

Méthodes spécifiques

Régis LEBRUN

Airbus Central Research & Technology
regis.lebrun@airbus.com

27 janvier 2020

Plan de l'exposé

- 1 Méthodes asymptotiques
- 2 Propagation exacte
- 3 Formule sommatoire de Poisson

Plan de l'exposé

- 1 Méthodes asymptotiques
- 2 Propagation exacte
- 3 Formule sommatoire de Poisson

Plan de l'exposé

- 1 Méthodes asymptotiques
- 2 Propagation exacte
- 3 Formule sommatoire de Poisson

Contexte : propagation probabiliste d'incertitudes

On se donne :

- Un **modèle numérique** $f : \mathbb{R}^n \rightarrow \mathbb{R}^p$
- Un **vecteur aléatoire** \mathbf{X} à valeurs dans \mathbb{R}^n : les **incertitudes**

On cherche des informations sur la loi de $\mathbf{Y} = g(\mathbf{X})$:

- Des moments $\mathbb{E}[h(\mathbf{Y})]$
- La probabilité de certains événements $\mathbb{P}(\mathbf{Y} \in B)$

Une situation typique : $p = 1$, calculer $\mathbb{P}(Y \geq s)$ pour un seuil s donné.

Méthodes FORM et SORM I

Dans la pratique industrielle :

- g coûteuse à évaluer
- Y scalaire
- Seuil s tel que $p = \mathbb{P}(Y \geq s) \ll 1$

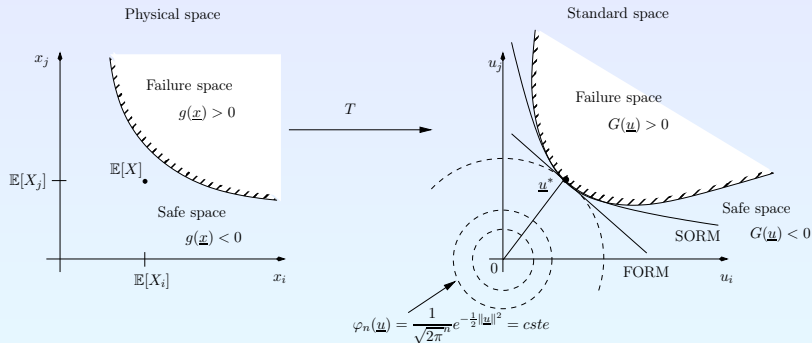
→ Méthode de Monte Carlo inefficace, précision relative $\mathcal{O}(1/\sqrt{Np})$

Alternatives : méthodes FORM (First Order Reliability Method) et SORM (Second Order Reliability Method) ([HasoferLind1974], [DitlevsenMadsen2005])

- 1 Construire $T : \mathbb{R}^n \rightarrow \mathbb{R}^n$ telle que $\mathbf{U} = T(\mathbf{X}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ et poser $G = g \circ T^{-1}$
- 2 Trouver $\mathbf{u}^* = \operatorname{argmin}_{G(\mathbf{u}) \geq s} \|\mathbf{u}\|$ et poser $\beta = \|\mathbf{u}^*\|$
- 3 Remplacer G par une approximation \tilde{G} linéaire (FORM) ou quadratique (SORM)
 - FORM : $\mathbb{P}(Y \geq s) \simeq \Phi(-\beta)$
 - SORM : $\mathbb{P}(Y \geq s) \simeq \Phi(-\beta) \prod_{i=1}^{n-1} \frac{1}{\sqrt{1-\beta\kappa_i}}$ où $(\kappa_i)_{i=1,\dots,n-1}$ sont les courbures principales de G en \mathbf{u}^*

où Φ désigne la fct de rép. de la loi Gaussienne standard.

Méthodes FORM et SORM II



Espace physique (\underline{X}), espace standard (\underline{U}), approximations FORM et SORM.

Rappel : changement de variable dans les intégrales

Théorème

Soit g une fonction d'un ouvert $\mathcal{O} \subset \mathbb{R}^d$ dans \mathbb{R}^n , mesurable intégrable. Soit ϕ un C^1 difféomorphisme de \mathcal{O} dans un ouvert $\mathcal{O}' \subset \mathbb{R}^d$. On a :

$$\begin{aligned} \int_{\mathcal{O}} g(\mathbf{x}) d\mathbf{x} &= \int_{\mathcal{O}'} \frac{g(\phi^{-1}(\mathbf{y}))}{|\text{Jac}[\phi](\phi^{-1}(\mathbf{y}))|} d\mathbf{y} \\ &= \int_{\mathcal{O}'} g(\psi(\mathbf{y})) |\text{Jac}[\psi](\mathbf{y})| d\mathbf{y} \end{aligned}$$

où $\psi = \phi^{-1}$.

Dans le cas $d = n = 1$, on a :

$$\begin{aligned} \int_{\mathcal{O}} g(x) dx &= \int_{\mathcal{O}'} \frac{g(\phi^{-1}(y))}{|\phi'(\phi^{-1}(y))|} dy \\ &= \int_{\mathcal{O}'} g(\psi(y)) |\psi'(y)| dy \end{aligned}$$

Rappel : calcul de courbure

Définition

Soit g une fonction de \mathbb{R}^n dans \mathbb{R} avec $n \geq 2$, telle que $\forall x \in \mathbb{R}^n, \nabla g(x) \neq \mathbf{0}$. Pour $x_0 \in \mathbb{R}^n$, on note :

- $\Sigma(x_0)$ la variété définie implicitement par $\Sigma(x_0) = \{x \in \mathbb{R}^n \mid g(x) = g(x_0)\}$;
- $N = \nabla g(x)/|\nabla g(x)|$ l'application qui à $x \in \mathbb{R}^n$ associe le vecteur normal unitaire en x à $\Sigma(x)$

Les **courbures principales** de $\Sigma(x_0)$ en x_0 sont les $(n-1)$ valeurs propres de plus grande valeur absolue de $W(x_0) = -\partial N / \partial x|_{x=x_0}$.

Transformations iso-probabilistes I

Littérature : 2 choix pour T

- Transformation conditionnelle [Rosenblatt1952], universelle mais coût numérique élevé ;
- Transformation de Nataf ([DerKiureghianLiu1986a], [DerKiureghianLiu1986b]), cas particulier mais coût numérique réduit.

Transformations iso-probabilistes II

Définition

Soit F une fct de rep. n -dimensionnelle. La **transformation de Rosenblatt** associée est donnée par $T^R = T_2^R \circ T_1^R$ avec :

$$T_1^R : \mathbb{R}^n \rightarrow \mathbb{R}^n \quad \left| \quad \begin{array}{l} \mathbf{x} \mapsto \mathbf{y} = \begin{pmatrix} F_1(x_1) \\ \vdots \\ F_{k|1,\dots,k-1}(x_k | x_1, \dots, x_{k-1}) \\ \vdots \\ F_{n|1,\dots,n-1}(x_n | x_1, \dots, x_{n-1}) \end{pmatrix} \end{array} \right.$$

$$T_2^R : \mathbb{R}^n \rightarrow \mathbb{R}^n \quad \left| \quad \begin{array}{l} \mathbf{y} \mapsto \mathbf{w} = \begin{pmatrix} \Phi^{-1}(y_1) \\ \vdots \\ \Phi^{-1}(y_k) \\ \vdots \\ \Phi^{-1}(y_n) \end{pmatrix} \end{array} \right.$$

où $F_{k|1,\dots,k-1}$ est la fonction de répartition de X_k sachant X_1, \dots, X_{k-1} avec \mathbf{X} de fct de rep. F .

Théorème

Soit \mathbf{X} un vecteur aléatoire de dimension n de fct. de rép. F et T^R la transformation de Rosenblatt associée. $\mathbf{U} = T^R \mathbf{X}$ suit la loi $\mathcal{N}(\mathbf{0}, I_n)$.

Transformations iso-probabilistes III

$\mathbf{Z} = T_1^R(\mathbf{X})$ prend ses valeurs dans $[0, 1]^n$. Soit ϕ une fct continue bornée sur $[0, 1]^n$:

$$\begin{aligned}\mathbb{E}[g(\mathbf{Z})] &= \mathbb{E}[g(T_1^R(\mathbf{X}))] \\ &= \int_{\mathbb{R}^n} g(T_1^R(\mathbf{x})) p_{\mathbf{X}}(\mathbf{x}) d\mathbf{x} \\ &= \int_{\Omega} g(T_1^R(\mathbf{x})) p_{\mathbf{X}}(\mathbf{x}) d\mathbf{x}\end{aligned}$$

où $\Omega = \{\mathbf{x} \in \mathbb{R}^n \mid p_{\mathbf{X}}(\mathbf{x}) > 0\}$.

- $\mathbf{z} = T_1^R(\mathbf{x})$ est un C^1 -difféomorphisme entre Ω et $(0, 1)^n$
- $d\mathbf{z} = |\det(\mathbf{J}(\mathbf{x}))| d\mathbf{x}$, où \mathbf{J} est la Jacobienne de T_1^R :

$$\begin{aligned}J_{i,j} &= \frac{\partial (T_1^R)_i(\mathbf{x})}{\partial x_j} \\ &= \frac{\partial F_{i|1,\dots,i-1}(x_i | x_1, \dots, x_{i-1})}{\partial x_j}\end{aligned}$$

Transformations iso-probabilistes IV

On remarque que pour $j > i$, $J_{i,j} = 0$ car $F_{i|1,\dots,i-1}$ ne dépend pas de x_j . \mathbf{J} est donc triangulaire inférieure de déterminant :

$$\begin{aligned}\det(\mathbf{J}(\mathbf{x})) &= \prod_{k=1}^n \frac{\partial F_{k|1,\dots,k-1}(x_k | x_1, \dots, x_{k-1})}{\partial x_k} \\ &= \prod_{k=1}^n p_{k|1,\dots,k-1}(x_k | x_1, \dots, x_{k-1}) \\ &= p_{\mathbf{X}}(\mathbf{x}) > 0\end{aligned}$$

donc $dz = p_{\mathbf{X}}(\mathbf{x})d\mathbf{x}$ et :

$$\begin{aligned}\mathbb{E}[g(\mathbf{Z})] &= \int_{[0,1]^n} g(\mathbf{z}) d\mathbf{z} \\ &= \int_{\mathbb{R}^n} g(\mathbf{z}) \mathbb{1}_{[0,1]^n}(\mathbf{z}) d\mathbf{z}\end{aligned}$$

- \mathbf{Z} suit la loi uniforme sur $[0, 1]^n$
- $\mathbf{Y} = T_2^R(\mathbf{Z})$ suit la loi $\mathcal{N}(\mathbf{0}, \mathbf{I}_n)$ cqfd.

Transformations iso-probabilistes V

Définition

Soient F_1, \dots, F_n n fcts de rép. continues et R_0 une matrice symétrique définie positive à diagonale unité. La transformation de Nataf s'écrit $T = T_2 \circ T_1$ avec :

$$\begin{aligned}
 T_1 : \quad & \left| \begin{array}{ll} \mathbb{R}^n & \rightarrow \mathbb{R}^n \\ \mathbf{x} & \mapsto \mathbf{w} = \begin{pmatrix} \Phi^{-1} \circ F_1(x_1) \\ \vdots \\ \Phi^{-1} \circ F_n(x_n) \end{pmatrix} \end{array} \right. \\
 T_2 : \quad & \left| \begin{array}{ll} \mathbb{R}^n & \rightarrow \mathbb{R}^n \\ \mathbf{w} & \mapsto \mathbf{u} = \Gamma \mathbf{w} \end{array} \right. \quad (1)
 \end{aligned}$$

avec $\Gamma^t \Gamma = R_0^{-1}$, R_0 **corrélations fictives**.

a

Transformations iso-probabilistes VI

Théorème ([DutfoyLebrun2009a])

Soit \mathbf{X} vecteur aléatoire de marginales F_1, \dots, F_n et T la transformation de Nataf associée avec $\mathbf{R}_0 = \text{Cor} [\mathbf{T}_1(\mathbf{X})]$. On a :

- $T_1(\mathbf{X})$ a des marginales $\mathcal{N}(0, 1)$
- $\text{Cor} [\mathbf{T}(\mathbf{X})] = \mathbf{I}$
- $\mathbf{T}(\mathbf{X}) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$ ssi \mathbf{X} a une copule Gaussienne

+ recommandations méthodologiques :

- Impact de l'hypothèse de copule Gaussienne pour \mathbf{X} sur la valeur de $\mathbb{P}(Y \geq s)$
- \mathbf{R}_0 ne peut pas être spécifiée indépendamment de $\text{Cor} [\mathbf{T}_1(\mathbf{X})]$!

Transformations iso-probabilistes VII

Définition

- \mathbf{X} a une **distribution sphérique** ssi :

$$\forall \mathbf{Q} \in \mathcal{O}_n(\mathbb{R}), \quad \mathbf{X} \stackrel{\mathcal{L}}{=} \mathbf{Q}\mathbf{X} \quad (2)$$

- \mathbf{X} a une **distribution elliptique** ssi $\mathbf{X} = \boldsymbol{\mu} + \mathbf{A}\mathbf{U}$ avec \mathbf{U} à loi sphérique, $\boldsymbol{\mu} \in \mathbb{R}^n$ et $\mathbf{A} \in \mathcal{M}_n(\mathbb{R})$.
- Une **copule elliptique** est la copule d'une loi elliptique.

Une copule elliptique est associée à une loi elliptique qui est associée à une loi sphérique.

Exemples : loi Gaussienne, loi de Student.

Remarque

- Les lois marginales d'une distribution sphérique sont toutes égales, de fct de rep. E .
- Une loi sphérique absolument continue a une densité de la forme $p(\mathbf{u}) = \theta(\|\mathbf{u}\|^2)$ avec $\theta : \mathbb{R}^+ \rightarrow \mathbb{R}^+$, appelé le **générateur de densité**.

Transformations iso-probabilistes VIII

Définition ([DutfoyLebrun2009b])

Soient F_1, \dots, F_n n fcts de rép. continues, R_0 une matrice symétrique définie positive à diagonale unité et E la marginale commune d'une loi sphérique. La **transformation de Nataf généralisée** associée s'écrit $T = T_2 \circ T_1'$ avec :

$$\begin{aligned}
 T_1' : \quad & \left\{ \begin{array}{l} \mathbb{R}^n \rightarrow \mathbb{R}^n \\ \mathbf{x} \mapsto \mathbf{w} = \begin{pmatrix} E^{-1} \circ F_1(x_1) \\ \vdots \\ E^{-1} \circ F_n(x_n) \end{pmatrix} \end{array} \right. \\
 T_2 : \quad & \left\{ \begin{array}{l} \mathbb{R}^n \rightarrow \mathbb{R}^n \\ \mathbf{w} \mapsto \mathbf{u} = \Gamma \mathbf{w} \end{array} \right.
 \end{aligned} \tag{3}$$

avec $\Gamma^t \Gamma = R_0^{-1}$, R_0 **corrélations fictives**.

Transformations iso-probabilistes IX

Théorème ([DutfoyLebrun2009c])

Soit \mathbf{X} un vecteur aléatoire continu à copule elliptique C , T et T^R les transformations de Nataf généralisée et de Rosenblatt associées. On a :

$$T(\mathbf{X}) \stackrel{\mathcal{L}}{=} T^R(\mathbf{X}) \text{ ssi } C \text{ Gaussienne} \quad (4)$$

et alors :

$$\forall \mathbf{x} \in \mathbb{R}^n, T(\mathbf{x}) = T^R(\mathbf{x}) \quad (5)$$

Proposition

Soit F une fct de rep. n -dimensionnelle continue. On a $\forall \mathbf{P}$ matrice de permutation, $\forall \mathbf{x} \in \mathbb{R}^n$, $T^R(\mathbf{P}\mathbf{x}) = \mathbf{P}T^R(\mathbf{x})$ ssi la copule de F est la copule indépendante.

Formule de Breitung et approximations FORM/SORM généralisées I

Théorème

Soit \mathbf{X} vecteur aléatoire de marginales F_1, \dots, F_n , E la marginale d'une loi sphérique et $R_0 = \text{Cor} [T'_1(\mathbf{X})]$.

$T(\mathbf{X})$ a une loi sphérique de marginale E ssi \mathbf{X} a une copule elliptique associée à cette loi sphérique.

Sous ces hypothèses les méthodes FORM et SORM se généralisent :

- FORM : $\mathbb{P}(Y \geq s) \simeq E(-\beta)$
- SORM : $\mathbb{P}(Y \geq s) \simeq E(-\beta) \prod_{i=1}^{n-1} \frac{1}{\sqrt{1-\beta\kappa_i}}$ où $(\kappa_i)_{i=1,\dots,n-1}$ sont les courbures principales de G en \mathbf{u}^* , sous hypothèses supplémentaires sur la copule de \mathbf{X} .

Formule de Breitung et approximations FORM/SORM généralisées II

Proposition (Ratio de Mill généralisé)

Sous les hypothèses de SORM généralisée on a :

$$E(-\beta) = \left(-\frac{\pi\theta(\beta^2)}{\theta'(\beta^2)} \right)^{(n+1)/2} \frac{\theta(\beta^2)}{2\pi\beta} (1 + o(1)) \text{ quand } \beta \rightarrow \infty \quad (6)$$

où θ est le générateur de la densité de E .

Avec OpenTURNS I

But : calculer $\mathbb{P}\left(\frac{1}{e^{-2X_1} + e^{-2X_2}} \geq 4\right)$ où (X_1, X_2) suit une loi Gaussienne centrée réduite 2D à composantes indépendantes, en utilisant les approximations FORM et SORM.

Listing 1 – formSorm.py

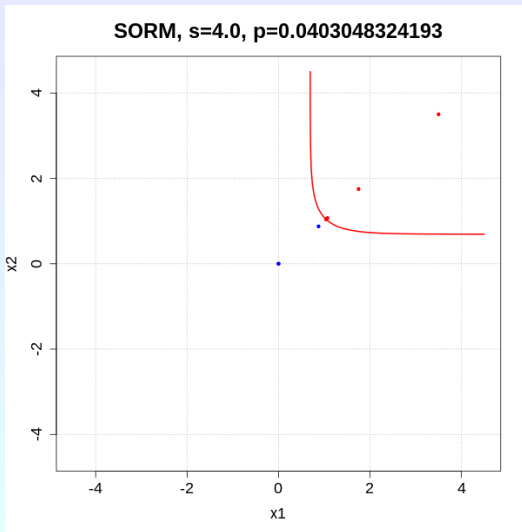
```
import openturns as ot
# Physical model
f = ot.SymbolicFunction(["x1", "x2"], ["1.0 / (exp(-2*x1)+exp(-2*x2))"])
# Input distribution
dist_x = ot.Normal(2)
# Input random vector
X = ot.RandomVector(dist_x)
# Output random vector
Y = ot.RandomVector(f, X)
# Event
s = 4.0
test = ot.Greater()
E = ot.Event(Y, test, s)
# FORM
form = ot.FORM(ot.AbdoRackwitz(), E, X.getMean())
form.run()
result = form.getResult()
```

Avec OpenTURNS II

```
# Probability
p = result.getEventProbability()
print("Probability FORM=", p)
# SORM
sorm = ot.SORM(ot.AbdoRackwitz(), E, X.getMean())
sorm.run()
result = sorm.getResult()
# Probability
p = result.getEventProbabilityBreitung()
print("Probability SORM=", p)
```

```
Probability FORM= 0.0707282995775788
Probability SORM= 0.04030483241925678
```

Avec OpenTURNS III

8 évaluations de f

Correction additive de l'approximation FORM : variable de contrôle I

On considère la variable $Z = \underline{u}^* \cdot \underline{u} - \beta^2$, qui est l'image de \underline{u} par l'application linéaire tangente à h en \underline{u}^* . On décompose la probabilité cherchée en :

$$\begin{aligned}\mathbb{P}[Y > s] &= \mathbb{P}[Z > 0] + \mathbb{P}[Z \leq 0, Y > s] - \mathbb{P}[Z > 0, Y \leq s] \\ &= \mathbb{P}[Z > 0] + \delta_1(s) - \delta_2(s)\end{aligned}\tag{7}$$

Dans le cas où l'approximation FORM est raisonnable, on a $\delta_1(s) \ll \mathbb{P}[Z > 0]$ et $\delta_2(s) \ll \mathbb{P}[Z > 0]$. On estime alors $\mathbb{P}[Y > s]$ par :

$$\begin{aligned}\hat{P}_n &= \mathbb{P}[Z > 0] + \\ &\quad \frac{1}{n} \sum_{i=1}^n \left(\mathbf{1}_{\{\underline{u}^* \cdot \underline{U}_i \leq \beta^2\} \cap \{h(\underline{U}_i) > s\}} - \mathbf{1}_{\{\underline{u}^* \cdot \underline{U}_i > \beta^2\} \cap \{h(\underline{U}_i) \leq s\}} \right)\end{aligned}$$

Réduction de variance de l'ordre de $\max(\delta_1(s), \delta_2(s))/\mathbb{P}[Z > 0] \rightarrow$ insuffisant si chaque évaluation de h coûte plusieurs minutes.

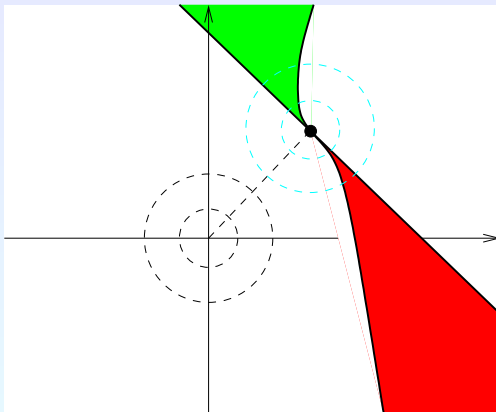
Correction additive de l'approximation FORM : variable de contrôle II

On combine alors la technique de la variable de contrôle avec un tirage préférentiel ayant pour densité de pondération la loi $\mathcal{N}(\underline{u}^*, \underline{I}_n)$. Le nouvel estimateur de $\mathbb{P}[Y > s]$ s'écrit :

$$\hat{P}_n = \mathbb{P}[Z > 0] + \frac{1}{n} \sum_{i=1}^n \left(\mathbf{1}_{\{\underline{u}^* \cdot \underline{U}_i \leq \beta^2\} \cap \{h(U_i) > s\}} - \mathbf{1}_{\{\underline{u}^* \cdot \underline{U}_i > \beta^2\} \cap \{h(U_i) \leq s\}} \right) \rho(\underline{U}_i)$$

où $\rho(\underline{U}_i) = \frac{\varphi(\underline{U}_i)}{\varphi^*(\underline{U}_i)}$, φ est la densité de $\mathcal{N}(0, \underline{I}_n)$ et φ^* celle de $\mathcal{N}(\underline{u}^*, \underline{I}_n)$.

Correction additive de l'approximation FORM : variable de contrôle III



δ_1 correspond à la probabilité portée par la zone rouge, δ_2 à celle portée par la zone verte.
Les iso-densités noires correspondent à la méthode de Monte Carlo, les bleues au tirage préférentiel.

Mise en œuvre I

On considère le modèle $g(x_1, x_2) = x_1 - \alpha x_2^2$, on choisit la loi $\mathcal{N}(\underline{0}, \underline{I}_2)$ pour le vecteur \underline{X} de telle sorte que la transformation T est l'identité : on a donc $h = f$.

On veut évaluer $\mathbb{P}[g(\underline{X}) > 2]$ par les méthodes suivantes :

M1 : Méthode FORM ;

M2 : Méthode de Monté Carlo standard ;

M3 : Méthode FORM avec correction par variable de contrôle ;

M4 : Méthode de Monté Carlo avec tirage préférentiel ;

M5 : Méthode FORM avec correction par variable de contrôle et tirage préférentiel

Mise en œuvre II

Test numérique $\alpha = 0.1$

On peut calculer la probabilité par une méthode d'intégration numérique type Gauss, qui donne $\mathbb{P}[g(X) > 2] \simeq 0.0186231$ (calcul Maple).

Résultats pour 10^4 tirages ($\text{Prec} = 1.96\sqrt{\hat{v}_n/n}$) :

	M1	M2	M3	M4	M5
Est. $\times 10^2$	2.275	1.91	1.855	1.864	1.895
Prec. $\times 10^3$	–	2.7	1.3	0.59	0.39
Acc.	–	1	2.1	4.5	6.8

On remarque qu'il s'agit d'un cas où la linéarisation est une approximation assez grossière (erreur de 22%).

Mise en œuvre III

Test numérique $\alpha = 0.02$

L'intégration de Gauss donne $\mathbb{P}[g(X) > 2] \simeq 0.0217319$ (calcul Maple).

Résultats pour 10^4 tirages ($\text{Prec} = 1.96\sqrt{\hat{v}_n/n}$) :

	M1	M2	M3	M4	M5
Est. $\times 10^2$	2.275	2.24	2.185	2.145	2.177
Prec. $\times 10^3$	–	2.9	0.59	0.65	0.22
Acc.	–	1	4.9	4.5	13.2

On remarque qu'il s'agit d'un cas où la linéarisation est une approximation très correcte (erreur de 2.6%).

Avec OpenTURNS I

But : calculer $\mathbb{P}\left(\frac{1}{e^{-2X_1} + e^{-2X_2}} \geq 4\right)$ où (X_1, X_2) suit une loi Gaussienne centrée réduite 2D à composantes indépendantes, en utilisant les méthodes d'échantillonnage basées sur les approximations FORM et SORM.

Listing 2 – analyticalSampling.py

```
import openturns as ot

# Physical model
f = ot.SymbolicFunction(["x1", "x2"], ["1.0 / (exp(-2*x1)+exp(-2*x2))"])
# Input distribution
dist_x = ot.Normal(2)
# Input random vector
X = ot.RandomVector(dist_x)
# Output random vector
Y = ot.CompositeRandomVector(f, X)
# Event
s = 4.0
test = ot.Greater()
E = ot.Event(Y, test, s)
# FORM
form = ot.FORM(ot.AbdoRackwitz(), E, X.getMean())
form.run()
```

Avec OpenTURNS II

```

result = form.getResult()
# Post-analytical controlled importance sampling
algo = ot.PostAnalyticalControlledImportanceSampling(result)
algo.setMaximumOuterSampling(100)
algo.setBlockSize(100)
algo.setMaximumCoefficientOfVariation(0.0)
algo.run()
result = algo.getResult()
# Probability
p = result.getProbabilityEstimate()
print("Probability estimate=", p)
l = result.getConfidenceLength(0.95)
print("Confidence interval 95\%=", ot.Interval(p - 0.5 * l, p + 0.5 * l))

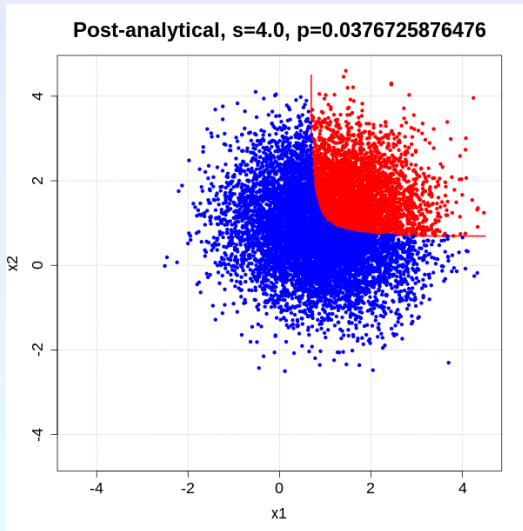
```

```

Probability estimate= 0.0381471661267695
Confidence interval 95\%= [0.0365189, 0.0397754]

```

Avec OpenTURNS III



$$\hat{\sigma}_n = 4.8 \cdot 10^{-9} \text{ (avec Monte Carlo, } \hat{\sigma}_n = 3.4 \cdot 10^{-6} \text{)}$$

Transformation de variable aléatoire I

Théorème

Soit X une variable aléatoire absolument continue de densité f_X et g une fonction réelle C^1 sans partie constante, telle que g' s'annule en un nombre au plus dénombrable de points $(x_n)_{n \in I}$.

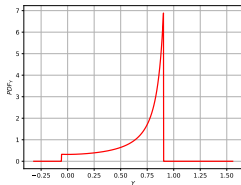
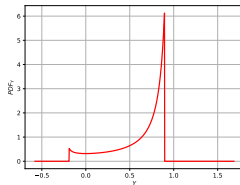
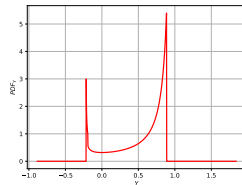
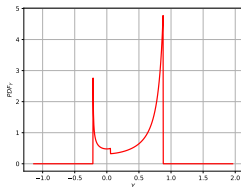
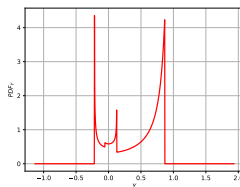
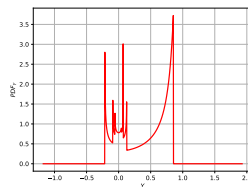
La variable aléatoire $Y = g(X)$ est absolument continue, de densité f_Y donnée par :

$$f_Y(y) = \sum_{n \in I} \frac{f_X(g^{-1}(y))}{|g' \circ g^{-1}(y)|} 1_{y \in g^{-1}([a_n, a_{n+1}])}$$

où $a_0 = \inf \text{supp } f_X$, $a_{\sup I} = \sup \text{supp } f_X$ et (a_1, \dots) tels que g soit monotone sur $[a_n, a_{n+1}[$.

Exercice : démontrer ce résultat.

Transformation de variable aléatoire II

 $Y = f(X \sin(1/X)), X = \text{Uniform}(a = 0.3, b = 1.3)$  $Y = f(X \sin(1/X)), X = \text{Uniform}(a = 0.25, b = 1.25)$  $Y = f(X \sin(1/X)), X = \text{Uniform}(a = 0.2, b = 1.2)$  $Y = f(X \sin(1/X)), X = \text{Uniform}(a = 0.15, b = 1.15)$  $Y = f(X \sin(1/X)), X = \text{Uniform}(a = 0.1, b = 1.1)$  $Y = f(X \sin(1/X)), X = \text{Uniform}(a = 0.05, b = 1.05)$ 

Transformation de variable aléatoire III

Exercices :

- Déterminer la loi de $Y = \arctan X$ où X suit la loi $\mathcal{N}(0, \sigma^2)$. Donner l'allure de la densité f_Y en fonction de σ .
- Déterminer la loi de $Y = \sin X$ où X suit la loi $\mathcal{N}(0, \sigma^2)$. Donner l'allure de la densité f_Y en fonction de σ .

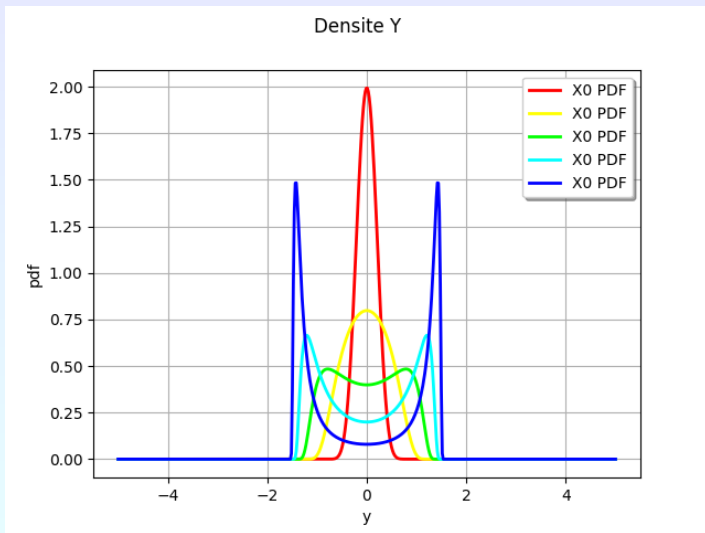
Avec OpenTURNS I

On souhaite déterminer les lois de $\arctan(X)$ et de $\sin(X)$ où $X \sim \mathcal{N}(0, \sigma^2)$.

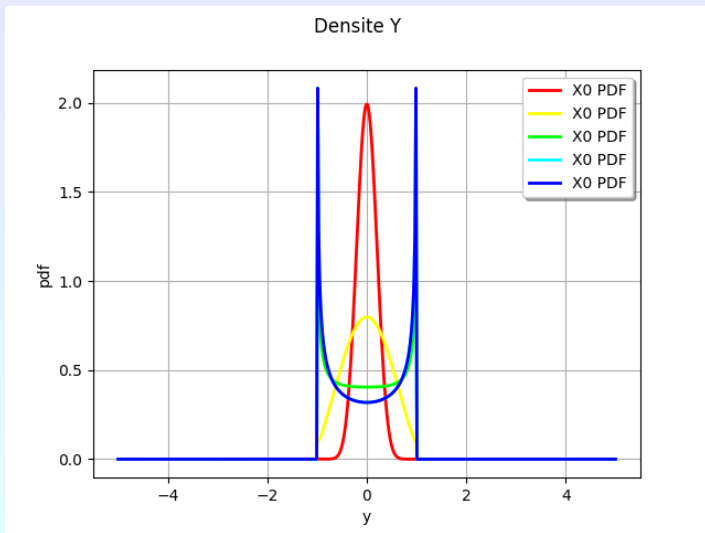
Listing 3 – compositeDistribution.py

```
import openturns as ot
import openturns.viewer as otv
# Les deux graphes à produire
graph_atan = ot.Graph("Densité Y", "y", "pdf", True, "topright")
graph_sin = ot.Graph("Densité Y", "y", "pdf", True, "topright")
# Pour chaque valeur de sigma
for sigma in [0.2, 0.5, 1.0, 2.0, 5.0]:
    # Distribution de l'arctangente d'une Normal
    dist_Y = ot.Normal(0.0, sigma).atan()
    graph_atan.add(dist_Y.drawPDF(-5.0, 5.0, 512))
    # Distribution du sinus d'une Normal
    dist_Y = ot.Normal(0.0, sigma).sin()
    graph_sin.add(dist_Y.drawPDF(-5.0, 5.0, 512))
    graph_atan.setColors(ot.Drawable.BuildDefaultPalette(5))
    graph_sin.setColors(ot.Drawable.BuildDefaultPalette(5))
    view = otv.View(graph_atan)
    view.save("Normal_atan.png")
    view.close()
    view = otv.View(graph_sin)
    view.save("Normal_sin.png")
    view.close()
```

Avec OpenTURNS II

Densité de Y en fonction de σ .

Avec OpenTURNS III

Densité de Y en fonction de σ .

Somme et produit de variables aléatoires indépendantes

Théorème

Soient X et Y deux variables aléatoires indépendantes absolument continues de densités respectives f_X et f_Y . Les variables aléatoires $W = X + Y$ et $Z = XY$ sont absolument continues de densité :

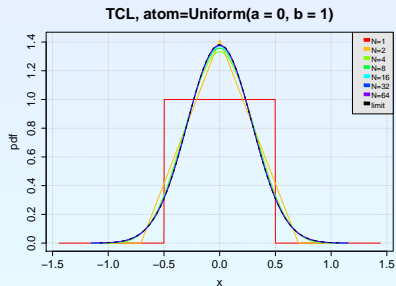
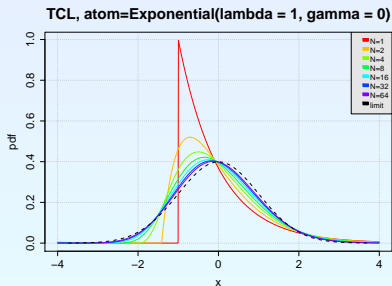
$$f_W(w) = \int_{\mathbb{R}} f_X(t) f_Y(w - t) dt$$
$$f_Z(z) = \int_{\mathbb{R}} f_X(t) f_Y(z/t) \frac{dt}{|t|}$$

Exercice : démontrer ce résultat.

Objectif

Développer une méthode numérique **précise**, **efficace** et **versatile** pour manipuler des combinaisons linéaires de variables aléatoires absolument continues.

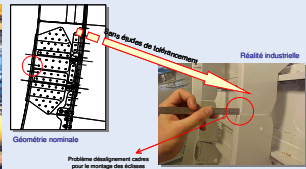
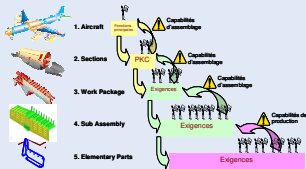
Diverses applications, comme par exemple vérifier le TCL :



Application : tolérancement

Objectifs

- Spécifier les pièces à produire ;
- Déterminer les séquences d'assemblage ;
- Gérer la complexité du produit ;
- Définir les moyens de contrôle de qualité.



Le tolérancement

Problématique industrielle

- Minimiser le taux de rebus et les dérogations ;
- Modèle hiérarchique très profond (jusqu'à 7 niveaux de détail) ;
- Nombre de paramètres incertains très élevé (10^5), non gaussiens ;
- Modèle numérique linéaire par rapport aux paramètres incertains ;
- Grand volume de données disponibles.

Modélisation mathématique du tolérancement

- On souhaite contrôler **une caractéristique majeure** Y d'un assemblage mécanique. Cet assemblage est défini par un ensemble de paramètres géométriques \mathbf{X} , et la valeur de la caractéristique majeure se déduit de celle du vecteur \mathbf{X} par une relation cinématique :

$$Y = g(\mathbf{X}) \quad (8)$$

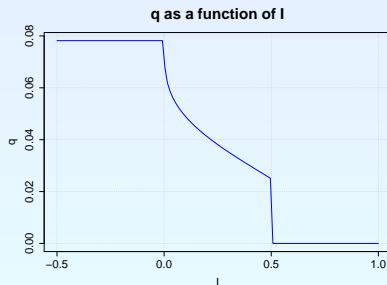
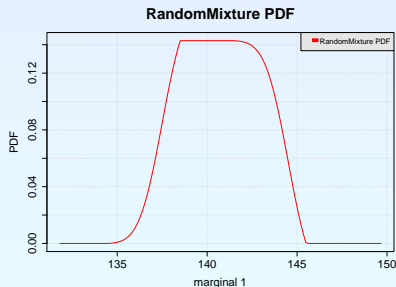
- Les étapes de fabrication induisent des dispersions dans les valeurs des paramètres : \mathbf{X} est modélisé par un vecteur aléatoire continu à composantes indépendantes.
- Dans le contexte aéronautique, ces dispersions sont de très faible amplitude : la relation (8) est linéarisée autour de la valeurs nominale \mathbf{x}_{nom} de \mathbf{X} (qui n'est **pas** $\mathbb{E}[\mathbf{X}]$).
- La grandeur Y est alors une variable aléatoire s'écrivant :

$$Y = \alpha_0 + \sum_{k=1}^n \alpha_k X_k \quad (9)$$

où $\alpha_0 = g(\mathbf{x}_{nom})$ et $\forall k \in \{1, \dots, n\}, \alpha_i = \frac{\partial f}{\partial x_k}(\mathbf{x}_{nom})$.

Modélisation mathématique du tolérancement

- On souhaite étudier ensuite Y , notamment la forme de sa densité et sa fonction de dispersion $\delta : [0, 1] \rightarrow \mathbb{R}^+$, $\delta(q) = (y_{1-q/2} - y_{q/2})/y_{nom}$ où $y_{1-q/2}$ et $y_{q/2}$ sont respectivement les quantiles d'ordre $1 - q/2$ et $q/2$ de Y et $y_{nom} = \alpha_0 + \sum_{k=1}^n \alpha_k x_{nomk}$.



Calculer la loi d'une somme de variables aléatoires indépendantes

Soit $Y = \alpha_0 + \sum_{k=1}^n \alpha_k X_k$ une variable aléatoire définie comme étant une combinaison affine de variables aléatoires indépendantes X_1, \dots, X_n . Les formules sommatoires de Poisson relient les valeurs prises par la loi de Y et une transformée intégrale de cette loi sur des réseaux (ie des sous-groupes discrets $h\mathbb{Z}$ de \mathbb{R}). Plus précisément :

- Dans le cas où Y est continue, la somme des valeurs prises par la densité de Y sur un réseau est égale à la somme des valeurs prises par sa fonction caractéristique sur un autre réseau judicieusement choisi ;
- Dans le cas où Y est discrète portée par un réseau entier ($h = 1$), la densité est remplacée par la fonction de probabilité et la fonction caractéristique par la fonction génératrice.

Dès lors qu'au moins un contributeur X_k est absolument continu, Y est absolument continue.

Le cas continu

Soit X une variable aléatoire réelle continue de densité p et de fonction caractéristique ϕ sommable, i.e $\int_{\mathbb{R}} |\phi(x)| dx < \infty$ où $\phi(u) = \mathbb{E}[e^{iuX}]$. Pour tout $h > 0$ et $x \in \mathbb{R}$ on a :

$$\sum_{k \in \mathbb{Z}} p\left(x + \frac{2k\pi}{h}\right) = \frac{h}{2\pi} \sum_{\ell \in \mathbb{Z}} \phi(h\ell) e^{-ih\ell x} \quad (10)$$

et par intégration, pour tout $h > 0$ et $x, y \in \mathbb{R}$ on a :

$$\sum_{k \in \mathbb{Z}} \left\{ F\left(x + \frac{2k\pi}{h}\right) - F\left(x - y + \frac{2k\pi}{h}\right) \right\} = \frac{hy}{2\pi} + \frac{h}{2\pi} \sum_{\ell \in \mathbb{Z}^*} \phi(h\ell) \frac{e^{-ih\ell x} - e^{-ih\ell(x-y)}}{ih\ell} \quad (11)$$

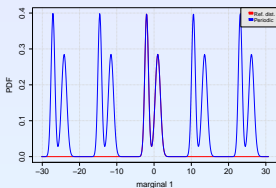
Démonstration

- On forme la densité **périodique** $\tilde{p}(x) = \sum_{k \in \mathbb{Z}} p\left(x + \frac{2k\pi}{h}\right)$;
- Cette densité est égale à sa décomposition en série de Fourier, qui converge par sommabilité de ϕ : $\tilde{p}(x) = \sum_{\ell \in \mathbb{Z}} c_{\ell} e^{ih\ell x}$;
- Le calcul du coefficient de rang ℓ de cette décomposition donne :

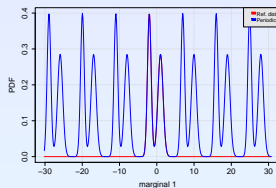
$$\begin{aligned}
 c_{\ell} &= \frac{h}{2\pi} \int_{-\pi/h}^{\pi/h} \tilde{p}(x) e^{-ih\ell x} dx \\
 &= \frac{h}{2\pi} \int_{-\pi/h}^{\pi/h} \sum_{k \in \mathbb{Z}} p\left(x + \frac{2k\pi}{h}\right) e^{-ih\ell x} dx \\
 &= \frac{h}{2\pi} \int_{-\infty}^{\infty} p(x) e^{-ih\ell x} dx \\
 &= \frac{h}{2\pi} \phi(-h\ell)
 \end{aligned}$$

A quoi ça sert ???

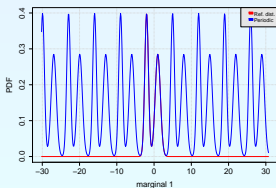
On compare p et \tilde{p} via $\epsilon = \max_{x \in [\mu - 3\sigma, \mu + 3\sigma]} |p(x) - \tilde{p}(x)|$:



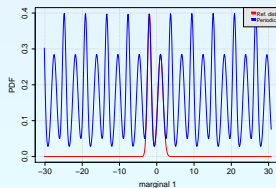
$$h = 0.5, \epsilon = 0.0$$



$$h = 0.7, \epsilon = 2.6 \cdot 10^{-4}$$



$$h = 0.9, \epsilon = 1.9 \cdot 10^{-1}$$



$$h = 1.1, \epsilon = 4.0 \cdot 10^{-1}$$

Vers une méthode numérique

Formule d'approximation

On réécrit la formule (10) :

$$p(x) = \frac{h}{2\pi} \sum_{|\ell| \leq L} \phi(h\ell) e^{-ih\ell x} + \eta(x, h, L) \quad (12)$$

où :

$$\eta(x, h, L) = \frac{h}{2\pi} \sum_{|\ell| > L} \phi(h\ell) e^{-ih\ell x} - \sum_{k \in \mathbb{Z}^*} p\left(x + \frac{2k\pi}{h}\right) \quad (13)$$

Vers une méthode numérique

Choix de h

L'objectif est de diminuer la valeur de $\sum_{k \in \mathbb{Z}^*} p\left(x + \frac{2k\pi}{h}\right)$ dans $\eta(x, h, L)$.

On note par μ et σ^2 l'espérance et la variance de X . On a :

$$\mu = \alpha_0 + \sum_{k=1}^n \alpha_k \mathbb{E}[X_k] \text{ et } \sigma^2 = \sum_{k=1}^n \alpha_k^2 \text{Var}[X_k]$$

On se donne un facteur $a > 0$ indiquant sur quel interval $[\mu - a\sigma, \mu + a\sigma]$ on souhaite approcher p par \tilde{p} , et un facteur $b > a$ tel que p soit négligeable hors de $[\mu - b\sigma, \mu + b\sigma]$ par rapport à $p_a^* = \max_{[\mu - a\sigma, \mu + a\sigma]} p(x)$

On choisit alors h tel que :

$$h = \frac{2\pi}{(b-a)\sigma} \quad (14)$$

et on a $\forall k \in \mathbb{Z}^*, \forall x \in [\mu - a\sigma, \mu + a\sigma], x \notin [\mu - b\sigma, \mu + b\sigma]$.

Dans les applications pratiques, Y est souvent assez proche d'une loi normale (TCL) donc le choix $a = 5$ et $b = 8$ garantit de calculer p à la précision machine sur un interval allant du quantile à $3 \cdot 10^{-7}$ à $1 - 3 \cdot 10^{-7}$

Vers une méthode numérique

Choix de L

L'objectif est de diminuer la valeur de $\frac{h}{2\pi} \sum_{|\ell| > L} \phi(h\ell) e^{-ih\ell x}$ dans $\eta(x, h, L)$.
 Il n'est pas évident de pouvoir borner à priori la valeur de ℓ , le calcul de L est donc incrémental. On part d'une valeur $L_0 \in \mathbb{N}^*$, puis on génère une suite $(L_j)_{j \geq 0}$ par $L_{j+1} = 2L_j$ tant que la contribution de la somme partielle :

$$\frac{h}{2\pi} \sum_{L_j < |\ell| \leq L_{j+1}} \phi(h\ell) e^{-ih\ell x}$$

contribue significativement à

$$\frac{h}{2\pi} \sum_{|\ell| \leq L_{j+1}} \phi(h\ell) e^{-ih\ell x}$$

Dans le cas où au moins un des contributeurs X_k a une fonction caractéristique à décroissance exponentielle (par exemple un contributeur de loi normale), la contribution du bloc $\{L_j + 1, \dots, L_{j+1}\}$ est de l'ordre de $e^{-(2^j L_0 + 1)h}$ donc décroît très rapidement avec j .

La méthode numérique finale

Si on souhaite utiliser la méthode numérique ébauchée précédemment pour le calcul d'une fonction de répartition par une utilisation directe de (11) les performances numériques sont mauvaises car on cherche à calculer une fonction bornée par correction successive de la fonction non bornée $\frac{hy}{2\pi}$.

La formule sommatoire de Poisson étant linéaire par rapport à p (et F), on introduit la loi normale «équivalente» $\mathcal{N}(\mu, \sigma^2)$ de même moyenne et variance que Y et on a :

$$p(x) = p^{\mathcal{N}}(x) + \frac{h}{2\pi} \sum_{|\ell| \leq L} (\phi - \phi^{\mathcal{N}})(h\ell) e^{-ih\ell x} + \sum_{k \in \mathbb{Z}^*} p^{\mathcal{N}}\left(x + \frac{2k\pi}{h}\right) + \tilde{\eta}(x, h, L) \quad (15)$$

où $\tilde{\eta}(x, h, L) = \eta(x, h, L) - \sum_{|\ell| > L} \phi^{\mathcal{N}}(h\ell) e^{-ih\ell x}$, et pour tout $h > 0$ et $x, y \in \mathbb{R}$ on a :

$$F(x) = F^{\mathcal{N}}(x) + F(x-y) - F^{\mathcal{N}}(x-y) + \frac{h}{2\pi} \sum_{\ell \in \mathbb{Z}^*} (\phi - \phi^{\mathcal{N}})(h\ell) \frac{e^{-ih\ell x} - e^{-ih\ell(x-y)}}{ih\ell} \quad (16)$$

La méthode numérique finale

Les avantages de cette formulation sont :

- Dans le cas où p est proche de $p^{\mathcal{N}}$, ce qui sera le cas quand $n \gg 1$, la correction à calculer entre p et $p^{\mathcal{N}}$ sera de très faible amplitude : dès L_1 on aura une excellente précision !
- Pour le calcul de la fonction de répartition, on a supprimé le terme $\frac{hy}{2\pi}$: la fonction calculée sera bornée par constuction !

Au bilan, on obtient une méthode dont le coût algorithmique est $\mathcal{O}(n|\log(\epsilon)|)$ pour calculer la densité ou la fonction de répartition de Y avec une précision absolue de ϵ , à comparer à une approche type Monte Carlo de coût $\mathcal{O}(n/\epsilon)$, inenvisageable pour $\epsilon \simeq 10^{-12}$ et $n \simeq 10^3$.

L'utilisation de la formule sommatoire de Poisson permet le calcul de la densité et de la fonction de répartition de Y essentiellement à la précision machine

Cette méthode souffre cependant d'un risque de cancellation (sensibilité aux erreurs d'arrondi) dont la prise en compte dépasse le cadre de cette présentation mais qui est pris en compte dans l'implémentation OpenTURNS.

Avec OpenTURNS I

On mesure le temps mis pour évaluer la densité de $W = \sum_{k=1}^n X_k$ en 10^3 points, avec X_k iid $X_1 \sim \mathcal{E}(1)$.

Listing 4 – randomMixture.py

```
import openturns as ot
import openturns.viewer as otv
from time import time

# Nombre d'évaluations de la pdf
N = 1000
data = ot.Sample(0, 2)
nMax = 0
tMax = 0.0
tMin = ot.SpecFunc.MaxScalar
for n in range(1, 1001):
    print("n=", n)
    distribution = ot.RandomMixture([ot.Exponential(i + 1.5) for i in range(n)])
    sample = distribution.getSample(N)
    t0 = time()
    pdf = distribution.computePDF(sample)
    t1 = time()
    data.add([n, (t1 - t0) / N])
    if data[n-1, 1] > tMax:
```

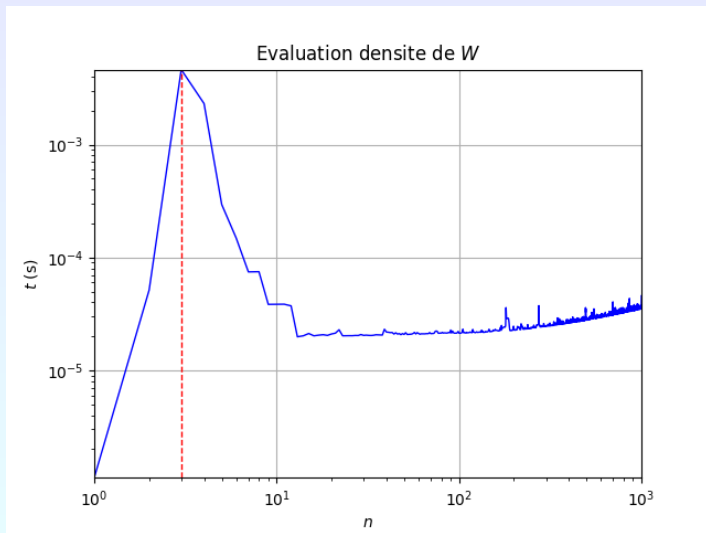
Avec OpenTURNS II

```

        tMax = data[n-1, 1]
        nMax = n
    if data[n-1, 1] < tMin:
        tMin = data[n-1, 1]
graph = ot.Graph("Evaluation densite de $$$", "$n$", "$t$ (s)", True, "", 1,
3)
graph.add(ot.Curve(data))
curve = ot.Curve([[nMax, tMin], [nMax, tMax]])
curve.setColor("red")
curve.setLineStyle("dashed")
graph.add(curve)
graph.setXMargin(0.0)
graph.setYMargin(0.0)
view = otv.View(graph)
view.save("bench_poisson.png")
view.close()

```

Avec OpenTURNS III





A. Der Kiureghian and P. L. Liu.

Structural reliability under incomplete probabilistic information.

Journal of Engineering Mechanics, 112(1) :85–104, 1986.



A. Der Kiureghian and Liu P. L.

Multivariate distribution models with prescribed marginals and covariances.

Probabilistic Engineering Mechanics, 1(2) :105–112, 1986.



O. Ditlevsen and H. O. Madsen.

Structural Reliability Methods.

John Wiley & Sons, 2005.



A. Dutfoy and R. Lebrun.

Practical approach to dependence modelling using copulas.

Proceedings of the Institution of Mechanical Engineers, Part O : Journal of Risk and Reliability, 223(4) :347–361, 2009.



A. Dutfoy and R. Lebrun.

An innovating analysis of the nataf transformation from the copula viewpoint.

Probabilistic Engineering Mechanics, 24(3) :312–320, 2009.



A. Dutfoy and R. Lebrun.

A generalization of the nataf transformation to distributions with elliptical copula.
Probabilistic Engineering Mechanics, 24 :172–178, 2009.



A. M. Hasofer and N. C. Lind.

An exact and invariant first order reliability format.
Journal of Engineering Mechanics, 100 :111–121, 1974.



M. Rosenblatt.

Remarks on a multivariate transformation.
The Annals of Mathematical Statistics, 23(3) :470–472, 1952.

- **Abate J., Whitt W.**, *The Fourier-series method for inverting transforms of probability distributions*, Queueing Systems, 10 :1-2, 5–87 (1992).
- **Feller W.** *An Introduction to Probability Theory and Its Applications Volume II*, Second edition, Wiley, 1971.
- **Lebrun R.** *Efficient time/space algorithm to compute rectangular probabilities of multinomial, multivariate hypergeometric and multivariate Pólya distributions*, Statistics and Computing, online June 2012.
- **Ma C.**, *Representations, bounds and approximations for tail probabilities of multivariate non-central hypergeometric and negative hypergeometric distributions*, Journal of Statistical Computation and Simulation, 62 :3, 237–258, 1999.