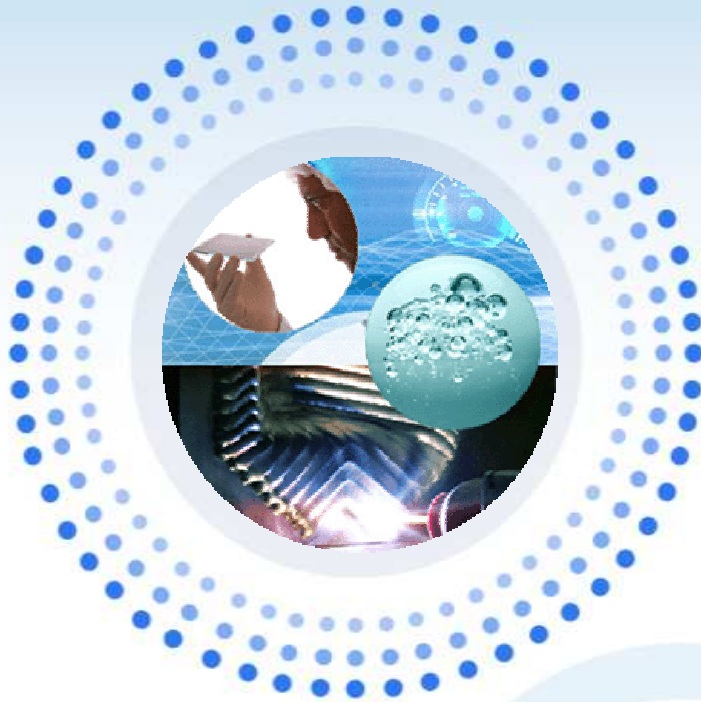


Analyses probabilistes avec Salome_Meca. Application au calcul de courbes de fragilité pour les études de sûreté sismique.



Irmela Zentner & Anne-Laure Popelin

EDF R&D AMA Analyses Mécaniques et Acoustique

EDF R&D MRI Management du Risque Industriel

Journée des Utilisateurs de Code_Aster

17 mars 2011, EDF R&D Clamart





Plan de l'exposé

- ◎ Introduction
 - Contexte général
 - Études probabilistes de sûreté (EPS) sismique
 - Analyses probabilistes avec Code_Aster

- ◎ Calcul de courbes de fragilité
 - Application à la maquette SMART

- ◎ Études probabilistes avec la plate-forme Salome_Meca
 - Code_Aster et Open TURNS dans Salome_Meca
 - Illustration sur l'étude SMART



Contexte et enjeux

- ◎ Analyses mécaniques probabilistes
 - Calculs mécaniques « best-estimate » pour évaluer les marges
 - Modèles mécaniques complexes
 - Chargement aléatoire
 - Prise en compte des incertitudes pour fiabiliser les résultats
 - Évaluer les conservatismes, déterminer des probabilités de défaillance
 - Maîtriser les risques
- ◎ Études probabilistes de sûreté (EPS) sismique
 - Quantifier le risque sismique des installations compte-tenu
 - du chargement aléatoire (séisme)
 - des incertitudes sur les modèles et les données



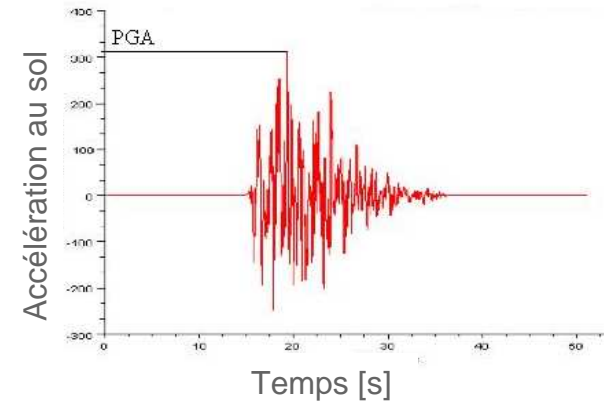
Études probabilistes de sûreté sismique

● Définition de courbes de fragilité

- Probabilité de défaillance P_f (d'une structure ou d'un composant) en fonction du niveau sismique α :

$$P_f(\alpha) = P(Y > Y_{seuil} / \alpha)$$

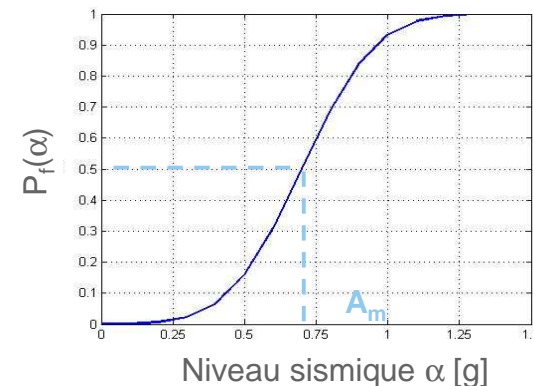
- α : paramètre caractérisant la nocivité du séisme, souvent le PGA (Peak Ground Acceleration)
- Y : variable d'intérêt (contrainte, déplacement)



- Expression analytique de la courbe de fragilité si modèle log-normal

$$P_f(\alpha) = \Phi\left(\frac{\ln(\alpha/A_m)}{\beta}\right)$$

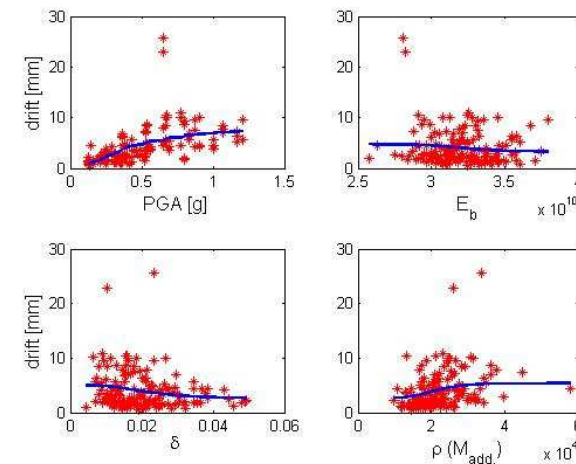
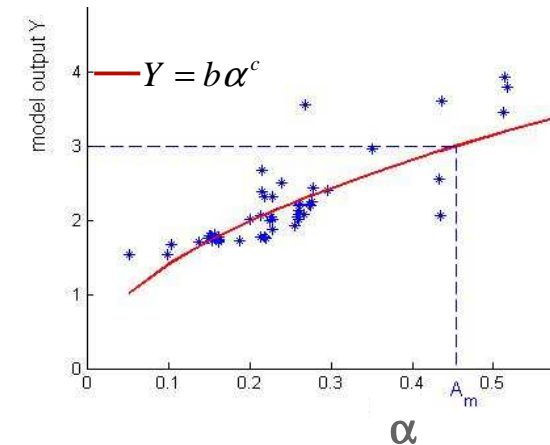
- Capacité médiane A_m de la structure
- Ecart-type logarithmique β (variabilité)





Études probabilistes de sûreté sismique

- Méthodes pour le calcul de courbes de fragilité
 - Modèle log-normal de courbe de fragilité (A_m, β)
 - Estimation des paramètres par maximum de vraisemblance
 - Régression linéaire
 - Construction d'un méta-modèle (surface de réponse)
 - Modèle HDMR (High Dimensional Model Representation)
 - Décomposition de la fonction en termes d'ordre croissant
 - Régression non paramétrique des termes de la décomposition





Études probabilistes avec Code_Aster

- ◎ Calculs paramétriques via ASTK (plan d'expériences)
 - Utile pour les plans d'expérience et les études de sensibilité paramétriques
 - Fichier de commande unique + jeu de paramètres
 - ✓ Distribution des calculs
 - ✓ Post-traitement statistique avec un logiciel dédié (comme OT)
 - ✓ Documentation [U2.08.07] « Distribution de calculs paramétriques »

- ◎ Couplage Code_Aster avec Open TURNS via la plateforme Salome_Meca
 - Open TURNS pour le traitement des incertitudes
 - ✓ Études probabilistes plus complètes
 - ✓ Plus d'ergonomie via l'interface



Application maquette SMART

◎ Benchmark SMART



SMART (**S**eismic design and best-estimate **M**ethods **A**ssessment for **R**einforced concrete buildings subjected to **T**orsion and non-linear effects)



◎ Essais et benchmark organisées par CEA et EDF

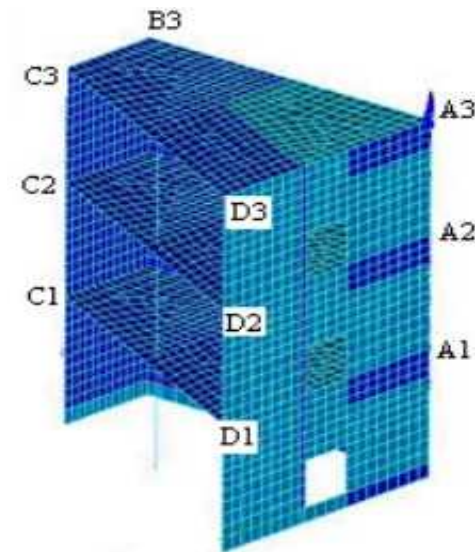
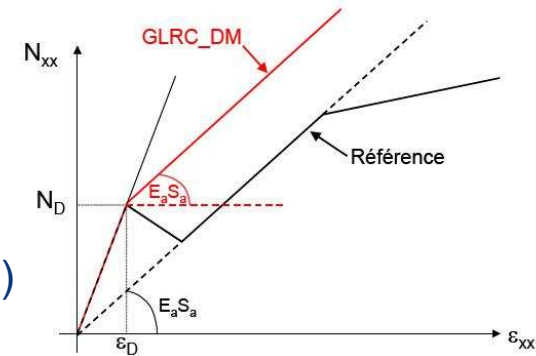
- Modélisation du comportement non linéaire en dynamique (phase 1)
 - Évaluation des méthodes d'ingénierie sismique
 - Prédiction aveugle et études « best-estimate »
- Études paramétriques et calcul de courbes de fragilité (phase 2)



Application maquette SMART

◎ Modèle EF coque et poutres

- Loi de comportement du béton armé
 - Modèle d'endommagement global GLRC_DM
- Calculs transitoires non linéaires (DYNA_NON-LINE)
 - Temps CPU pour un calcul : 4 –7 h
- Variables caractérisant l'endommagement
 - Drift Y (déplacement diff. entre deux étages)
 - Décalage des premières fréquences propres





Application maquette SMART

⊙ Aléa sismique et incertitudes

- 50 séismes artificiels (base de donnée)
- 3 paramètres modélisés par des variables aléatoires: M_{add} , E_b , δ

Parameter	distribution	mean	std
Add. Mass	Log-normal	Nom. value	5% of no. value
Concrete Young's modulus E_b	Log-normal	32000MPa	2200MPa
Damping δ	Log-normal	0.02	0.01

⊙ Propagation des incertitudes

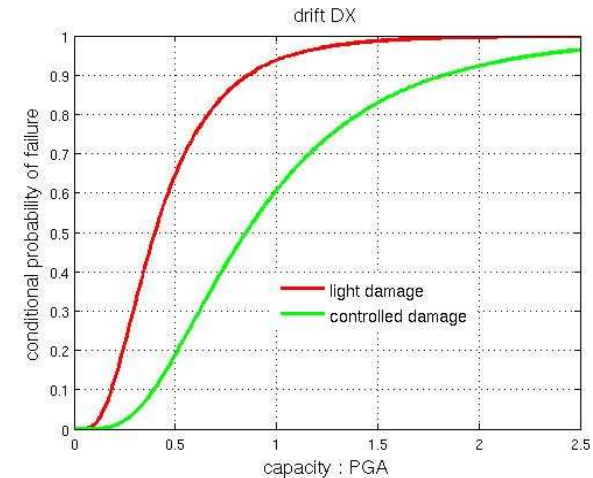
- Latin Hypercube Sampling (LHS)
- 4 réplifications de l'échantillon initial de taille 50 : $N_s=200$
- Temps de calcul total: ~120h sur 10 processeurs



Application maquette SMART

● Comparaison des résultats

- Régression simple - Maximum de vraisemblance – méta-modèle (HDMR)
- Indicateur du mouvement sismique α : PGA
- 2 niveaux de dommage pour le critère drift
 - Endommagement léger : « light »
 - Endommagement contrôlé : « controlled »



		“light”			“controlled”	
method	regression	max. likelihood	HDMR	regression	max. likelihood	HDMR
A_m	0.41	0.40	0.36	0.89	0.83	0.90
β	0.57	0.64	0.63	0.57	0.76	0.91

- Résultats en bon accord, variabilité (β) sous-estimée par la régression simple



Application maquette SMART

⊙ Analyse de sensibilité globale

- Décomposition de la variance : indices de Sobol'
- Résultats dérivés directement du méta-model :

S_{α}	$S_{\mathbf{B}}$	S_{δ}	S_{ρ}	S_U	$S_{\alpha U}$
0.307	0.013	0.035	0.047	0.482	0.115

• Les sources de variabilité les plus importantes sont

- Le niveau sismique α
- L'aléa sismique (variabilité non expliquée par les paramètres) U
- Interactions entre ces derniers



Études probabilistes avec la plate-forme Salomé-Méca

© Salomé-Méca 2011.1, diffusée depuis mars 2011, propose désormais :

- Open TURNS, outil pour les études probabilistes

- Développé par EDF, EADS et Phiméca depuis 2005
- Déclinaison logicielle de la méthodologie de traitement des incertitudes partagée par de nombreux industriels (partenaires, SNCF, Airbus, ...)
- Accès à de nombreuses méthodologies avancées pour les études probabilistes
 - Méthodes FORM, SORM, tirage d'importance, ...
 - Chaos polynomial
 - Modélisation de la dépendance par les copules
 - ...
- Intègre les résultats de travaux de recherche récents
- Diffusé en licence LGPL pour une transparence et un partage accru
- Plus d'informations : www.openturns.org

SALOME 5.1.5 - [Study1]

File Edit View OPENTURNS Tools Window Help

Object Browser

Name	Value
Aster	ASTER
new_case	-
Data	-
frag_smart_mod.comm	/local00/Demo_Salome_Meca/frag_sm...
DTM.med	/local00/Demo_Salome_Meca/DTM.med
interactiv-follow-up	True
has-base-result	True
Astk parameters	-
astk-action	make_etude
aster-version	STA9.8
name	new_case
debug	nodebug
mode	interactif
proc-nb	1
memory	128
time	120
login	popelin
server	localhost
build-script	oui
submit-script	oui
origin	salomemeca_asrun 1.8.1
Results	-
Post-Pro	VISU
OpenTURNS	OPENTURNS
variables	/local00/JUASTER/variables.comm
study	/local00/JUASTER/study.comm

Python Console

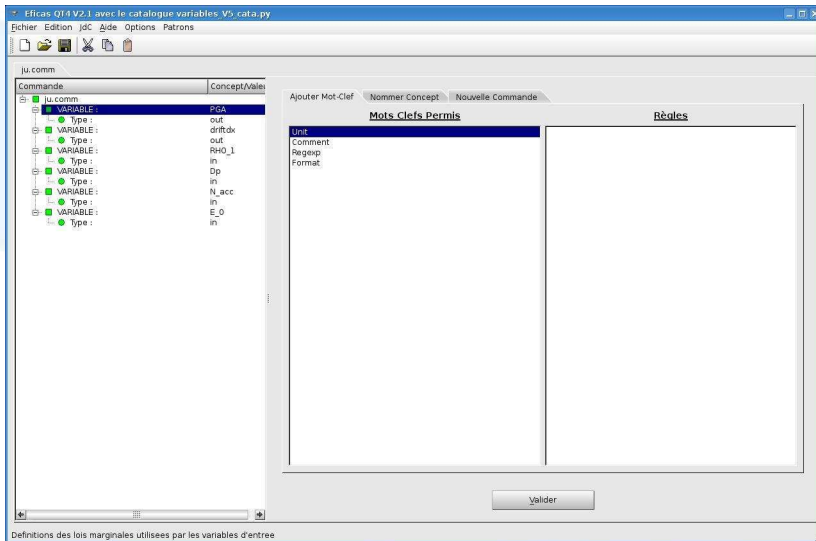
```
Python 2.4.4 (#1, Nov 13 2009, 17:32:46)
[GCC 4.1.2 20061115 (prerelease) (Debian 4.1.1-21)] on linux2
type help to get general information on environment
>>>
```

Message Window

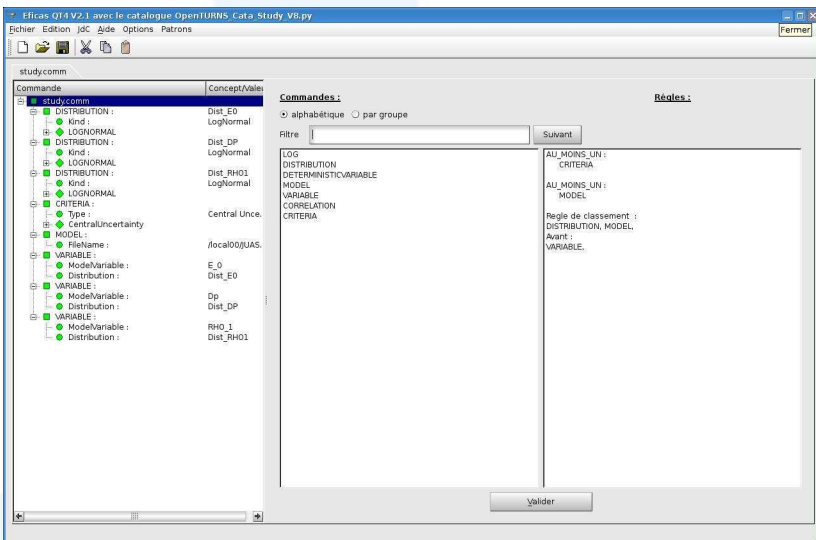
Message Log



Études probabilistes avec la plate-forme Salomé-Méca



```
variables.xml - KWrite
Eichier  Édition  Affichage  Signets  Outils  Configuration  Aide
<?xml version="1.0" encoding="UTF-8"?>
<wrapper>
  <library>
    <path></path>
    <description>
      <variable-list>
        <variable id="driftdx" type="out"/>
        <variable id="RHO_1" type="in"/>
        <variable id="Dp" type="in"/>
        <variable id="N_acc" type="in"/>
        <variable id="E_0" type="in"/>
      </variable-list>
      <function provided="no"></function>
      <gradient provided="no"></gradient>
      <hessian provided="no"></hessian>
    </description>
  </library>
  <external-code>
    <data>
      <framework>
```



```
study.py - KWrite
Eichier  Édition  Affichage  Signets  Outils  Configuration  Aide
<!-- study-id -->
<!-- study-case -->
<!-- component-name -->
#! /usr/bin/env python
# Chargement du module systeme
import sys
sys.path[:0]=['']
# Chargement du module math
import math
# Chargement du module Open TURNS
from openturns import *
results = {}
from openturns.viewer import ViewImage, StopViewer, WaitForViewer
# Definit le niveau d'affichage de la log
flags = Log.NONE
Log.Show( flags )
# Definit la methode d'evaluation (evaluation par YACS)
model = yacseval
n = model.getInputNumericalPointDimension()
# Definit la loi jointe des variables d'entree
collection = DistributionCollection( n )
description = Description( n )
# Definit la loi marginale de la composante 0
marginal_0 = LogNormal( 0.02, 0.01, 0, LogNormal.MUSIGMA )
marginal_0.setName( 'Dist_DP' )
description[ 0 ] = 'Dp'
collection[ 0 ] = Distribution( marginal_0 )
# Definit la loi marginale de la composante 1
marginal_1 = LogNormal( 3.2e+10, 2.2e+09, 0, LogNormal.MUSIGMA )
marginal_1.setName( 'Dist_EO' )
description[ 1 ] = 'E_0'
```





Études probabilistes avec la plate-forme Salomé-Méca

- Permet la distribution des calculs via YACS
 - Indispensable pour les études probabilistes qui nécessitent un nombre élevé de calculs
 - Plusieurs schémas possibles
 - Interface graphique pour la définition du schéma

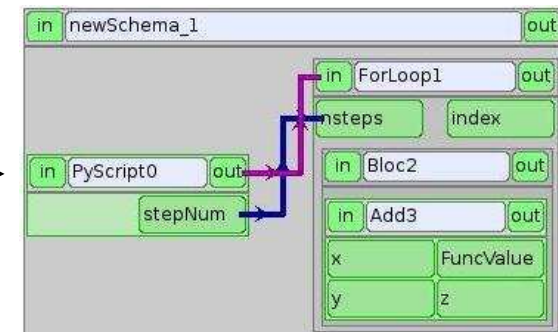
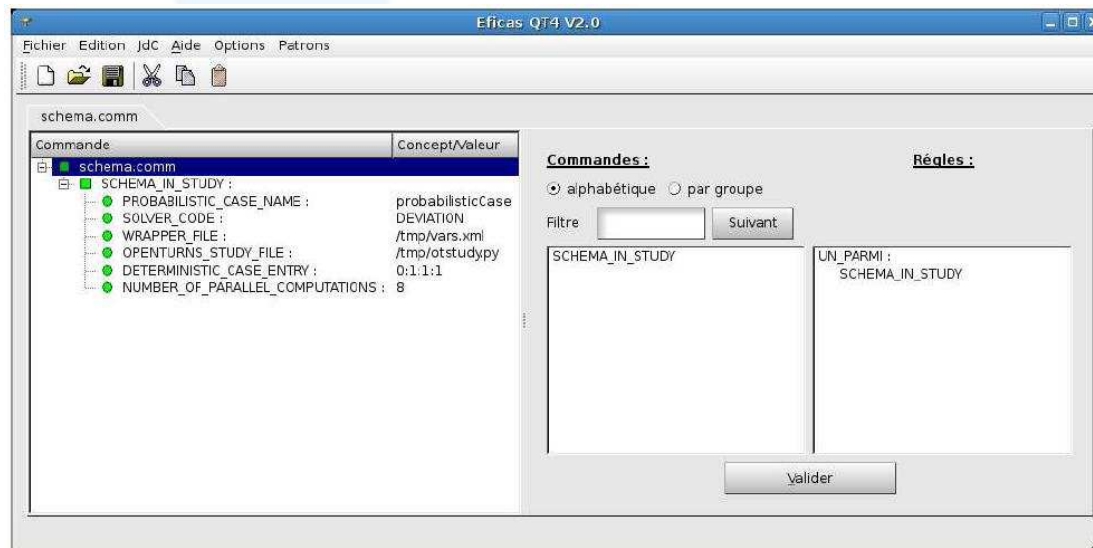




Illustration sur l'étude SMART

Communication Code_Aster – Open TURNS

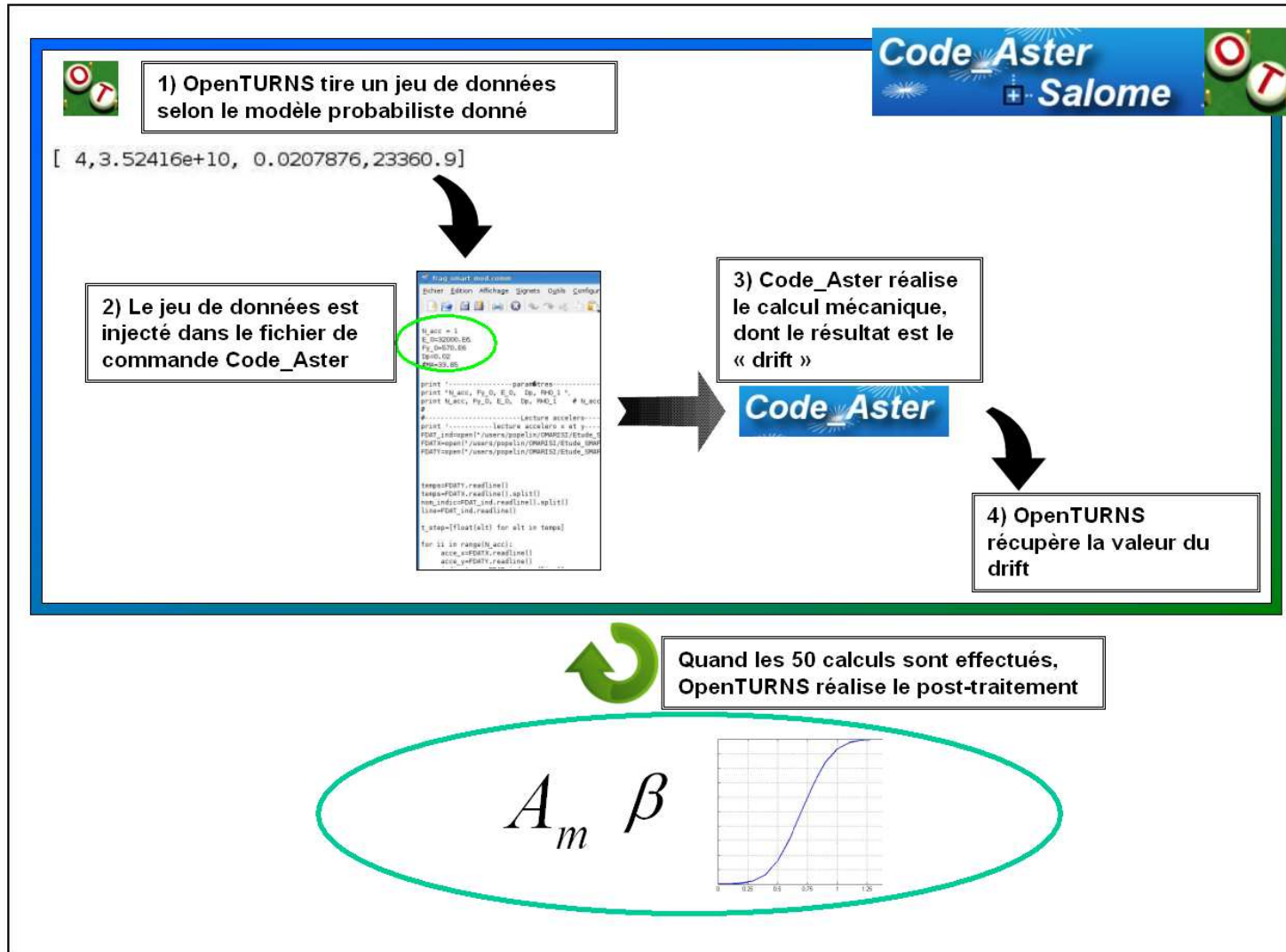




Illustration sur l'étude SMART

- ◎ Faisabilité des calculs de courbe de fragilité sous Salome_Meca
- ◎ Premier retour d'expérience sur l'utilisation de la plate-forme
- ◎ Intégration de ce retour d'expérience dans la version diffusée



Merci de votre attention