^{n}\left[-\left(\frac{\partial G}{\partial\theta}(x_i,\theta)\right)^2 + (y_i - G(x_i,\theta)(x_i,\theta))\left(\frac{\partial^2 G}{\partial\theta^2}\right)\right]\\
\frac{\partial^2 log\,\mathcal{L}}{\partial\theta\partial\tau}(y|\nu) &= \sum_{i=1}^{n}(y_i - G(x_i,\theta))\left(\frac{\partial G}{\partial\theta}(x_i,\theta)\right)\\
\frac{\partial log\,\mathcal{L}}{\partial\tau}(y|\nu) &= \frac{n}{2}\frac{1}{\tau} - \frac{1}{2}\sum_{i=1}^{n}(y_i - G(x_i,\theta))^2\\
\frac{\partial^2 log\,\mathcal{L}}{\partial\tau^2}(y|\nu) &= -\frac{n}{2\tau^2}
\end{aligned}$$

Donc la matrice d'information de Fisher est :

$$\begin{aligned}
I_{1,1}(\theta,\tau) &= -E\left[\frac{\partial^2 log\,\mathcal{L}}{\partial\theta^2}\right] = \tau\sum_{i=1}^{n}\left(\left(\frac{\partial G}{\partial\theta}(x_i,\theta)\right)^2 - \left(\frac{\partial^2 G}{\partial\theta^2}(x_i,\theta)\right)E\left[y_i - G(x_i,\theta)\right]\right)\\
&= \tau\sum_{i=1}^{n}\left(\frac{\partial G}{\partial\theta}(x_i,\theta)\right)^2 \quad \text{car } (E\left[y_i - G(x_i,\theta)\right] = E\left[\epsilon_i\right] = 0)\\
I_{1,2}(\theta,\tau) &= I_{2,1}(\theta,\tau) = 0 \quad \text{pour la même raison}\\
I_{2,2}(\theta,\tau) &= \frac{n}{2\tau^2}
\end{aligned}$$

$$\Rightarrow I(\theta,\tau) = \begin{pmatrix} \tau\sum_{i=1}^{n}\left(\frac{\partial G}{\partial\theta}(x_i,\theta)\right)^2 & 0 \\ 0 & \frac{n}{2\tau^2}\end{pmatrix}$$

D'autre part, on a :

$$\phi_x(\nu) = \phi_x(\theta,\tau) = G(x,\theta)$$

alors :

$$\nabla\phi_x(\nu) = \begin{pmatrix}\frac{\partial G}{\partial\theta}(x,\theta)\\ 0\end{pmatrix}$$

Donc :

$$\widehat{var}\left(\phi_x(\hat{\nu})\right) = \frac{1}{n}\left(\frac{\partial G}{\partial\theta}(x,\theta)\right)^2 \Big/ \left[\tau\sum_{i=1}^{n}\left(\frac{\partial G}{\partial\theta}(x,\theta)\right)^2\right]$$

2.5 Procédures de Bayes :

Le principe de l'estimation bayésienne est très différent de celui des méthodes précédentes. Dans la méthode du maximum de vraisemblance, le vecteur des paramètre δ est inconnu mais reste constant, déterministe. L'estimation est menée en considérant qu'on ignore tout de δ, mis à part son ensemble de définition.

Or parfois, on dispose d'une connaissance sur δ. Cette information, dite *a priori*, peut provenir d'expériences similaires effectuées auparavant ou d'avis d'experts du phénomène étudié. Le principe de l'estimation bayésienne est de considérer que le vecteur δ est la réalisation d'un vecteur aléatoire, et d'intégrer dans sa loi de probabilité toutes les informations *a priori* dont on dispose sur lui.

En appliquant le théorème de Bayes, si δ est supposé suivre une loi *a priori* $\pi(\delta)$ et si $\mathcal{L}(y|\delta)$ est la loi conditionnelle de y conditionnellement à δ, la loi de δ conditionnelle à y, appelée la loi *a posteriori* et notée $\pi(\delta|y)$, est définie par :

$$\pi(\delta|y) = \frac{\mathcal{L}(y|\delta)\pi(\delta)}{m(y)} = \frac{\mathcal{L}(y|\delta)\pi(\delta)}{\int \mathcal{L}(y|\delta)\pi(\delta)\, d\delta} \tag{2.4}$$

Ainsi que l'explique *"Le raisonnement bayésien"* comme mentionné dans [2], notons que le dénominateur de (2.4) sert à la fois de constante de normalisation, pour que $\pi(\delta|y)$ soit effectivement une densité de probabilité, et de vraisemblance marginale pour y, utile dans la comparaison de modèles et les tests d'hypothèses. Comme ce dénominateur est uniquement défini en fonction des données y, on utilise souvent la notation de proportionnalité $\pi(\delta|y) \propto \pi(\delta)\mathcal{L}(y|\delta)$, qui signifie que la densité en δ, $\pi(\delta|y)$, est égale au produit $\pi(\delta)\mathcal{L}(y|\delta)$ à une constante près et cette constante s'obtient par l'intégration en δ : $m(y) = \int \pi(\delta)\mathcal{L}(y|\delta)d\delta$.
Dans ce travail, nous avons supposé que le coefficient de Strickler K_s suit une loi *a priori* uniforme $U([10, 100])$, que la précision τ suit une loi *a priori* exponentielle $\varepsilon(1)$ et que ces deux paramètres sont indépendants l'un à l'autre *a priori* :

$$\pi(\delta) = \pi(\theta) \times \pi(log(\tau))$$

Ici, nous allons faire un changement de variable car si on effectue directement l'estimation bayésienne sur τ, on aura une contradiction sur le support entre la loi *a priori* et la loi *a posteriori*. Donc au lieu d'effectuer les calculs sur τ, nous les avons réalisé sur $\gamma = log(\tau)$. Comme τ suit la loi exponentielle $\varepsilon(1)$, nous pouvons déterminer la loi de γ par la méthode suivante :

$$\begin{aligned} \text{Soit } \phi : \mathbb{R} \longrightarrow \mathbb{R}, \quad E\left[\phi(Y)\right] &= E\left[\phi\left(log(\tau)\right)\right] = \int_0^{+\infty} \phi\left(log(x)\right) e^{-x} dx \\ &= \int_{-\infty}^{+\infty} \phi\left(y\right) exp\left(y - e^y\right) dy \quad \text{(changement de variable } y = log(x)\text{)} \end{aligned}$$

En effet, la loi *a priori* de γ est $\pi(y) = exp(y - e^y)$ avec $y \in \mathbb{R}$. En revanche, on peut vérifier si $\pi(y)$ est bien une densité par deux étapes suivantes :

1. Positivité :
$$\forall y \in \mathbb{R}, \quad \pi(y) = exp(y - e^y) > 0,$$
2. Intégrabilité avec l'intégrale égale à 1 :

$$\begin{aligned} \int_{-\infty}^{+\infty} \pi(y)dy &= \int_{-\infty}^{+\infty} exp(y - e^y)dy \\ &= \int_0^{+\infty} exp\left(log(z) - z\right)\frac{1}{z}dz \quad \text{(changement de variable } y = log(z)\text{)} \\ &= \int_0^{+\infty} e^{-z}dz = 1 \end{aligned}$$

Donc $\pi(y)$ est bien une densité de γ et $\pi(\delta) = \pi(\theta)\pi(\gamma) = \frac{1}{90}\mathbb{I}_{[10,100]}(\theta) \times exp\left(\gamma - e^{\gamma}\right)$.

2.5.1 Approximation de Laplace

Le calcul de la constante $m(y)$ n'est pas toujours évident, ce qui rend compliquer le calcul de la loi *a posteriori*. Dans un premier temps, nous avons utilisé l'approximation de Laplace pour évaluer la loi *a posteriori* du vecteur δ.

Définition 1. ***Approximation de Laplace***
L'approximation de Laplace de $\pi(\delta|y)$ est la loi normale $N(\hat{\delta}_{MAP}; \hat{\Sigma})$ définie par :

— $\hat{\delta}_{MAP}$ *:* ***Mode a posteriori*** *(valeur de δ qui maximise $\pi(\delta)\mathcal{L}(y|\delta)$)*

— $\hat{\Sigma} = \left(-H(\hat{\delta}_{MAP})\right)^{-1}$ *où $H(\delta)$ est la matrice Hessienne :* $H_{i,j}(\delta) = \dfrac{\partial^2 log\left(\pi(\delta)\mathcal{L}(y|\delta)\right)}{\partial\delta_i\partial\delta_j}$

— *On peut montrer que cette approximation est d'autant plus précise que n est grand (normalité asymptotique de la loi a posteriori) [6]*

Un inconvénient de cette approximation est que la précision est bonne quand le nombre de données est important, tandis que dans ce cas test, nous n'avons que huit couples de données. De plus, la méthode sera plus facile à appliquer si les dérivées de la vraisemblance et la loi *a priori* sont faciles à calculer. Sinon, nous devrons calculer les dérivées $\frac{\partial^2 log\left(\pi(\delta)\mathcal{L}(y|\delta)\right)}{\partial\delta_i\partial\delta_j}$, pour tout i, j, par la méthode de différences finies. En comparant avec l'information de Fisher dans le principe du maximum de vraisemblance, le calcul de la matrice hessienne de l'approximation de Laplace ne contient pas l'espérance, qui est un peu compliqué à calculer si on utilise les différences finies.
Dans cet exemple, nous avons obtenu :

$$\mathcal{L}(y|\delta)\pi(\delta) = \left(\frac{e^\gamma}{2\pi}\right)^{n/2} exp\left[-\frac{e^\gamma}{2}\sum_{i=1}^{n}\left(y_i - G(x_i,\theta)\right)^2\right] \times \frac{1}{90} \times exp\left(\gamma - e^\gamma\right) \quad (\text{avec } \delta = (\theta,\gamma))$$

$$\Rightarrow \quad log\left(\mathcal{L}(y|\delta)\pi(\delta)\right) = \frac{n}{2}\gamma - \frac{n}{2}log(2\pi) - \frac{e^\gamma}{2}\sum_{i=1}^{n}\left(y_i - G(x_i,\theta)\right)^2 - log(90) + \gamma - e^\gamma$$

$$= \left(\frac{n}{2}+1\right)\gamma - \frac{n}{2}log(2\pi) - \frac{e^\gamma}{2}\sum_{i=1}^{n}\left(y_i - G(x_i,\theta)\right)^2 - log(90) - e^\gamma$$

Pour trouver $\hat{\delta}_{MAP}$, il faut maximiser $\pi(\delta)\mathcal{L}(y|\delta)$, cela revient à minimiser $-log\ (\pi(\delta)\mathcal{L}(y|\delta))$. Pour calculer $\hat{\Sigma}_{MAP}$, il faut calculer les dérivées partielles de seconde degré de la fonction $log\ (\pi(\delta)\mathcal{L}(y|\delta)))$:

$$\frac{\partial log(\mathcal{L}\pi)}{\partial\theta}(\theta,\gamma) = e^\gamma\sum_{i=1}^{n}\left(y_i - G(x_i,\theta)\right)\left(\frac{\partial G}{\partial\theta}(x_i,\theta)\right)$$

$$\Rightarrow \quad \frac{\partial^2 log(\mathcal{L}\pi)}{\partial\theta^2}(\theta,\gamma) = e^\gamma\sum_{i=1}^{n}\left[-\left(\frac{\partial G}{\partial\theta}(x_i,\theta)\right)^2 + \left(y_i - G(x_i,\theta)\right)\left(\frac{\partial^2 G}{\partial\theta^2}(x_i,\theta)\right)\right]$$

$$\Rightarrow \quad \frac{\partial^2 log(\mathcal{L}\pi)}{\partial\theta\partial\gamma}(\theta,\gamma) = e^\gamma\sum_{i=1}^{n}\left(y_i - G(x_i,\theta)\right)\left(\frac{\partial G}{\partial\theta}(x_i,\theta)\right)$$

$$\frac{\partial log(\mathcal{L}\pi)}{\partial\gamma}(\theta,\gamma) = \left(\frac{n}{2}+1\right) - \frac{e^\gamma}{2}\sum_{i=1}^{n}\left(y_i - G(x_i,\theta)\right)^2 - e^\gamma$$

$$\Rightarrow \quad \frac{\partial^2 log(\mathcal{L}\pi)}{\partial\gamma^2}(\theta,\gamma) = -\frac{e^\gamma}{2}\sum_{i=1}^{n}\left(y_i - G(x_i,\theta)\right)^2 - e^\gamma$$

Donc, on a :

$$\hat{\Sigma}_{MAP} = \begin{pmatrix} \frac{\partial^2 log\left(\mathcal{L}\pi\right)}{\partial\theta^2}(\theta,\gamma) & \frac{\partial^2 log\left(\mathcal{L}\pi\right)}{\partial\theta\partial\gamma} \\ \frac{\partial^2 log\left(\mathcal{L}\pi\right)}{\partial\theta\partial\gamma}(\theta,\gamma) & \frac{\partial^2 log\left(\mathcal{L}\pi\right)}{\partial\gamma^2}(\theta,\gamma) \end{pmatrix}$$

Pour inverser la matrice Hessienne, nous avons utilisé la fonction np.linalg.inv de PYTHON. Sinon, comme c'est une matrice 2×2, nous pourrions toujours appliquer la formule suivante :

$$\text{Soit } A = \begin{pmatrix} a & b \\ c & d \end{pmatrix}. \text{ Si } ad - bc \neq 0 \text{ alors } A \text{ possède un inverse, noté } A^{-1} \text{ et} \tag{2.5}$$

$$A^{-1} = \frac{1}{ad-bc}\begin{pmatrix} d & -b \\ -c & a \end{pmatrix} \tag{2.6}$$

Jusqu'à ce moment, nous avons vu les méthodes d'estimation par moindres carrés et par maximum de vraisemblance, qui nous permettent de calculer un estimateur ponctuel du paramètre inconnu. L'approximation de Laplace nous permet d'approcher la loi *a posteriori* de ces paramètres par une loi normale. Cependant, cette méthode ne fonctionne bien que dans le cas où on a beaucoup de données.

2.5.2 Méthode d'acceptation - rejet

La méthode d'acceptation - rejet permet aussi de simuler des variables ou des vecteurs aléatoires dont la loi est à densité par rapport à la mesure de Lebesgue, en s'aidant d'une autre densité, appelé *loi instrumentale*, dont on sait simuler facilement. Commençons par la description de l'algorithme.
On voudrait simuler une variable aléatoire X suivant la loi f. Il faut d'abord choisir une loi instrumentale g telle que le support de g est le même que celui de la loi f et telle que :

$$f(x) \leq M\, g(x)$$

Cette loi s'appelle la loi de proposition. Alors on peut simuler suivant f avec l'algorithme suivant :

1. Générer $X \sim g$ et $U \sim \mathscr{U}([0,1])$
2. Si $U \leq \dfrac{f(X)}{M\, g(X)}$, Accepter $Y = X$ car X suit la distribution f.
3. Sinon, rejeter X et revenir à l'étape 1

Pour suivre le raisonnement, on s'intéresse à la figure (2), abordée dans [2] et à la zone en gris sur cette figure.

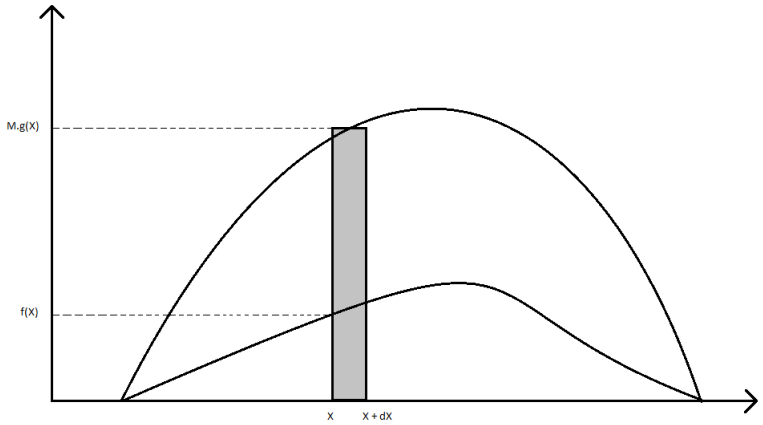


FIGURE 2: Méthode AR Acceptation - rejet

Nous pouvons vérifier que cette procédure génère effectivement une réalisation de a loi f en écrivant la fonction de répartition de la variable Y associée à la valeur acceptée. Nous remarquons tout d'abord qu'en moyenne, la probabilité de rejeter un x suivant g sera égale à :

$$\int P\left(U > \frac{f(x)}{Mg(x)}\right) g(x)\, dx = \int \left(1 - \frac{f(x)}{Mg(x)}\right) g(x)\, dx = 1 - \frac{1}{M}$$

Nous pouvons donc exprimer la densité de Y par :

$$P\left(Y | U \leq \frac{f(Y)}{Mg(Y)}\right) = \frac{P\left(U \leq \frac{f(Y)}{Mg(Y)} | Y\right) g(Y)}{P\left(U \leq \frac{f(Y)}{Mg(Y)}\right)} = \frac{\frac{f(Y)}{Mg(Y)} g(Y)}{\frac{1}{M}} = f(Y)$$

qui, tout calcul fait, vaut $f(x)$.
Ce type d'algorithme est particulièrement intéressant en analyse bayésienne. En effet, il s'applique lorsqu'on a connaissance de la densité d'intérêt à une constante près. Nous pouvons donc nous affranchir dans le cas d'une loi *a posteriori* du calcul de la prédictive en cherchant une fonction g imitant $\pi(\delta)\mathcal{L}(y|\delta)$.
Dans notre cas, nous voulons générer une variable aléatoire de la loi *a posteriori* $\pi(\delta|y)$. Or d'après le théorème de Bayes, on sait que la loi *a posteriori* est proportionnelle à $\pi(\delta)\mathcal{L}(y|\delta)$. Donc nous avons appliqué l'acceptation

- rejet avec $f = \pi(\delta)\mathcal{L}(y|\delta)$ car cette méthode fonctionne avec la densité *a posteriori* à une constance près. Nous avons choisi la loi *a priori* comme la loi de proposition, alors le quotient $\frac{f}{g}$ devient $\mathcal{L}$, dont les calculs précis seront détaillés dans la partie 3.4.

2.5.3 Méthode Importance Sampling (ou Échantillonnage préférentiel)

Dans la méthode d'acceptation - rejet, à chaque itération, on doit générer une fois le paramètre inconnu et après on vérifie si on peut garder cette valeur. Le temps de calcul sera très long si on veut générer un grand échantillon de la variable. Dans cette partie, nous présentons une deuxième méthode de génération d'un échantillon du paramètre, l'*Importance Sampling* (*Échantillonnage préférentiel* en français). Contrairement à la méthode précédente, dans l'*Importance Sampling*, on conserve toutes les valeurs du paramètre, puis affectées d'un poids.
D'après [2], cette méthode nous permet de calculer toute quantité de la forme $E_X[\phi(X)]$, où X est un vecteur aléatoire à valeur dans $\mathbb{R}^d$ de densité f par rapport à la mesure de Lebesgue sur $\mathbb{R}^d$. Elle consiste à tirer les variables aléatoires selon une distribution instrumentale g, et à compenser numériquement le résultat *a posteriori*. Nous appelons $\Phi = E_X[\phi(X)]$, alors Φ peut s'écrire comme suivant :

$$\begin{aligned}\Phi &= \int_{\mathbb{R}^d} \phi(x)f(x)dx \\ &= \int_{supp(g)} \frac{\phi(x)f(x)}{g(x)} g(x)dx \\ &= E_g\left[\frac{\phi(x)f(x)}{g(x)}\right]\end{aligned}$$

L'estimateur d'Importance Sampling est défini par :

$$\text{Pour toute densité } g,\ \hat{\Phi} = \frac{1}{n}\sum_{i=1}^{n} \phi(Y_i)\omega_i$$

$$\text{où } \omega_i = \frac{f(Y_i)}{g(Y_i)} \text{ et les variables } Y_i,\ \forall i \in \{1, \cdots, n\}, \text{ sont i.i.d de densité } g$$

Pourtant, les poids w_i peuvent des fois être supérieurs à 1. Donc pour calculer la densité de X suivant f, il faut normaliser les poids précédents en faisant :

$$\forall i \in \{1, \cdots, n\}, \quad p_i = \frac{w_i}{\sum_{k=1}^{n} w_k}$$

L'intérêt de cette méthode réside dans la possibilité d'effectuer un changement de loi qui permet de réduire la variance de l'estimateur de Monte Carlo usuel.
En revanche, si nous ne nous intéressons pas à calculer l'espérance ou la variance, mais voulons simuler un échantillon de la variable X suivant la densité f, il est possible d'appliquer une étape de ré-échantillonnage abordée dans *"Le raisonnement bayésien"* [2], qui s'appelle **Sampling Importance Resampling** (SIR). Donc l'algorithme final sera écrit comme ci-dessous :

1. Tirer n points indépendants sur Y_i suivant la densité g,
2. Tirer (avec remises) un m-échantillon de la population précédente, dont la probabilité de chaque élément est le poids normalisé calculé précédemment.

Attention : Il est déconseillé de réaliser un nombre M trop élevé de ré-échantillonnages au risque d'obtenir beaucoup de répétitions. C'est pourquoi nous avons choisi $M = ESS$, où l'ESS (Equivalence Sample Size) est donné par : $ESS = \left[\sum_{i=1}^{n}(W_i)^2\right]^{-1}$ où W_i est le poids normalisés du *ième* élément.

2.6 Plan d'expérience de l'échantillonnage par hypercube latin

Pour créer le méta-modèle, il nous faut un plan d'expériences des données. D'après l'article [7], les plans développés pour le krigeage ont été construits de façon à ce que leur points représentent au mieux le domaine expérimental. En pratique, les plans les plus utilisés sont les hypercubes latins.

En effet, l'échantillon par hypercube latin ou *Latin Hypercube Sampling* (LHS) est une méthode statistique pour générer un échantillon de collections des valeurs d'une distribution multidimensionnelle. La méthode d'échantillonnage est souvent utilisée pour construire des plans d'expériences sur l'ordinateur. Le LHS a été décrit par McKay en 1979, une autre technique équivalente a été proposée par Eglajs en 1977.
Dans le contexte de l'échantillonnage statistique, une grille carrée contenant des positions d'échantillon est un carré latin s'il n'y a qu'un seul échantillon dans chaque ligne et chaque colonne. Un hypercube latin est la généralisation de ce concept à un nombre arbitraire de dimensions, de sorte que chaque échantillon est le seul dans chaque hyperplan (voir la figure 3).

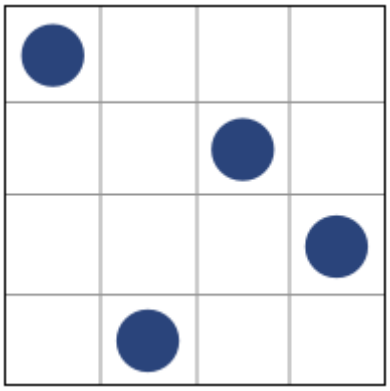


FIGURE 3: Illustration d'un hypercube latin

Les hypercubes latins présentent beaucoup d'avantages :

- Ils sont simples à construire. En effet, chaque colonne d'une hypercube est une permutation de $\{1, \cdots, n\}$.
- Les points sont uniformément distribués sur chaque axe du domaine.

La distribution uniforme sur chaque axe n'assure pas l'uniformité sur le domaine expérimental. Cependant, pour n fixé, ils existent $(n!)^d$ hypercubes latins possibles, avec d est la dimension de l'expérience. Il est donc possible de sélectionner le plan optimisant un critère d'uniformité ou bien un critère statistique. Ici, nous avons créé le plan d'expériences par le package **otlhs** sous python en utilisant le critère d'optimisation par Monte Carlo (annexe (6.8)).

Pour fabriquer le plan d'expériences par hypercube latin, nous avons supposé que le coefficient de frottement Ks suit une loi uniforme $\mathcal{U}(25, 45)$ et que le débit suit une loi uniforme $\mathcal{U}(40, 60)$. Grâce aux package **scikit** et **otlhs**, nous avons créer un plan d'expériences de la manière suivante :

```
from sklearn.guassian_process import GaussianProcess
import openturns as ot
import otlhs

bounds = ot.Interval([40,25],[60,45])
lhs = otlhs.LHSDesign(bounds, data_size)
nSimu = 50000
algo = otlhs.MonteCarloLHS(lhs,nSimu)
result  = algo.generate()
design  = result.getOptimalDesign()
```

Enfin, cela nous donne un schéma d'un plan d'expériences qui est représenté par la figure (4).

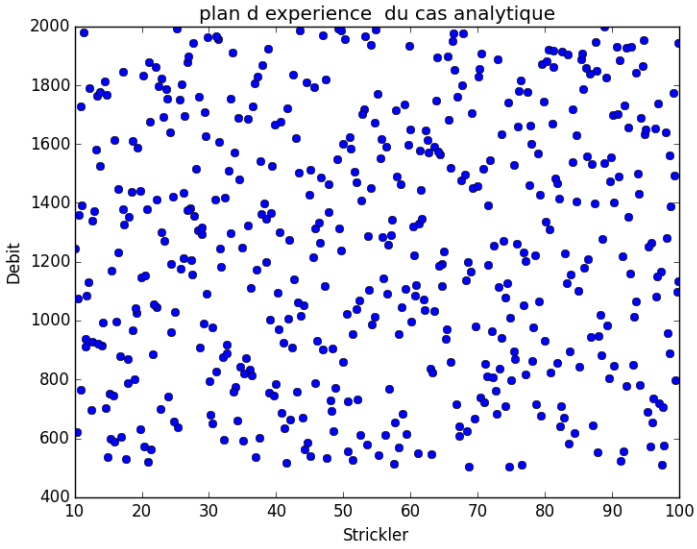


FIGURE 4: Plan d'expériences par l'échantillonnage par hypercube latin

En revanche, ces plans ne sont pas uniques car il existe de nombreuses combinaisons possibles pour les plans LHS "simples". Ils pleuvent aussi être de très mauvaise qualité, par example qu'il y a plusieurs points sur la même colonne ou même ligne. Ainsi, pour pallier ce problème, des plans LHS optimisés ont été mis en place. Plusieurs critères d'optimisation existent et les deux plus connus sont le minimax et maximin :

— Minimax : la distance maximale entre un point **x** du domaine et le point de notre plan d'expérience le plus proche de **x** est minimisée, et c'est pour tous les points du domaine **D**. Le minimax s'écrit de la façon suivante :

$$\min_{x_1,\cdots,x_N} \left[\max_{x \in D} \left(\min_i d(x_i, x) \right) \right]$$

où **D** est le domaine de l'ensemble des valeurs d'entrées possibles.

— Maximin : on va chercher à maximiser la distance minimale entre deux points de notre plan d'expérience, soit à effectuer :

$$\max_{x_1,\cdots,x_N} \left(\min_{i \neq j} d(x_i, x_j) \right)$$

2.7 Méta-modèle par la méthode de krigeage

Le krigeage est une technique géostatistique de modélisation spatiale permettant d'obtenir une représentation homogène des informations étudiées. Il est souvent utilisé pour remplacer un modèle très coûteux en temps de calcul à partir d'un certain nombre de points déjà évalués par le vrai code.
Historiquement, le terme *krigeage* provient du nom de famille de l'ingénieur minier sud-africain Daniel G.Krige. Il a été formalisé pour la prospection minière par Georges Matheron (1930 - 2000). Depuis, le domaine de ses application a largement été étendu, touchant notamment la météorologie, les sciences de l'environnement et l'électromagnétisme.
Supposons que X est l'ensemble des points du plan d'expériences créé par un hypercube latin et Y est l'ensemble des valeurs obtenues par le vrai modèle sur ces points.

$$X = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix} = \begin{pmatrix} (Q_1, Ks_1) \\ \vdots \\ (Q_n, Ks_n) \end{pmatrix}$$

$$Y = \begin{pmatrix} y_1 \\ \vdots \\ y_n \end{pmatrix} = \begin{pmatrix} G(Q_1, Ks_1) \\ \vdots \\ G(Q_n, Ks_n) \end{pmatrix}$$

Le méta-modèle par krigeage gaussien est défini par :

$$y(x) = \beta F(x) + Z(x) \tag{2.7}$$

où βF est appelé la partie de régression et Z est la partie stochastique du modèle, $Z \sim \mathcal{N}(0, c)$ avec c est une fonction de corrélation. D'après les cours de Amandine MARREL (CEA Cadarache) [11] et Bertrand IOOSS (EDF Chatou) [12], il existe plusieurs types de régression et également de la partie stochastique.

Choix de la fonction "moyenne" de régression :

- Constante : $\beta F(x) = c$,
- Polynôme de degré 1 ou Linéaire : $\beta F(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2 = \beta_0 + \beta_1 Q + \beta_2 Ks$,
- Polynôme d'ordre supérieur.

Choix de la fonction de corrélation de la partie stochastique :

- Covariance isotrope : $c(x, u) = c(\|x - u\|)$,
- Covariance anisotrope : $c(x, u) = \prod_{i=1}^{d} c_i(x_i, u_i)$ où d est la dimension du domaine,
  - Exponentielle : $c(x, u) = \sigma^2 \prod_{i=1}^{d} exp\left(-\frac{x_i - u_i}{\theta_i}\right) = \sigma^2 exp\left(-\sum_{i=1}^{d} \frac{x_i - u_i}{\theta_i}\right)$,
  - Gaussienne : $c(x, u) = \sigma^2 exp\left(-\sum_{i=1}^{d} \left(\frac{x_i - u_i}{\theta_i}\right)^2\right)$,
  - Exponentielle carrée : $c(x, u) = \sigma^2 exp\left(-\sum_{i=1}^{d} \theta_i |x_i - u_i|^2\right)$.

Ici, nous avons utilisé la fonction de régression linéaire et la fonction de corrélation exponentielle carrée. Continuons vers la prédiction de la méthode krigeage, on peut définir l'estimateur du modèle par :

Définition 2.

1. *Un prédicteur $\hat{y}(x)$ de $y(x)$ est un prédicteur linéaire s'il est de la forme $\hat{y}(x) = \sum_{i=1}^{n} c_i(x) y_i$,*
2. *Un prédicteur $\hat{y}(x)$ est un prédicteur sans biais si $\mathbb{E}(\hat{y}(x)) = \mathbb{E}(y(x))$,*
3. *Un prédicteur $\hat{y}(x)$ est le mailleur prédicteur linéaire sans biais (BLUP) s'il minimise l'erreur quadratique moyenne (Mean Squared Error ou MSE), $\mathbb{E}(\hat{y}(x) - y(x))^2$.*

Le résultat par krigeage nous donne :

$$\hat{y}(x) = \mathbb{E}(y(x)|(X, Y)) = \beta F(x) + r(x) R_{LS}^{-1} (Y - \beta F(X)) \tag{2.8}$$

$$Cov\left(y(u)|_{(X,Y)}, y(v)|_{(X,Y)}\right) = \sigma^2 \left(R(u, v) - {}^t r(u) R_{LS}^{-1} r(v)\right) \tag{2.9}$$

avec :

- $R(x, u) = \frac{c(x, u)}{\sigma^2} = exp\left(-\sum_{i=1}^{d} \theta_i |x_i - u_i|^2\right)$,
- $r(x) = (R(x_1, x), \cdots, R(x_n, x))^t$,
- $(R_{LS})_{i,j} = R(x_i, x_j)$.

La démonstration des deux équations (2.8) et (2.9) est détaillée dans l'annexe (6.10) et (6.11).

Notons que pour trouver le méta-modèle par la méthode de krigeage, nous avons besoin des paramètres (θ, β, σ). En s'inspirant du chapitre *"Metamodeling"* de [9], nous pouvons trouver ces paramètres par le principe du maximum de vraisemblance :

$$log\mathcal{L}(Y, \beta, \theta, \sigma^2) = -\frac{n}{2} log(2\pi\sigma^2) - \frac{1}{2} log(det\, R_{LS}(\theta)) - \frac{1}{2\sigma^2}\, {}^t (Y - \beta F(X))\, R_{LS}^{-1}(\theta)\, (Y - \beta F(X))$$

avec $F(x) = F((Q, Ks)) = \begin{pmatrix} 1 \\ Q \\ Ks \end{pmatrix}$.

Pour trouver l'estimation conjointe des paramètres β et σ^2, il suffit de fixer le paramètre de covariance θ et de résoudre les équations suivantes :

$$
\begin{aligned}
&\begin{cases} \dfrac{\partial log\mathcal{L}}{\partial \beta}(Y,\beta,\theta,\sigma^2) &= 0 \\ \dfrac{\partial log\mathcal{L}}{\partial \beta}(Y,\beta,\theta,\sigma^2) &= 0 \end{cases} \\
\Leftrightarrow &\begin{cases} \dfrac{1}{2\sigma^2}\,{}^t F(X) R_{LS}^{-1}(\theta)\,(Y-\beta F(X)) + \dfrac{1}{2\sigma^2}\,{}^t (Y-\beta F(X))\,R_{LS}^{-1}(\theta) F(X) = 0 \\ -\dfrac{n}{2\sigma^2} + \dfrac{1}{2(\sigma^2)^2}\,{}^t (Y-\beta F(X))\,R_{LS}^{-1}(\theta)\,(Y-\beta F(X)) = 0 \end{cases} \\
\Leftrightarrow &\begin{cases} {}^t F(X) R_{LS}^{-1}(\theta) Y - {}^t F(X) R_{LS}^{-1}(\theta)\beta F(X) + {}^t Y R_{LS}^{-1}(\theta) F(X) - {}^t(\beta F(X)) R_{LS}^{-1}(\theta) F(X) = 0 \\ \dfrac{1}{\sigma^2}\left[-n + \dfrac{1}{\sigma^2}\,{}^t (Y-\beta F(X))\,R_{LS}^{-1}(\theta)\,(Y-\beta F(X))\right] = 0 \end{cases} \\
\Leftrightarrow &\begin{cases} 2\beta\left({}^t F(X) R_{LS}^{-1}(\theta) F(X)\right) = 2\,{}^t F(X) R_{LS}^{-1}(\theta) Y \\ \dfrac{1}{\sigma^2}\,{}^t (Y-\beta F(X))\,R_{LS}^{-1}(\theta)\,(Y-\beta F(X)) = n \end{cases} \\
\Leftrightarrow &\begin{cases} \beta^* &= \left({}^t F(X)\,R_{LS}^{-1}(\theta)\,F(X)\right)^{-1}\,{}^t F(X)\,R_{LS}^{-1}(\theta)\,Y \\ \sigma^{2\,*} &= \dfrac{1}{n}\,{}^t (Y-\beta^* F(X))\;R_{LS}^{-1}(\theta)\;(Y-\beta^* F(X)) \end{cases}
\end{aligned}
\tag{2.10}
$$

Après avoir trouvé les paramètres estimés de β et de σ^2, nous pouvons trouvé l'estimateur du paramètre de corrélation θ par une des méthodes abordées dans le rapport de Amandine MARREL [14]. D'abord, on remplace les valeurs trouvées de β et de σ^2 dans le log - vraisemblance pour obtenir :

$$log\mathcal{L}(Y,\beta^*,\theta,\sigma^{2*}) = -\frac{n}{2}log(2\pi) - \frac{n}{2}log(\sigma^{2*}) - \frac{1}{2}log\left(det\, R_{LS}(\theta)\right) - \frac{n}{2}$$

Le problème devient à maximiser la fonction $\varphi(\theta)$ qui est de la forme :

$$\varphi(\theta) = -\frac{1}{2}\left[n\, log(\sigma^{2*}) + log(det\, R_{LS}(\theta))\right]$$

Au lieu de maximiser la fonction $\varphi(\theta)$, on peut minimiser la fonction $\Psi(\theta) = \sqrt[n]{exp(-2\varphi(\theta))}$. Ce problème consiste donc en une minimisation numérique dont la méthodologie utilisée est décrite dans la section 3.2 du rapport [14]. Le résultat de cette optimisation peut se reformuler ainsi :

$$\theta^* \quad = \quad \underset{\theta}{Argmin}\;\Psi(\theta) \quad \text{avec } \Psi(\theta) = (det\, R_{LS})^{1/n}\sigma^{2\,*}$$

En revanche, d'après le livre [9], un autre algorithme a été proposé par FANG, LI et SUDJIANTO dans [9] de la manière suivante :

Étape 1 : Initialiser β à la valeur $\left({}^t F(X)F(X)\right)^{-1} F(X)Y$,

Étape 2 : Pour un β donné, on met à jour le couple $\left(\sigma^2,\theta\right)$ en utilisant (2.10) et en résolvant l'équation :

$$\frac{\partial log\mathcal{L}(Y,\beta,\sigma,\theta)}{\partial\theta} = 0$$

Cette étape a besoin d'un certain algorithme d'optimisation comme *Newton-Raphson* ou *Fisher scoring*,

Étape 3 : Pour un θ donné, on met à jour β en utilisant (2.10),

Étape 4 : Répéter l'étape 2 et 3 jusqu'à la convergence.

Le krigeage est une méthode d'interpolation exacte, c'est-à-dire qu'il restitue les valeurs régionalisées mesurées aux sites d'observations. Il ne sera donc pas possible d'analyser directement les résidus du modèle car ils sont tous nuls, voir la figure (5), source *scikit-learn.org*. C'est pour cette raison qu'on utilise la validation croisée présentée dans le paragraphe (2.7.1) pour valider un modèle ou comparer deux modèles.

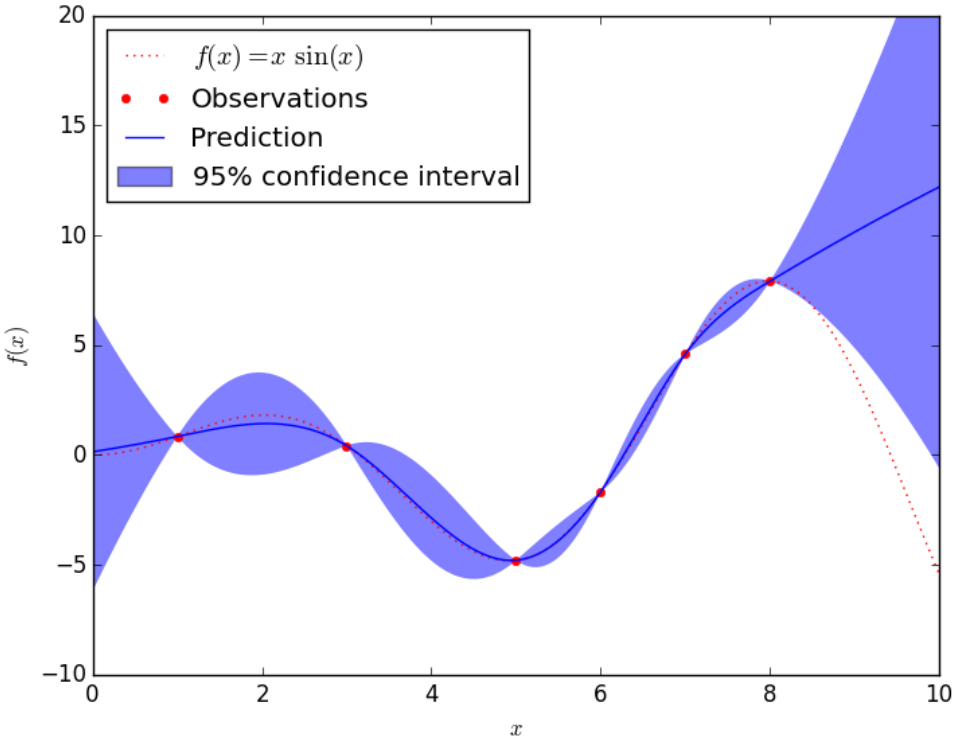


FIGURE 5: Exemple d'un méta-modèle de la fonction $f(x) = x\ sin(x)$ [13]

2.7.1 Validation du méta - modèle

Lors de ce stage, nous avons utilisé la validation croisée dite *"leave-one-out"*.
Supposons que l'on ait un échantillon d'observation $\{Y_i,\ i = 1, \cdots, n\}$.
Le processus de la validation croisée *"leave-one-out"* est le suivant :

- On retire l'observation i, notée Y_i de l'échantillon et on lance le modèle avec le nouvel échantillon,
- Puis à l'aide de ce nouveau modèle, on calcule la valeur prédite de l'observation retirée i que l'on note $\hat{Y}_i$.

On réitère ainsi ce processus sur toutes les observation dont on dispose, voir la figure (6). On obtient alors une série de valeurs prédites que l'on peut comparer aux valeurs observées en calculant le résidu du modèle.

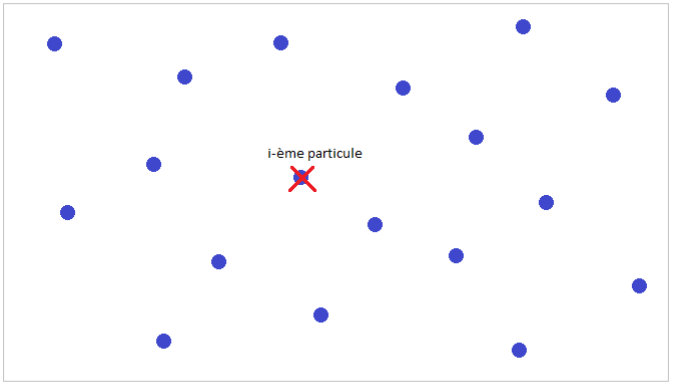


FIGURE 6: Exemple de la validation croisée d'un méta-modèle

Il est ainsi possible d'étudier les erreurs commises sur l'ensemble des données en regardant leur moyenne et leur écart-type. Cependant, l'étude des erreurs n'est pas toujours satisfaisante. En effet, l'amplitude des erreurs commises sur un site de données ne dépend pas seulement de la qualité du modèle, mais aussi de l'éloignement par rapport aux autres sites. Ainsi, un site isolé tend à produire une erreur d'estimation élevée. Cet effet peut être corrigé en examinant les erreurs standardisées, c'est-à-dire les erreurs diviser par les écart-type de krigeage. Celles-ci sont alors ramenées à la même échelle et ne sont plus sensibles à la configuration géométrique des échantillons.
Une fois la validation croisée effectuée, le calcul des différents critères permet de comparer deux modèles entre eux. Voici la liste des critères les plus utilisés dans la validation croisée :

- **Biais moyen :** $\frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i - Y_i\right)$,
- **MSE :** $\frac{1}{n}\sum_{i=1}^{n}\left(\hat{Y}_i - Y_i\right)^2$,
- **Coefficient de prédictivité :** $Q_2(Y,\hat{Y}) = 1 - \frac{\sum_{i=1}^{n}\left(Y_i - \hat{Y}_i\right)^2}{\left(\bar{Y} - Y_i\right)^2}$ où $\bar{Y}$ est la moyenne des valeurs observées,
- **Critère d'adéquation :** $\frac{1}{n}\sum_{i=1}^{n}\left(\frac{\hat{Y}_i - Y_i}{\sigma_i}\right)^2$ où σ_i^2 correspond à la variance de krigeage, $\sigma_i^2 = Var\left(\hat{Y}_i - Y_i\right)$.

2.7.2 Dérivée analytique du méta-modèle

En étudiant le package scikit learn, nous avons réussi à calculer la dérivée analytique du méta-modèle, créé par le krigeage. Pour construire le méta-modèle $\hat{y}(x)$ avec $x = (Ks, Q)$, scikit learn a basé sur les étapes suivantes :

- Normaliser l'entrée :
$$x_{norm} = \frac{x - X_{mean}}{X_{std}}$$
où x_{mean} est la moyenne normalisée des points du plan DOE et x_{std} est l'écart - type de ces points,
- Calculer la partie de régression :
$$a = \beta_0^* + \beta_1^* x_{norm}^1 + \beta_2^* x_{norm}^2 = \beta_0^* + \beta_1^* Ks + \beta_2^* Q$$
,
- Calculer la distance manhattan de l'entrée normalisée x_{norm} avec les points du plan d'expérience DOE,
$$d = d_{manhattan}(x_{norm}, X) = |x_{norm} - X|$$
- Calculer la partie de processus gaussien : on appelle $\gamma = R_{LS}^{-1}(Y - \beta^* F(X))$, qui ne dépend pas de l'entrée x. Donc la partie stochastique est calculée de la manière suivante :
$$r(x) = \begin{pmatrix} r_1 \\ \vdots \\ r_n \end{pmatrix} = \begin{pmatrix} r(x_1) \\ \vdots \\ r(x_n) \end{pmatrix} = \begin{pmatrix} exp\left(-\theta^* \sum_{i=1}^{2}(d_1)_i^2\right) \\ \vdots \\ exp\left(-\theta^* \sum_{i=1}^{2}(d_n)_i^2\right) \end{pmatrix}$$
$$= \begin{pmatrix} exp\left(-\theta^* \mid Ks_1^{norm} - Ks_1^{DOE} \mid^2 - \theta^* |Q_1^{norm} - Q_1^{DOE}|^2\right) \\ \vdots \\ exp\left(-\theta^* |Ks_n^{norm} - Ks_n^{DOE}|^2 - \theta^* |Q_n^{norm} - Q_n^{DOE}|^2\right) \end{pmatrix}$$
avec $Ks^{norm} = \frac{Ks - Ks_{mean}}{Ks_{std}}$ et $Q^{norm} = \frac{Q - Q_{mean}}{Q_{std}}$.
On pose :
$$b = r(x)^T \gamma$$

— Calculer la prédiction $\hat{y}(x)$:

$$\hat{y}(x) = Y_{mean} + Y_{std}(a + b) \tag{2.11}$$

où Y_{mean} est la moyenne normalisée des valeurs correspondants à DOE et Y_{std} est l'écart - type normalisé de ces valeurs.

Calcul de la dérivée du premier ordre :
A partir de l'équation (2.11), nous pouvons écrire la dérivée du premier ordre du méta - modèle sous la forme :

$$\frac{\partial \hat{y}}{\partial Ks}(x) = \frac{\partial \hat{y}}{\partial Ks}(Ks, Q) = Y_{std}\left(\frac{\partial a}{\partial Ks}(Ks, Q) + \frac{\partial b}{\partial Ks}(Ks, Q)\right)$$

La dérivée partielle de a par rapport à Ks peut être calculé facilement. D'autre part, celle b par rapport à Ks demande beaucoup de rigueur :

$$\frac{\partial b}{\partial Ks}(Ks, Q) = \frac{\partial r(x)^T \gamma}{\partial Ks} = \left(\frac{\partial r}{\partial Ks}(Ks, Q)\right)^T \gamma$$

avec :

$$\frac{\partial r}{\partial Ks}(Ks, Q) = \begin{pmatrix} exp\left(-\theta^* |Ks_1^{norm} - Ks_1^{DOE}|^2 - \theta^* |Q_1^{norm} - Q_1^{DOE}|^2\right) \times \left(-2\theta^*(Ks_1^{norm} - Ks_1^{DOE})\right) \times (1/Ks_{std}) \\ \vdots \\ exp\left(-\theta^* |Ks_n^{norm} - Ks_n^{DOE}|^2 - \theta^* |Q_n^{norm} - Q_n^{DOE}|^2\right) \times \left(-2\theta^*(Ks_n^{norm} - Ks_n^{DOE})\right) \times (1/Ks_{std}) \end{pmatrix}$$

où $X_{std} = (Ks_{std}, Q_{std})$. Donc la dérivée du premier ordre du méta - modèle s'écrit :

$$\frac{\partial \hat{y}}{\partial Ks}(x) = Y_{std}\left(\beta_1 + \left(\frac{\partial r}{\partial Ks}(x)\right)^T \gamma\right)$$

Calcul de la dérivée du deuxième ordre :
De la même façon, nous avons calculé la dérivée seconde du méta - modèle et obtenu :

$$\frac{\partial^2 \hat{y}}{\partial Ks^2}(x) = Y_{std}\left(\frac{\partial^2 a}{\partial Ks^2}(Ks, Q) + \frac{\partial^2 b}{\partial Ks^2}(Ks, Q)\right) = Y_{std}\left(\frac{\partial^2 b}{\partial Ks^2}(Ks, Q)\right)$$

et :

$$\begin{aligned} \frac{\partial^2 b}{\partial Ks^2}(Ks, Q) &= \left(\frac{\partial^2 r}{\partial Ks^2}(Ks, Q)\right)^T \gamma \\ &= \begin{pmatrix} exp(-\theta \sum_{i=1}^2 (d_1)_i^2) \times \left(\frac{-2\theta^*(Ks_1^{norm} - Ks_1^{DOE})}{Ks_{std}}\right)^2 + exp(-\theta^* \sum_{i=1}^2 (d_1)_i^2) \times \left(\frac{-2\theta^*}{Ks_{std}^2}\right) \\ \vdots \\ exp(-\theta \sum_{i=1}^2 (d_n)_i^2) \times \left(\frac{-2\theta^*(Ks_n^{norm} - Ks_n^{DOE})}{Ks_{std}}\right)^2 + exp(-\theta^* \sum_{i=1}^2 (d_n)_i^2) \times \left(\frac{-2\theta^*}{Ks_{std}^2}\right) \end{pmatrix}^T \gamma \end{aligned}$$

2.7.3 Convergence du méta-modèle

Le méta-modèle est créé à la base d'un plan d'expérience et nous nous intéressons à la convergence du méta-modèle vers le vrai modèle quand le nombre de point du plan d'expérience augmente. Dans le cas analytique, nous avons effectué quelques tests de la validation croisée avec des certains nombres de points différents $\{100, 500, 1000, 2000\}$. Dans la figure (7), nous avons fait varier le nombre de points du plan d'expérience et calculé l'erreur, en norme $\mathcal{L}^2$, entre la hauteur calculée par le modèle exact et celle calculée par le krigeage sur tous les points de chaque plan. Immédiatement, le test nous montre que plus on augmente le nombre de point du plan, plus l'erreur entre le modèle exact et le méta - modèle est petit.

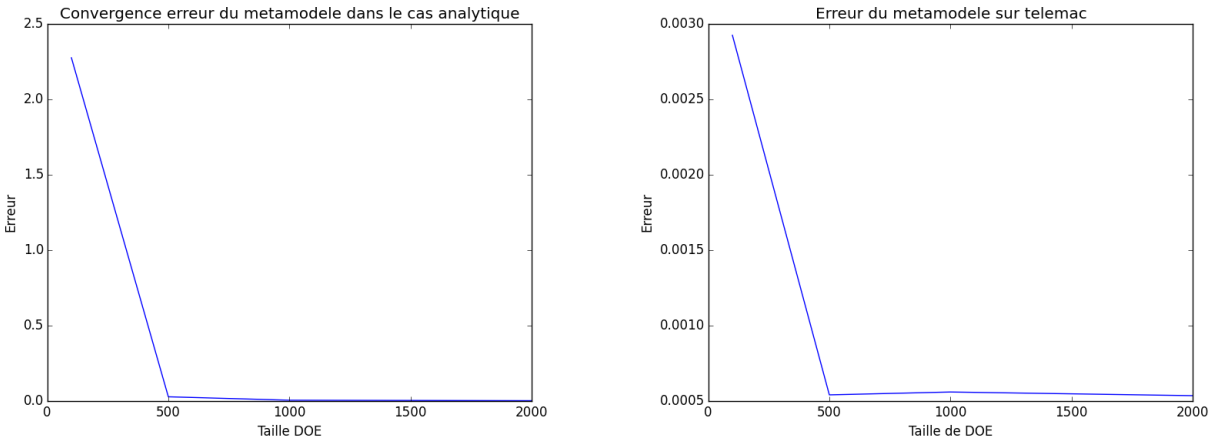


FIGURE 7: Convergence de l'erreur du méta - modèle dans le cas analytique (à gauche) et dans le cas estimation du TELEMAC2D

Plus exactement, le résultat numérique que nous avons obtenu est le suivant :

Taille de DOE	500	1000	2000
Erreur cas analytique	0.0271	0.0043	0.0012
Erreur cas estimation TELEMAC	0.0005337	0.0005339	0.0005334

On a remarqué que dans le cas estimation de TELEMAC, l'erreur du méta - modèle a légèrement augmenté quand la taille du plan d'expériences égale à 1000 points et elle redescend quand la taille atteint 2000 points, en sachant que toutes les deux erreurs sont de l'ordre 10^{-4}. Ce phénomène vient de la régularité du modèle et de la stochastique dans la création du méta - modèle. Autrement dit, si on refais deux fois le même méta - modèle, on n'aura pas exactement le même résultat.

De plus, nous avons également effectué un test sur le coefficient de prédictivité du méta - modèle. Le résultat de ce test est présenté par la figure (8), qui affirme que plus on augmente le nombre de points du plan d'expérience, plus le coefficient de prédictivité est proche de 1 et plus la prédiction du méta - modèle est bonne.

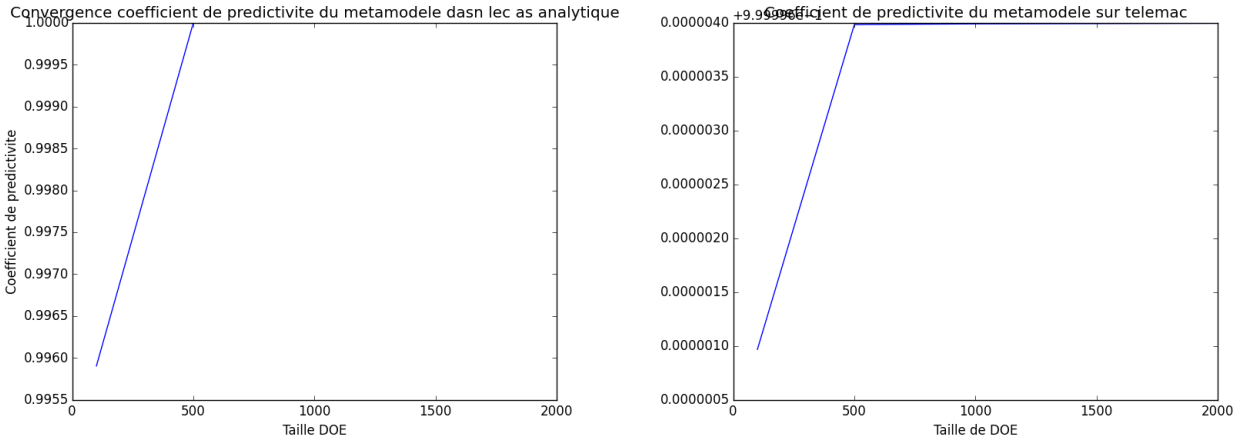


FIGURE 8: Convergence du coefficient de prédictivité du méta - modèle dans le cas analytique (à gauche) et dans le cas estimation du TELEMAC2D

3 Application dans le modèle hydraulique simplifié

Dans le premier temps, nous allons nous limiter à un tronçon de rivière dont le tracé peut être raisonnablement considéré comme rectiligne. Ainsi que l'explique *Aide mémoire d'hydraulique à surface libre* comme

mentionné dans [1], on suppose qu'il existe une seule direction de l'écoulement appelée axe de l'écoulement. Par conséquence, la surface est supposée horizontale et les composantes verticales de l'écoulement sont donc négligeables. Les paramètres géométriques essentiels de ce tronçon sont le tirant d'eau y, la section mouillée S, la largeur du tronçon L et le périmètre mouillé P, ils sont définis sur la figure (9). On note que le périmètre mouillé est la longueur de paroi en contact avec l'eau (berges et fond), mais ne comporte pas le contact eau - atmosphère.

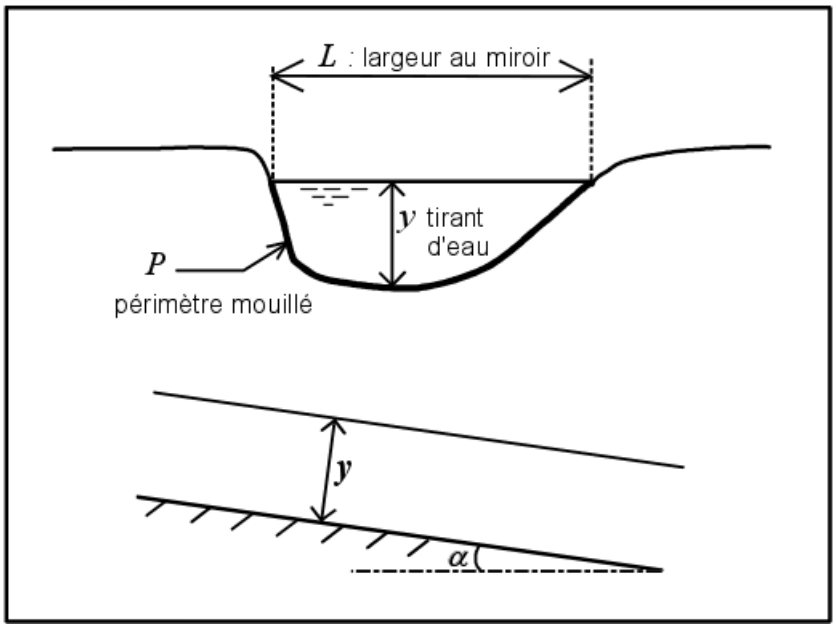


FIGURE 9: Définitions des paramètres du tronçon d'une rivière, source [1]

Le rayon hydraulique est le rapport entre la section mouillée et le périmètre mouillé, $R = \frac{S}{P}$. Pour un canal rectangulaire, $R = \frac{L.y}{L + 2.y}$. Pour un canal infiniment large, $R = y$. La pente du tronçon est la pente de son fond, mesurée tout le long de son axe, et comptée positivement si le tronçon est descendant. Elle est notée i ($i = sin\,\alpha$). Si z désigne la côte du fond, alors $i = -\frac{dz}{dx}$.

3.1 Résolution du régime permanent uniforme d'écoulement

Il existe plusieurs types de régime des cours d'eau comme régime transitoire, permanent uniforme, permanent varié, etc. Ici, nous étudions le cas le plus simple dont les caractéristiques géométriques restent constants tout le long du tronçon, le régime permanent uniforme. Un écoulement est uniforme lorsque la géométrie, la pente et la nature des parois restent inchangées et lorsque le tirant d'eau y garde une valeur constante. On appelle la charge hydraulique l'énergie par unité de poids de liquide. Par définition, en utilisant le théorème de Bernoulli (6.7), la charge à un point P d'une ligne de courant s'écrit comme :

$$\begin{aligned} H_P &= \frac{\varepsilon_P}{\rho g} = \frac{p_P + \frac{1}{2}\rho v_P^2 + \rho g z_P}{\rho g} \\ &= \frac{p_P}{\gamma} + \frac{1}{2g} v_P^2 + z_P \quad (\gamma = \rho g) \end{aligned}$$

où p_P est la pression hydrostatique au point P, ρ est la masse volumique d'eau ($1000 kg/m^3$), g est l'accélération de la pesanteur, v_P est la vitesse d'eau au point P et z_P est la profondeur du point P par rapport à la surface d'eau.

On appelle Δz la différence d'altitude entre le point P et la surface libre, y_P la distance du point P à la surface et α l'angle du fond par rapport à l'horizontale. Alors la pression au point P est $p_P = \gamma\,\Delta z$ et

$y_P = \dfrac{\Delta z}{cos(\alpha)}$ (voir la figure (10)). Donc $p_P = \dfrac{\gamma\, y_P}{cos(\alpha)}$. En réalité, la pente du fond d'une rivière est très faible ($cos(\alpha) \approx 1$), la relation précédente devient :

$$p_P = \gamma\, y_P$$

Donc, la charge hydraulique au point P devient :

$$H_P = z_P + y_P + \frac{v_P^2}{2g}$$

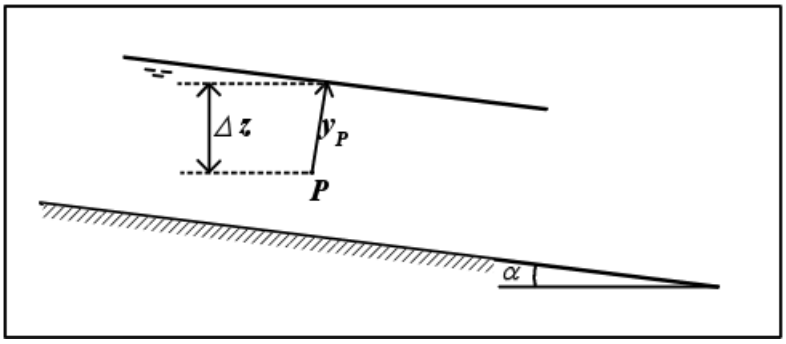


FIGURE 10: Pression en point P, source [1]

Après avoir eu la charge d'eau en un point, nous nous intéressons maintenant à la charge d'eau moyenne d'une section du tronçon. En intégrant $H_P = z_P + y_P + \dfrac{v_P^2}{2g}$ dans une section, nous obtenons $H = z_f + y + \dfrac{\beta V^2}{2g}$, où z_f désigne la côte du fond et y désigne le tirant d'eau pour la section. Le coefficient β vaut 1 si la répartition des vitesses dans la section est uniforme. En général, β est compris entre $[1, 1.2]$ dans une rivière. Dans le cas de régime permanent et uniforme, la vitesse reste constante tout le long du tronçon, alors β vaut 1.

D'autre part, la pente de la surface est aussi égale à i car le tirant d'eau est constant dans l'espace. En réalité, le fluide n'est jamais parfait alors entre une section 1 et une section 2, il existe toujours une perte de charge, notée $\Delta H = j\Delta x$. La perte de charge linéaire est donc :

$$j = -\frac{dH}{dx} = -\frac{d}{dx}\left(z_f + y + \frac{V^2}{2g}\right) = -\frac{d\, z_f}{dx}$$

(car y et V sont constants tout le long du tronçon)

On en déduit que $j = i$. Ensuite, nous allons chercher la relation entre le tirant d'eau y et le débit Q. En sachant que l'écoulement dans notre cas est uniforme, nous pouvons écrire l'équation d'équilibre des forces appliquées sur la masse d'eau entre deux sections d'une distance l :

$$\sum_i \boldsymbol{F_i} = \mathbf{0}$$

où $\boldsymbol{F_i}$ est l'ensemble des forces appliquées sur la masse d'eau considérée. Dans ce cas, ces forces comprennent le poids de la masse d'eau, le frottement sur les parois et la résistance du lit (voir la figure (11)). Comme le poids de la masse d'eau est vertical, on peut le projeter sur un axe parallèle à la pente, appelons la projection tangentielle du poids, et sur un axe perpendiculaire à la pente, appelons la projection normale. Effectivement, la projection normale du poids et la résistance du fond s'annulent. Notons $\mathcal{P}$ le poids cette masse d'eau, alors $\mathcal{P} = mg$ avec m la masse. Avec des changements de variables, on obtient $m = \rho S\, l$ alors $\mathcal{P} = \rho S\, l\, g = \gamma S\, l$, et la projection tangentielle du poids sera $\mathcal{P}_t = \mathcal{P} sin(\alpha) = \gamma S\, l\, i$ car i est la pente descendante.

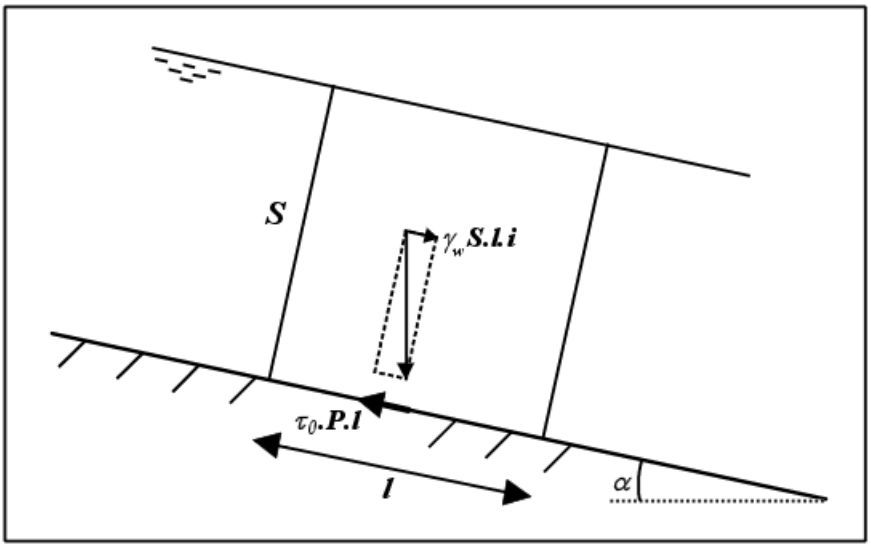


FIGURE 11: Forces appliquées sur une masse d'eau, source [1]

D'autre part, sur le même volume d'eau, le frottement sur les parois s'écrit comme $\tau_0 Pl$, où τ_0 est le frottement par unité de surface, P est le périmètre mouillé. Avec les deux relations précédentes, nous en déduisons :

$$\begin{aligned} & \gamma S\, l\, i = \tau_0 P\, l \\ \Leftrightarrow \quad & \tau_0 = \frac{\gamma S\, l\, i}{P\, l} \\ \Leftrightarrow \quad & \tau_0 = \frac{\gamma S\, i}{P} \\ \Leftrightarrow \quad & \tau_0 = \gamma R\, i \quad (\text{car } R = \frac{S}{P}) \end{aligned}$$

Or en utilisant le paragraphe 1.4 de l'article [1], nous avons :

$$\begin{aligned} & \tau_0 = \gamma \left(\frac{V}{C}\right)^2 \quad (C = KsR^{1/6} \text{ est le coefficient de Chézy qui s'exprime en } m^{1/2}s^{-1}) \\ \Leftrightarrow \quad & R\, i = \left(\frac{V}{C}\right)^2 \\ \Leftrightarrow \quad & V = Ks\, R^{2/3} i^{1/2} \\ \Leftrightarrow \quad & Q = Ks\, S\, R^{2/3} i^{1/2} \quad (\text{car } Q = \frac{S}{V}) \end{aligned}$$

Dans cette relation, R et S sont des fonctions du tirant d'eau y. Pour une rivière très large et de forme rectangulaire, le rayon hydraulique vaut à peu près le tirant d'eau. On en déduit :

$$\begin{aligned} & Q = Ks\, L\, y^{5/3} i^{1/2} \\ \Leftrightarrow \quad & y = Q^{3/5} Ks^{-3/5} L^{-3/5} i^{-3/10} \end{aligned}$$

Cela aboutit à l'expression (2.1) que l'on va utiliser dans le premier cas test du calage. Avant de réaliser le calage sur TELEMAC 2D, nous avons mis en place tous les algorithmes précédents dans le cas beaucoup plus simple du 2.1. L'étude du modèle simplifié nous permet d'avoir une idée sur le fonctionnement de ces algorithmes et de comparer leurs résultats.

3.2 Comparaison de la méthode des moindres carrés et du maximum de vraisemblance

Dans le cas test simplifié, nous avons travaillé sur un ensemble de huit couples de mesures de débit/hauteur d'eau qui sont donnés dans le tableau ci-dessous :

Débit (m^3/s)	616	724	835	925	1060	1200	1470	1560
Hauteur (m)	102.85	103.25	103.65	103.95	104.35	104.75	107.94	105.23

Après l'optimisation de la fonction *MoindreCarre* avec un point initial $\theta_0 = Ks_0 = 50.33$, nous avons obtenu l'estimateur du coefficient de frottement suivant :

$$\widehat{Ks_{OLS}} = 59.33$$

C'est la valeur qui décrit le mieux les données à travers le modèle précédent car elle minimise la somme quadratique des déviations des mesures au prédictions de $G(x, \theta)$.
Ensuite, nous avons également réalisé les estimations du coefficient de frottement et de la précision des mesures $\tau = \frac{1}{\sigma^2} = e^{\gamma}$ par le principe du maximum de vraisemblance. De la même façon, nous avons minimisé à partir du point initial $\delta_0 = (\theta_0, \tau_0) = (50.33, 3.5)$ l'opposé de la vraisemblance $\mathcal{L}(y|\delta)$ qui est décrit par 2.3 : Le résultat obtenu est très proche de celui obtenu par les moindres carrés :

$$\widehat{\delta_{MLE}} = \left(\widehat{Ks_{MLE}}, \widehat{\tau_{MLE}}\right) = (59.33, 1.45)$$

La coïncidence de la valeur du coefficient de frottement est bien expliquée quand on compare les deux expressions du logarithme de la vraisemblance et des moindres carrés. La formule du logarithme de la vraisemblance contient la somme quadratiques des déviations des moindres carrés, en raison de cela, nous avons obtenu la même valeur dans les deux cas tests.
De plus, en appliquant les formules du paragraphe (2.4), nous avons mis en œuvre les calculs de l'intervalle de confiance de l'estimateur $\widehat{Ks_{MLE}}$ de niveau 95%, dont le résultat est donné par la figure (12) ci-dessous.

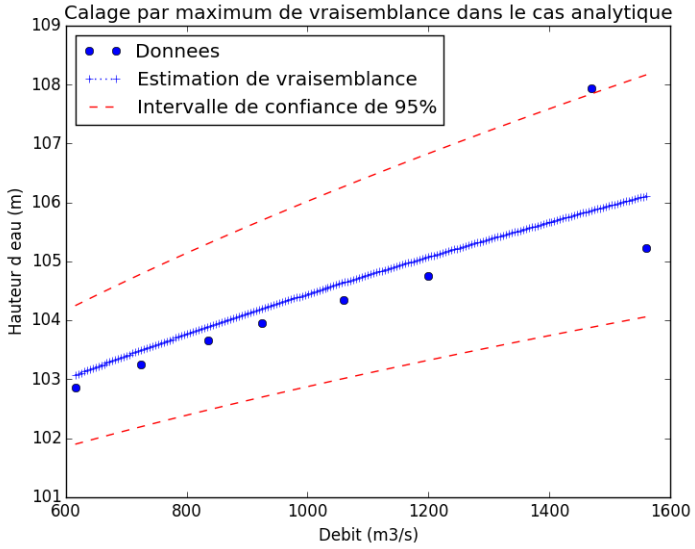


FIGURE 12: Comparaison des données avec les calculs par le calage du maximum de vraisemblance

Dans cette partie, nous avons également calculé l'intervalle de confiance de niveau 95%. L'article [3] a parlé de la formule suivante :

$$IC_{95\%}(Ks) = \left[Ks_{MLE} - 1.96\sqrt{\frac{1}{n}I^{-1}_{1,1}(Ks_{MLE})}; Ks_{MLE} + 1.96\sqrt{\frac{1}{n}I^{-1}_{1,1}(Ks_{MLE})}\right]$$

où I est la matrice information de Fisher que l'on a calculée précédemment. Avec cette formule, nous avons obtenu l'intervalle de confiance du paramètre de Strickler suivant :

$$IC^{MLE}_{95\%}(Ks) = [30.79103, 87.87586]$$

3.3 Résultat par approximation de Laplace

L'approche bayésienne fournit une distribution d'incertitudes du paramètre mal connu et une des méthodes la plus facile à appliquer pour l'obtenir c'est l'approximation de Laplace. En appliquant le résultat du paragraphe (2.5.1), nous avons trouvé facilement le couple :

$$\widehat{\delta_{MAP}} = (\widehat{Ks_{MAP}}, \widehat{\tau_{MAP}}) = (59.33,\ 1.453)$$

et la matrice de covariance :

$$\widehat{\Sigma_{MAP}} = \begin{pmatrix} 28.908 & 1.5079 \times 10^{-8} \\ 1.5079 \times 10^{-8} & 0.199 \end{pmatrix}$$

Il est immédiat de voir que les coefficients de frottement Ks obtenus par le maximum de vraisemblance (ou également par les moindres carrés) et par l'approximation de Laplace sont les mêmes. Cependant, la précision a été légèrement modifiée. Les figures (13) et (14) représentent les distributions du couple (θ, τ) par l'approximation de Laplace.

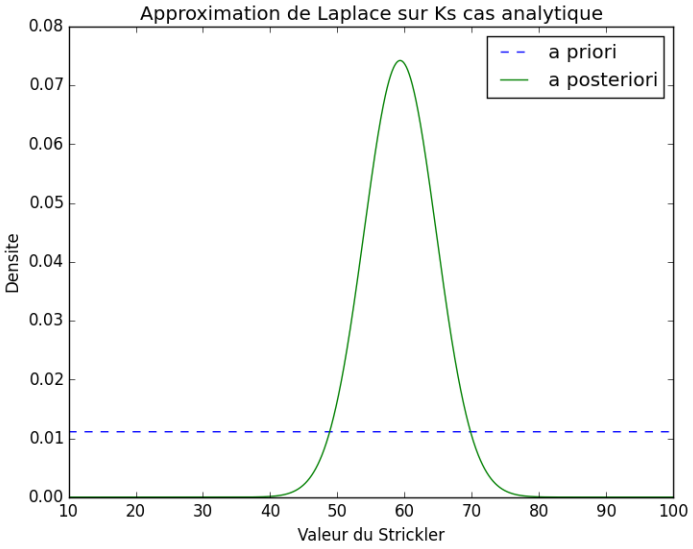


FIGURE 13: Approximation de Laplace du coefficient de frottement Ks

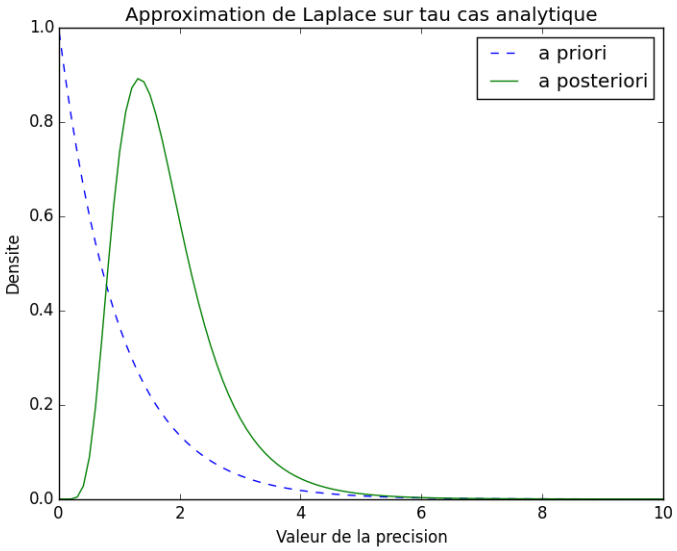


FIGURE 14: Approximation de Laplace de la précision τ

La loi *a posteriori* du couple (Ks, γ) d'après Laplace suit une loi gaussienne bidimensionnelle. Cela signifie que chacun suit une également une loi normale de paramètre :

$$\begin{aligned} Ks &\sim \mathcal{N}\left(\widehat{Ks_{MAP}}, \widehat{\Sigma_{MAP}}(1,1)\right) = \mathcal{N}\left(59.33, 28.908\right) \\ \tau &\sim \mathcal{N}\left(\widehat{\tau_{MAP}}, \widehat{\Sigma_{MAP}}(2,2)\right) = \mathcal{N}\left(1.453, 0.199\right) \end{aligned}$$

Donc, on peut calculer facilement l'intervalle de confiance du coefficient de Strickler en appliquant la formule :

$$\begin{aligned} IC_{95\%}^{MAP}(Ks) &= \left[\widehat{Ks_{MAP}} - 1.96\sqrt{\frac{\widehat{\Sigma_{MAP}}(1,1)}{n}}; \widehat{Ks_{MAP}} + 1.96\sqrt{\frac{\widehat{\Sigma_{MAP}}(1,1)}{n}}\right] \\ &= [48.795, 69.872] \end{aligned}$$

Pour bien observer l'intervalle de confiance, nous avons coloré la zone limité par cet intervalle et la courbe de distribution approchée par la définition de Laplace dans le figure (15) ci - dessous :

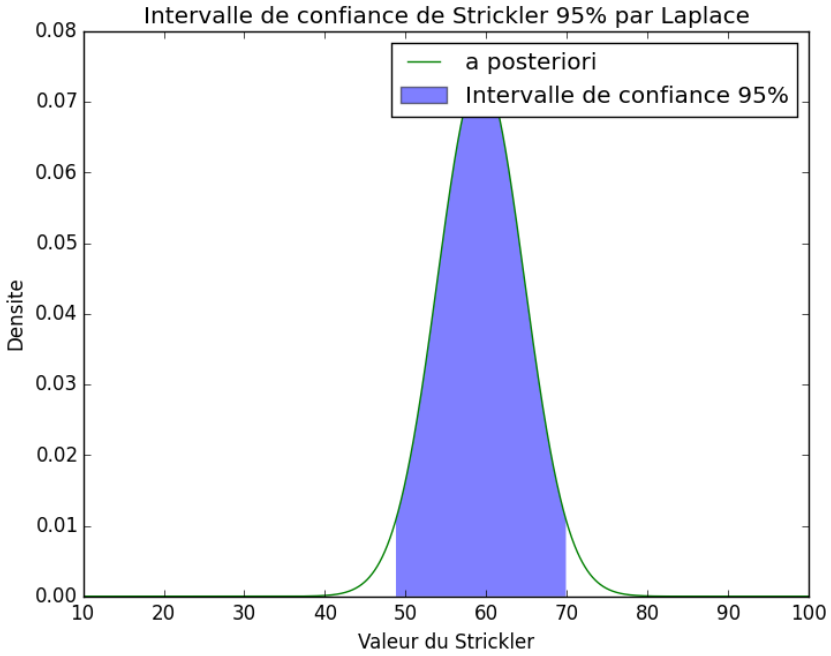


FIGURE 15: Intervalle de confiance au niveau 95% avec approximation de Laplace dans le cas analytique

3.4 Résultat de l'acceptation - rejet

Comme nous l'avons dit dans la partie précédente, les estimations par moindres carrés et par maximum de vraisemblance nous permettent seulement de trouver la valeur qui "approche" le mieux les données dans certains sens. Dans le paragraphe 2.5.2, nous avons présenté la méthode *acceptation - rejet* dont l'intérêt c'est qu'on n'a pas besoin d'un grand nombre de données pour simuler les paramètres inconnus.
Le code, que nous avons réalisé, est basé sur les mêmes étapes que nous avions écrit dans la partie 2.5.2. Par contre, pour éviter la facilité de calcul avec la vraisemblance, nous avons passé par le logarithme. En effet, nous avons trouvé la valeur $log(M) = -9.86$ qui est le maximum de la fonction $log(\mathcal{L})$.
Pour générer un échantillon de δ suivant la loi *a posteriori*, il faut d'abord simuler un vecteur uniforme U de la loi $\mathcal{U}(0,1)$ et un vecteur X suivant la loi *a priori* de la même taille. Après avoir calculé $log(U)$, on applique l'algorithme ci-dessous :

Pour $i = 1 \cdots, m$
  Si $log(U[i]) \leq log(\mathcal{L}(D, X[i])) - log(M)$, accepter $\delta = X[i]$

Un inconvénient de cette méthode est que le taux d'acceptation est parfois très faible. Supposons que nous souhaiterons simuler un échantillon de la variable δ de taille nb, le nombre total de simulation de la variable U et le vecteur X est beaucoup plus grand que nb. Dans ce cas test, nous avons obtenu le taux d'acceptation qui vaut $\frac{100000}{1530057} \approx 0.06535$, cela signifie que pour obtenir un 100000-échantillon de δ, il faut simuler un 1530057-échantillon de U et de X, ce qui va prendre beaucoup de temps de calcul.

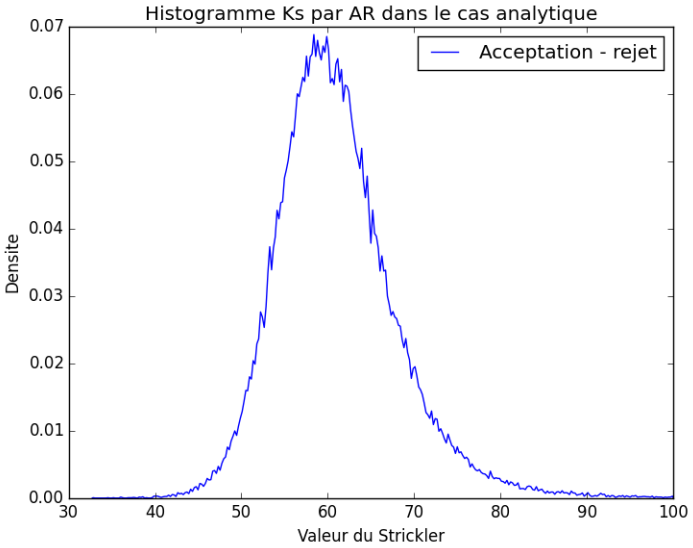


FIGURE 16: Génération du coefficient de frottement par acceptation - rejet

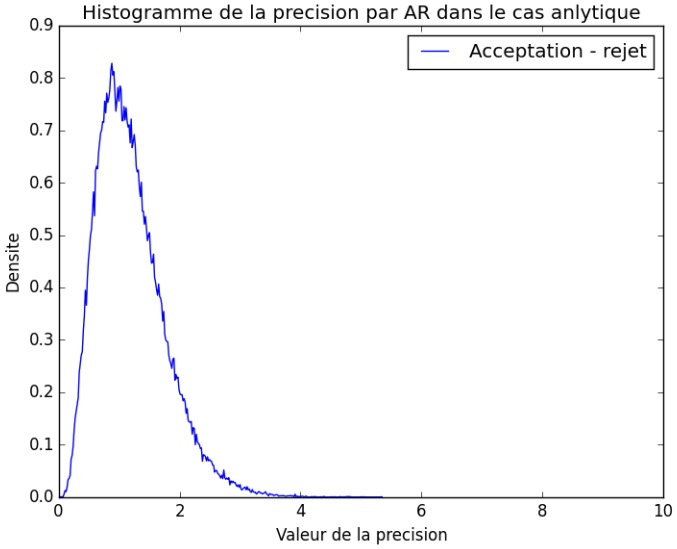


FIGURE 17: Génération de la précision par acceptation - rejet

Les figures (16) et (17) représentent les histogrammes d'un 100000-échantillon de δ par la méthode d'acceptation - rejet. Dans ces figures, on peut voir que le point $(\theta, \tau) = (59.33, 1.45)$ est le point qui a la probabilité la plus élevée, ce qui coïncide avec le résultat du maximum de vraisemblance et des moindres carrés. D'autre part, nous avons remarqué également que les distributions obtenues ressemblent beaucoup plus au résultat de l'approximation de Laplace que les lois *a priori*.
D'autre part, pour calculer l'intervalle de crédibilité du Strickler, nous avons calculé les quantiles empiriques au niveau de 2.5% et 97.5%. Ici, nous avons utilisé la fonction *computeQuantile* de OPENTURNS, mais l'algorithme

qui permet de calculer les quantiles empiriques est expliqué dans l'annexe (6.6). Du coup, l'intervalle de crédibilité obtenu par *acceptation - rejet* est le suivant :

$$IC_{95\%}^{AR}(Ks) = [49.417, 77.2134]$$

De la même façon, nous avons obtenu la figure (18) qui représente l'emplacement de cet intervalle de crédibilité sur la distribution du paramètre Ks par l'acceptation rejet.

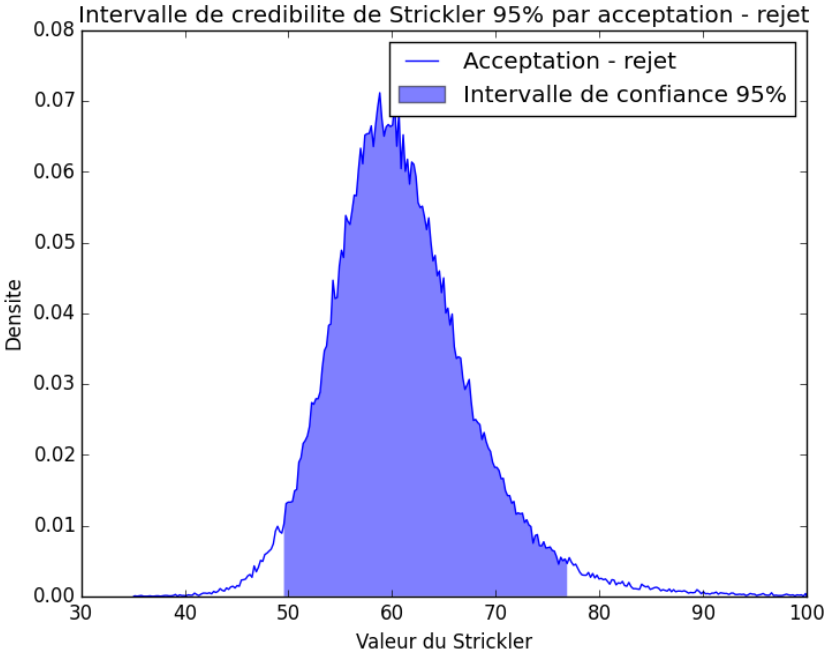


FIGURE 18: Intervalle de crédibilité de niveau 95% par acceptation - rejet dans le cas analytique

3.5 Adaptation à l'Importance Sampling

La méthode d'acceptation - rejet nous permet d'obtenir une distribution du paramètre inconnu mais elle n'est pas très efficace en terme de temps de calcul. Pour cette raison, nous avons appliqué une nouvelle méthode bien connue qui s'appelle *Importance Sampling* qui a été détaillée dans la partie 2.5.3. Dans ce cas test, nous avons choisi la loi *a priori* comme la loi instrumentale g et la loi *a posteriori* qui est égale à $\mathcal{L}(y|\delta)\pi(\delta)$ à une constance près.

Donc le quotient $\frac{f}{g}$ devient :

$$\begin{aligned} \frac{f}{g} &= \frac{\mathcal{L}(y|\delta)\pi(\delta)}{\pi(\delta)} \\ &= \mathcal{L}(y|\delta) \end{aligned}$$

Ensuite, comme on ne travaille jamais directement avec la vraisemblance, nous avons réalisé tous les calculs sous le logarithme comme avec l'acceptation - rejet. Après avoir généré un échantillon de taille ρ du vecteur δ suivant la loi g, on peut calculer le poids de chaque particule $log(w_r) = log(\mathcal{L}(y|\delta_r)\pi(\delta_r)) - log(g(\delta_r)), \quad \forall r \in \{1, \cdots, \rho\}$ et les poids normalisés par la formule 6.1 (voir l'annexe 6.3).

Passons à l'étape de ré-échantillonnage avec remise. Dans cette étape, nous avons besoin d'un vecteur qui contient des poids normalisés cumulés, noté $P = [log(p_1), \log(p_1 + p_2), \cdots, log(p_1 + \cdots + p_\rho)]$. En appliquant la

même méthode que les calculs précédents, nous avons :

$$\begin{aligned} \forall t \in \{1, \cdots, \rho\}, \quad log\,(p_1 + \cdots + p_t) &= log\left[\max_{r\in\{1,\cdots,t\}}(p_r)\left(\sum_{r=1}^{t}\frac{p_r}{\max\limits_{r\in\{1,\cdots,t\}}(p_r)}\right)\right] \\ &= \max_{r\in\{1,\cdots,t\}}(log(p_r)) + log\left(\sum_{r=1}^{t} exp\left(log(p_r) - \max_{r\in\{1,\cdots,t\}}(log(p_r))\right)\right) \end{aligned}$$

Ensuite, nous avons généré un ω-échantillon d'une variable uniforme U, avec ω calculé par l'expression ESS, et le ré-échantillonnage se déroule de la manière suivante :

$$\forall r \in \{1, \cdots, \omega\},\ si\ P_\alpha < log(U_r) < P_{\alpha+1},\ on\ accepte\ le\ couple\ \delta_{\alpha+1} = (\theta_{\alpha+1}, \gamma_{\alpha+1}).$$

On peut résumer l'étape de ré-échantillonnage par la figure 19.

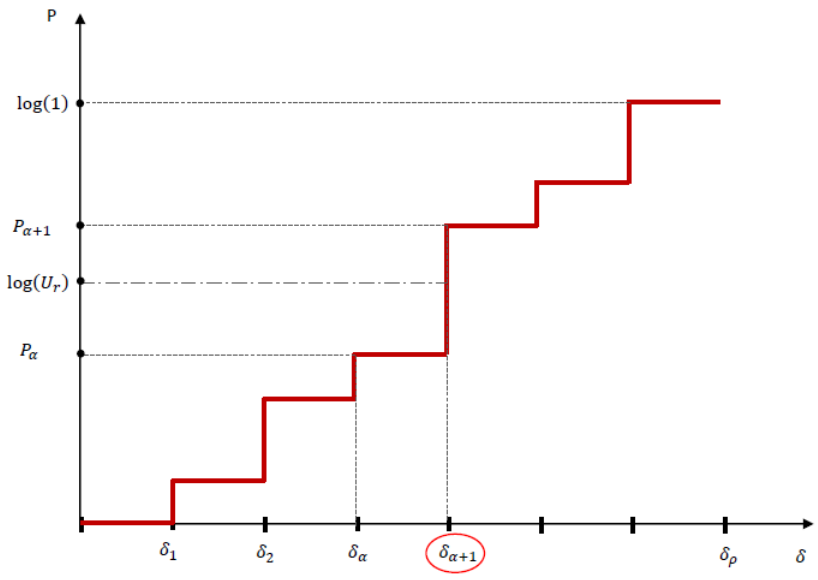


FIGURE 19: Ré-échantillonnage par Importance sampling

Dans le cas test simplifié, nous avons généré un ensemble de 1530057-échantillon du couple (θ, γ). Nous avons choisi ce nombre pour la facilité de comparaison entre l'acceptation - rejet et l'Importance sampling. Le résultat du code montre que l'Importance sampling est beaucoup plus efficace en temps de calcul et le taux d'acceptation est à 100%. Après avoir un échantillon de γ, nous n'avons pas tracé directement la distribution de γ mais celle de la précision $\tau = e^\gamma$ afin de respecter la cohérence avec les paragraphes précédents.Les figures (20) nous montrent une distribution "complète" du coefficient Ks et de la précision τ, dont la probabilité de chaque point est le poids normalisé que l'on a calculé précédemment.

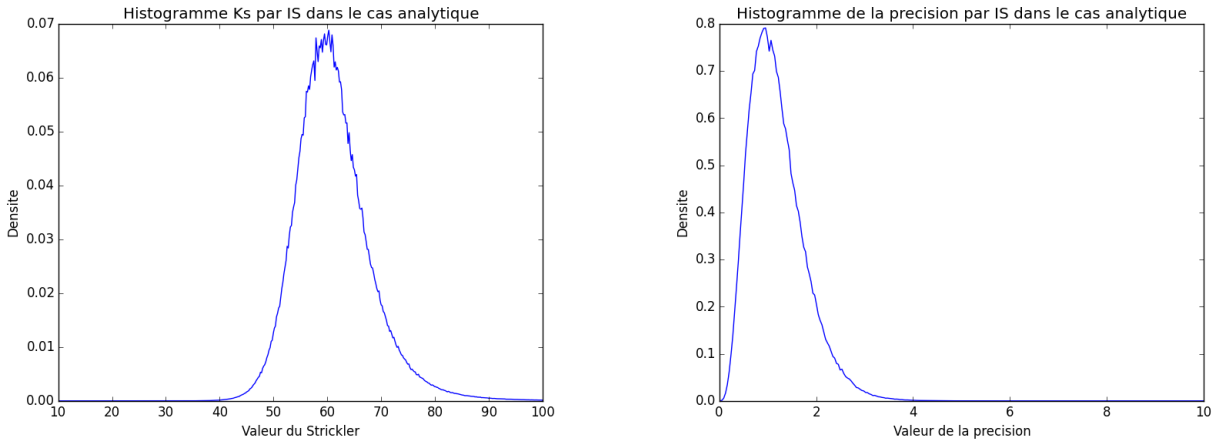


FIGURE 20: Distribution de Ks et de τ par *Importance Sampling*

Après avoir réalisé l'étape de ré-échantillonnage, nous avons obtenu les figures (21). D'après [2], l'avantage de cette méthode est d'obtenir un "véritable" échantillon (tous le point générés retrouvent le même poids). Son défaut est de produire un grand nombre de *doublons*, ce qui appauvrit les possibilités d'exploitation du support de la loi *a posteriori*.

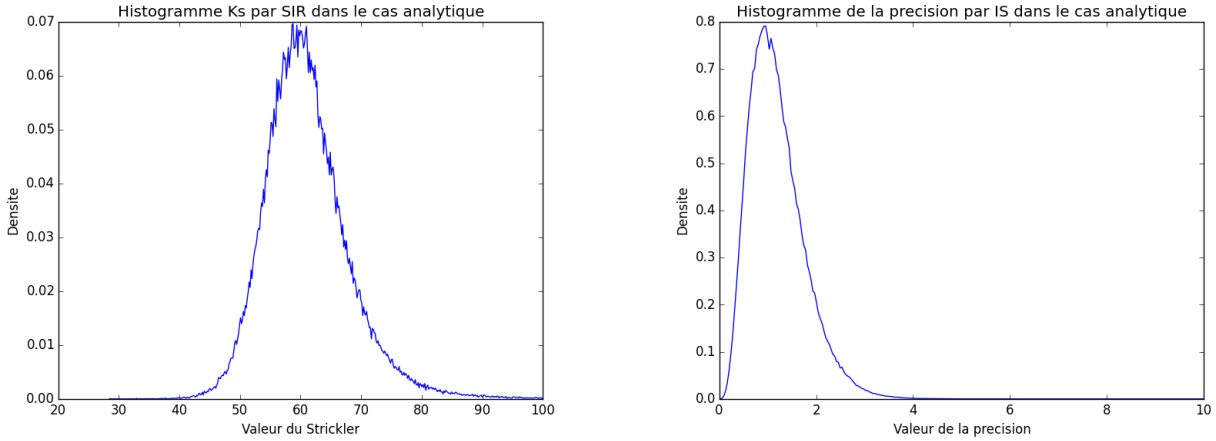


FIGURE 21: Distribution de Ks et de τ par *Sampling Importance Resampling*

Avec *Importance Sampling*, le calcul de l'intervalle de crédibilité est un peu différent car chaque élément dans l'échantillon ne porte pas le même poids comme *l'acceptation - rejet*. Ici, pour calculer cet intervalle, il faut d'abord mettre dans l'ordre croissant l'échantillon Ks que nous avons obtenu. Ensuite, nous avons réorganisé le poids suivant le nouvel échantillon croissant, cela nous donne une distribution de cet échantillon. A partir de là, nous pouvons calculer le poids normalisé cumulé P_{reorg} qui correspond à la fonction de répartition de cet échantillon. Enfin, nous avons calculé les quantiles empiriques de l'ordre 0.025 et 0.975 en sachant que :

$$q_{0.025} = P^{-1}_{reorg}(0.025) = 49.4452170641$$
$$q_{0.975} = P^{-1}_{reorg}(0.975) = 77.1521253695$$

Cela a abouti à la figure (22) ci - dessous.

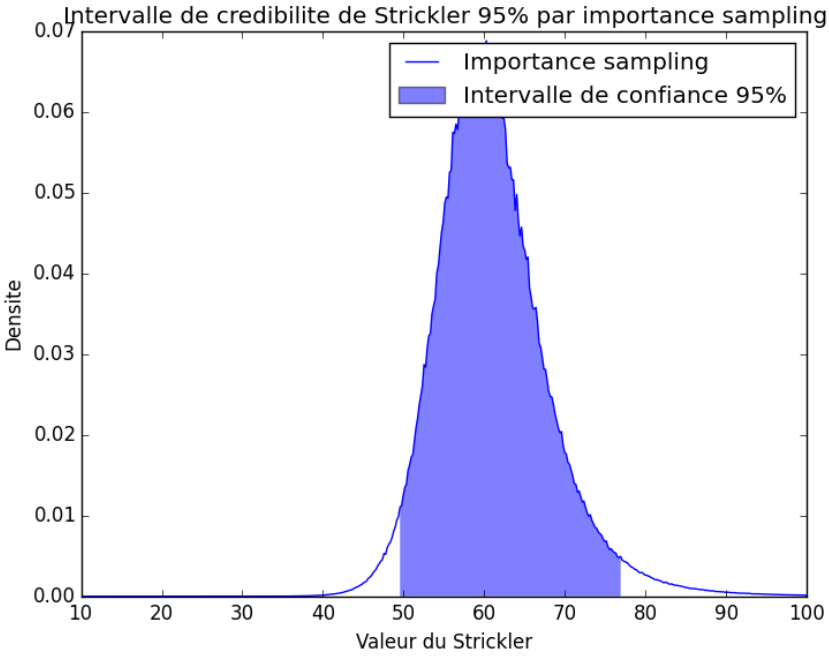


FIGURE 22: Intervalle de crédibilité de niveau 95% par importance sampling dans le cas analytique

3.6 Comparaison entre les résultats du méta-modèle et du vrai code dans le cas analytique

Dans un premier temps, nous avons construit un méta-modèle du cas simplifié (2.1). Cela nous permet de bien vérifier la qualité du méta-modèle par le frigeage avant d'appliquer sur un grand code comme TELEMAC.

3.6.1 Construction du méta-modèle

D'abord, nous avons créé un plan d'expériences, par *"Échantillonnage par hypercube latin"*, dont le résultat est donné par la figure (4). Ensuite, à l'aide du package SCIKIT LEARN sous PYTHON, nous avons construit un méta-modèle par krigeage avec les paramètres suivants :

- Fonction de régression linéaire : $\beta F(x) = \beta_0 + \beta_1 x_1 + \beta_2 x_2$,
- Fonction de corrélation : $r(\theta, x - u) = exp\left(-\theta_1 (x_1 - u_1)^2. - \theta_2 (x_2 - u_2)^2.\right)$,
- $\theta_0 = 0.4$,
- $nugget = 0$, avec $nugget_i = \left(\frac{\sigma_i}{y_i}\right)^2$

Le résultat obtenu est présenté par la figure (23). Ici, nous avons appliqué exactement la même méthode au vrai code pour calculer l'intervalle de confiance. En effet, nous avons approché la dérivée du méta-modèle par les différences finies avec un pas de 0.2.

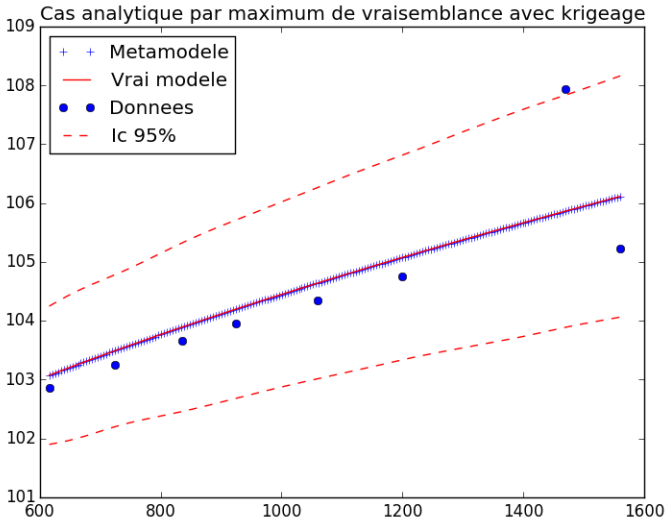


FIGURE 23: Approximation des données du méta-modèle

Remarquons que l'estimation obtenue par le méta-modèle est proche de celle du vrai code. Pour contrôler la qualité de ce méta-modèle, nous avons calculé un intervalle de confiance de ce méta-modèle au niveau 95%, voir la figure (**??**), et le coefficient de prédictivité Q_2.

$$Q_2 = 0.999718 \text{ pour } 100 \text{ points de DOE}$$
$$Q_2 = 0.999950 \text{ pour } 200 \text{ points de DOE}$$

Plus la valeur de Q_2 est proche de 1, plus l'estimation de méta-modèle est bonne. Ici, nous avons obtenu une valeur de coefficient de prédictivité très proche de 1, cela signifie que le méta-modèle représente bien le vrai modèle. Une autre méthode, pour vérifier la qualité du méta-modèle, qui est beaucoup plus simple est de tracer les hauteurs d'eau estimées par le méta-modèle en fonction des valeurs calculées par le vrai code. Si cela donne une diagonale du repère, le méta-modèle représente bien son code associé (voir la figure.

3.6.2 Résultat du calage statistique par krigeage dans le cas analytique

Après avoir validé le méta-modèle, nous avons réalisé les mêmes calculs au vrai modèle pour pouvoir comparer les deux résultats. La table (1) nous montre que les paramètres obtenus, par les moindres carrés, le maximum de vraisemblance et par l'approximation de Laplace, avec le méta-modèle sont assez proches de ceux calculés avec le vrai modèle. D'autre part, plus on augmente le nombre de point du plan d'expériences, plus le résultat obtenu sera exact.

	Vrai modèle		Méta-modèle (100)		Méta-modèle (200)	
	$\hat{Ks}$	$\hat{\tau}$	$\hat{Ks}$	$\hat{\tau}$	$\hat{Ks}$	$\hat{\tau}$
Moindres Carrés	59.3334		59.39233398		59.3364	
Maximum de vraisemblance	59.3334	1.4528	59.3922214945	0.372927134644	59.3353	1.4527
Approximation de Laplace	59.3334	1.3322	59.3922744821	0.286377226975	59.3361	1.3321

TABLE 1: Comparaison des résultats entre le vrai modèle et le méta-modèle

Ensuite, de la même façon, avec les figures (24), (25), (26) et (27), on peut voir clairement la coïncidence entre les résultats du vrai modèle et les résultats du méta - modèle dans la procédure bayésienne.

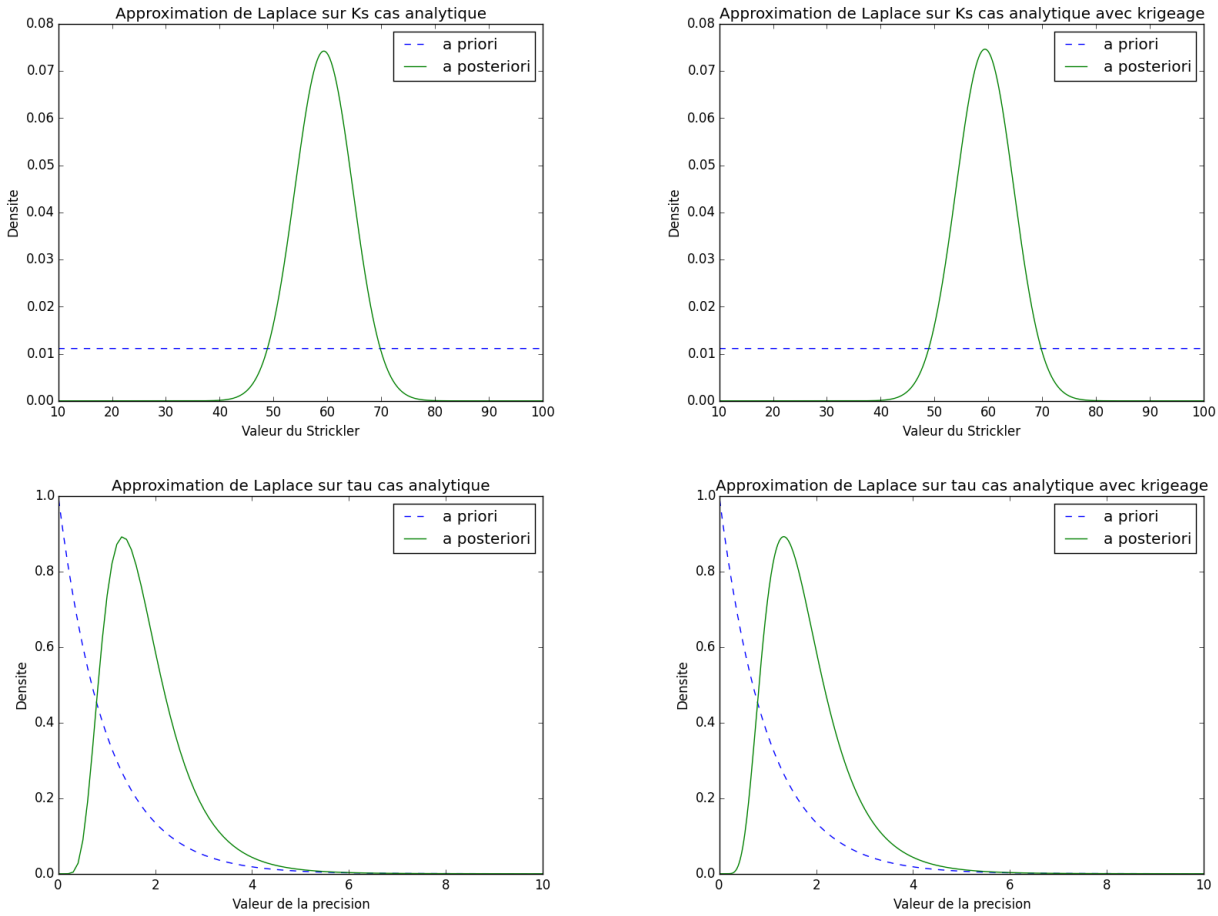


FIGURE 24: Approximation de Laplace par le vrai modèle et le méta - modèle

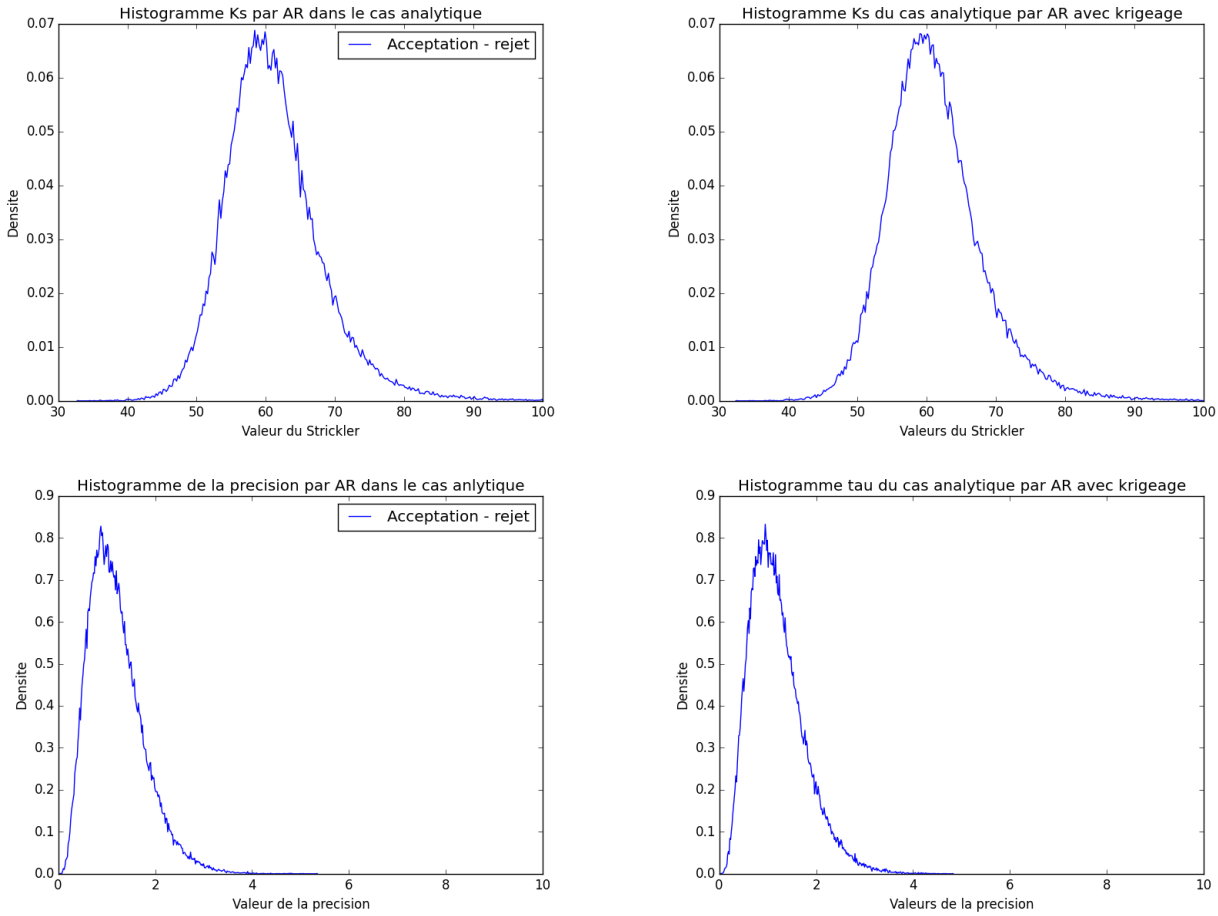
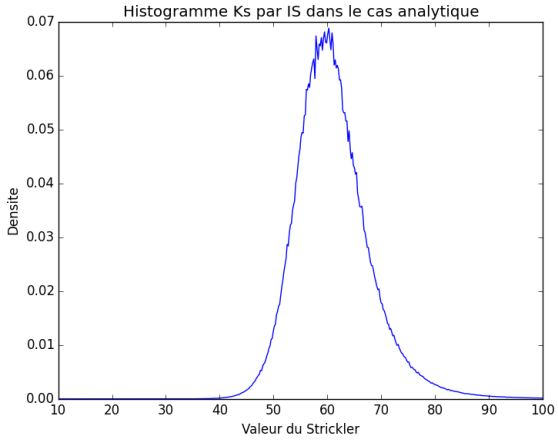
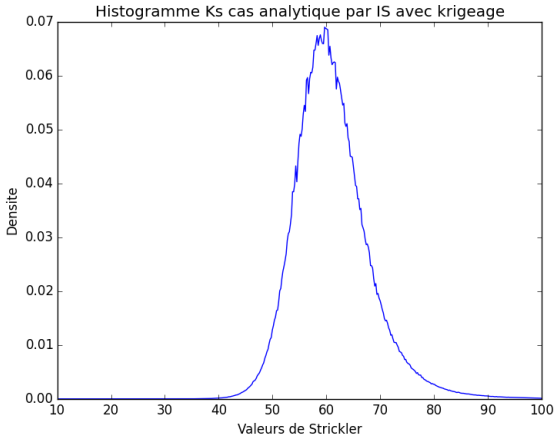
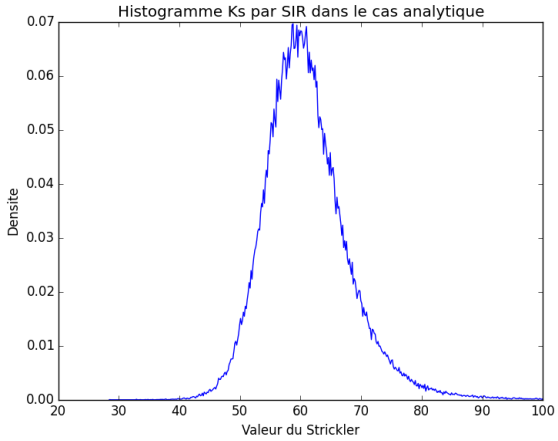
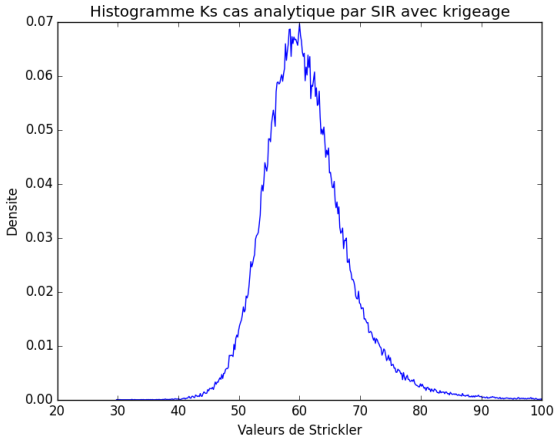


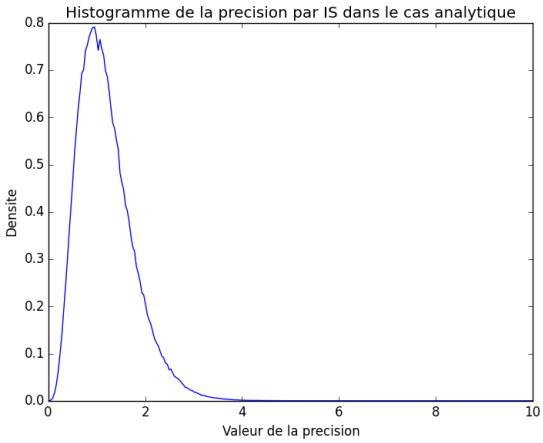
FIGURE 25: Acceptation - rejet par le vrai code et par le méta-modèle

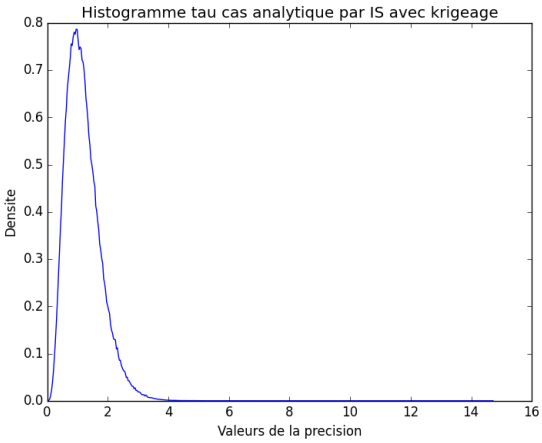

Histogramme Ks par IS dans le cas analytique
Densite
Valeur du Strickler

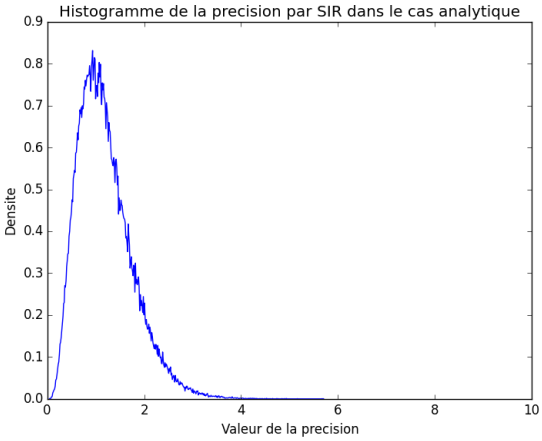

Histogramme Ks cas analytique par IS avec krigeage
Densite
Valeurs de Strickler


Histogramme Ks par SIR dans le cas analytique
Densite
Valeur du Strickler


Histogramme Ks cas analytique par SIR avec krigeage
Densite
Valeurs de Strickler


Histogramme de la precision par IS dans le cas analytique
Densite
Valeur de la precision


Histogramme tau cas analytique par IS avec krigeage
Densite
Valeurs de la precision



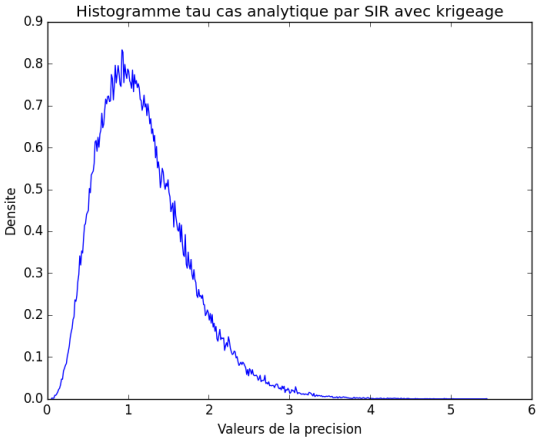
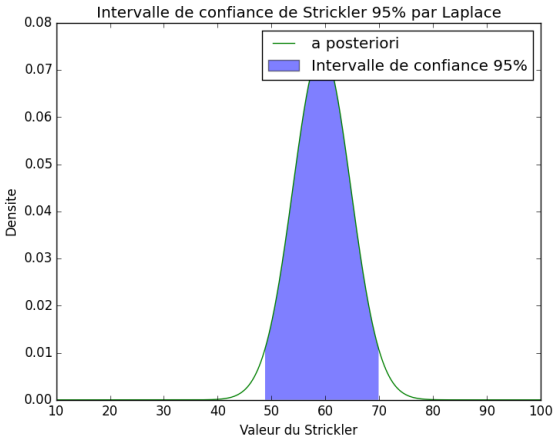
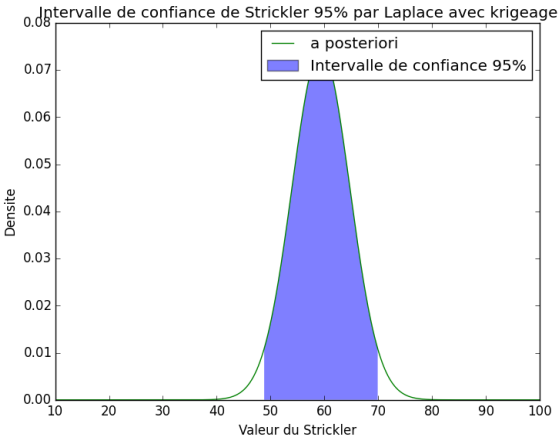
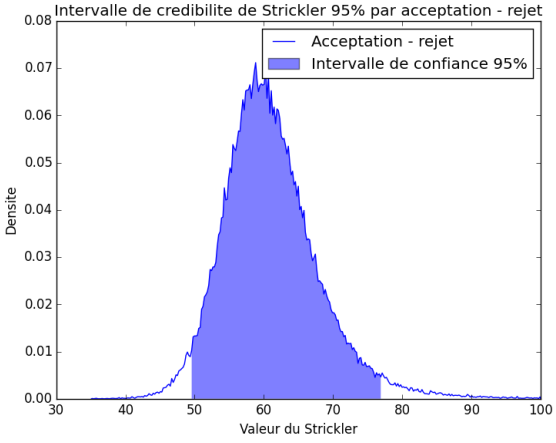
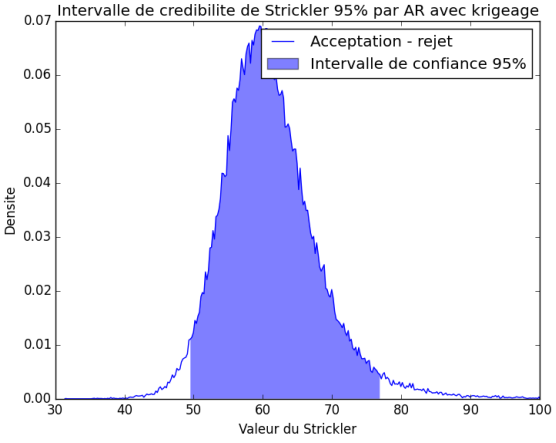


FIGURE 26: Importance Sampling et Sampling Importance Resampling par le vrai code et par le méta - modèle









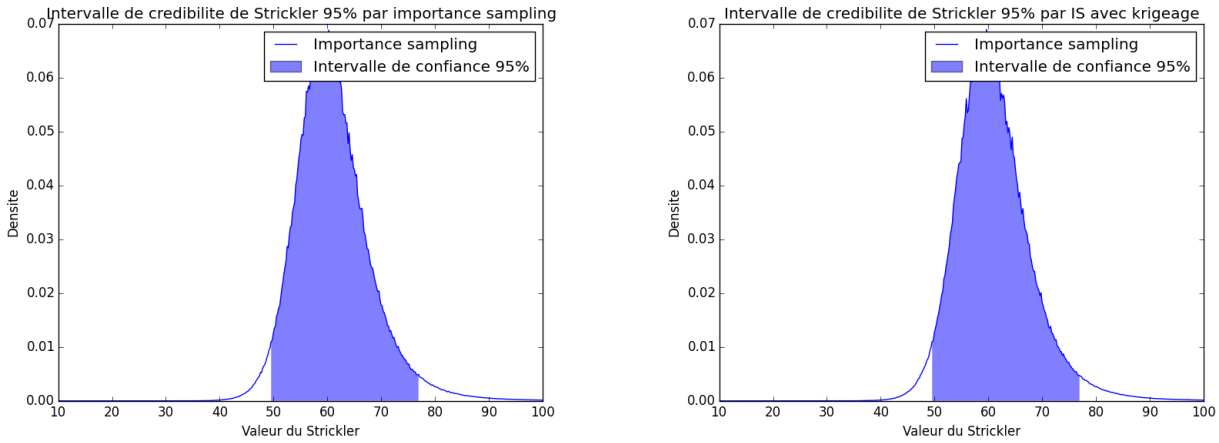


FIGURE 27: Intervalle de confiance et de crédibilité par le vrai code et par le méta - modèle

La construction du méta-modèle peut nous permettre à réduire le coût de calcul quand on applique les méthodes acceptation - rejet, importance sampling (IS) ou sampling importance resampling (SIR) sur TELEMAC. Une fois que le méta-modèle est fait, l'appel du code à chaque itération de la simulation sera beaucoup plus rapide. Les tableaux (3) et (4) représentent les résultats obtenus en utilisant le vrai code et le méta - modèle. Cela nous permet d'avoir une idée, pas seulement sur l'exactitude du méta - modèle, mais aussi sur son efficacité. Pour comparer les temps de calculs des deux modèles, nous avons utilisé le package **time** du PYTHON. Dans le cas analytique, comme le modèle déterministe est linéaire, le temps d'effectuer les méthodes de calage ne prend pas beaucoup de temps. En plus, pour créer un méta - modèle, nous avons besoin d'un plan d'expérience. Si le nombre de point sur le plan d'expérience est grand, les ordinateurs vont prendre beaucoup de temps pour créer le plan d'expérience et après calculer le méta - modèle. C'est pour cette raison, dans le cas analytique, l'utilisation du méta - modèle n'est pas très nécessaire.
Le tableau (2) suivant représente le temps nécessaire pour créer des plans d'expérience de taille 100, 500, 1000 et 5000 et les méta - modèles correspondants :

	Temps de création de DOE	Temps de création du méta - modèle
100	8.2221	0.1041
500	35.0025	0.4924
1000	94.6237	1.1894
5000	1514.5726	29.033

TABLE 2: Temps de création du plan d'expérience et du méta - modèle

Méthode			Temps de calcul	Nombre d'appel code	Valeur optimal de Ks	Différence avec valeur référence	IC 95%	
							Inf	Sup
MC	Vrai modèle		0.001909	66	59.333445	0.003445	0	0
	Méta - modèle	100	0.03739	83	59.295654	0.034346	0	0
		500	0.049732	83	59.3331238	0.0031238	0	0
		1000	0.076967	88	59.335091	0.005091	0	0
		5000	0.176984	92	59.345947	0.015947	0	0
MLE	Vrai modèle		0.004735	149	59.333445	0.003445	30.791032	87.875859
	Méta - modèle	100	0.071731	177	59.295643	0.034357	30.735414	87.855871
		500	0.091611	177	59.329477	0.000523	30.806400	87.852553
		1000	0.105055	165	0.002564	0.017436	30.778895	87.875978
		5000	0.359282	186	59.334119	0.004119	30.788101	87.880138
MAP	Vrai modèle		0.00585	151	59.333446	0.003446	48.795128	69.871763
	Méta - modèle	100	0.0052701	175	59.295576	0.034424	48.794031	69.797120
		500	0.089229	176	59.3239	0.0061	48.794348	69.876209
		1000	0.120511	184	59.333079	0.003079	48.789948	69.876209
		5000	0.336501	167	59.348755	0.018755	48.809218	69.888292

TABLE 3: Comparaison sur MC, MLE et MAP dans le cas analytique

Méthodes			Temps de calculs	Nb simulations nécessaire	Nb simulations total	Taux d'acceptation	ESS	IC inf 95%	IC sup 95%
AR	Vrai modèle		1470731742.61	100 000	1536006	0.065104		49.3427	77.1906
	Méta - modèle	100	1470742229.99	100 000	1530061	0.065357		49.3389	77.2201
		500	1470752425.92	100 000	1539429	0.064959		49.4508	77.1879
		1000	1470811964.66	100 000	1529034	0.065401		49.3757	77.4198
		5000	1470826547.61	100 000	1532397	0.065257		49.4496	77.1789
IS	Vrai modèle		27.793854		1536006		206764.7957	49.4147	77.22128
	Méta - modèle	100	1470742586.62		1530061		205775.7004	49.404348	77.202723
		500	1470753022.05		1539429		207429.3035	49.43.7558	77.153408
		1000	1470812830.66		1529034		2066.963838	49.432458	77.174009
		5000	1470829506.08		1532397		206697.8746	49.435676	77.200961

TABLE 4: Comparaison sur acceptation - rejet et importance sampling dans le cas analytique

4 TELEMAC2D et résultat du cas "Estimation"

4.1 Présentation globale de TELEMAC2D

Le système de modélisation **TELEMAC** est un logiciel développé par le département LNHE (Laboratoire national d'Hydraulique et Environnement) de EDF R&D depuis 1987 (figure (28)). C'est un système open - source qui permet de modéliser les problèmes de dimensions deux et trois.

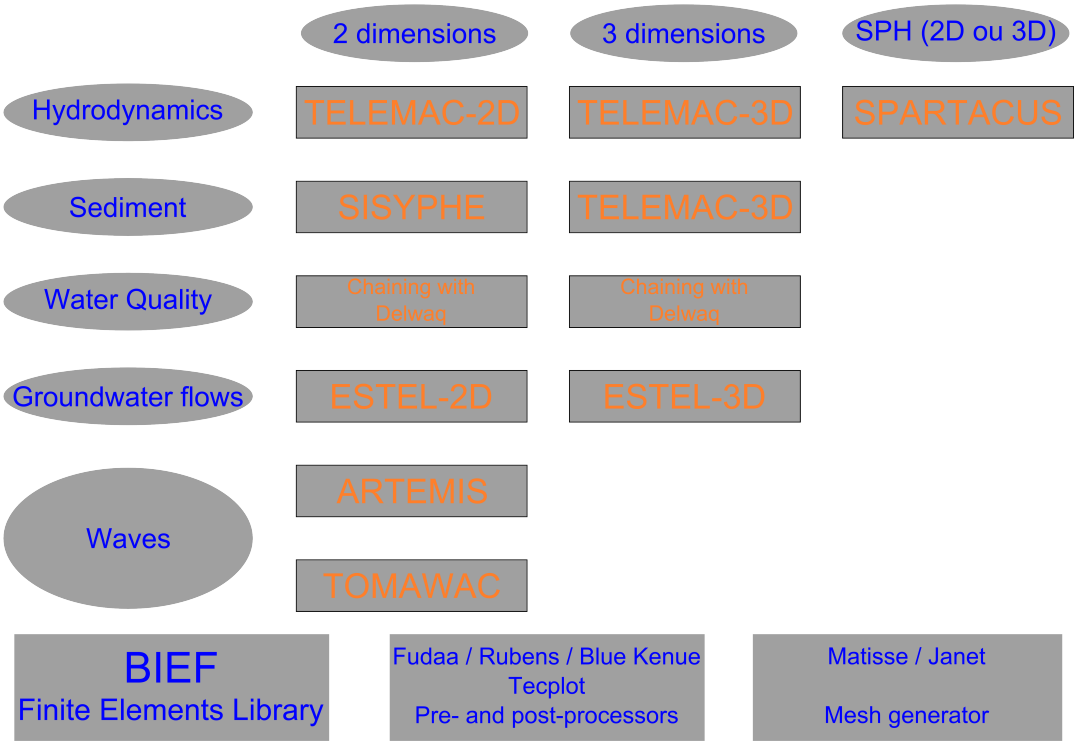


FIGURE 28: Composition du système **TELEMAC**, source [20]

D'après [21], **TELEMAC2D** est un module du système de modélisation **TELEMAC - MASCARET**, qui permet de traiter les écoulements non permanents à surface libre. Il permet d'étudier aussi bien des domaines côtiers que fluviaux, estuariens ou lacustres. Il est adapté en particulier à la simulation des courants de marée. **TELEMAC2D** résout, sur des maillages non structurés constitués d'éléments triangulaires, les équations de **Barré de Saint - Venant** à deux dimensions horizontales d'espace. Dans la suite de cette section, nous allons étudier l'établissement de ces équations à partir du système des équations de Navier - Strokes dans la mécanique des fluides.

4.1.1 Définition du système de Saint - Venant

Il existe plusieurs forme du système de Saint - Venant, dans le cadre de ce travail, nous l'appelons le système suivant :

$$\frac{\partial h}{\partial t} + \frac{\partial (hu_1)}{\partial x_1} + \frac{\partial (hu_2)}{\partial x_2} = 0 \tag{4.1}$$

$$\frac{\partial (hu_1)}{\partial t} + \frac{\partial \left(hu_1^2 + gh^2/2\right)}{\partial x_1} + \frac{\partial (hu_1u_2)}{\partial x_2} = -gh\frac{\partial z}{\partial x_1} + \nu_t\left[\frac{\partial}{\partial x_1}\left(h\frac{\partial u_1}{\partial x_1}\right) + \frac{\partial}{\partial x_2}\left(h\frac{\partial u_1}{\partial x_2}\right)\right] - gh\frac{n_M^2 u_1\sqrt{u_1^2+u_2^2}}{h^{4/3}}$$

$$\frac{\partial (hu_2)}{\partial t} + \frac{\partial (hu_1u_2)}{\partial x_1} + \frac{\partial (hu_2^2 + gh^2/2)}{\partial x_2} = -gh\frac{\partial z}{\partial x_2} + \nu_t\left[\frac{\partial}{\partial x_1}\left(h\frac{\partial u_2}{\partial x_1}\right) + \frac{\partial}{\partial x_2}\left(h\frac{\partial u_2}{\partial x_2}\right)\right] - gh\frac{n_M^2 u_2\sqrt{u_1^2+u_2^2}}{h^{4/3}}$$

où $u = (u_1, u_2)^T$ est le vecteur vitesse, $h = z(x_1, x_2, t) - Z(x_1, x_2)$ est la hauteur d'eau, $z(x_1, x_2, t)$ est la côte de la surface libre, $Z(x_1, x_2)$ est le fond de la rivière (ici, on suppose que le fond ne change pas au cours du

temps), $n_M = \dfrac{1}{Ks}$ est le coefficient de Manning et ν_t est la viscosité turbulente.

4.1.2 Obtention du système de Saint - Venant à partir de Navier - Stokes incompressible

Soit le système de Navier - Stokes dans la dimension trois suivant :

$$\begin{cases} div(\boldsymbol{u}) = 0 \\ \dfrac{\partial \boldsymbol{u}}{\partial t} + div(\boldsymbol{u}) \otimes \boldsymbol{u} + \dfrac{1}{\rho}\nabla P = \nu \Delta \boldsymbol{u} + \boldsymbol{f} \end{cases} \tag{4.2}$$

où $\boldsymbol{u} = (u_1, u_2, u_3)^T$, ρ est la masse volumique de l'eau, $\boldsymbol{P}$ est la pression, ν est la viscosité et $\boldsymbol{f} = (f_1, f_2, f_3)^T$ représente les forces volumiques extérieures qui peuvent être réduites à la force de pesanteur et la force de Coriolis.

En coordonnées cartésiennes, à partir du système (4.2), nous obtenons :

$$\begin{cases} \dfrac{\partial u_1}{\partial x_1} + \dfrac{\partial u_2}{\partial x_2} + \dfrac{\partial u_3}{\partial x_3} = 0 \\ \dfrac{\partial u_1}{\partial t} + u_1\dfrac{\partial u_1}{\partial x_1} + u_1\dfrac{\partial u_2}{\partial x_2} + u_1\dfrac{\partial u_3}{\partial x_3} + \dfrac{1}{\rho}\dfrac{\partial P}{\partial x_1} = \nu\left(\dfrac{\partial^2 u_1}{\partial x_1^2} + \dfrac{\partial^2 u_1}{\partial x_2^2} + \dfrac{\partial^2 u_1}{\partial x_3^2}\right) + f_1 \\ \dfrac{\partial u_2}{\partial t} + u_2\dfrac{\partial u_1}{\partial x_1} + u_2\dfrac{\partial u_2}{\partial x_2} + u_2\dfrac{\partial u_3}{\partial x_3} + \dfrac{1}{\rho}\dfrac{\partial P}{\partial x_2} = \nu\left(\dfrac{\partial^2 u_2}{\partial x_1^2} + \dfrac{\partial^2 u_2}{\partial x_2^2} + \dfrac{\partial^2 u_2}{\partial x_3^2}\right) + f_2 \\ \dfrac{\partial u_3}{\partial t} + u_3\dfrac{\partial u_1}{\partial x_1} + u_3\dfrac{\partial u_2}{\partial x_2} + u_3\dfrac{\partial u_3}{\partial x_3} + \dfrac{1}{\rho}\dfrac{\partial P}{\partial x_3} = \nu\left(\dfrac{\partial^2 u_3}{\partial x_1^2} + \dfrac{\partial^2 u_3}{\partial x_2^2} + \dfrac{\partial^2 u_3}{\partial x_3^2}\right) + f_3 \end{cases}$$

Dans la première étape, nous intégrons l'équation de conservation de la masse sur la verticale de la rivière. Cela nous permet d'éliminer la vitesse verticale qui ne nous intéresse pas. Ensuite, en appliquant la formule d'intégration de Leibnitz (annexe (6.14)), nous obtenons :

$$\begin{aligned} & \int_{Z(x_1,x_2)}^{z(x_1,x_2,t)} \left(\frac{\partial u_1}{\partial x_1}(x_1,x_2,x_3) + \frac{\partial u_2}{\partial x_2}(x_1,x_2,x_3) + \frac{\partial u_3}{\partial x_3}(x_1,x_2,x_3)\right) dx_3 = 0 \\ \Leftrightarrow \quad & \int_{Z(x_1,x_2)}^{z(x_1,x_2,t)} \frac{\partial u_1}{\partial x_1}(x_1,x_2,x_3)dx_3 + \int_{Z(x_1,x_2)}^{z(x_1,x_2,t)} \frac{\partial u_2}{\partial x_2}(x_1,x_2,x_3)dx_3 + u_3\left(x_1,x_2,z(x_1,x_2,t)\right) - u_3\left(x_1,x_2,Z(x_1,x_2)\right) = 0 \\ \Leftrightarrow \quad & \frac{\partial}{\partial x_1}\int_{Z(x_1,x_2)}^{z(x_1,x_2,t)} u_1(x_1,x_2,x_3)dx_3 + u_1\left(x_1,x_2,Z(x_1,x_2)\right)\frac{\partial Z}{\partial x_1}(x_1,x_2) - u_1\left(x_1,x_2,z(x_1,x_2,t)\right)\frac{\partial z}{\partial x_1}(x_1,x_2,t) \\ & + \frac{\partial}{\partial x_2}\int_{Z(x_1,x_2)}^{z(x_1,x_2,t)} u_2(x_1,x_2,x_3)dx_3 + u_2\left(x_1,x_2,Z(x_1,x_2)\right)\frac{\partial Z}{\partial x_2}(x_1,x_2) - u_2\left(x_1,x_2,z(x_1,x_2,t)\right)\frac{\partial z}{\partial x_2}(x_1,x_2,t) \\ & + u_3\left(x_1,x_2,z(x_1,x_2,t)\right) - u_3\left(x_1,x_2,Z(x_1,x_2)\right) = 0 \quad \text{(en utilisant l'intégration de Leibnitz)} \end{aligned} \tag{4.3}$$

avec $z(x_1, x_2, t)$ étant la côte de la surface libre et $Z(x_1, x_2)$ étant le fond de la rivière.

Ici, nous définissons une fonction F qui représente l'interface avec :

$$F(x_1, x_2, x_3) = z(x_1, x_2, t) - x_3$$

Au fond de la rivière, les vitesses et l'interface sont nulles $u_1 = u_2 = u_3 = 0$ et $F(x_1, x_2, 0) = 0$. Cette interface est conservée pendant le mouvement du fluide, donc :

$$\begin{aligned} & \frac{dF}{dt} = 0 \\ \Leftrightarrow \quad & \frac{\partial F}{\partial t} + u_1\frac{\partial F}{\partial x_1} + u_2\frac{\partial F}{\partial x_2} + u_3\frac{\partial F}{\partial x_3} = 0 \\ \Leftrightarrow \quad & \frac{\partial z}{\partial t} + u_1(x_1,x_2,z)\frac{\partial z}{\partial x_1}(x_1,x_2,t) + u_2(x_1,x_2,z)\frac{\partial z}{\partial x_2}(x_1,x_2,t) - u_3(x_1,x_2,z) = 0 \\ \Leftrightarrow \quad & u_3(x_1,x_2,z) = \frac{\partial z}{\partial t} + u_1(x_1,x_2,z)\frac{\partial z}{\partial x_1}(x_1,x_2,t) + u_2(x_1,x_2,z)\frac{\partial z}{\partial x_2}(x_1,x_2,t) \end{aligned} \tag{4.4}$$

En appliquant l'expression (4.4) dans l'équation (4.3), nous obtenons l'équation suivante :

$$\frac{\partial}{\partial x_1}\int_{Z(x_1,x_2)}^{z(x_1,x_2,t)} u_1(x_1,x_2,x_3)dx_3 + \frac{\partial}{\partial x_2}\int_{Z(x_1,x_2)}^{z(x_1,x_2,t)} u_2(x_1,x_2,x_3)dx_3 + \frac{\partial z}{\partial t}(x_1,x_2,t) = 0$$

On pose $h = z(x_1,x_2,t) - Z(x_1,x_2)$, la hauteur d'eau du point (x_1,x_2) au temps t, et $\bar{u} = \frac{1}{h}\int_Z^z u\, dx_3$. Comme Z ne dépend pas du temps t, on peut calculer la dérivée $\frac{\partial(z-Z)}{\partial t} = \frac{\partial h}{\partial t}$ au lieu de $\frac{\partial z}{\partial t}$. Alors l'équation précédente devient :

$$\frac{\partial h}{\partial t} + \frac{\partial(h\bar{u}_1)}{\partial x_1} + \frac{\partial(h\bar{u}_2)}{\partial x_2} = 0$$

Cela nous donne la première équation du système de Saint - Venant.

Pour obtenir les trois dernières équations de ce système, nous devons d'abord réécrire les trois équations de conservation de la quantité du mouvement du système de Navier - Stokes sous la forme d'un tenseur de contraintes de termes T_{ij}.

$$\frac{\partial u_i}{\partial t} + \sum_{i=1}^{3}\frac{\partial(u_i u_j)}{\partial x_j} + \frac{1}{\rho}\frac{\partial P}{\partial x_i} = \frac{1}{\rho}\sum_{i=1}^{3}\frac{\partial T_{ij}}{\partial x_j} + f_i \quad \text{pour } i = 1,2,3$$

Ensuite, de la même façon, nous avons intégré ces trois équations sur la verticale et nous obtenons :

$$h\frac{\partial \bar{u}_i}{\partial t} + \sum_{j=1}^{2}\bar{u}_j\frac{\bar{u}_i}{\partial x_j} + \sum_{j=1}^{2}\frac{\partial\left(\int_Z^z (u_i-\bar{u}_i)(u_j-\bar{u}_j)\,dx_3\right)}{\partial x_j} + gh\frac{\partial z}{\partial x_i} = \frac{T_i}{\rho} \quad (\text{pour } i = 1,2)$$

avec $T_i = \sum_{j=1}^{2}\frac{\partial(h\bar{T}_{ij})}{\partial x_j} + \left(T_{i3} - \sum_{j=1}^{2}T_{ij}\frac{\partial z}{\partial x_j}\right)_{surface} - \left(T_{i3} - \sum_{j=1}^{2}T_{ij}\frac{\partial Z}{\partial x_j}\right)_{fond}$. En s'inspirant du document [22], nous pouvons expliciter les quatres termes suivant :

- Le terme lié à la dispersion des vitesses sur la verticale $\sum_{j=1}^{2}\frac{\partial\left(\int_Z^z (u_i-\bar{u}_i)(u_j-\bar{u}_j)\,dx_3\right)}{\partial x_j}$ est souvent réécrit sous la forme $\frac{\partial\left(t_i h\bar{u}_i^2\right)}{\partial x_i} + D_i\sum_{j=1}^{2}\frac{\partial\left(h\frac{\partial\bar{u}_i}{\partial x_j}\right)}{\partial x_j}$ où t_i et D_i sont des coefficients constants dans le temps.
- Le terme lié aux contraintes à l'intérieur du fluide $\frac{1}{\rho}\sum_{j=1}^{2}\frac{\partial\left(h\bar{T}_{ij}\right)}{\partial x_j}$ est remplacé par un terme $D_i'\sum_{j=1}^{2}\frac{\partial\left(f\frac{\partial\bar{u}_i}{\partial x_j}\right)}{\partial x_j}$ où D_i' est aussi une constante dans le temps.
- le terme de frottement au fond $-\frac{1}{\rho}\left(T_{i3} - \sum_{j=1}^{2}T_{ij}\frac{\partial Z}{\partial x_j}\right)_{fond}$ est classiquement remplacé par la formule de Chézy : $-g\bar{u}_i\frac{\sqrt{\left(\sum_{j=1}^{2}\bar{u}_j^2\right)}}{C^2}$ où $C = Ks\, h^{1/6}$ constant dans le temps et Ks est le coefficient de Strickler.
- Le terme de frottement à la surface $\frac{1}{\rho}\left(T_{i3} - \sum_{j=1}^{2}T_{ij}\frac{\partial z}{\partial x_j}\right)_{surface}$ est souvent noté τ qui peut être estimé par différentes formules, par exemple quand il s'agit de l'action du vent.

La décomposition de ces expressions nous permet d'obtenir les deux dernières équations du système de Saint - Venant.

4.2 Présentation de Salome

La plate - forme SALOME a été co - développé par EDF R&D, le CEA et de nombreux partenaires industriels et académiques. C'est un logiciel open - source (`http://www.salome-platform.org/`) qui fournit une plate - forme de développement générique de pré/post - traitement et de couplage de codes pour la simulation numérique. Il est basé sur une architecture ouverte et flexible en composants réutilisables disponibles en tante que logiciel libre (figure (29)).

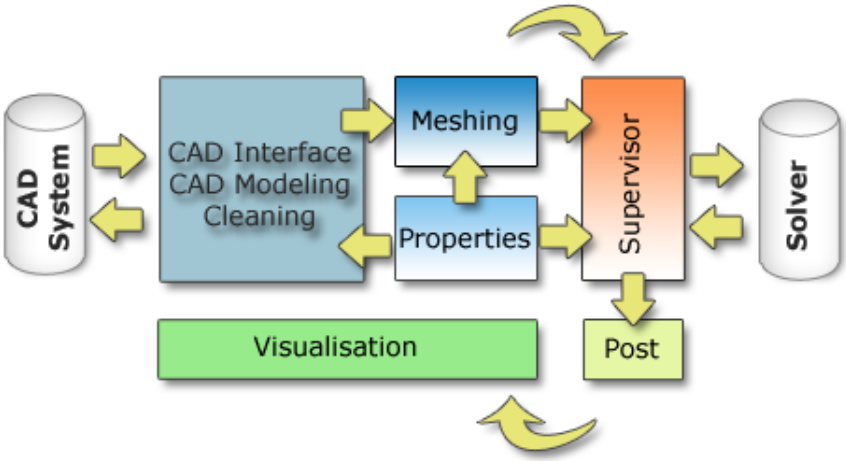


FIGURE 29: Fonctionnalité de la plate - forme SALOME, source `http://www.salome-platform.org/user-section/about/history`

SALOME a été créé en 2000 pour deux grands objectifs :

- faciliter l'inter - fonctionnement entre la modélisation CAO (Conception Assistée par Ordinateur)et les codes de calculs, et la mise en œuvre de couplage entre les codes de calculs en environnement hétérogène,
- regrouper la production des développements non - critiques (pré et post - traitement) dans une base commune de simulation numérique.

Les diverses modules de SALOME
SALOME est basé que le modèle de composants distribués construits sur CORBA (Common Object Request Broker Architecture) comme une architecture d'objets distribués. En général, nous pouvons distinguer deux niveaux principaux :

- **Lower layer** incorpore des fonctionnalités de base du noyau (communication entre les modules distribués), l'interface graphique et la gestion des études. Ces services sont réalisés par KERNEL et GUI (figure (30)).
- **Modules layer** qui fournissent des services spécialisés qui sont nécessaires pour atteindre l'objectif général. Nous citons ici les modules principaux, les services proposés par chaque module sont détaillé sur le site officiel de la plate - forme.
  - ▷ GEOM
  - ▷ MESH
  - ▷ MED
  - ▷ SUPERV
  - ▷ POST-PRO
  - ▷ YACS

Dans lesquels, YACS est un outil de gestion des simulations multidisciplinaires grâce à des programmes de calcul. Ce module peut être utilisé pour construire, modifier et exécuter des programmes de calcul. Un schéma de calcul est un ensemble plus ou moins complexe de composants de calcul (composants de SALOME, codes de calcul ou des scripts PYTHON). Par conséquent, un schéma de calcul est un moyen de définition d'une chaîne ou d'un couplage de codes de calcul. Dans ces études, nous avons utilisé le module YACS pour construire et exécuter le couplage du code TELEMAC2D et les script de calage en langage PYTHON.

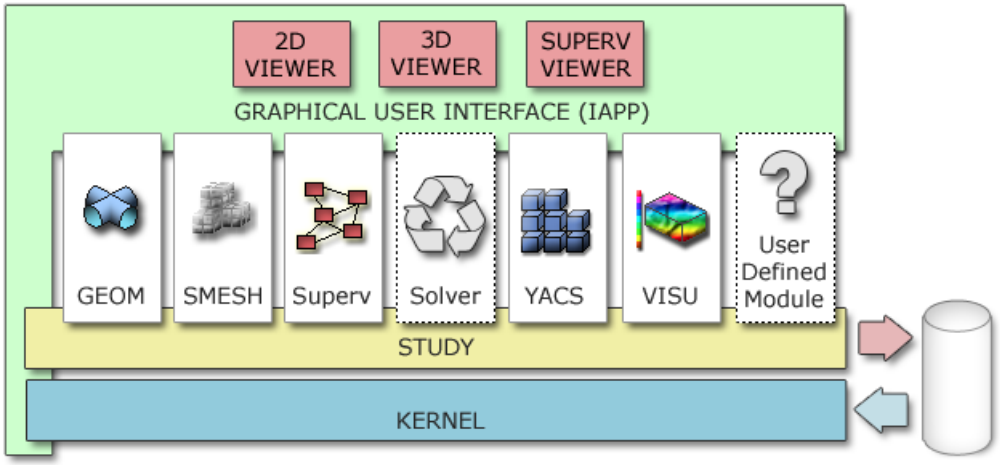


FIGURE 30: Architecture de SALOME, source `http://www.salome-platform.org/user-section/about/architecture`

4.3 Résultat du cas "Estimation"

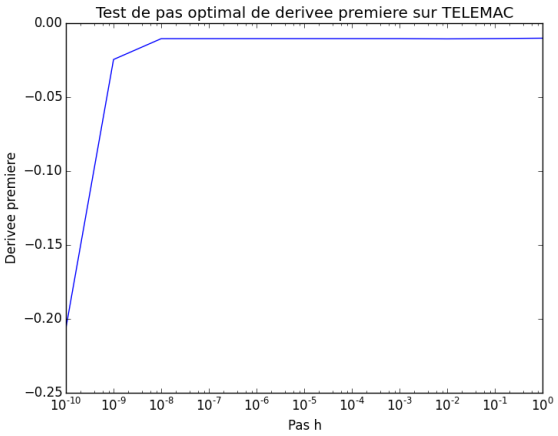
Dans le cas avec le modèle analytique, nous avons vu que les méthodes de calage précédentes fonctionnent bien. Par la suite, nous allons appliquer toutes ces méthodes dans un cas simple du TELEMAC à l'aide de la plate-forme Salome Hydro.

4.3.1 Dérivée de TELEMAC par différences finies

Nous allons commencer l'étude du calage de TELEMAC par un test sur la dérivée du modèle. Dans le cas analytique, on peut calculer facilement à la main la dérivée du modèle déterministe mais avec TELEMAC, ce n'est plus le cas. C'est pour cette raison, nous avons effectué le calcul de la dérivée du TELEMAC par différences finies.
Pour appliquer la formule de la dérivée par différences finies (6.3), nous avons besoin d'un pas h, qui peut être déterminé par les méthodes dans le cours [10]. Ici, nous avons effectué un test très simple pour déterminer le pas optimal de différences finies :

- Faire varier le pas $h \in \{10^0, \cdots, 10^{-10}\}$,
- A chaque valeur de h, calculer la quantité $q = \dfrac{f(x+h) - f(x)}{h}$,
- Tracer q en fonction de $log_{10}(h)$

En appliquant les étapes précédentes, nous avons obtenu la figure (31), respectivement correspondant aux tests de la dérivée première et de la dérivée seconde du TELEMAC.



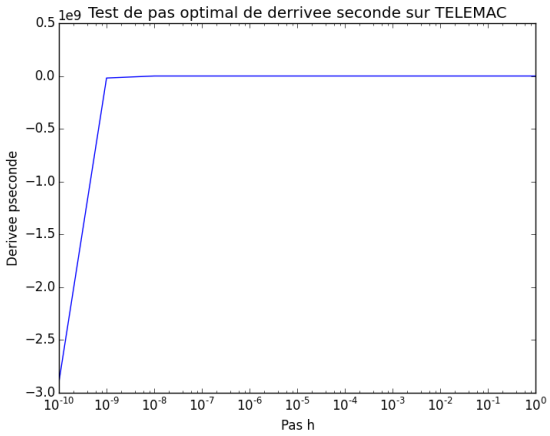


FIGURE 31: Test de pas optimal de différences finies dans TELEMAC

Ici, nous pouvons voir clairement que le pas minimal h pour la dérivée première et seconde est de l'ordre 10^{-8}. La perturbation de la dérivée quand $h < 10^{-8}$ vient de deux erreurs dans le calcul de la dérivée. Premièrement, quand on applique la formule $q = \frac{f(x+h) - f(x)}{h}$, nous avons créé une erreur de troncature d'ordre $\frac{h}{2} f''(x)$. La deuxième cause vient de l'erreur d'arrondissement des machines. Par example, la formule de différences finies $q = \frac{f(x+h) - f(x)}{h}$ est calculé par la machine comme $\tilde{q} = \frac{\tilde{f}(\tilde{x}+\tilde{h}) - \tilde{f}(\tilde{x})}{\tilde{h}}$. Si $\tilde{h}$ est trop petit, la somme $\tilde{x}+\tilde{h}$ sera égale à $\tilde{x}$. D'autre part, si $\tilde{h}$ est trop grand, $\tilde{x}+\tilde{h}$ sera égale à $\tilde{h}$. Donc l'erreur associée à la formule de différences finies ci - dessus est :

$$E(h) = \frac{r(x)\,|f(x)|}{h} + \frac{h}{2}|f''(x)|$$

où $r(x) = |\frac{\tilde{f}(x) - f(x)}{f(x)}|$. Cette erreur est équilibrée entre deux fonctions positives $\frac{r(x)\,|f(x)|}{h}$ et $\frac{h\,|f''(x)|}{2}$:

- Quand $h \longrightarrow 0$, $E(h)$ est dominée par l'erreur de troncature $\frac{h\,|f''(x)|}{2}$,
- Quand $h \longleftrightarrow 0$, $E(h)$ est dominée par l'erreur d'arrondissement $\frac{r(x)\,|f(x)|}{h}$.

Dans cette étude, nous avons choisi le pas $h = 10^-3$ pour les études de calage sur TELEMAC.

4.3.2 Création de données dans le cas test de TELEMAC2D

Pour réaliser les calculs de calage, nous avons besoin des couples de données Débit/Hauteur d'eau, mais il n'existe pas de données pour ce cas test. Alors il faut qu'on crée des données, en introduisant un certain nombre de débit et en fixant une valeur référence du Strickler. Ici, nous avons choisi 10 valeurs de débit qui varient entre 40 et $70 m^3/s$. Ensuite, pour faire un bruit sur les données, nous avons généré un vecteur d'erreur de taille 10, qui suit la loi normale centrée de variance σ^2. Le choix de σ est un peu délicat car pour retrouver le coefficient de référence, l'écart entre les valeurs calculées et les valeurs bruitées doit être assez faible. la valeur de σ représente l'erreur moyenne de l'échantillon par rapport à l'espérance, dans ce cas c'est 0. Donc si on choisit une grande valeur de σ, il risque d'avoir des valeurs de hauteur d'eau très éloignées des valeurs calculées par TELEMAC2D. Cela pourra perturber le résultat du calage. Du coup, nous avons fixé la valeur de référence du Strickler à 35 et σ à 0.01, cela nous permet d'obtenir le tableau de données (32).

Débits (m^3/s)	61.43	51.5	51.21	62.12	66.51	48.85	67.86	64.62	60.54	64.84
Hauteurs exactes (m)	0.829	0.77	0.769	0.833	0.858	0.754	0.866	0.848	0.824	0.849
Hauteurs bruitées (m)	0.833	0.748	0.764	0.819	0.841	0.75	0.857	0.848	0.836	0.837

FIGURE 32: Données Débits/Hauteurs d'eau pour le cas test de TELEMAC

Or on peut présenter les données du tableau (32) par la figure (33) suivante.

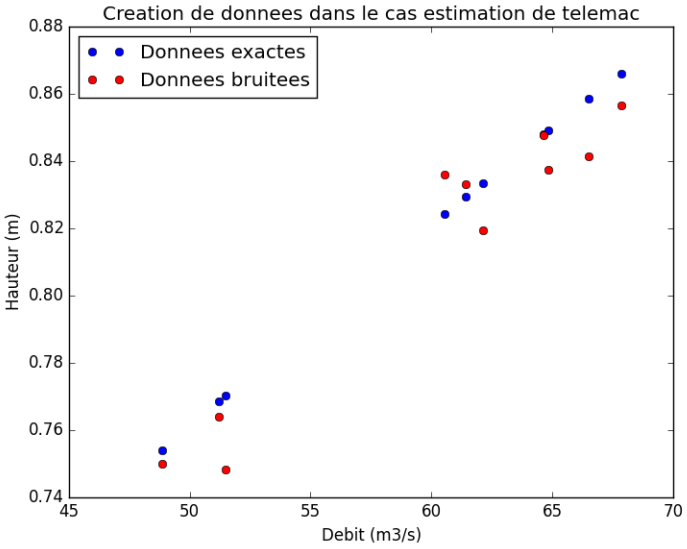


FIGURE 33: Représentation des données bruitées de TELEMAC2D

D'autre part, comme nous avons modifié les hauteurs d'eau calculées par TELEMAC2D, la valeur obtenue du Strickler peut être légèrement différente de celle de référence, $Ks = 35$. Pour la précision τ, nous utilisons la variable $\gamma = log(\tau) = -2log(\sigma)$, la valeur de référence de gamma sera $-2\,log(0.01) \approx 9.2103$.

4.3.3 Calage avec TELEMAC par moindres carrés et maximum de vraisemblance

Après avoir créé les données, nous avons mis en place le calage du coefficient de Strickler par moindres carrés et par maximum de vraisemblance. En appliquant les mêmes étapes, nous avons obtenu les valeurs suivantes :

Paramètre	Moindres Carrés		Maximum de vraisemblance	
	Valeur obtenue	Erreur	Valeur obtenue	Erreur
Strickler (θ) ($m^{1/3}/s$)	35.77428294	0.77428294	35.73277955	0.73277955
log précision $\gamma = log(tau)$			9.27162817	0.06128779

A partir de la valeur de Strickler obtenu, nous pouvons tracer la prédiction de hauteur d'eau de TELEMAC sur les débits observés et également l'intervalle de confiance des ces valeurs. Le résultat du calage par maximum de vraisemblance est présenté par la figure (34). On peut voir immédiatement que, avec la valeur estimée par cette méthodes $Ks = 35.7327$, TELEMAC2D représente plutôt bien les données. De plus, la plupart des données appartient à l'intervalle de confiance de niveau 95%, borné par les traits rouges et pointillés. Donc la valeur () est acceptable.

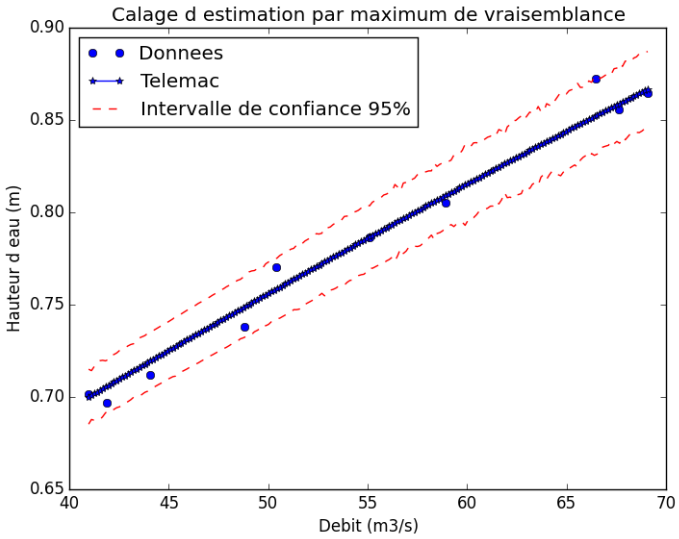


FIGURE 34: Calage de TELEMAC par moindres carrés et maximum de vraisemblance

4.3.4 Calage avec TELEMAC par approximation de Laplace

Le calage du coefficient de Strickler dans le cas de TELEMAC se déroulement exactement de la même façon que dans le cas analytique sauf le calcul de la dérivée du modèle. Dans le cas analytique, nous avons pu calculer explicitement la dérivée de la fonction G car cela ne prend pas beaucoup de temps. Par contre, nous n'avons pas de formule analytique du modèle de TELEMAC, c'est pour cette raison que nous avons calculé la dérivée de TELEMAC avec la méthode des différences finies, qui a été abordée dans le paragraphe (4.3.1). En effet, les valeurs estimées du coefficient de Strickler et de γ sont :

Paramètre	Approximation de Laplace	Erreur
Strickler (θ)	35.7580375	0.7580375
log précision $\gamma = log(tau)$	6.35098793	2.85935244

Donc, d'après la définition (1), la loi *a posteriori* du couple paramètre (θ, γ) suit une loi normale de paramètre :

$$(\theta, \gamma) \sim \mathcal{N}\left(\hat{\delta}_{MAP}, \hat{\Sigma}_{MAP}\right)$$

avec :

$$\begin{aligned} \hat{\delta}_{MAP} &= \begin{pmatrix} 35.7580375 \\ 6.35098793 \end{pmatrix} \\ \hat{\Sigma}_{MAP} &= \begin{pmatrix} 0.29923792 & -0.00075271 \\ -0.00075271 & 0.16666744 \end{pmatrix} \end{aligned}$$

La figure (35) représente le résultat de l'approximation de Laplace sur le couple $(\theta, \tau) = (\theta, e^{\gamma})$ avec TELEMAC.

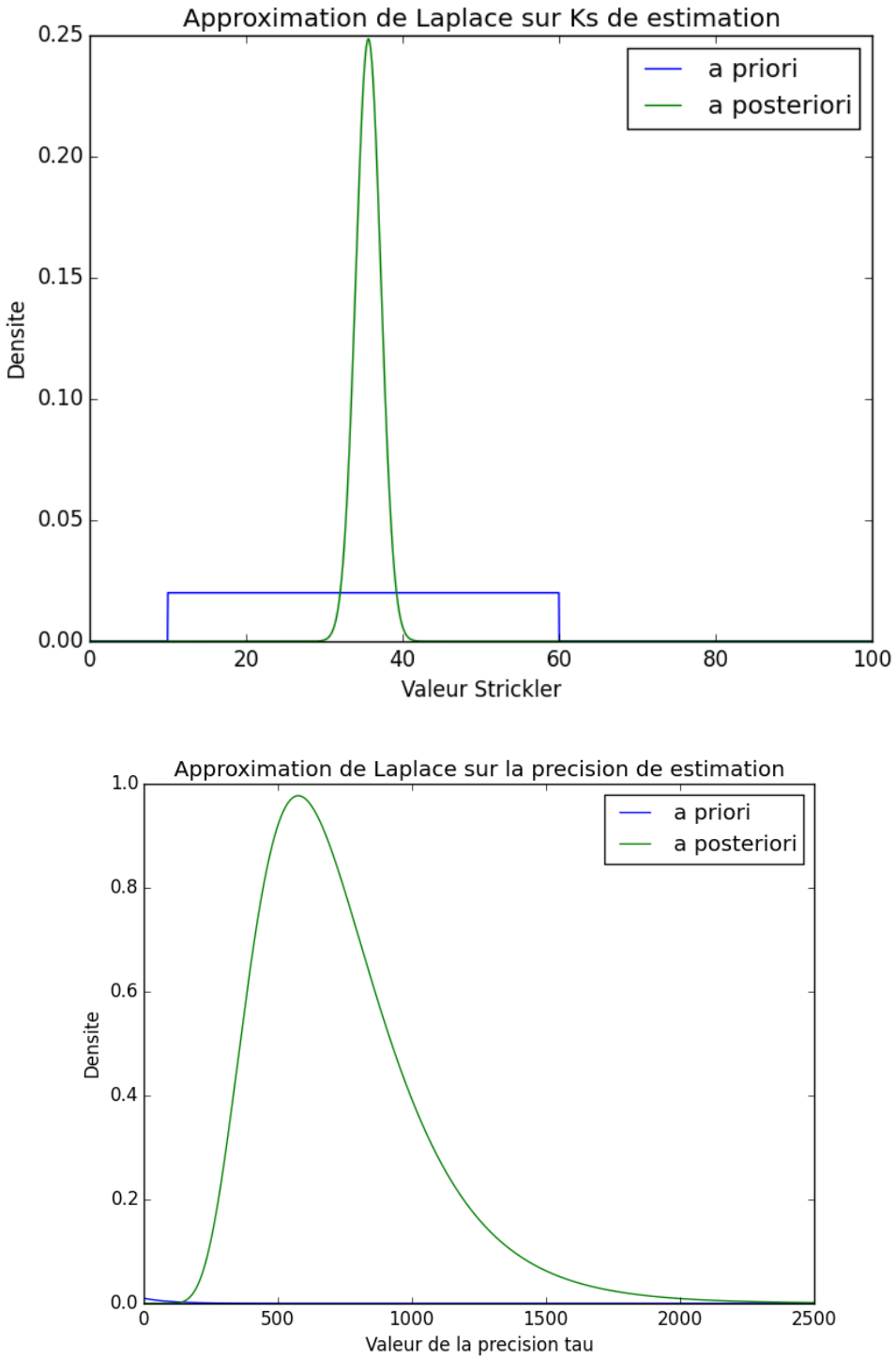


FIGURE 35: Approximation de Laplace sur le cas estimation de TELEMAC

De la même façon, nous avons aussi calculé l'intervalle de confiance du paramètre de Strickler par l'approximation de Laplace :

$$IC_{95\%}^{MAP}(Ks) = [32.7756, 38.7333]$$

A partir de ces calculs, nous pouvons représenter l'emplacement de cet intervalle de confiance sur la distribution du paramètre, voir la figure (36).

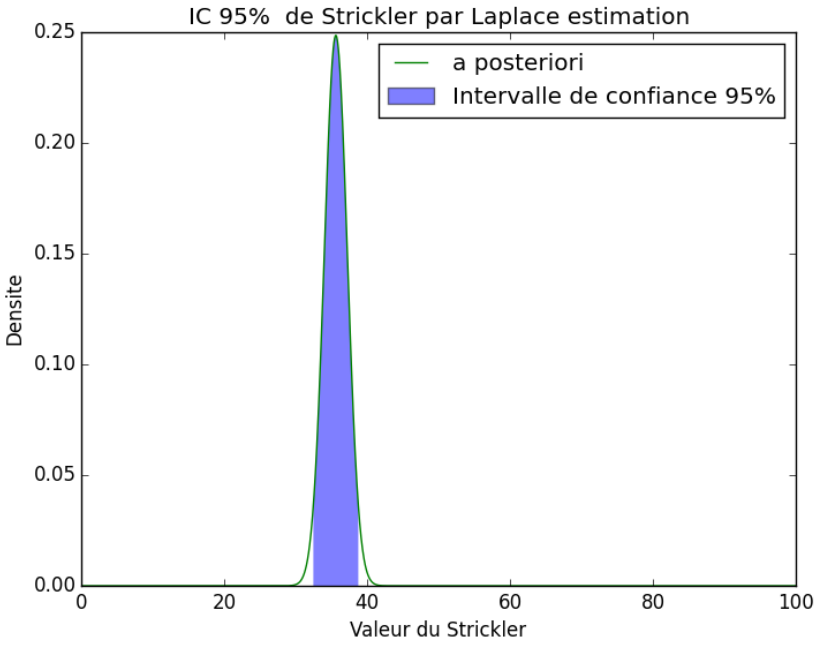
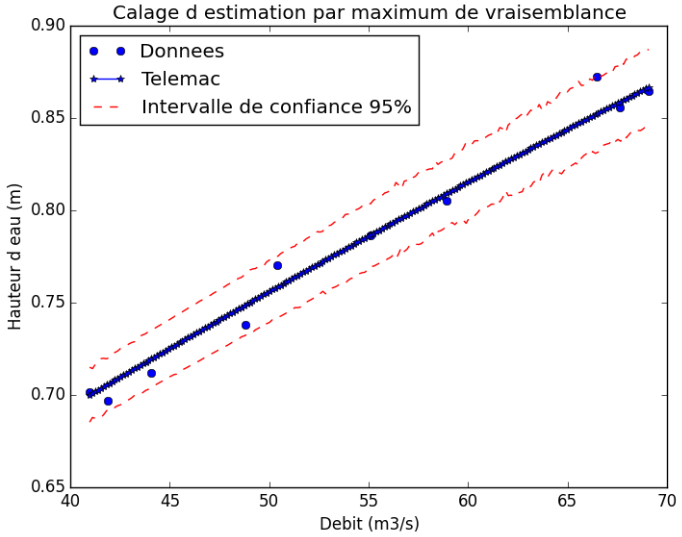


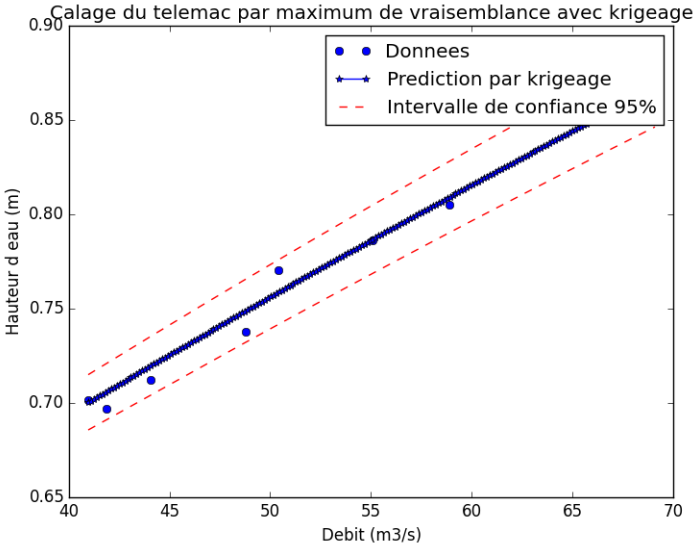
FIGURE 36: L'emplacement de l'intervalle de confiance du coefficient de Strickler sur sa distribution

Dans le cas "estimation" de TELEMAC2D avec le vrai code, nous n'avons pas appliqué les méthodes d'acceptation - rejet et impotance sampling car les calculs prennent énormément de temps. Pourtant, nous allons toujours utiliser ces méthodes pour le cas avec le méta - modèle dans la partie (4.4).

4.4 Résultat du krigeage dans la cas estimation de TELEMAC

Dans ce paragraphe, nous allons ré-appliquer le calage sur le cas estimation de TELEMAC mais avec le méta - modèle. Nous avons rencontré un problème de création du méta - modèle par krigeage dans cette partie, l'installation du package *scikit - learn* dans Salome Hydro était impossible à réaliser. De plus, le module *KrigingAlgorithm* de *Openturns 1.6.1* ne converge pas, on ne peut pas créer le méta - modèle directement sous Salome. Du coup, nous avons créé seulement le plan d'expérience sous Salome, à l'aide du module *otlhs* de *Openturns* et calculé les valeurs de hauteurs d'eau correspondant à ce plan. Puis, on les a sauvegardés dans un fichier *.csv* pour que l'on puisse créer le méta - modèle à l'extérieur de Salome Hydro. Une fois le méta - modèle est établi, nous pouvons mettre en place facilement toutes les méthodes de calage comme dans le cas analytique. Différent au cas analytique, les appels de TELEMAC sont coûteux alors on peut réduire de le temps de calcul en utilisant le méta - modèle. Mais la question est quel est le nombre idéal de point du plan d'expérience ? Comme nous avons dit dans la partie précédente, plus le nombre de point du plan d'expérience est élevé, plus le méta - modèle est proche du vrai code. Alors si on crée le méta - modèle sur un grand nombre de point, le méta - modèle obtenu représentera parfaitement TELEMAC mais le temps de création de ce méta - modèle sera aussi énorme. C'est pourquoi nous avons effectué plusieurs tests en faisant varier le nombre de point du plan d'expérience pour comparer les temps total d'exécution suivant les différentes tailles du plan.


Calage d estimation par maximum de vraisemblance
Donnees
Telemac
Intervalle de confiance 95%
Hauteur d eau (m)
Debit (m3/s)
0.90
0.85
0.80
0.75
0.70
0.65
40
45
50
55
60
65
70


Calage du telemac par maximum de vraisemblance avec krigeage
Donnees
Prediction par krigeage
Intervalle de confiance 95%
Hauteur d eau (m)
Debit (m3/s)
0.90
0.85
0.80
0.75
0.70
0.65
40
45
50
55
60
65
70

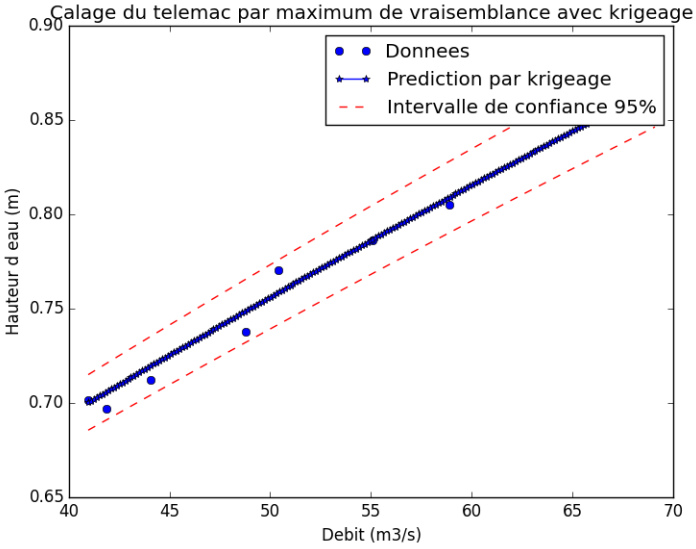

Calage du telemac par maximum de vraisemblance avec krigeage
Donnees
Prediction par krigeage
Intervalle de confiance 95%
Hauteur d eau (m)
Debit (m3/s)
0.90
0.85
0.80
0.75
0.70
0.65
40
45
50
55
60
65
70

FIGURE 37: Calage du vrai telemac, du méta - modèle de 100 points et de 500 par maximum de vraisemblance

Pour montrer la variation du résultat avec les différentes tailles du plan d'expérience, nous avons comparer les résultats obtenus du calage par le vrai code et les méta - modèle de 100 points et de 500 points (voir la figure (37)). Ici, on peut voir immédiatement que dès qu'avec la taille de 100, le résultat obtenu par le méta - modèle ressemble beaucoup à celui du vrai code. De plus, par le tableau (5), nous pouvons comparer le temps de création des plan d'expérience et du méta - modèle, le mieux est de choisir la taille du plan telle que le temps total de calcul soit inférieur à celui du vrai code mais le résultat reste acceptable.

	Temps de création de DOE	Temps de création du méta - modèle
100	60.2876	0.1436
500	265.138	0.5769
1000	587.9159	2.0971
2000	2126.6712	7.3422

TABLE 5: Temps de création du plan d'expérience et du méta - modèle pour le cas estimation de TELEMAC

Méthode			Temps de calcul	Nombre d'appel code	Valeur optimal de Ks	Différence avec valeur référence	IC 95%	
							Inf	Sup
MC	Telemac		486.337612	90	35.609724	0.609724	0	0
	Méta - modèle	100	0.075646	83	35.623047	0.623047	0	0
		500	0.056963	84	35.623058	0.623058	0	0
		1000	0.088900	85	35.625002	0.625002	0	0
		2000	0.105136	87	35.623779	0.623779	0	0
MLE	Telemac		1129.382034	223	35.628051	0.628051	33.452477	37.803624
	Méta - modèle	100	0.100133	184	35.625640	0.625640	33.447145	37.804136
		500	0.106989	195	35.624243	0.624243	33.443578	37.804908
		1000	0.184336	259	35.621343	0.621343	33.441269	37.801417
		2000	0.284988	246	35.627973	0.627973	33.452781	37.803166
MAP	Telemac		572.414194	206	35.628051	0.628051	32.482496	38.773606
	Méta - modèle	100	0.063094	168	35.624042	0.624042	32.478080	38.770004
		500	0.107639	187	35.624315	0.624315	32.477394	38.771237
		1000	0.127095	167	35.614739	0.614739	32.471650	38.757828
		2000	0.20874	170	35.613371	0.613371	32.467309	39.759432

TABLE 6: Comparaison sur MC, MLE et MAP dans le cas estimation de TELEMAC

Méthodes			Temps de calculs	Nb simulations effectuées	ESS	IC inf 95%	IC sup 95%
IC	Méta - modèle	100	882.289132	1 500 000	304 181.6974	32.381877	39.824254
		500	982.974242	1 500 000	169 799.4357	32.413380	39.778202
		1000	1316.857385	1 500 000	413 155.7734	32.397054	39.785222
		5000	1980.083733	1 500 000	129 262.6913	32.340605	39.858561

TABLE 7: Comparaison sur importance sampling dans le cas estimation sur TELEMAC

Dans les figures (38), (39) et (40) suivantes, nous pouvons comparer les résultats obtenus par le vrai modèle de TELEMAC et son méta - modèle de 1000 points. Ici, pour effectuer la méthode importance sampling, nous avons choisi le résultat de l'approximation de Laplace comme la loi instrumentale car le coefficient ESS dans ce cas est beaucoup plus élevé que le cas de la loi *a priori*. Dans le cas analytique, nous avons utilisé la loi *a priori* comme la loi instrumentale car ça simplifie les calculs de la fraction $\frac{f}{g}$ et ça donne un bon résultat.

Dans la figure (39), on voit que la simulation par importance sampling du paramètre de Strickler nous a donné une distribution très proche de celle de l'approximation de Laplace. Cela vient de la régularité du modèle, qui intervient dans le calcul de la matrice hessienne de la définition (1).

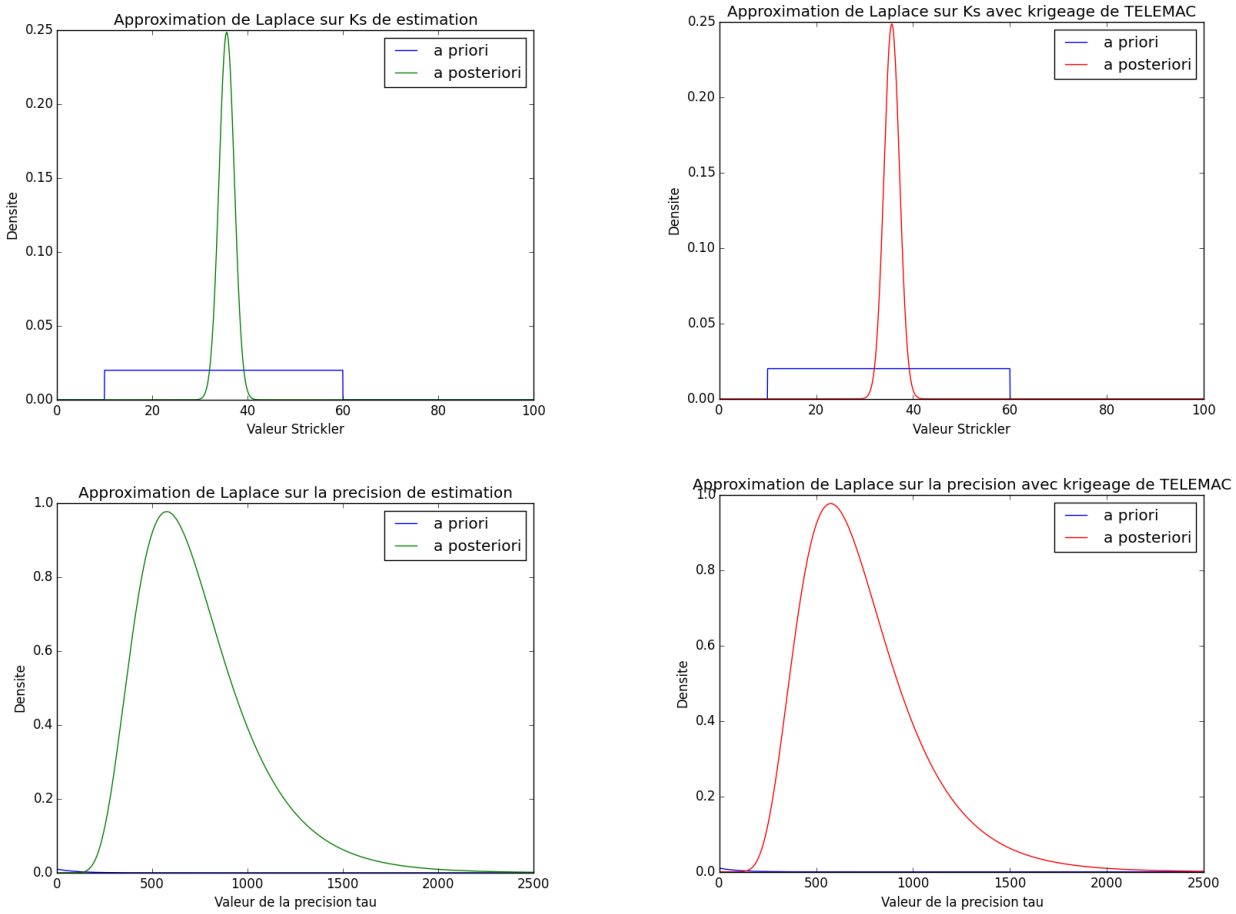


FIGURE 38: Approximation de Laplace par le vrai modèle et le méta - modèle de 1000 points sous TELEMAC

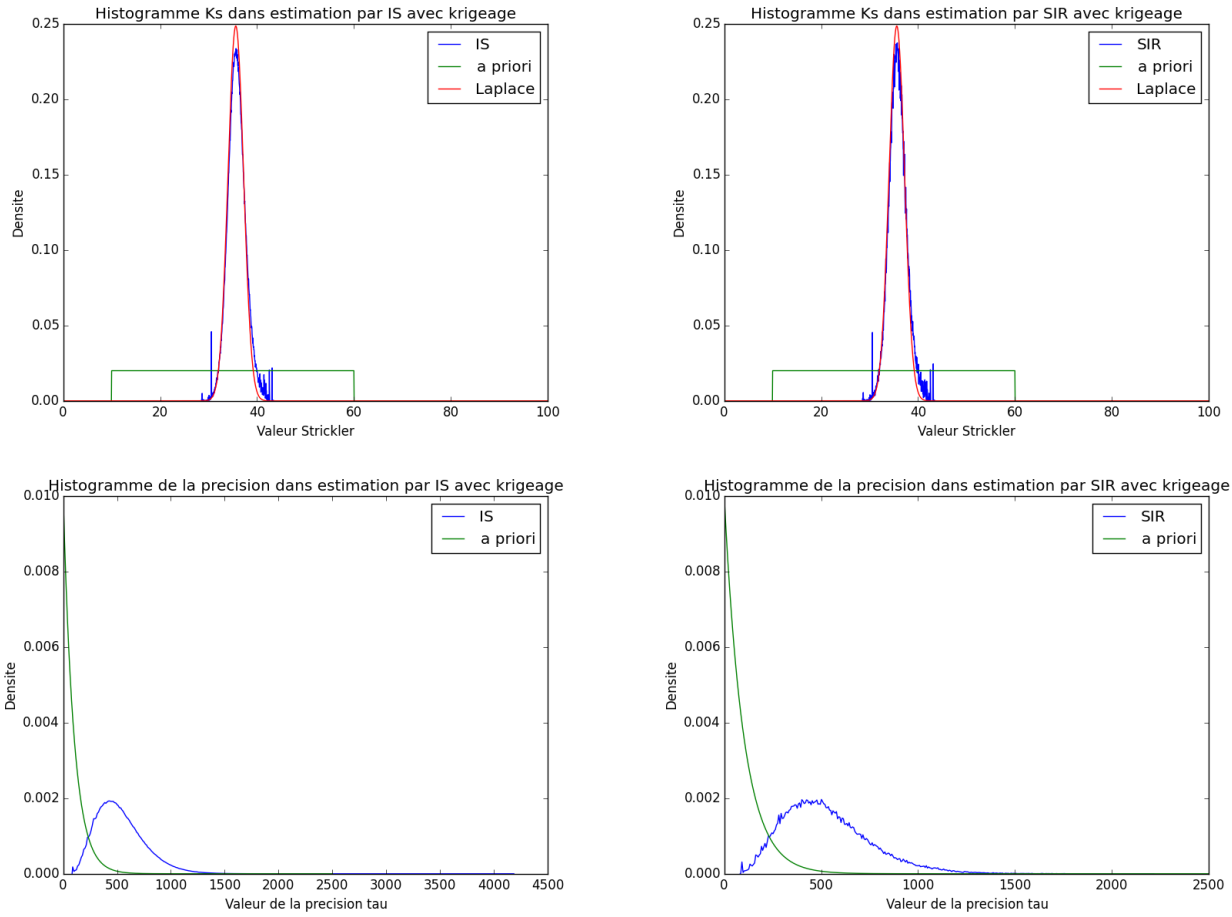


FIGURE 39: Importance Sampling et Sampling Importance Resampling par le méta - modèle sous TELEMAC

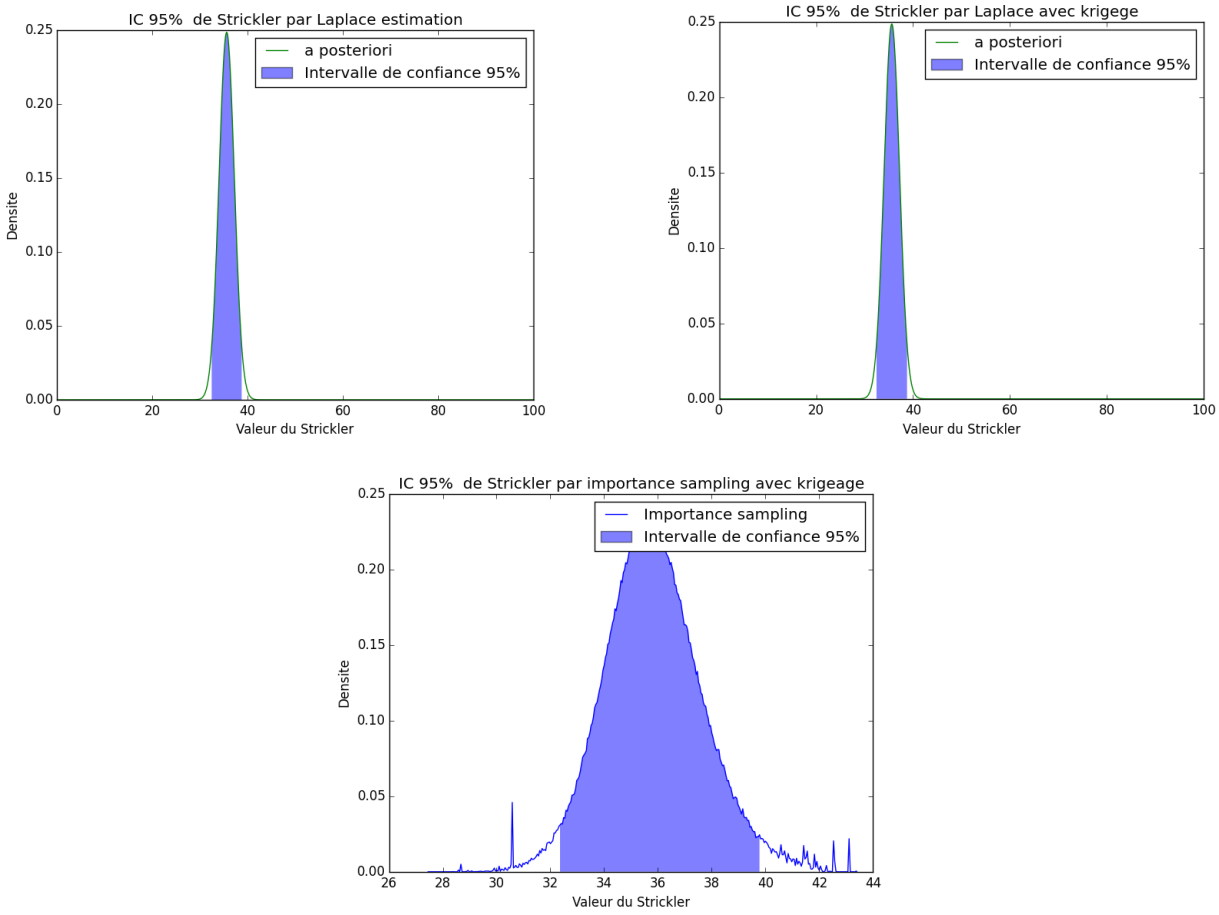


FIGURE 40: Intervalle de confiance et de crédibilité au niveau de 95% dans le cas estimation du TELEMAC2D

5 Application des méthodes de calage statistique dans le cas de la Garonne

Adaptation des formules

Dans le cas analytique ou le cas estimation, nous avons supposé qu'il existe une seule zone de frottement dans l'intégralité du fond. Cela signifie que la structure du lit est uniforme sur tout le canal. Cependant, dans le cas de la Garonne, le fond du fleuve est composé par cinq zones de frottement différentes (voir la figure (41)) :

— Zone 1 : lit mineur en aval de Tonneins,
— Zone 2 : lit mineur entre Mas d'Agenais et Marmande,
— Zone 3 : lit mineur en amont de la Réole,
— Zone 4 : lit majeur rive droite,
— Zone 5 : lit majeur rive gauche.

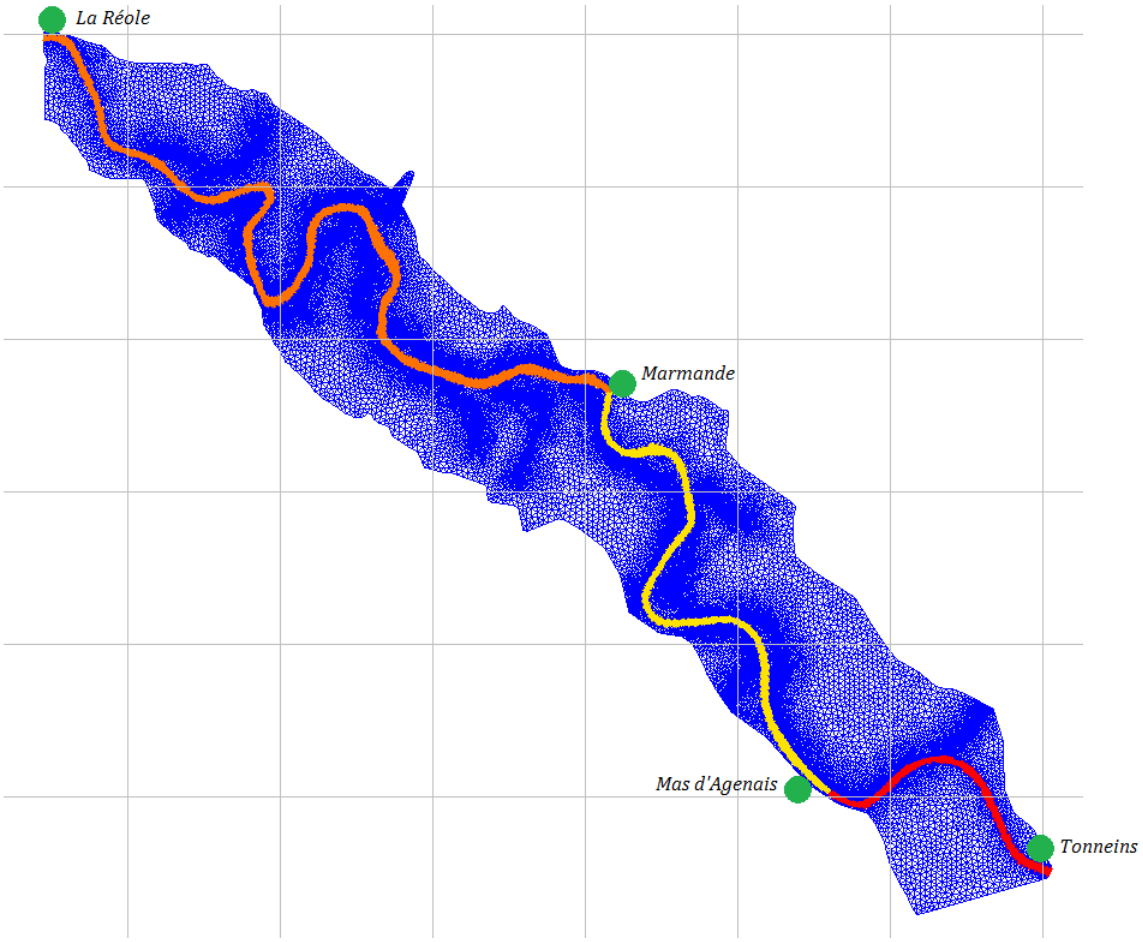


FIGURE 41: Distribution des zones de frottement de la Garonne entre Tonneins et la Réole

Moindres carrés :
A chaque station (La Réole, Marmande, Mas d'Agenais et Tonneins), nous disposons un certain nombre de mesures de débit et hauteur d'eau. Avec les méthodes de calage, nous allons caler le vecteur $(Ks_1, Ks_2, Ks_3, Ks_4, Ks_5)$ simultanément avec toutes les données des quatre ville. Cela signifie que l'entrée du code TELEMAC maintenant n'est plus un vecteur de dimension deux mais un vecteur de dimension six qui contient les cinq valeurs du coefficient de Strickler et le débit du fleuve. Par conséquent, nous pouvons écrire l'erreur de quadratique pour les moindres carrés comme suivant :

$$SC(\theta) = \sum_{i=1}^{m} \left(h_i^R - G^R(q_i^R, \theta)\right)^2 + \sum_{i=1}^{n} \left(h_i^M - G^M(q_i^M, \theta)\right)^2 + \sum_{i=1}^{p} \left(h_i^{MA} - G^{MA}(q_i^{MA}, \theta)\right)^2 + \sum_{i=1}^{q} \left(h_i^T - G^T(q_i^T, \theta)\right)^2$$

où :

- $H^R = \left(h_1^R, \cdots, h_m^R\right)$, $H^M = \left(h_1^M, \cdots, h_n^M\right)$, $H^{MA} = \left(h_1^{MA}, \cdots, h_p^{MA}\right)$ et $H^T = \left(h_1^T, \cdots, h_q^T\right)$ sont respectivement les hauteurs d'eau observées à La Réole, Marmande, Mas d'Agenais et Tonneins,
- $Q^R = \left(q_1^R, \cdots, q_m^R\right)$, $Q^M = \left(q_1^M, \cdots, q_n^M\right)$, $Q^{MA} = \left(q_1^{MA}, \cdots, q_p^{MA}\right)$ et $Q^T = \left(q_1^T, \cdots, q_q^T\right)$ sont respectivement les débits mesurés à La Réole, Marmande, Mas d'Agenais et Tonneins,
- G^R, G^M, G^{MA} et G^T sont respectivement les hauteurs d'eau calculées par TELEMAC à La Réole, Marmande, Mas d'Agenais et Tonneins,
- $\theta = \left(Ks_1, Ks_2, Ks_3, Ks_4, Ks_5\right)$ est un vecteur qui contient les valeurs de Strickler sur les zones de frottement,
- m, n, p et q sont respectivement les nombres de données à La Réole, Marmande, Mas d'Agenais et Tonneins car on n'a pas les mêmes nombres de données pour toutes les stations.

Maximum de vraisemblance :
De la même façon, nous pouvons réécrire la vraisemblance de manière suivante :

$$\begin{aligned}
\mathcal{L}(y|\delta) &= \prod_{i=1}^{m} f_{\epsilon_i}\left(h_i^R - G^R(q_i^R,\theta)\right) \prod_{i=1}^{n} f_{\epsilon_i}\left(h_i^M - G^M(q_i^M,\theta)\right) \prod_{i=1}^{p} f_{\epsilon_i}\left(h_i^{MA} - G^{MA}(q_i^{MA},\theta)\right) \times \cdots \\
&\quad \prod_{i=1}^{q} f_{\epsilon_i}\left(h_i^T - G^T(q_i^T,\theta)\right) \\
&= \left(\frac{e^\gamma}{2\pi}\right)^{(m+n+p+q)/2} exp\left[-\frac{e^\gamma}{2}\left(\sum_{i=1}^{m}\left(h_i^R - G^R(q_i^R,\theta)\right)^2 + \sum_{i=1}^{n}\left(h_i^M - G^M(q_i^M,\theta)\right)^2 + \cdots\right.\right. \\
&\quad \left.\left.\sum_{i=1}^{p}\left(h_i^{MA} - G^{MA}(q_i^{MA},\theta)\right)^2 + \sum_{i=1}^{q}\left(h_i^T - G^T(q_i^T,\theta)\right)^2\right)\right] \\
\Rightarrow log\mathcal{L}(y|\delta) &= \frac{m+n+p+q}{2}\gamma - \frac{m+n+p+q}{2}log(2\pi) - \frac{e^\gamma}{2}\left[\sum_{i=1}^{m}\left(h_i^R - G^R(q_i^R,\theta)\right)^2 + \cdots\right. \\
&\quad \left.\sum_{i=1}^{n}\left(h_i^M - G^M(q_i^M,\theta)\right)^2 + \sum_{i=1}^{p}\left(h_i^{MA} - G^{MA}(q_i^{MA},\theta)\right)^2 + \sum_{i=1}^{q}\left(h_i^T - G^T(q_i^T,\theta)\right)^2\right] \qquad (5.1)
\end{aligned}$$

avec $\delta = (\theta,\gamma) = (Ks_1, Ks_2, Ks_3, Ks_4, Ks_5, \gamma)$. Donc, nous allons chercher le vecteur δ_{MLE} qui maximise le logarithme de la vraisemblance ci-dessus. Une fois que ce vecteur est déterminé, nous pouvons réaliser les calculs de l'intervalle de confiance en commençant par calculer l'information de Fisher. Dans ce cas, la matrice de l'information de Fisher deviendra une matrice de dimension 6×6 définie par les formules suivantes :

$$\begin{aligned}
\forall k,l \in \{1,\cdots,6\}, \quad I_{k,l} &= -\mathbb{E}_{y|\delta_{MLE}}\left[\frac{\partial log\mathcal{L}}{\partial\delta_k}(y|\delta_{MLE}) \times \frac{\partial log\mathcal{L}}{\partial\delta_l}(y|\delta_{MLE})\right] \\
&= -\mathbb{E}_{y|\delta_{MLE}}\left[\frac{\partial^2 log\mathcal{L}}{\partial\delta_k\partial\delta_l}(y|\delta_{MLE})\right]
\end{aligned}$$

Or,

$\forall k,l \in \{1,\cdots,5\}$,

$$\begin{aligned}
\frac{\partial^2 log\mathcal{L}}{\partial Ks_k^2} &= e^{\gamma_{MLE}}\left[\sum_{i=1}^{m}\left[-\left(\frac{\partial G^R}{\partial Ks_k}(q_i^R,\theta_{MLE})\right)^2 + \left(h_i^R - G^R(q_i^R,\theta_{MLE})\right)\left(\frac{\partial^2 G^R}{\partial Ks_k^2}(q_i^R,\theta_{MLE})\right)\right] + \cdots\right. \\
&\quad \left.+\sum_{i=1}^{q}\left[-\left(\frac{\partial G^T}{\partial Ks_k}(q_i^T,\theta_{MLE})\right)^2 + \left(h_i^T - G^T(q_i^T,\theta_{MLE})\right)\left(\frac{\partial^2 G^T}{\partial Ks_k^2}(q_i^T,\theta_{MLE})\right)\right]\right] \\
\Rightarrow I_{k,k} &= e^{\gamma_{MLE}}\left(\sum_{i=1}^{m}\left(\frac{\partial G^R}{\partial Ks_k}(q_i^R,\theta_{MLE})\right)^2 + \cdots + \sum_{i=1}^{q}\left(\frac{\partial G^T}{\partial Ks_k}(q_i^T,\theta_{MLE})\right)^2\right) \\
&\quad (\text{car } \mathbb{E}_{y|\delta_{MLE}}\left[h^R - G^R\right] = \cdots = \mathbb{E}_{y|\delta_{MLE}}\left[h^T - G^T\right] = 0)
\end{aligned}$$

$$\begin{aligned}
\frac{\partial^2 log\mathcal{L}}{\partial Ks_k\partial Ks_l} &= e^{\gamma_{MLE}}\left[\sum_{i=1}^{m}\left[\left(\frac{\partial G^R}{\partial Ks_k}\right)\left(\frac{\partial G^R}{\partial Ks_l}\right) + \left(h_i^R - G^R\right)\left(\frac{\partial^2 G^R}{\partial Ks_k\partial Ks_l}\right)\right] + \cdots\right. \\
&\quad \left.+\sum_{i=1}^{q}\left[\left(\frac{\partial G^T}{\partial Ks_k}\right)\left(\frac{\partial G^T}{\partial Ks_l}\right) + \left(h_i^T - G^T\right)\left(\frac{\partial^2 G^T}{\partial Ks_k\partial Ks_l}\right)\right]\right] \\
\Rightarrow I_{k,l} &= e^{\gamma_{MLE}}\left[\sum_{i=1}^{m}\left(\frac{\partial G^R}{\partial Ks_k}(q_i^R,\theta_{MLE})\right)\left(\frac{\partial G^R}{\partial Ks_l}(q_i^R,\theta_{MLE})\right) + \cdots\right. \\
&\quad \left.+\sum_{i=1}^{q}\left(\frac{\partial G^T}{\partial Ks_k}(q_i^T,\theta_{MLE})\right)\left(\frac{\partial G^T}{\partial Ks_l}(q_i^T,\theta_{MLE})\right)\right]
\end{aligned}$$

et :

$$\forall k \in \{1, \cdots, 5\},$$

$$\begin{aligned}
\frac{\partial^2 log\mathcal{L}}{\partial \gamma^2} &= -\frac{e^{\gamma_{MLE}}}{2}\left[\sum_{i=1}^{m}\left(h_i^R - G^R(q_i^R, \theta_{MLE})\right)^2 + \cdots + \sum_{i=1}^{q}\left(h_i^T - G^T(q_i^T, \theta_{MLE})\right)^2\right] \\
\Rightarrow I_{6,6} &= \frac{m+n+p+q}{2} \\
&\quad (\text{car } \mathbb{E}_{y|\delta_{MLE}}\left[\left(h^R - G^R(q^R, \theta_{MLE})\right)^2\right] = \cdots = \mathbb{E}_{y|\delta_{MLE}}\left[\left(h^T - G^T(q^T, \theta_{MLE})\right)^2\right] = e^{\gamma_{MLE}}) \\
\frac{\partial^2 log\mathcal{L}}{\partial \gamma \partial Ks_k} &= e^{\gamma_{MLE}}\left[\sum_{i=1}^{m}\left(h_i^R - G^R(q_i^R, \theta_{MLE})\right)\left(\frac{\partial G^R}{\partial Ks_k}\right) + \cdots + \sum_{i=1}^{q}\left(h_i^T - G^T(q_i^T, \theta_{MLE})\right)\left(\frac{\partial G^T}{\partial Ks_k}\right)\right] \\
\Rightarrow I_{6,k} &= 0
\end{aligned}$$

Les formules ci-dessus nous permettent de calculer l'information de Fisher ainsi que l'intervalle de confiance par maximum de vraisemblance. Dans la suite, nous pouvons récupérer ces calculs pour obtenir la matrice hessienne dans la méthode approximation de Laplace.

Approximation de Laplace :
Comme nous avons plusieurs zones de frottement, le coefficient de Strickler de chaque zone a des bornes de valeurs différentes. La plage des valeurs est donnée par le tableau (8) suivant :

Zone de frottement	Valeur
Zone 1	$I_1 = [40; 50]$
Zone 2	$I_2 = [33; 43]$
Zone 3	$I_3 = [35; 45]$
Zone 4	$I_4 = [12; 22]$
Zone 5	$I_5 = [12; 22]$

TABLE 8: Plage de valeur par zone de frottement

A partir de ces informations, nous déduisons la loi *"a priori"* du vecteur δ définie par :

$$\begin{aligned}
\pi(\delta) &= \frac{1}{10}\mathbb{I}_{[40;50]}(Ks_1) \times \frac{1}{10}\mathbb{I}_{[33;43]}(Ks_2) \times \frac{1}{10}\mathbb{I}_{[35;45]}(Ks_3) \times \frac{1}{10}\mathbb{I}_{[12;22]}(Ks_4) \times \frac{1}{10}\mathbb{I}_{[12;22]}(Ks_5) \times exp(\gamma - e^{\gamma}) \\
&= \frac{1}{10^5} exp(\gamma - e^{\gamma})\, \mathbb{I}_{[40;50]}(Ks_1)\mathbb{I}_{[33;43]}(Ks_2)\mathbb{I}_{[35;45]}(Ks_3)\mathbb{I}_{[12;22]}(Ks_4)\mathbb{I}_{[12;22]}(Ks_5)
\end{aligned}$$

En utilisant l'expression (5.1), nous obtenons la formule suivante qui permet de calculer la matrice hessienne de l'approximation de Laplace :

$$\begin{aligned}
log\left(\mathcal{L}(y|\delta)\pi(\delta)\right) &= log\mathcal{L}(y|\delta) + log\pi(\delta) \\
&= \begin{cases} log\mathcal{L}(y|\delta) - 5log(10) + \gamma - e^{\gamma} & \text{si } Ks1 \in I_1,\ Ks_2 \in I_2,\ Ks_3 \in I_3 \\ & \text{et } Ks_4, Ks_5 \in I_4 = I_5 \\ -\infty & \text{sinon} \end{cases} \qquad (5.2)
\end{aligned}$$

La matrice hessienne dans ce cas est aussi une matrice de taille 6×6 dont les coefficients peuvent être calculés facilement par l'expression (5.2).

Conclusions et perspectives

Le but de cette étude est d'utiliser les modèles de calage statistique pour retrouver les valeurs du coefficient de rugosité au fond des écoulements. A l'aide de la plate-forme SALOME Hydro, nous avons pu coupler les calculs hydrauliques de TELEMAC - 2D et les méthodes de calage s'écrivant en langage PYTHON. Dans cette étude, les variables incertaines considérées sont tous les coefficients de frottement du lit mineur et du lit majeur ainsi que le débit entrant du modèle.

L'application des différentes méthodes de calage présentée dans ce rapport est divisée en trois grandes étapes : application dans le cas analytique avec la solution du système de Saint - Venant 1D, application dans le cas estimation du TELEMAC - 2D sur un canal de section rectangulaire et mise en place pour le cas réel de la Garonne entre Tonneins et La Réole. Le fait de partir d'un cas test simple nous permet d'avoir une observation globale sur l'efficacité de ces méthodes. De plus, pour réduire le temps de calcul, nous avons utilisé le méta - modèle créé par les processus gaussiens et l'efficacité de ce méta - modèle a été validé dans les deux premiers cas tests.

Dans ce document, la première partie rassemble l'explication de l'ensemble des méthodes de calage que l'on a utilisé. Les deux premières méthodes, moindres carrés et maximum de vraisemblance, sont simple à appliquées. De plus, le principe du maximum de vraisemblance nous permet d'obtenir un intervalle de confiance, qui donne une vue sur la degré de précision du résultat. Par contre, ces algorithmes fonctionnent bien seulement quand on dispose un grand nombre de données. Dans le cas contraire, le résultat obtenu ne seront plus exact. D'autre part, avant de commencer le calage, nous disposons des prévisions expertes sur les variables incertaines, par exemple, le coefficient de Strickler Ks est compris dans l'intervalle $[a, b]$ ou la précision τ suit une loi exponentielle de paramètre λ. Les algorithmes de moindres carrés et du maximum de vraisemblance ne tiennent pas compte de ces connaissances ci-dessus tandis que la procédure bayésienne considère ces connaissances expertes comme une loi *a priori* des variables incertaines. Parmi les méthodes bayésiennes, l'approximation de Laplace est la plus facile à appliquer. Elle a pourtant un inconvénient, c'est d'avoir besoin d'un nombre de données assez élevé pour bien approcher la loi *a posteriori*. La deuxième méthode de la procédure de Bayes est l'acceptation - rejet. L'avantage de cet algorithme est qu'on n'a pas besoin de beaucoup de données pour réaliser les calculs et à la fin, qu'on peut obtenir un vrai échantillon de la loi *a posteriori*. L'algorithme d'acceptation - rejet est basé sur la sélection des éléments d'un échantillon généré par une loi instrumentale donc il existe un taux d'acceptation. Selon la loi instrumental choisie, si ce taux est faible, la génération de l'échantillon de la loi *a posteriori* est très lent. Dans ce travail, le taux d'acceptation du cas analytique est 0.0666, cela signifie que pour obtenir un échantillon de taille 100000 de la loi *a posteriori*, il faut un échantillon de taille 1500000 de la loi instrumentale. Pour le cas estimation de TELEMAC, ce taux est encore beaucoup plus faible. Enfin, la méthode Importance Sampling est assez similaire à la précédente mais le temps de calcul est beaucoup plus court. Elle impose directement sur chaque élément de l'échantillon un poids normalisé qui correspond à la densité *a posteriori*.

Dans la deuxième partie, nous avons effectué respectivement les cas tests du plus simple, modèle analytique, au plus complexe, modèle de TELEMAC - 2D avec plusieurs zones de frottement. Grâce aux différents tests, nous pouvons confirmer que les hauteurs d'eau, calculées avec la valeur de Strickler obtenue et les débits mesurés, sont proches de celles observées dans les stations hydrologiques. Pour les cas tests de TELEMAC, l'utilisation du méta - modèle nous permet de réduire énormément le temps de calcul en restant assez proche du résultat final.

Ce travail visait à la réalisation d'une synthèse de différentes méthodes de calage statistique et à leurs applications sur le code TELEMAC - 2D via la plate - forme SALOME Hydro. Il serait intéressant d'appliquer ces méthodes dans les autres modules du système TELEMAC, par exemple pour SISYPHE pour les transports sédimentaires. Dans ce travail, nous avons arrêté sur Importance Sampling car il fonctionne bien avec TELEMAC - 2D mais il existe également les méthodes avancées comme méthode de Monte-Carlo par chaîne de Markov, qui permet d'obtenir un échantillon d'une variable aléatoire s'il n'est pas possible d'utiliser les méthodes usuelles. De plus, il pourrait être intéressant de mettre en place ces méthodes dans Salome pour automatiser le calage de TELEMAC.

6 Annexe

6.1 Relation avec le théorème de Bayes

Étant donné un modèle paramétrique d'observation $y \sim f(y|\delta)$, où $\delta \in \Theta$, un espace de dimension finie, l'analyse statistique bayésienne vise à exploiter la plus efficacement possible l'information apportée par y sur le paramètre δ, pour ensuite construire des procédures d'inférence sur δ. Bien δ ne soit qu'une réalisation d'une loi gouvernée par δ, elle apporte une actualisation aux informations préalablement recueillies par l'expérimentateur. L'information fournie par l'observation y est contenue dans la densité $f(y|\delta)$. Cette notation signifie qu'il s'agit d'une fonction de δ, qui est *inconnu*, dépendant de la valeur observée y.Il reflète le premier but de la statistique qui est de reconstruire le paramètre δ au vu de la réalisation aléatoire y. C'est donc pourquoi elle est naturellement liée au *Théorème de Bayes* qui formalise l'inversion des conditionnements dans les probabilités :

Théorème 1 (Théorème de Bayes)**.**
Si A et E sont des événements tels que $\mathbb{P}(E) \neq 0$, $\mathbb{P}(A|E)$ et $\mathbb{P}(E|A)$ sont reliés par :

$$\mathbb{P}(A|E) = \frac{\mathbb{P}(E|A)\mathbb{P}(A)}{\mathbb{P}(E)}$$

Une version continue de ce résultat permet d'inverser les densités conditionnelles,

$$g(y|x) = \frac{f(x|y)g(y)}{\int f(x|y)g(y)dy}$$

Le lien entre ces propriétés probabilistes et l'inférence bayésienne est que le paramètre inconnu δ n'est plus considéré comme inconnu et déterministe, mais comme une variable aléatoire. On considère ainsi que l'incertitude sur le paramètre δ d'un modèle peut être décrite par une distribution de probabilité π sur Θ, appelée *loi a priori*, par opposition à la *loi a posteriori* qui inclut l'information contenue dans l'observation y, ce qui revient à supposer que δ est distribué suivant $\pi(\delta)$, $\delta \sim \pi(\delta)$, avant que y soit généré par suivant $f(y[\delta)$.

6.2 Normalité asymptotique de l'estimateur du maximum de vraisemblance

Soient $X_1, \cdots, X_n$ les variables aléatoires f_θ - i.i.d, pour tout $\theta \in \Theta$. Nous appelons la vraisemblance des variables X_i du paramètre θ la probabilité suivante :

$$\mathcal{L}\left(X_1 = x_1, \cdots, X_n = x_n|\theta\right) = \prod_{i=1}^{n} f_\theta\left(X_i = x_i\right)$$

Nous appelons aussi l'estimateur du paramètre θ par maximum de vraisemblance la valeur :

$$\hat{\theta} = \underset{\theta\in\Theta}{argmin}\, \mathcal{L}\left(X_1 = x_1, \cdots, X_n = x_n|\theta\right)$$

La propriété asymptotiquement normale de cet estimateur donne la convergence en loi suivante :

$$\sqrt{n}\left(\hat{\theta} - \theta\right) \underset{n\to+\infty}{\longrightarrow} \mathcal{N}\left(0, I(\theta)^{-1}\right)$$

où $I(\theta)$ est la matrice d'information de Fisher définie par :

$$I_{i,j}(\theta) = \mathbb{E}_\theta\left[\frac{\partial log\, \mathcal{L}(x_1, \cdots, x_n|\theta)}{\partial\theta_i} \times \frac{\partial log\, \mathcal{L}(x_1, \cdots, x_n|\theta)}{\partial\theta_j}\right]$$

La démonstration de cette convergence a été détaillée dans le cours statistique de Marie - Claude VIANO et Charles SUQUET [18].

6.3 Calcul du logarithme de poids normalisé

$$\begin{aligned}
\forall r \in \{1, \cdots, \rho\}, \quad log(p_r) &= log\left(\frac{w_r}{\sum_{r=1}^{\rho} w_r}\right) = log(w_r) - log\left(\sum_{r=1}^{\rho} w_r\right) \\
&= log(w_r) - log\left(\underset{r}{max}(w_r) \sum_{r=1}^{\rho}\left(\frac{w_r}{\underset{r}{max}(w_r)}\right)\right) \\
&= log(w_r) - log\left(\underset{r}{max}(w_r)\right) - log\left(\sum_{r=1}^{\rho}\left(\frac{w_r}{\underset{r}{max}(w_r)}\right)\right) \\
&= log(w_r) - \underset{r}{max}\left(log(w_r)\right) - log\left[\sum_{r=1}^{\rho} exp\left(log(w_r) - \underset{r}{max}\left(log(w_r)\right)\right)\right] \quad (6.1)
\end{aligned}$$

où ρ est la taille de l'échantillon souhaité.

6.4 Rappel de la formule de Taylor et de différences finies

La formule de Taylor, établit par Brook Taylor en 1712, nous permet d'approcher d'une fonction plusieurs dérivable au voisinage d'un point par un polynôme dont les coefficients dépendent uniquement de la dérivée de cette fonction en ce point.
Notations : Soient I un intervalle de $\mathbb{R}$, x_0 un point intérieur à I, et $f : I \longrightarrow \mathbb{R}$ une fonction. On fixe un entier naturel n.

Théorème 2. *(Taylor - Young)*
Supposons que f soit de classe $\mathcal{C}^n$ sur I. Alors, pour tout $h \in \mathbb{R}$ tel que $x_0 + h$ appartienne à I, on peut écrire :

$$\begin{aligned}
f(x_0 + h) &= f(x_0) + hf'(x_0) + \frac{h^2}{2!} f^{(2)}(x_0) + \cdots + \frac{h^n}{n!} f^{(n)}(x_0) + \mathcal{O}(h^{n+1}) \\
&= \sum_{k=0}^{n} \frac{h^k}{k!} f^{(k)}(x_0) + \mathcal{O}(h^{n+1}) \quad (6.2)
\end{aligned}$$

Le formule (6.2) est appelée formule de Taylor d'ordre n pour f au voisinage de x. Ici, $f^{(k)}(x)$ désigne la dérivée k^{ime} de f, et $\mathcal{O}(h^{n+1})$ est une quantité qui tend vers 0 quand h tend vers 0. La formule de Taylor donne une approximation de $f(x+h)$, en connaissant f et ses dérivées au point x. Réciproquement, en connaissant la fonction f au voisinage de x, on peut en déduire une approximation des dérivées successives de f au point x.

Proposition 1.
Les formules suivantes donnent des approximations des dérivées de f au point x :

$$f'(x) = \frac{f(x+h) - f(x)}{h} + \mathcal{O}(h) \quad (6.3)$$

$$f'(x) = \frac{f(x) - f(x-h)}{h} + \mathcal{O}(h) \quad (6.4)$$

$$f'(x) = \frac{f(x+h) - f(x-h)}{2h} + \mathcal{O}(h^2) \quad (6.5)$$

$$f''(x) = \frac{f(x+h) - 2f(x) + f(x-h)}{h^2} + \mathcal{O}(h^2) \quad (6.6)$$

Ces formules sont à base des méthodes de différences finies utilisées pour résoudre les équations aux dérivées partielles de la physique. Elles se démontrent en écrivant la formule de Taylor à un ordre précis, aux points $x - h$ et $x + h$, et en faisant des combinaisons linéaires de ces équations.

Preuve proposition 1.
La démonstration de l'équation (6.3) se déroule dans l'ordre suivant :

$$\begin{aligned} & f(x+h) = f(x) + hf'(x) + \mathcal{O}(h^2) \\ \Rightarrow \quad & f'(x) = \frac{f(x+h) - f(x)}{h} + \frac{\mathcal{O}(h^2)}{h} \end{aligned}$$

d'où, en remarquant que $\frac{\mathcal{O}(h^2)}{h} = \mathcal{O}(h)$*, on déduit la formule annoncée.*
Pour l'équation (6.4), on utilise exactement la même méthode dans la démonstration précédente.
Pour l'équation (6.5), on utilise (6.2) aux points $x+h$ *et* $x-h$ *:*

$$\begin{aligned} & f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \mathcal{O}(h^3) \\ & f(x-h) = f(x) - hf'(x) + \frac{h^2}{2}f''(x) + \mathcal{O}(h^3) \\ \Rightarrow \quad & 2hf'(x) = f(x+h) - f(x-h) + \mathcal{O}(h^3) \\ \Leftrightarrow \quad & f'(x) = \frac{f(x+h) - f(x-h)}{2h} + \mathcal{O}(h^2) \end{aligned}$$

De la même façon, pour l'équation (6.6), on aura :

$$f(x+h) = f(x) + hf'(x) + \frac{h^2}{2}f''(x) + \frac{h^3}{6}f^3(x) + \mathcal{O}(h^4) \tag{6.7}$$

$$f(x) = f(x) \tag{6.8}$$

$$f(x-h) = f(x) - hf'(x) + \frac{h^2}{2}f''(x) - \frac{h^3}{6}f^3(x) + \mathcal{O}(h^4) \tag{6.9}$$

$$\Rightarrow \quad f(x+h) - 2f(x) + f(x-h) = h^2 f''(x) + \mathcal{O}(h^4) \tag{6.10}$$

$$\Leftrightarrow \quad f''(x) = \frac{f(x+h) - 2f(x) + d(x-h)}{h^2} + \mathcal{O}(h^2) \tag{6.11}$$

6.5 Conditionnement du vecteur gaussien

Pour démontrer l'expression de la covariance du krigeage (2.9), nous avons besoin de la proposition sur le conditionnement des vecteurs gaussiens dans le rapport de thèse de Clément Chevalier [15].

Proposition 2.
Soit $\begin{pmatrix} Y_1 \\ Y_2 \end{pmatrix}$ *un vecteur gaussien* $(Y_1 \in \mathbb{R}^p, Y_2 \in \mathbb{R}^q)$ *de densité :*

$$\mathcal{N}_{p+q}\left(\begin{pmatrix} m_1 \\ m_2 \end{pmatrix}, \begin{pmatrix} \Sigma_{11} & \Sigma_{12} \\ \Sigma_{21} & \Sigma_{22} \end{pmatrix}\right),$$

avec $m_1 \in \mathbb{R}^p$*,* $m_2 \in \mathbb{R}^q$*,* $\Sigma_{11} \in \mathbb{R}^{p\times p}$*,* $\Sigma_{12} = \Sigma_{21}^T \in \mathbb{R}^{p\times q}$*,* $\Sigma_{22} \in \mathbb{R}^{q\times q}$*, dont* Σ_{11}*,* Σ_{22} *sont semi - définie positives et* Σ_{22} *n'est pas singulier.*
Alors :

— *L'espérance conditionnelle de* Y_1 *sachant* Y_2 *coïncide avec l'espérance linéaire :*

$$\exists a \in \mathbb{R}^p, \boldsymbol{B} \in \mathbb{R}^{p\times q},\ \mathbb{E}(Y_1|Y_2) = a + \boldsymbol{B}Y_2 \tag{6.12}$$

— *La loi conditionnelle de* Y_1 *sachant* $Y_2 = y_2$ *est :*

$$\mathfrak{L}(Y_1|Y_2 = y_2) = \mathcal{N}_p\left(m_1 + \Sigma_{12}\Sigma_{22}^{-1}(y_2 - m_2),\ \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21}\right) \tag{6.13}$$

Cette proposition a été démontrée dans plusieurs livres. Ici, nous allons la démontrer dans un cas simple où $m = \begin{pmatrix} m_1 \\ m_2 \end{pmatrix} = 0$:

Preuve proposition 2.
Soit $\epsilon = Y_1 - \Sigma_{12}\Sigma_{22}^{-1}Y_2$, ϵ est un vecteur gaussien qui vérifie $cov(Y_2, \epsilon) = 0$:

$$\begin{aligned} cov(Y_2, \epsilon) &= cov\left(Y_2, Y_1 - \Sigma_{12}\Sigma_{22}^{-1}Y_2\right) \\ &= cov\left(Y_2, Y_1\right) - \Sigma_{12}^T\Sigma_{22}^{-1}var\left(Y_2\right) \quad \textit{(par bilinéarité de la covariance et } cov(u,u) = var(u)\textit{)} \\ &= \Sigma_{21} - \Sigma_{21}\Sigma_{22}^{-1}\Sigma_{22} \\ &= 0 \end{aligned}$$

Ce résultat signifie que Y_2 et ϵ sont indépendants. Autrement dit, le vecteur ϵ est orthogonal avec tous les variable $\}(Y_2)$ avec $\}$ une fonction de Borel. Alors :

$$\begin{aligned} \mathbb{E}(Y_1|Y_2) &= \mathbb{E}(\epsilon + \Sigma_{12}\Sigma_{22}^{-1}Y_2|Y_2) \\ &= \mathbb{E}(\epsilon|Y_2) + \mathbb{E}(\Sigma_{12}\Sigma_{22}^{-1}Y_2|Y_2) \quad \textit{(par linéarité de l'espérance)} \\ &= \mathbb{E}(\epsilon) + \Sigma_{12}\Sigma_{22}^{-1}Y_2 \quad \textit{(par propriété de l'espérance conditionnelle)} \end{aligned}$$

Cela entraîne le résultat (6.12).
Dans le deuxième temps, nous allons calculer la variance conditionnelle de Y_1 sachant Y_2. En décomposant Y_1, nous obtenons :

$$\begin{aligned} var\left(Y_1|Y_2\right) &= var\left(\epsilon + \Sigma_{12}\Sigma_{22}^{-1}Y_2|Y_2\right) \\ &= var(\epsilon|Y_2) + 2cov\left(\epsilon, \Sigma_{12}\Sigma_{22}^{-1}Y_2|Y_2\right) + var\left(\Sigma_{12}\Sigma_{22}^{-1}Y_2|Y_2\right) \\ &\quad \textit{(par propriété de la variance)} \\ &= var(\epsilon) + \mathbb{E}\left[\left(\Sigma_{12}\Sigma_{22}^{-1}Y_2\right)^2|Y_2\right] - \left[\mathbb{E}\left(\Sigma_{12}\Sigma_{22}^{-1}Y_2|Y_2\right)\right]^2 \\ &\quad \textit{(}cov\left(\epsilon, \Sigma_{12}\Sigma_{22}^{-1}Y_2|Y_2\right) = 0 \textit{ car } \epsilon \textit{ et } Y_2 \textit{ sont indépendants)} \\ &= var(\epsilon) + \left(\Sigma_{12}\Sigma_{22}^{-1}Y_2\right)^2 - \left(\Sigma_{12}\Sigma_{22}^{-1}Y_2\right)^2 \\ &= var(\epsilon) \\ &= var(Y_1 - \Sigma_{12}\Sigma_{22}^{-1}Y_2) \\ &= var(Y_1) - 2cov\left(Y_1, \Sigma_{12}\Sigma_{22}^{-1}Y_2\right) + var\left(\Sigma_{12}\Sigma_{22}^{-1}Y_2\right) \\ &= \Sigma_{11} - 2\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} + \Sigma_{12}\Sigma_{22}^{-1}var(Y_2)\left(\Sigma_{12}\Sigma_{22}^{-1}\right)^T \\ &\quad \textit{(car } var(AX) = AXA^T\textit{, avec } A \in \mathbb{R}^{n\times p} \textit{ et } X \in \mathbb{R}^p\textit{)} \\ &= \Sigma_{11} - 2\Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} + \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{22}\Sigma_{22}^{-1}\Sigma_{12}^T \quad \textit{(car } \Sigma_{22}^{-T} = \Sigma_{22}^{-1}\textit{)} \\ &= \Sigma_{11} - \Sigma_{12}\Sigma_{22}^{-1}\Sigma_{21} \end{aligned}$$

d'où vient le résultat (6.13).

6.6 Quantile d'une loi et quantile empirique

Le chapitre 4 du cours statistique de Léonard GALLARDO [16] nous a donné la définition du quantile dans le cas de la loi de probabilité connue et aussi le quantile empirique. La définition de la fonction quantile est basée sur la fonction de répartition de manière suivante :

Définition 3. *Quantiles d'une loi*
Soit $\mathcal{F}$ une fonction de répartition continue et strictement croissante. Pour tout $p \in]0,1[$, on appelle quantile d'ordre p et on note q_p la solution unique de l'équation $\mathcal{F}(x) = p$, i.e $q_p = \mathcal{F}^{-1}(p)$. En particulier :

- *si $p = \frac{1}{2}$, $q_{1/2}$ est appelé la médiane de la loi $\mathcal{F}$,*
- *si $p = \frac{1}{4}$, $q_{1/4}$ est appelé le premier quartile de la loi $\mathcal{F}$.*

Du coup, la fonction quantile empirique de l'échantillon $X_1, \cdots, X_n$ est donnée par :

$$Q_n(p) = F_n^{-1}(p) = inf\{x \in \mathbb{R},\ F_n(x) \geq p\}, \quad 0 < p < 1$$

6.7 Théorème de Bernoulli

Le théorème de Bernoulli est une application de la conservation de l'énergie au cas des fluides en mouvement. Un certain travail est fourni au fluide lorsqu'il passe d'un point à un autre et ce travail est égal à la variation d'énergie mécanique. Dans le cas d'un fluide incompressible, nous avons la relation suivante :

$$p_1 + \frac{1}{2}\rho v_1^2 + \rho g z_1 = p_2 + \frac{1}{2}\rho v_2^2 + \rho g z_2 + \Delta p_{1,2}$$

où p_i est la pression au point 1 ou 2, $\frac{1}{2}\rho v_i^2$ est l'énergie mécanique au point 1 ou 2, $\rho g z_i$ est l'énergie potentielle au point 1 ou 2 et $\Delta p_{1,2}$ désigne la perte d'énergie pendant le déplacement du point 1 au point 2.
Pour un fluide parfait, le terme $\Delta p_{1,2}$ est nul et l'équation de Bernoulli se réduit à :

$$p_1 + \frac{1}{2}\rho v_1^2 + \rho g z_1 = p_2 + \frac{1}{2}\rho v_2^2 + \rho g z_2 = constant$$

6.8 Optimisation du plan d'expérience par Monte Carlo

Dans la création du plan d'expérience, nous nous intéressons à maximiser l'espace de remplissage des paramètres d'entrée (dans ce cas ce sont les couples $(Ks_1, Q_1), \cdots, (Ks_n, Q_n)$). Un problème s'est posé maintenant est qu'une fois, on a un plan d'expérience, comment peut - on savoir que c'est celui qui maximise l'espace de remplissage dans le domaine imposé $[Ks_{inf}, Ks_{sup}] \times [Q_{inf}, Q_{sup}]$? Pour répondre à ce problème, il existe deux techniques qui ont été implémentées par OPENTURNS :

- Optimisation par Monte Carlo,
- Optimisation par recuit simulé.

Dans ce travail, nous avons utilisé la première méthode, optimisation par Monte Carlo dont l'algorithme est donné par :

Algorithm 1 Function MONTECARLOLHS. Algorithme d'optimisation du plan d'expérience LHS par la méthode de Monte Carlo, source [19]

Input :

$bounds$	:	les bornes de distributions uniformes
N	:	taille du plan d'expérience
ϕ	:	critère de remplissage de l'espace $(SpaceFillingC2, SpaceFillingMinDist, SpaceFillingPhiP)$ ▷ voir la documentation de OTLHS [19]
MC	:	nombre de simulations

Output :

LHS_{opt}	:	un plan optimal
ω_{opt}	:	valeur du critère

Function $\{LHS_{opt}, \omega_{opt}\} \leftarrow$ MONTECARLOLHS$(bounds, N, \phi, MC)$
  $\omega_{opt} \leftarrow 10^{308}$
  for $i \leftarrow 1$ **to** MC **do**
    $LHS \leftarrow$ LHSGENERATE$(bounds, N)$
    $\omega \leftarrow \phi(LHS)$
    if $\omega < \omega_{opt}$ **then**
      $LHS_{opt} \leftarrow LHS$
      $\phi_{opt} \leftarrow \omega$
    end if
  end for
end Function

Pour créer les plans d'expérience, nous avons choisi cette méthode car elle est simple à appliquée. Ici, les paramètres d'entrée de cette fonction sont $SpaceFillingMinDist$ comme critère de remplissage du domain et $MC = 50000$. Le critère $MinDist$, signifie *"distance minimale entre deux points du plan"* est donné par :

$$\phi(X) = min \|x^{(i)-x^{(j)}}\|_{L^2},\ \forall i \neq j = 1, \cdots, N$$

et ce critère doit être maximisé.

L'un des inconvénients majeurs de Monte Carlo échantillonnage est la consommation du temps de calcul, car le nombre de dessins générés doit être élevé.

6.9 Calcul intégral de Leibnitz

$$\frac{\partial}{\partial x}\int_{Z(x)}^{s(x)} f(x,y)dy = \int_{Z(x)}^{z(x)} \frac{\partial f}{\partial x}(x,y)dy - f(x,Z(x))\frac{\partial Z(x)}{\partial x} + f(x,z(x))\frac{\partial z(x)}{\partial x} \tag{6.14}$$

6.10 Démonstration de la prédiction du krigeage (2.8)

Supposons que le processus stochastique gaussien peut s'écrire sous la forme :

$$y_i \equiv y(x_i) = \sum_{j=1}^{d} f_j(x_i)\beta_j + \epsilon_i = \boldsymbol{f}^T(x_i)\boldsymbol{\beta} + \epsilon_i, \quad \text{pour } 1 \leq i \leq n$$

où f_i est la fonction de régression, $\boldsymbol{\beta} = (\beta_1, \cdots, \beta_p)^T$ est inconnu, et les erreurs de mesure $\{\epsilon_i\}$ ne sont pas corrélée avec une moyenne commune nulle et une variance commune σ^2. On considère le BLUP de $Y_0 = Y(x_0)$, le prédicteur $\widehat{Y_0} = a_0 + \boldsymbol{a}^T\boldsymbol{Y}$ est sans biais pour $Y(x_0)$ en condition de :

$$E\left[a_0 + \boldsymbol{a}^T Y\right] = a_0 + \boldsymbol{a}^T\boldsymbol{F}\boldsymbol{\beta} = E\left[Y_0\right] = \boldsymbol{f}_0^T\boldsymbol{\beta}$$

pour tout (β, σ^2), où $\boldsymbol{f}_0 = \boldsymbol{f}(x_0)$. Ce qui est équivalent à :

$$a_0 = 0 \quad \text{et} \quad \boldsymbol{F}^T\boldsymbol{a} = \boldsymbol{f}_0$$

Soient $\boldsymbol{Z} = \begin{pmatrix} Z(x_1) \\ \vdots \\ Z(x_n) \end{pmatrix}$ et $Z_0 = Z(x_0)$ sont respectivement les parties stochastiques de Y et de Y_0 de (2.7). Pour $\boldsymbol{\beta}$ et σ^2, le MSPE (Mean Squared Predictor Error ou Moyenne quacratique de l'erreur du prédicteur) de $\boldsymbol{a}^T\boldsymbol{Y}$ est :

$$\begin{aligned} E\left[\left(\boldsymbol{a}^T Y - Y_0\right)^2\right] &= E\left[\left(\boldsymbol{a}^T(\boldsymbol{F}\boldsymbol{\beta} + \boldsymbol{Z}) - (\boldsymbol{f}_0^T\boldsymbol{\beta} + Z_0)\right)^2\right] \\ &= E\left[\left((\boldsymbol{a}^T\boldsymbol{F} - \boldsymbol{f}_0^T)\boldsymbol{\beta} + \boldsymbol{a}^T\boldsymbol{Z} - Z_0\right)^2\right] \\ &= E\left[\boldsymbol{a}^T\boldsymbol{Z}\boldsymbol{Z}^T\boldsymbol{a} - 2\boldsymbol{a}^T\boldsymbol{Z}Z_0 + Z_0^2\right] \\ &= \sigma^2\boldsymbol{a}^T R_{LS}\boldsymbol{a} - 2\sigma^2\boldsymbol{a}^T r_0 + \sigma^2 \\ &= \sigma^2\left(\boldsymbol{a}^T R_{LS}\boldsymbol{a} - 2\boldsymbol{a}^T r_0 + 1\right) \end{aligned}$$

avec $r_0 = r(x_0)$. Donc le BLUP choisit $\boldsymbol{a}$ qui minimise la fonction :

$$\boldsymbol{a}^T R_{LS}\boldsymbol{a} - 2\boldsymbol{a}^T r_0 \tag{6.15}$$

avec la condition :

$$\boldsymbol{F}^T\boldsymbol{a} = \boldsymbol{f}_0 \tag{6.16}$$

En utilisant la méthode des multiplicateurs de Lagrange, nous pouvons minimiser la fonction (6.15) sous la contrainte (6.16). Ce problème devient :

$$\text{Trouver } (\boldsymbol{a}, \boldsymbol{\lambda}) \in \mathbb{R}^{n+p} \text{ qui minimise } \boldsymbol{a}^T R_{LS}\boldsymbol{a} - 2\boldsymbol{a}^T r_0 + 2\boldsymbol{\lambda}^T(\boldsymbol{F}^T\boldsymbol{a} - \boldsymbol{f}_0) \tag{6.17}$$

Pour résoudre (6.17), il faut d'abord calculer sont gradient et résoudre le système $\nabla\Phi = 0$ avec $\Phi(\boldsymbol{a}, \boldsymbol{\lambda}) = \boldsymbol{a}^T R_{LS}\boldsymbol{a} - \boldsymbol{a}^T r_0 + \boldsymbol{\lambda}^T(\boldsymbol{F}^T\boldsymbol{a} - \boldsymbol{f}_0)$:

$$\begin{aligned} & \begin{cases} \boldsymbol{F}^T\boldsymbol{a} - f_0 = 0 \\ R_{LS}\boldsymbol{a} - r_0 + \boldsymbol{F}\boldsymbol{\lambda} = 0 \end{cases} \\ \Leftrightarrow \quad & \begin{pmatrix} 0 & \boldsymbol{F}^T \\ \boldsymbol{F} & R_{LS} \end{pmatrix} \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{a} \end{pmatrix} = \begin{pmatrix} \boldsymbol{f}_0 \\ r_0 \end{pmatrix} \\ \Leftrightarrow \quad & \begin{pmatrix} \boldsymbol{\lambda} \\ \boldsymbol{a} \end{pmatrix} = \begin{pmatrix} 0 & \boldsymbol{F}^T \\ \boldsymbol{F} & R_{LS} \end{pmatrix}^{-1} \begin{pmatrix} \boldsymbol{f}_0 \\ r_0 \end{pmatrix} \end{aligned}$$

En appliquant la formule (2.6), nous avons le résultat suivant :

$$\begin{aligned} \boldsymbol{\lambda} &= \left(\boldsymbol{F}^T R_{LS}^{-1}\boldsymbol{F}\right)^{-1}\left(\boldsymbol{F}R_{LS}^{-1}r_0 - f_0\right) \\ \boldsymbol{a} &= R_{LS}^{-1}\left(r_0 - \boldsymbol{F}\boldsymbol{\lambda}\right) \end{aligned}$$

A partir de ce résultat, nous avons obtenir l'expression suivante :

$$\begin{aligned} \hat{Y}_0 &= R_{LS}^{-1}\left[r_0 - \boldsymbol{F}\left(\boldsymbol{F}^T R_{LS}^{-1}\boldsymbol{F}\right)^{-1}\left(\boldsymbol{F}R_{LS}^{-1}r_0 - f_0\right)\right]\boldsymbol{Y} \\ &= r_0^T R_{LS}^{-1}\boldsymbol{Y} - r_0^T R_{LS}^{-1}\boldsymbol{F}\beta + f_0\beta \\ &= f_0\beta + r_0^T R_{LS}^{-1}\left(\boldsymbol{Y} - \boldsymbol{F}\beta\right) \end{aligned}$$

avec $\beta = \left(\boldsymbol{F}^T R_{LS}^{-1}\boldsymbol{F}\right)^{-1}\boldsymbol{F}R_{LS}^{-1}\boldsymbol{Y}$.

6.11 Démonstration de la covariance (2.9)

Pour démontrer les expressions de la prédiction et de la covariance du krigeage, nous allons appliquer directement le résultat de la proposition dans l'annexe (6.5). Le méta - modèle est créé à base d'un plan d'expériences donné que l'on connaît la hauteur d'eau sur chaque point, Y. L'algorithme du krigeage dit que le méta - modèle obtenu est la moyenne des processus gaussiens, qui vérifient la loi conditionnelle (6.13) avec $Y_2 = Y$ et $Y_1 = y(x)$. Cela signifie que :

$$\mathfrak{L}\left(y(x)|Y\right) = \mathcal{N}\left[\mathbb{E}\left(y(x)|Y\right), cov\left(y(u), y(v)|Y\right)\right]$$

avec :

$$\begin{aligned} & \mathbb{E}\left(y(x)|Y\right) = \mathbb{E}(y(x)) + cov\left(y(x), Y\right)\left[cov\left(Y, Y\right)\right]^{-1}\left(Y - \mathbb{E}(Y)\right) \\ & cov\left(y(u), y(v)|Y\right) = cov\left(y(u), y(v)\right) - cov\left(y(u), Y\right)\ cov^{-1}\left(Y, Y\right)\ cov\left(y(v), Y\right) \end{aligned}$$

où :

$$\begin{aligned} & \mathbb{E}\left(y(x)\right) = \mathbb{E}\left(\beta F(x) + Z(x)\right) = \beta F(x) \quad \text{(car } Z \text{ suit une loi gaussienne centrée)} \\ & cov\left(y(x), Y\right) = cov\left(Z(x), Z\right) = \sigma^2 r(x) \quad (Y = \beta F(X) + Z) \\ & cov\left(Y, Y\right) = cov\left(Z, Z\right) = \sigma^2 R_{LS} \\ & \mathbb{E}(Y) = \mathbb{E}\left(\beta F(X) + Z\right) = \beta F(X) \\ & cov\left(y(u), y(v)\right) = cov\left(Z(u), Z(v)\right) = \sigma^2 R(u, v) \end{aligned}$$

A partir de tous ces résultats, nous obtenons :

$$\begin{aligned} & \hat{y}(x) = \mathbb{E}\left(y(x)|Y\right) = \beta F(x) + \sigma^2\ {}^t r(x) \times \left(\sigma^2 R_{LS}\right)^{-1}\left(Y - \beta F(X)\right) \\ = \quad & \beta F(x) +^t r(x) \times \mathbb{R}_{LS}^{-1}\left(Y - \beta F(X)\right) \quad \text{(une deuxième démonstration de (2.8))} \\ & cov\left(y(u), y(v)|Y\right) = \sigma^2 R(u, v) - \left(\sigma^2\ {}^t r(u)\right)\left(\sigma^2 R_{LS}\right)^{-1}\left(\sigma^2 r(v)\right) \\ = \quad & \sigma^2 R(u, v) - \sigma^2\ {}^t r(u) R_{LS}^{-1} r(v) \end{aligned}$$

Cela nous donne la démonstration de (2.9).

Références

[1] Pierre HUBERT, Aide mémoire d'hydraulique à surface libre,www.hydrologie.org/

[2] Eric PARENT et Jacques BERNIER, Le raisonnement bayésien, Modélisation et inférence

[3] Merlin KELLER, Calibration d'un modèle Modelica/Dymola du circuit secondaire du palier N4 pour le CEF, EDF R&D, MRI

[4] Merlin KELLER, Alberto PASANISI et Eric PARENT, Réflexions sur l'analyse d'incertitudes dans un contexte industriel : information disponible et enjeux décisionnels, Journal de la Société Française de Statistique, Vol. 152 No. 4, 2011

[5] Félix DEMANGEON, Cédric GOEURY, Fabrice ZAOUI, Nicole GOUTAL, Valérie PASCUAL et Laurent HASCOËT, Algorithmic differentiation applied to the optimal calibration of a shallow water model

[6] Kass, Tierrey et Kadane, The validity of posterior asymptotic approximations based on Laplace's method, 1990

[7] Astrid Jourdan, Planification d'expériences numériques, Revue MODULAD, 2005, https ://www.rocq.inria.fr/axis/modulad/archives/numero-33/jourdan-33/jourdan-33.pdf

[8] A. W. van der Vaart, Asymptotic Statistics, June 2000

[9] Kai-Tai FANG, Runge LI et Agus SUDJIANTO, Design and modeling for computer experiments, Computer Science and Data Analysis Series

[10] Michaël BAUDIN, Numerical Derivatives in Scilab, May 2009

[11] Amandine MARREL, Méta - modélisation par processus gaussien, 11 Avril 2013

[12] Bertrand IOOSS, Modelling of computer experiments by kriging, July 2015

[13] Scikit - learn, Gaussian Process regression : basic introductory example

[14] Amandine MARREL et Bertrand IOOSS, Rapport de stage - Modélisation des codes de calcul dans le cadre des processus gaussiens, 05/10/2005

[15] Clément CHEVALIER, Fast uncertainty reduction strategies relying on Gaussian process models, 18 September 2013

[16] Léonard GALLARDO, Cours statistique université de Tours, 2008

[17] Département Vois Navigables et Eau, Groupe d'Hydraulique Fluviale, Hydraulique des cours d'eau, la Théorie et sa mise en pratique, Août 2001

[18] Marie - Claude VIANO et Charles SUQUET, Eléments de statistique asymptotique, année 2010 - 2011

[19] Openturns, Documentation of the Otlhs module, December 17 2014, *http ://autobuilder.openturns.org/openturns/tags/openturns − 1.5rc1_r4237/modules/openturns − lhs/tags/otlhs − 1.0_r767/OTLHS_Documentation.pdf*

[20] Cédric GOEURY, Modélisation du transport des nappes d'hydrocarbures en zone continentale et estuarienne, 22 octobre 2012

[21] Système de modélisation TELEMAC, Manuel de l'utilisateur, octobre 2010.

[22] Rabih GHOSTINE, Contribution à la résolution numérique des équations de Barré de Saint - Venant bidimensionnelles par une méthode de type éléments finis discontinus : application à la simulation des écoulements au sein des carrefours dans la ville, 04 novembre 2009.

[23] Cedric GOEURY et Thomas DAVID, Quantification des incertitudes dans les modèles hydrauliques sous la plate-forme SALOME : Application au cas bidimensionnel de la Garonne, 09 Novembre 2015.