Optimisation sous incertitudes

G. Perrin

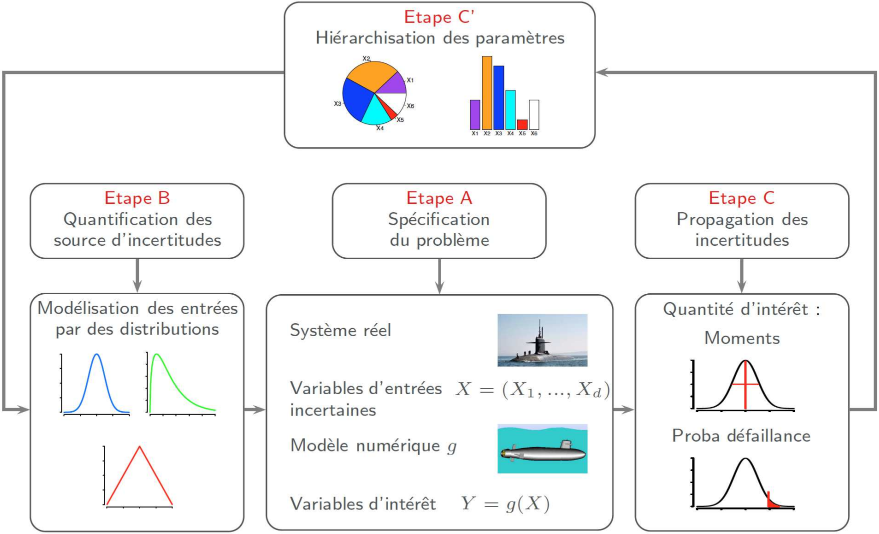
guillaume.perrin@univ-eiffel.org

Année scolaire 2022-2023

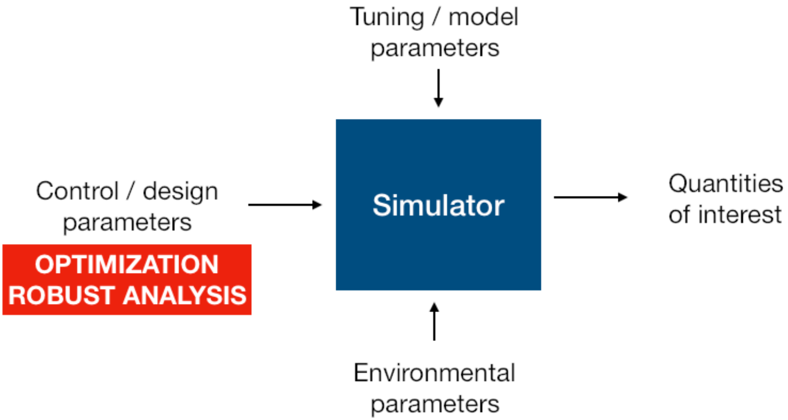
     

Plan de la séance

La conception robuste au sein de la démarche incertitudes



La conception robuste au sein des problèmes d'UQ



Plan de la séance

1. Conception robuste
2. Différences entre simulateurs "rapides" et simulateurs "chers"
3. Robustesse et fiabilité
4. Conclusions

Définition 1

Conception (numérique)

Action d'élaborer quelque chose, de le concevoir, en privilégiant la simulation par rapport à la mesure (parce qu'elle est coûteuse ou impossible).

Exemples :

- satellites, robots exploratoires (Sojourner, Curiosity...),
- centrales nucléaires (scénario d'accident nucléaire et anticipation des conséquences...),
- voitures (résistance au crash...),
- ...

Conception et optimisation

Si l'on définit :

- $\boldsymbol{x} \leftrightarrow$ paramètres de conception (géométrie, matériaux, ...),
- $\boldsymbol{y}(\boldsymbol{x}) \leftrightarrow$ critères de conception (coût, performance...),

concevoir un objet par la simulation revient à utiliser un code prédictif reliant $\boldsymbol{x}$ et $\boldsymbol{y}$ pour résoudre le problème d'optimisation suivant :

$$\boldsymbol{x}^\star = \arg\min_{\boldsymbol{x} \in \mathbb{X}} \mathcal{C}(\boldsymbol{y}(\boldsymbol{x})),$$

où $\mathcal{C}$ est une fonction coût à minimiser (agrégeant potentiellement plusieurs critères) et $\mathbb{X} \subset \mathbb{R}^d$ un domaine de recherche (intégrant potentiellement des contraintes).

Définition 2

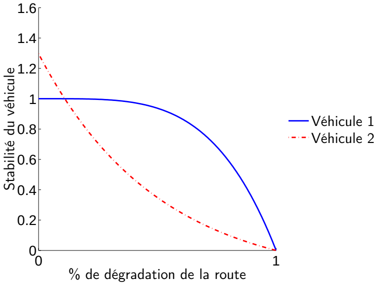
Robustesse (numérique)

Un système est dit **robuste** s'il est capable de bien se comporter dans des conditions **non nominales**.

Quel est le véhicule **le plus robuste** ?

Robustesse vs Performance

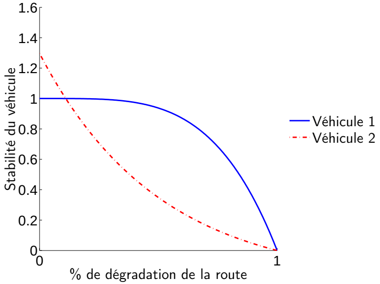
Robustesse numérique



Quel est le véhicule **le plus robuste** ?

Robustesse vs Performance

Recherche-t-on toujours la robustesse ?



Quel est le "meilleur" véhicule ? (Formule 1 / partir en vacances / pays différents...)

Conception et robustesse

L'intégration de la robustesse passe le plus souvent par la **prise en compte des incertitudes** dans le processus d'optimisation. Ces incertitudes peuvent porter sur :

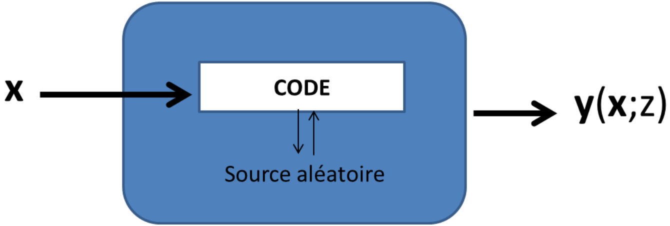
- les entrées de simulation (tolérances constructeur, méconnaissance des conditions limites/initiales, aléa météo...),
- les sorties de simulation (erreurs de modèle, incertitudes paramétriques, erreurs numériques,...).

$$\boldsymbol{g}(\boldsymbol{x}+\boldsymbol{\delta};\boldsymbol{\beta})+\boldsymbol{\varepsilon}(\boldsymbol{x})=\boldsymbol{y}(\boldsymbol{x};z), \quad z=(\boldsymbol{\beta},\boldsymbol{\delta},\boldsymbol{\varepsilon}),$$

avec $\boldsymbol{g}$ le code numérique, et $\boldsymbol{\beta}$ (paramètres du code), $\boldsymbol{\delta}$ (incertitudes sur les entrées), $\boldsymbol{\varepsilon}$ (incertitudes sur les sorties) des quantités potentiellement aléatoires.

Deux configurations classiques

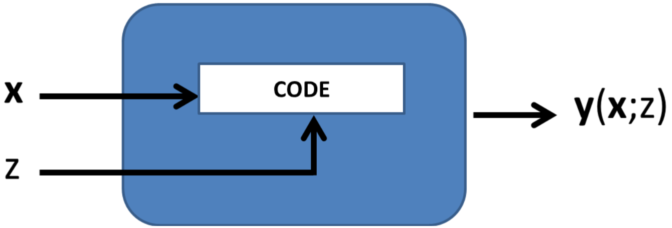
Cas 1 : le **simulateur stochastique**.



- La source aléatoire n'est pas accessible directement.
- Seul $\boldsymbol{x}$ peut être choisi : $\boldsymbol{y}(\boldsymbol{x}; z)$ est observé sans contrôle de z.
- La loi de z n'est pas connue explicitement.

Deux configurations classiques

Cas 2 : la présence de **variables d'environnement**.

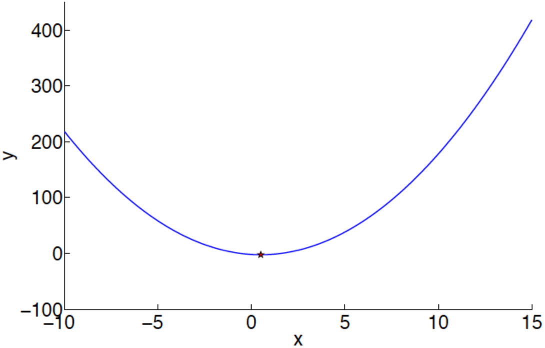


- Le simulateur reste déterministe.
- La loi de z est connue.
- Il est possible de spécifier à la fois les valeurs de $\boldsymbol{x}$ et z.

Optimisation avec et sans incertitudes

Exemple : $y(x;\boldsymbol{z}) = z_1 x^2 + z_2 x + z_3$, $z_1 \sim \mathcal{N}(2,1)$, $z_2 \sim \mathcal{N}(-2,1)$, $z_3 \sim \mathcal{N}(-2,1)$, z_1, z_2, z_3 indépendants. ("$\min_{x,\boldsymbol{z}} y(x;\boldsymbol{z})$" $= -\infty$)

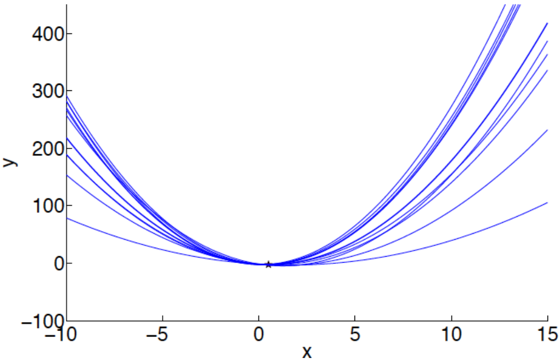
$$\widehat{x}^\star = \arg\min_{x\in\mathbb{R}} y(x;\boldsymbol{z}=\boldsymbol{z}(\theta)).$$



Optimisation avec et sans incertitudes

Exemple : $y(x;\boldsymbol{z}) = z_1 x^2 + z_2 x + z_3$, $z_1 \sim \mathcal{N}(2,1)$, $z_2 \sim \mathcal{N}(-2,1)$, $z_3 \sim \mathcal{N}(-2,1)$, z_1, z_2, z_3 indépendants. ("$\min_{x,\boldsymbol{z}} y(x;\boldsymbol{z})$" $= -\infty$)

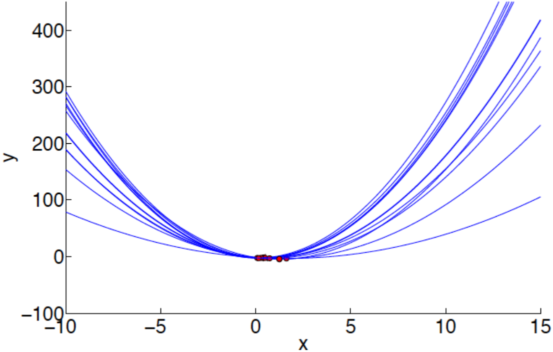
$$\widehat{x}^{\star} = \arg\min_{x\in\mathbb{R}} y(x;\boldsymbol{z}=\boldsymbol{z}(\theta)).$$



Optimisation avec et sans incertitudes

Exemple : $y(x;\boldsymbol{z}) = z_1 x^2 + z_2 x + z_3$, $z_1 \sim \mathcal{N}(2,1)$, $z_2 \sim \mathcal{N}(-2,1)$, $z_3 \sim \mathcal{N}(-2,1)$, z_1, z_2, z_3 indépendants. ("$\min_{x,\boldsymbol{z}} y(x;\boldsymbol{z})$" $= -\infty$)

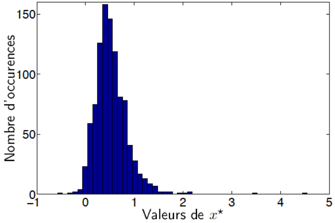
$$\widehat{x}^\star = \arg\min_{x\in\mathbb{R}} y(x;\boldsymbol{z}=\boldsymbol{z}(\theta)).$$



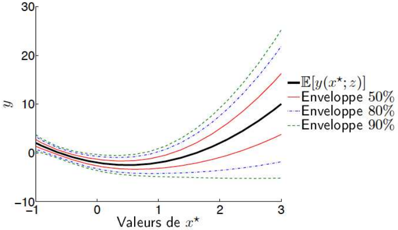
Optimisation avec et sans incertitudes

Exemple : $y(x;\boldsymbol{z}) = z_1 x^2 + z_2 x + z_3$, $z_1 \sim \mathcal{N}(2,1)$, $z_2 \sim \mathcal{N}(-2,1)$, $z_3 \sim \mathcal{N}(-2,1)$, z_1, z_2, z_3 indépendants. ("$\min_{x,\boldsymbol{z}} y(x;\boldsymbol{z})$" $= -\infty$)

$$\widehat{x}^\star = \arg\min_{x\in\mathbb{R}} y(x;\boldsymbol{z}=\boldsymbol{z}(\theta)).$$

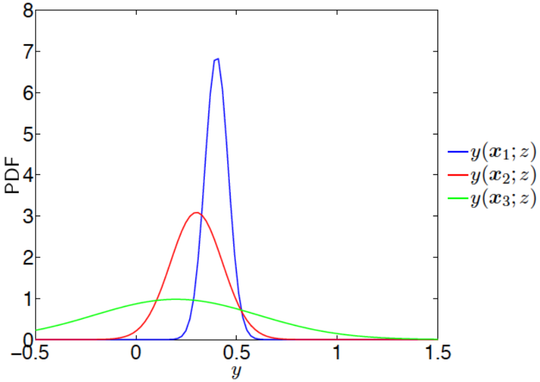


(a) Plusieurs minima possibles en fonction de $\boldsymbol{z}(\theta)$



(b) Une sortie dispersée au niveau de chaque valeur de $x^\star$

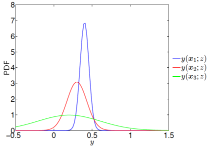
Définir une relation d'ordre



⇒ Comment choisir parmi $\boldsymbol{x}_1$, $\boldsymbol{x}_2$ et $\boldsymbol{x}_3$?

⇒ Quel sens donner à "$\boldsymbol{x}$ est meilleur que $\boldsymbol{x}'$" ?

Définir une relation d'ordre



$$\boldsymbol{x}^\star = \arg\min_{\boldsymbol{x}\in\mathbb{X}} \mathcal{C}(\boldsymbol{y}(\boldsymbol{x};z)),$$

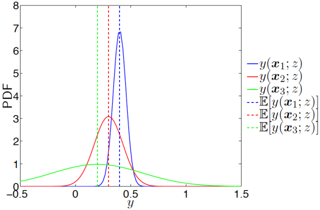
avec z une quantité aléatoire.

Quelle fonction $\mathcal{C}$ pour exprimer "$\boldsymbol{x}$ est meilleur que $\boldsymbol{x}'$" ?

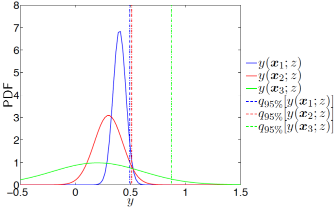
- Approche **"cas pire"** : $\max_z[\boldsymbol{y}(\boldsymbol{x};z)] \leq \max_z[\boldsymbol{y}(\boldsymbol{x}';z)]$.
- Moyenne : $\mathbb{E}[\boldsymbol{y}(\boldsymbol{x};z)] \leq \mathbb{E}[\boldsymbol{y}(\boldsymbol{x}';z)]$.
- Quantile : $q_\alpha[\boldsymbol{y}(\boldsymbol{x};z)] \leq q_\alpha[\boldsymbol{y}(\boldsymbol{x}';z)]$, $\mathbb{P}(\boldsymbol{y}(\boldsymbol{x};z) \leq q_\alpha[\boldsymbol{y}(\boldsymbol{x};z)]) = \alpha$.
- Conditional value-at-risk :
  $\mathbb{E}[\boldsymbol{y}(\boldsymbol{x};z) \mid \boldsymbol{y}(\boldsymbol{x};z) \geq q_\alpha[\boldsymbol{y}(\boldsymbol{x};z)]] \leq \mathbb{E}[\boldsymbol{y}(\boldsymbol{x}';z) \mid \boldsymbol{y}(\boldsymbol{x}';z) \geq q_\alpha[\boldsymbol{y}(\boldsymbol{x}';z)]]$.
- Approche multicritère $(\mathbb{E}[\boldsymbol{y}(\boldsymbol{x};z)], \mathrm{Var}[\boldsymbol{y}(\boldsymbol{x};z)])$.
- ... (108 métriques différentes recensées dans Göhler et al., Journal of Mechanical Design, 2016.)

Définir une relation d'ordre

Le choix de la relation d'odre est primordial !



(c) Moyenne : $x_3 > x_2 > x_1$

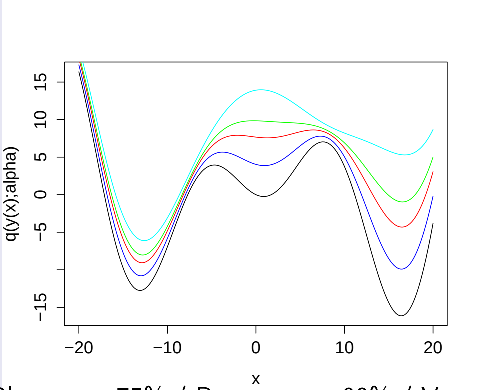


(d) Quantile 95% : $x_1 > x_2 > x_3$

L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 1 : $y(\boldsymbol{x}; z) = y(x) + z(x)$, $\mathcal{C} \leftrightarrow q_\alpha$, $z \leftrightarrow$erreur de modèle

$$x^\star = \arg \min_{x \in [-20,20]} q_\alpha \left[y(x; z) \right],$$

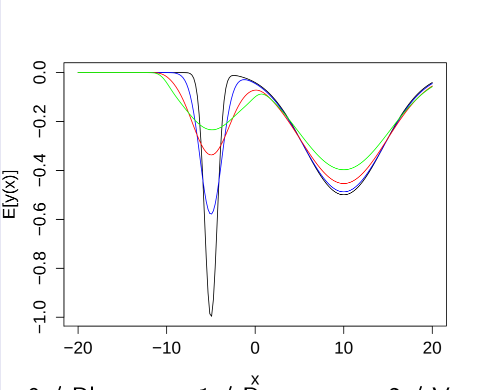


Noir : $\alpha = 50\%$ / Bleu : $\alpha = 75\%$ / Rouge : $\alpha = 90\%$ / Vert : $\alpha = 95\%$ / Cyan : $\alpha = 99\%$.

L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 2 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow \mathbb{E}$, $z \leftrightarrow$tolérance constructeur

$$x^\star = \arg \min_{x \in [-20,20]} \mathbb{E}\left[y(x+z)\right], \quad z \sim \mathcal{N}(0, \sigma^2)$$

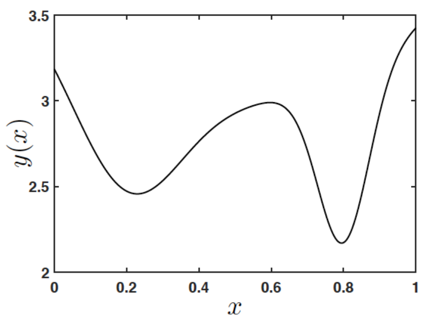


Noir : $\sigma = 0$ / Bleu : $\sigma = 1$ / Rouge : $\sigma = 2$ / Vert : $\sigma = 3$.

L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 3 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow$???, $z \leftrightarrow$tolérance constructeur

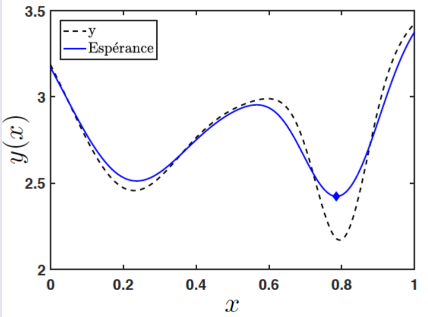
$$x^\star = \arg \min_{x \in [-20,20]} \mathcal{C}\left[y(x+z)\right], \quad z \sim \mathcal{U}(-0.1, 0.1)$$



L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 3 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow$???, $z \leftrightarrow$tolérance constructeur

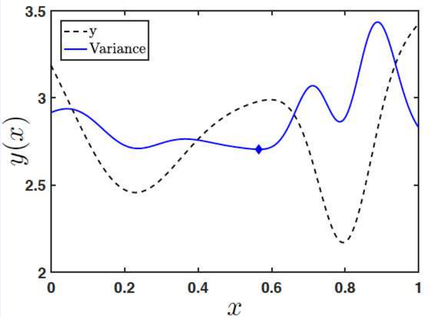
$$x^\star = \arg \min_{x \in [-20,20]} \mathcal{C}\left[y(x+z)\right], \quad z \sim \mathcal{U}(-0.1, 0.1)$$



L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 3 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow$???, $z \leftrightarrow$tolérance constructeur

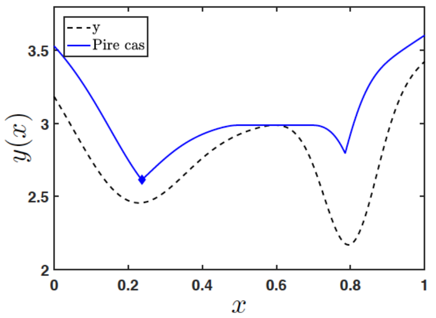
$$x^\star = \arg \min_{x \in [-20,20]} \mathcal{C}\left[y(x+z)\right], \quad z \sim \mathcal{U}(-0.1, 0.1)$$



L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 3 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow$???, $z \leftrightarrow$tolérance constructeur

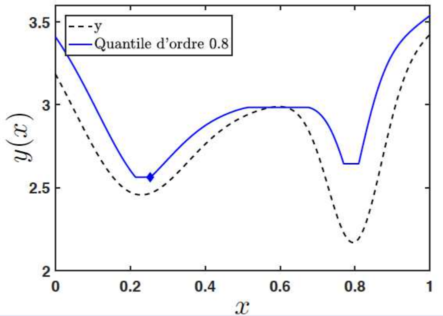
$$x^\star = \arg \min_{x \in [-20,20]} \mathcal{C}\left[y(x+z)\right], \quad z \sim \mathcal{U}(-0.1, 0.1)$$



L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 3 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow$???, $z \leftrightarrow$tolérance constructeur

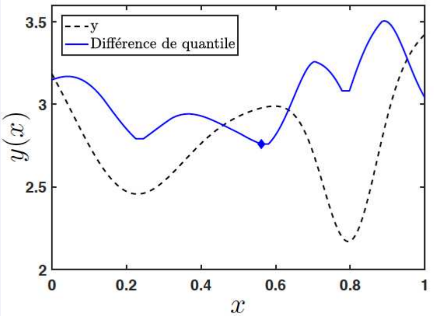
$$x^\star = \arg \min_{x \in [-20,20]} \mathcal{C}\left[y(x+z)\right], \quad z \sim \mathcal{U}(-0.1, 0.1)$$



L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 3 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow$???, $z \leftrightarrow$tolérance constructeur

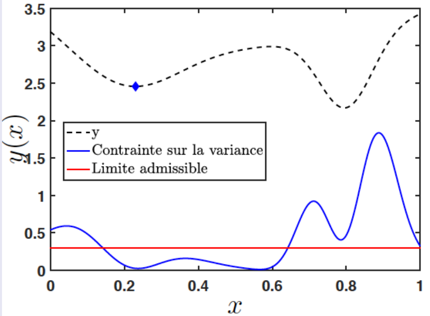
$$x^\star = \arg \min_{x \in [-20,20]} \mathcal{C}\left[y(x+z)\right], \quad z \sim \mathcal{U}(-0.1, 0.1)$$



L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 3 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow$???, $z \leftrightarrow$tolérance constructeur

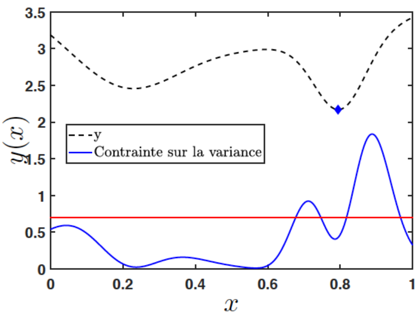
$$x^\star = \arg \min_{x \in [-20,20]} \mathcal{C}\left[y(x+z)\right], \quad z \sim \mathcal{U}(-0.1, 0.1)$$



L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 3 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow$???, $z \leftrightarrow$tolérance constructeur

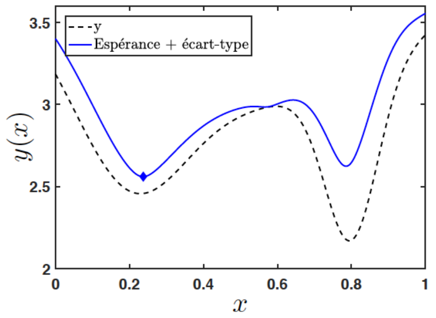
$$x^\star = \arg \min_{x \in [-20,20]} \mathcal{C}\left[y(x+z)\right], \quad z \sim \mathcal{U}(-0.1, 0.1)$$



L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 3 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow$???, $z \leftrightarrow$tolérance constructeur

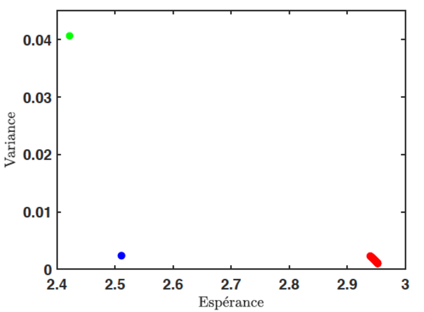
$$x^\star = \arg \min_{x \in [-20,20]} \mathcal{C}\left[y(x+z)\right], \quad z \sim \mathcal{U}(-0.1, 0.1)$$

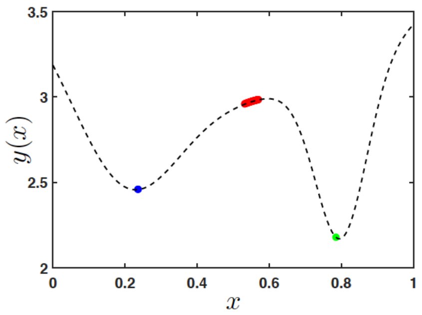


L'optimum peut varier en fonction du niveau d'incertitudes

Exemple 3 : $y(\boldsymbol{x}; z) = y(x+z)$, $\mathcal{C} \leftrightarrow$???, $z \leftrightarrow$tolérance constructeur

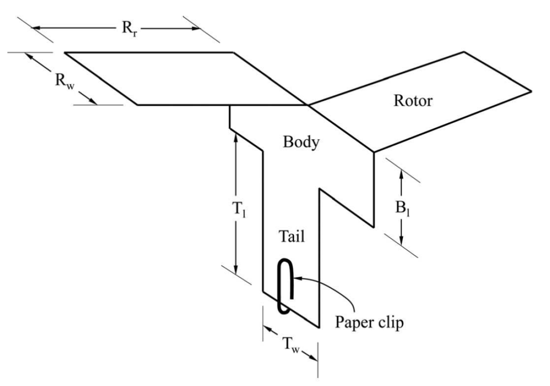
$$x^\star = \arg \min_{x \in [-20,20]} \mathcal{C}\left[y(x+z)\right], \quad z \sim \mathcal{U}(-0.1, 0.1)$$





Un dernier exemple

Optimisation du temps de chute d'un hélicoptère en papier



Paramètres :

- Dimensions R_r, R_w, T_l, B_l, T_w.
- Masses de l'hélicoptère et du trombone.

Partant d'une feuille A4 standard, comment maximiser le temps de chute d'un hélicoptère en papier, auquel est accroché un trombone ?

Plan de la séance

Schéma de résolution classique

$$x^\star = \arg\min_{x \in \mathbb{X}} \mathcal{C}(y(x;z)),$$

- z regroupe les sources d'incertitudes,
- $\mathcal{C}$ est une relation d'ordre choisie.

L'approche classique repose sur une "double boucle"

$C^\star = +\infty, \ n = 1$

While $n \leq N$, (**boucle 1**)

$\rightarrow$ "choisir" $x \in \mathbb{X}$,

$\rightarrow$ "évaluer" $\mathcal{C}(y(x;z))$ par des calculs répétés (**boucle 2**)

If $\mathcal{C}(y(x;z)) \leq C^\star, \ x^\star = x, \ C^\star = \mathcal{C}(y(x;z))$.

end **If**

end **While**

Simulateurs rapides à évaluer

Dans le cas de simulateurs "rapides" à évaluer, l'incertitude liée à l'estimation du critère $\mathcal{C}(\boldsymbol{y}(\boldsymbol{x}; z))$ peut être **négligée** (en effectuant suffisamment de répétitions).

$\Rightarrow$ on se ramène à une optimisation déterministe classique...

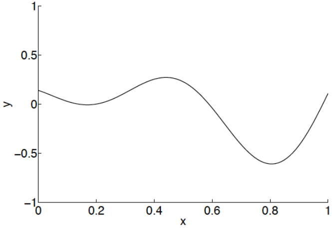
Principales difficultés

- $\boldsymbol{x} \mapsto \mathcal{C}(\boldsymbol{y}(\boldsymbol{x}; z))$ est souvent non-linéaire et non convexe $\Rightarrow$ utilisation d'heuristiques pour la résolution, sans convergence garantie...
- L'ensemble de définition $\mathbb{X}$ peut présenter des contraintes d'égalité ou d'inégalité...

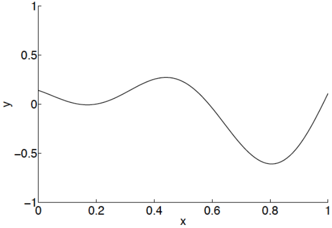
Simulateurs "chers"

Dans le cas de simulateurs "chers" à évaluer, il est **hors de question** d'effectuer un grand nombre de répétitions aux mêmes valeurs de $\boldsymbol{x}$.

- $\Rightarrow$ l'incertitude liée à l'estimation du critère $\mathcal{C}(\boldsymbol{y}(\boldsymbol{x}; z))$ doit maintenant être intégrée dans la procédure d'optimisation :
- $\Rightarrow$ avant chaque nouveau calcul, se pose la question : **"dupliquer ou explorer"** ?



(e) Duplication



(f) Exploration

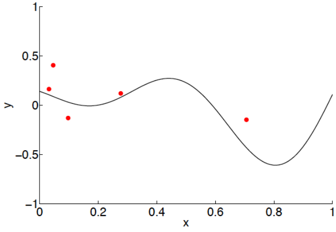
FIGURE: Fonction $\boldsymbol{x} \mapsto \boldsymbol{y}(\boldsymbol{x}; z = 0)$ à minimiser.

Simulateurs "chers"

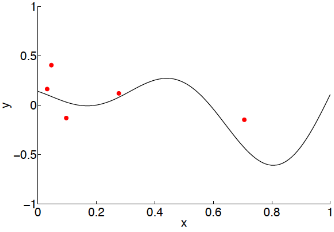
Dans le cas de simulateurs "chers" à évaluer, il est **hors de question** d'effectuer un grand nombre de répétitions aux mêmes valeurs de $\boldsymbol{x}$.

$\Rightarrow$ l'incertitude liée à l'estimation du critère $\mathcal{C}(\boldsymbol{y}(\boldsymbol{x}; z))$ doit maintenant être intégrée dans la procédure d'optimisation :

$\Rightarrow$ avant chaque nouveau calcul, se pose la question : **"dupliquer ou explorer"** ?



(a) Duplication



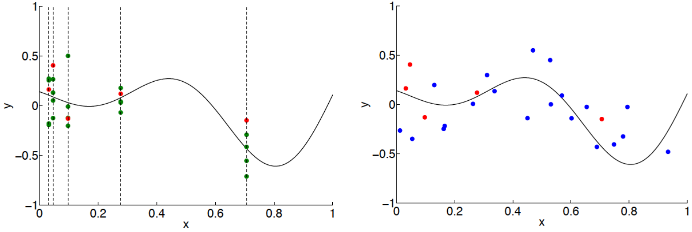
(b) Exploration

FIGURE: Information après 5 évaluations bruitées.

Simulateurs "chers"

Dans le cas de simulateurs "chers" à évaluer, il est **hors de question** d'effectuer un grand nombre de répétitions aux mêmes valeurs de $\boldsymbol{x}$.

- $\Rightarrow$ l'incertitude liée à l'estimation du critère $\mathcal{C}(\boldsymbol{y}(\boldsymbol{x}; z))$ doit maintenant être intégrée dans la procédure d'optimisation :
- $\Rightarrow$ avant chaque nouveau calcul, se pose la question : **"dupliquer ou explorer"** ?



(a) Duplication (b) Exploration

FIGURE: Nouvelles évaluations pour la minimisation.

Simulateurs "chers"

Pourquoi dupliquer en optimisation ?

- **En théorie**, une mauvaise idée : le gain en information est nécessairement égal ou inférieur au calcul effectué au même point.
- **En pratique**, en facilitant la gestion du "bruit", dupliquer augmente les chances de rechercher de nouveaux points dans les bonnes directions.

$\Rightarrow$ un compromis relativement efficace consiste à augmenter graduellement le nombre de duplications lors de l'optimisation.

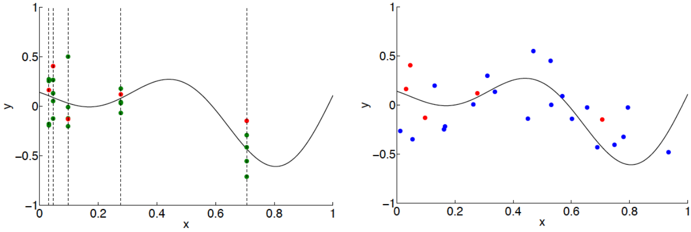
Simulateurs "chers"

Plus généralement, l'optimisation de simulateurs "chers" repose sur l'utilisation de **méthodes d'apprentissage** (statistiques ou non), permettant de **maximiser l'exploitation** des calculs disponibles, et d'**estimer la valeur du simulateur** en des points non calculés pour guider la recherche de minimum.

- Dans le cas des **variables d'environnement** (l'aléa est contrôlable), on peut chercher à approcher la fonction (déterministe) $(\boldsymbol{x}, z) \mapsto \boldsymbol{y}(\boldsymbol{x}; z)$, puis déduire la valeur de $\mathcal{C}$ en tout $\boldsymbol{x}$, en effectuant les post-traitements en z à partir de l'approximation obtenue.
- Dans le cas des **simulateurs stochastiques** (l'aléa n'est pas contrôlable), on distingue deux grands courants :
  - $\rightarrow$ les méthodes basées sur des duplications : l'apprentissage se fait alors sur la valeur approchée de $\mathcal{C}$.
  - $\rightarrow$ les méthodes sans duplications : dans ce cas, on apprend directement la loi (statistique) de $y(\boldsymbol{x}; z)$ en tout $\boldsymbol{x}$, que l'on post-traite ensuite pour orienter les recherches.

Simulateurs "chers"

Retour à l'exemple précédent.

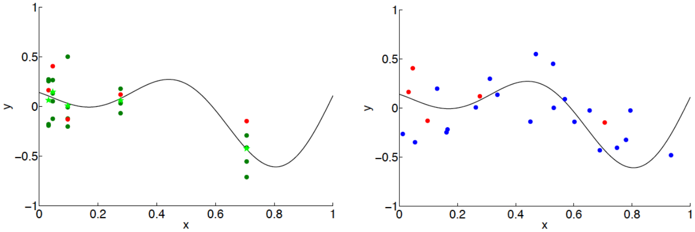


(a) Duplication (b) Exploration

FIGURE: Information disponible initiale.

Simulateurs "chers"

Retour à l'exemple précédent.

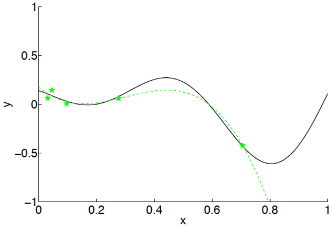


(a) Duplication (b) Exploration

FIGURE: Condensation de l'information dans le cas dupliqué en prenant la moyenne des observations (par exemple).

Simulateurs "chers"

Retour à l'exemple précédent.



(a) Duplication (b) Exploration

FIGURE: Prédiction polynômiale pour l'orientation des nouveaux calculs.

Plan de la séance

Robustesse et fiabilité

- Jusqu'à présent, on a appelé "optimisation robuste" la résolution du problème :

$$\boldsymbol{x}^\star = \arg\min_{\boldsymbol{x} \in \mathbb{X}} \mathcal{C}(\boldsymbol{y}(\boldsymbol{x}; z)),$$

  avec $\mathcal{C}$ une fonction coût à minimiser ($\leftrightarrow$ **relation d'ordre**) et $\mathbb{X} \subset \mathbb{R}^d$ un domaine de recherche pouvant intégrer des contraintes.

- En fonction de la nature des contraintes, qui peuvent être **déterministes** ou **probabilistes**, de la présence ou non d'incertitudes sur les sorties de simulation, ce type de problème peut être intégré dans une décomposition plus générale des problèmes d'optimisation sous incertitudes.

Robustesse et fiabilité

En particulier, on distingue souvent fiabilité et robustesse, au sens où l'on parle plutôt de :

- → **fiabilité** lorsque les incertitudes portent sur les contraintes,
- → **robustesse** lorsque les incertitudes portent sur les fonctions objectifs.

ROBUSTESSE

FIABILITE

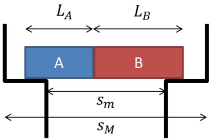
	Pas de fonction objectif	Fonction objectif déterministe	Fonction objectif probabiliste
Pas de contraintes		« design optimal »	« design robuste »
Contraintes déterministes	« domaine admissible »	« design optimal et admissible»	« design robuste et admissible»
Contraintes probabilistes	« domaine fiable »	« design optimal et fiable»	« design robuste et fiable»

Robustesse et fiabilité

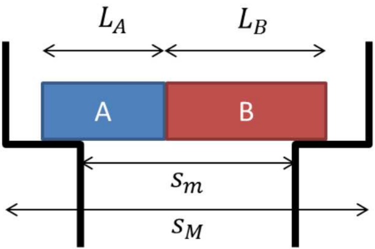
Un exemple "classique" de fiabilité

- On s'intéresse au dimensionnement d'une poutre constituée d'un assemblage de deux éléments, A et B (voir dessin).
- De manière artificielle, on suppose que $L_A \sim \mathcal{N}(4, x^2)$, $L_B \sim \mathcal{N}(6, x^2)$ ($x \leftrightarrow$ tolérances de fabrication).
- En partant du principe que "plus on contraint les tolérances de fabrication, plus la construction coûte chère", on cherche :

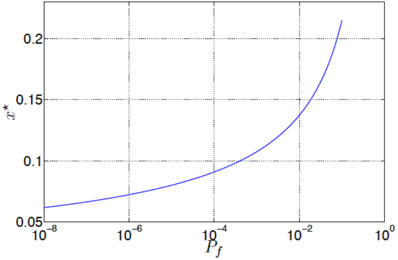
$$x^\star = \arg \min_{\mathbb{P}(L_A+L_B \notin [s_m, s_M]) \leq P_f} \frac{1}{x^2}, \quad \frac{1}{x^2} \leftrightarrow \text{"coût de fabrication".}$$



Robustesse et fiabilité



(a) Exemple de fiabilité



(b) Evolution de $x^\star$

FIGURE: Evolution de $x^\star$ en fonction de P_f, pour $s_m = 9.5$ et $s_M = 10.5$.

$$x^\star = \arg \max_{\mathbb{P}(L_A + L_B \notin [s_m, s_M]) \leq P_f} x^2 = -\frac{1}{2\sqrt{2}\Phi^{-1}(P_f/2)}.$$

Robustesse et fiabilité

Un petit exercice de vocabulaire pour finir

On cherche à résoudre $x^\star = \arg\min_{\mathbb{P}(x+z\geq 7)\geq 95\%} \mathbb{E}[y(x;z)]$, $z \sim \mathcal{N}(0,\sigma^2)$.

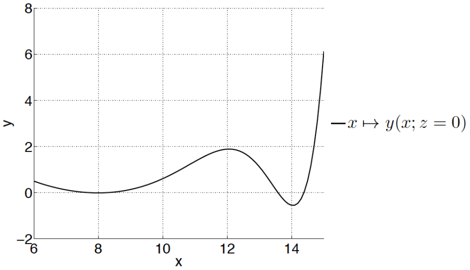


FIGURE: Si $\sigma = 0$, $\mathbb{E}[y(x;z)] = y(x;z=0)$.

Robustesse et fiabilité

Un petit exercice de vocabulaire pour finir

On cherche à résoudre $x^\star = \arg\min_{\mathbb{P}(x+z \geq 7) \geq 95\%} \mathbb{E}[y(x;z)]$, $z \sim \mathcal{N}(0, \sigma^2)$.

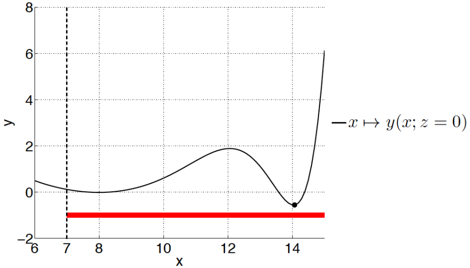


FIGURE: Que représente le trait rouge ? Le point noir ?

Robustesse et fiabilité

Un petit exercice de vocabulaire pour finir

On cherche à résoudre $x^{\star} = \arg\min_{\mathbb{P}(x+z\geq 7)\geq 95\%} \mathbb{E}[y(x;z)]$, $z \sim \mathcal{N}(0,\sigma^2)$.

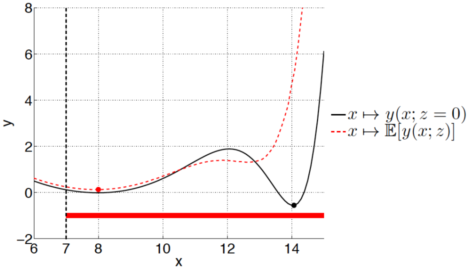


FIGURE: Que représente le point rouge ?

Robustesse et fiabilité

Un petit exercice de vocabulaire pour finir

On cherche à résoudre $x^\star = \arg\min_{\mathbb{P}(x+z\geq 7)\geq 95\%} \mathbb{E}[y(x;z)]$, $z \sim \mathcal{N}(0,\sigma^2)$.

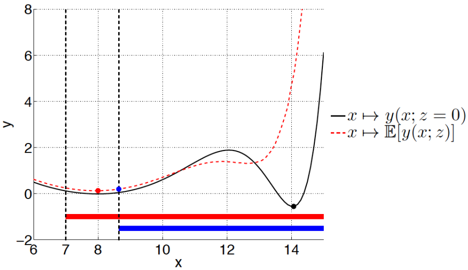


FIGURE: Que représente le trait bleu ? Le point bleu ?

Plan de la séance

Conclusions

- Intégrer les incertitudes dans le processus de conception modifie fortement le problème d'optimisation à résoudre.
- La relation d'ordre à minimiser doit être soigneusement adaptée au problème.
- En première approche, la résolution d'un problème d'optimisation robuste nécessite la duplication de calculs, et implique un coût numérique élevé.
- Un certain nombre de travaux actuels portent sur la mise en place de méthodes d'optimisation permettant de s'affranchir de ces répliques (en se basant notamment sur l'exploitation de métamodèles).

Quizz

- **L'optimisation est dite robuste car elle**
  ☐ est rapide ☐ est répétable ☐ donne de bons résultats ☐ intègre les incertitudes.
- **En introduisant les incertitudes dans mon problème d'optimisation, je**
  ☐ prends des risques ☐ contrôle mes marges ☐ me complique la vie ☐ dimensionne au juste besoin.
- **Une structure robuste est une structure optimisée.**
  ☐ oui ☐ non ☐ ça dépend de ce qu'on entend par "optimisée".
- **Afin d'adapter au mieux la méthode d'optimisation**
  ☐ je liste les contraintes du problème ☐ je recense les sources d'incertitudes ☐ je réfléchis à la relation d'ordre ☐ j'évalue le coût d'une simulation.