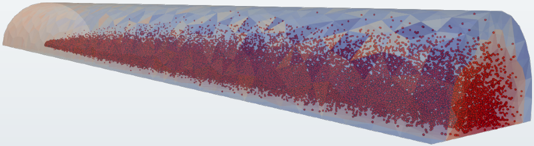
Inria - Sophia Antipolis Méditerranée



Uncertainty quantification in CFD simulations of multiphase flow

Aurore Dupré

14th January 2021 - **Calisto seminar**

Context of the post-doc

- ➙ Post-doc from 1^{st} February 2020 to 31^{st} January 2021
- ➙ European Project: VIrtual Materials Market Place (VIMMP)
  - Project started the 1^{st} January 2018 and ends the 31^{st} December 2021
  - VIMMP facilitates and promotes the exchange between all materials modelling stakeholders for the benefit of increased innovation in European manufacturing industry.

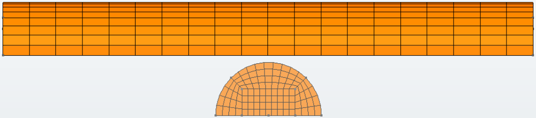
In this context, we define a CFD workflow to quantify uncertainty.

Context of the post-doc

- Post-doc from 1^{st} February 2020 to 31^{st} January 2021
- European Project: VIrtual Materials Market Place (VIMMP)
  - Project started the 1^{st} January 2018 and ends the 31^{st} December 2021
  - VIMMP facilitates and promotes the exchange between all materials modelling stakeholders for the benefit of increased innovation in European manufacturing industry.

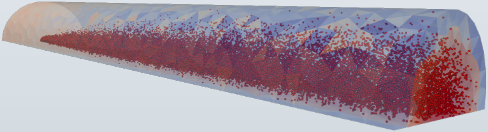
In this context, we define a CFD workflow to quantify uncertainty

Case



➙ Turbulent pipe flow

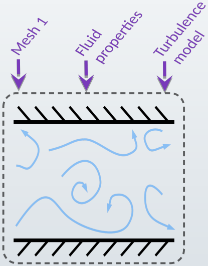
➙ Particle injection when statistical stationnary state is reached



The workflow

'Particles in pipe' workflow

Input
(user defined)

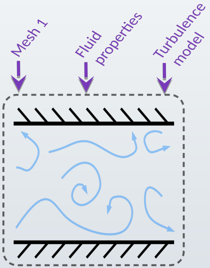


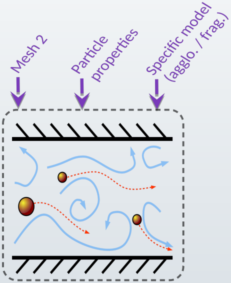
Tool | *Code_Saturne*

Code | CFD

'Particles in pipe' workflow

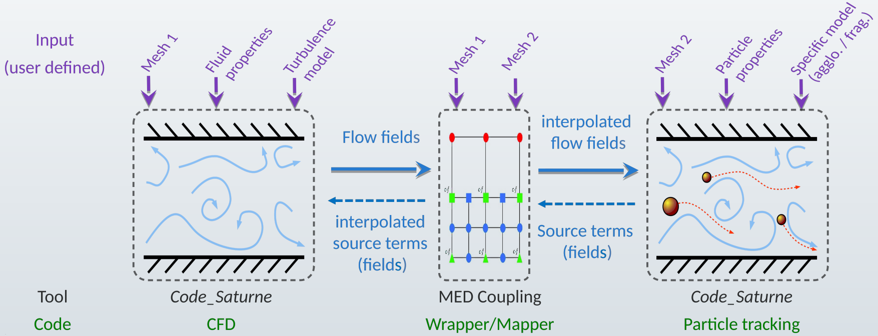
Input
(user defined)





Tool	*Code_Saturne*	*Code_Saturne*
Code	CFD	Particle tracking

'Particles in pipe' workflow



Fluid instance (1): Physical parameters

➞ Fluid properties:

- $\rho_f = 1.17862$ kg·m^{-3}
- $T = 293.15$ K
- $|U_{\text{volumetric}}| = 1$ m·s^{-1}

Fluid instance (2): Model parameters

➙ Turbulence model: $R_{ij} - \varepsilon$

$$\frac{\partial\langle u_f^i u_f^j\rangle}{\partial t} + \langle U_f^k\rangle \frac{\partial\langle u_f^i u_f^j\rangle}{\partial x_k} + \frac{\partial\langle u_f^i u_f^j u_f^k\rangle}{x_k}$$

$$= \quad -\left(\langle u_f^i u_f^k\rangle \frac{\partial\langle U_f^j\rangle}{\partial x_k} + \langle u_f^j u_f^k\rangle \frac{\partial\langle U_f^i\rangle}{\partial x_k}\right)$$

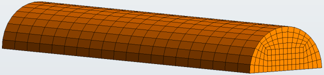
$$- \frac{1}{\rho_f}\left(\frac{\partial\langle \mathscr{P} u_f^i\rangle}{\partial x_j} + \frac{\partial\langle \mathscr{P} u_f^j\rangle}{\partial x_i}\right) + \frac{1}{\rho_f}\left(\langle \mathscr{P}\frac{\partial u_f^i}{\partial x_j}\rangle + \langle \mathscr{P}\frac{\partial u_f^j}{\partial x_i}\rangle\right)$$

$$+ \nu_f \frac{\partial^2\langle u_f^i u_f^j\rangle}{\partial x_k^2} - 2\nu_f\langle \frac{\partial u_f^i}{x_k}\frac{\partial u_f^j}{x_k}\rangle$$

➙ Gradient reconstruction: Green-Gauss

Fluid instance (3): Numerical parameters

➙ Half pipe: 1 m length, 0.1 m radius, hexahedric mesh



➙ Time resolution: dt = 0.1 s, 100 iterations (10 s physical time)

➙ Tool: Code_Saturne

Particle instance (1): Physical parameters

- Fluid properties: 'frozen' flow
- Particle properties: $\varnothing = 10^{-4}$ m, ρ_p, spherical shape, volumetric flow rate
- Injection area: point-source injection

Particle instance (2): Model parameters

➙ Stochastic Lagrangian model

$$
\begin{cases}
dx_p = U_p dt, \\
dU_p = \dfrac{(U_s(t) - U_p(t))}{\tau_p} dt, \qquad \tau_p = \dfrac{\rho_p}{\rho_f} \dfrac{d_p^2}{18\nu_f} \\
dU_s^i = -\dfrac{1}{\rho_f} \partial_i \langle \mathscr{P} \rangle (t, x_p(t)) dt \\
\qquad + (\langle U_p^j \rangle - \langle U_f^j \rangle) \dfrac{\partial \langle U_f^j \rangle}{\partial x_i} dt + G_{ij}^* (U_s^j - \langle U_f^j \rangle (t, x_p(t)) dt \\
\qquad + \sigma_{ij}(t, x_p(t)) dB_j
\end{cases}
$$

Particle instance (3): Numerical parameters

- ➙ Half pipe: 1 m length, 0.1 m radius, hexahedric mesh
- ➙ Particle injection: 200 particles at each time step
- ➙ Time resolution: dt = 0.01 s, 500 iterations (5 s physical time)
- ➙ Tool: Code_Saturne

Wrapper/Mapper

- Coupling 2 components: CFD fluid (tool: Code_Saturne) - Particle tracking (tool: Code_Saturne)
- Model: $\mathbb{P}_0$-interpolation, frozen fields
- Exchanged variables: pressure, velocity field, R_{ij}, turbulent viscosity ν_t, ε
- Exchange frequency: initialization
- Tool: MEDCoupling

Sensitivity analysis

Variables of interest (1)

→ **Physical parameters:**

- Fluid properties: ρ_f, T, $|U_{\text{volumetric}}|$
- Particle properties: $\varnothing$, ρ_p, shape, flow rate
- Injection area

→ **Model parameters:**

- Turbulence model
- Gradient reconstruction
- Agglomeration

→ **Numerical parameters:**

- Spatial discretization
- Temporal discretization
- Number of particle injected

Variables of interest (1)

- **Physical parameters:**
  - Fluid properties: ρ, T, $|\boldsymbol{U}_{\text{volumetric}}|$
  - Particle properties: $\varnothing$, ρ_p, shape, flow rate
  - Injection area
- **Model parameters:**
  - Turbulence model
  - Gradient reconstruction
  - Agglomeration
- **Numerical parameters:**
  - Spatial discretization
  - Temporal discretization
  - **Number of particle injected**

Variables of interest (1)

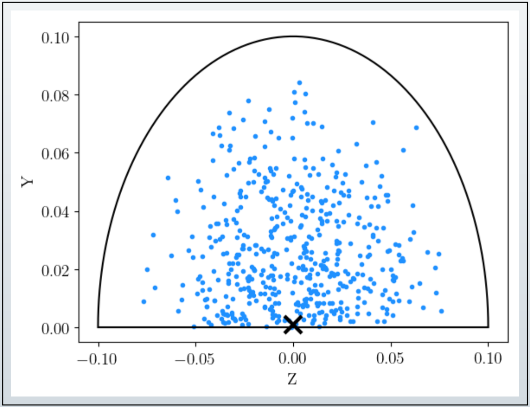
- **Physical parameters:**
  - Fluid properties: ρ, T, $|\boldsymbol{U}_{\text{volumetric}}| \sim \text{Uniform}(\{1, 2\})$
  - Particle properties: $\varnothing \sim \text{Uniform}([10^{-6}, 10^{-3}])$, ρ_p, shape, flow rate
  - Injection area
- **Model parameters:**
  - Turbulence model
  - Gradient reconstruction
  - Agglomeration
- **Numerical parameters:**
  - Spatial discretization
  - Temporal discretization
  - **Number of particle injected** $\sim \text{Uniform}([100, 1000])$

Variables of interest (2)

$$
\begin{cases}
dx_p = U_p dt, \\
dU_p = \dfrac{(U_s(t) - U_p(t))}{\tau_p} dt, \qquad \tau_p = \dfrac{\rho_p}{\rho_f} \dfrac{d_p^2}{18\nu_f} \\
dU_s^i = -\dfrac{1}{\rho_f} \partial_i \langle \mathscr{P} \rangle (t, x_p(t)) dt \\
\qquad + (\langle U_p^j \rangle - \langle U_f^j \rangle) \dfrac{\partial \langle U_f^j \rangle}{\partial x_i} dt + G_{ij}^* (U_s^j - \langle U_f^j \rangle (t, x_p(t)) dt \\
\qquad + \sigma_{ij}(t, x_p(t)) dB_j
\end{cases}
$$

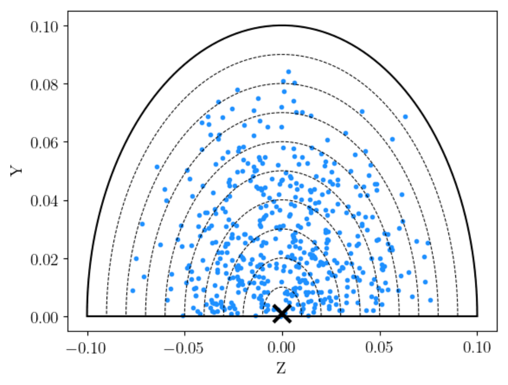
Observable (1)

- Debit at output
- Concentration at output
- Particle diameter



Observable (1)

- Debit at output
- **Concentration at output**
- Particle diameter



Observable (2)

➙ **Concentration at output:**

- Non-deterministic observable: depends on particles trajectories

- Time cumulative over the last 100 iterations

SA Methods (1): Overview

1. Non-parametric techniques
   - ➙ Input-output correlation
     - Pearson correlation coefficient
     - Standardized regression coefficient
   - ➙ Computation of several esitmators

2. Variance-based indicators
   - ➙ Sobol' sensitivity indices
   - Rests on the assumption of independant inputs
   - Implicity assumes that the second moment is sufficient to describe output variability

3. Looking at the entire distribution

First order Sobol indice

First order indice: $$S_i = \frac{\mathrm{Var}[\mathbb{E}[Y|X_i]]}{\mathrm{Var}[Y]}$$

- Variable ranking
- $S_i \in [0,1]$ are directly interpreted as measures of sensitivity: the larger it is, the more influential the i^{th} input X_i
- Common methods for assessing uncertainty imporance
- Requires a high number of simulations (computational cost, memory space, ..)

Total order Sobol indice

Total order Sobol indice:

$$S_i^{\text{tot}} = \sum_{\mathbf{u} \subset \{1,\ldots,d\}, \mathbf{u} \neq \emptyset, i \in \mathbf{u}} S_{\mathbf{u}}$$

- ➙ Absolute ranking
- ➙ Measures the contribution to the output variance X_i, including all variance caused by its interactions, of any order, with any other input variables

SA Methods (3): Metamodeling

➙ Generate a surrogate model: alternative model that mimic the solver and which is **less** costly

Polynomial chaos expansion

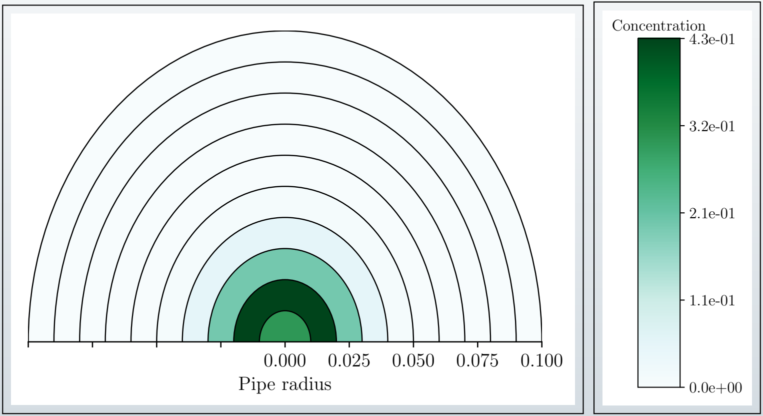
➙ Polynomial chaos expansion is a polynomial spectral decomposition of random variables on the basis of orthogonal polynomials

Gaussian process regression (Kriging)

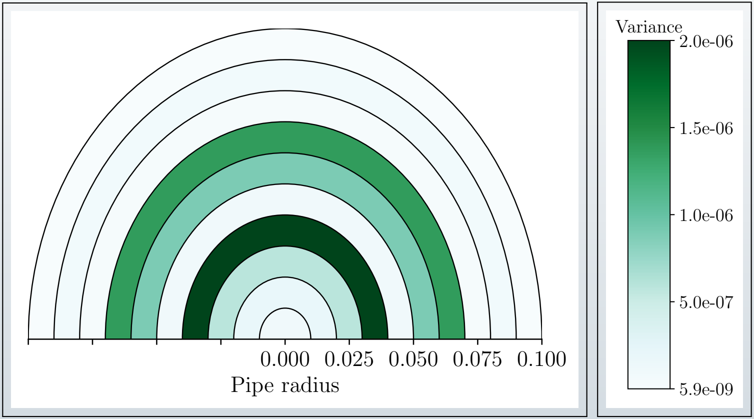
➙ Gaussian process regression is a method of interpolation where the interpolated values are modeled by a Gaussian process

Numerical results

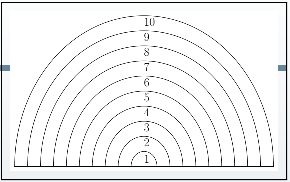
Average concentration at output



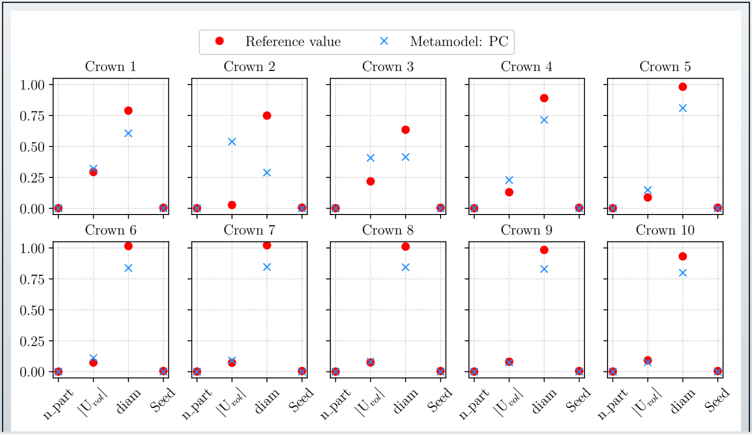
Intrinsic variance



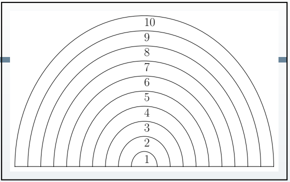
First Sobol indice

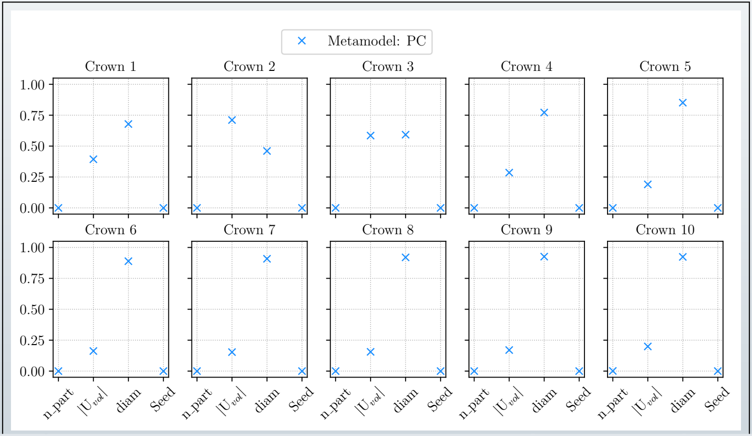


- **Tool:** Python library OpenTurns
- **Sample size:** 13200



Total Sobol indice





Plateform

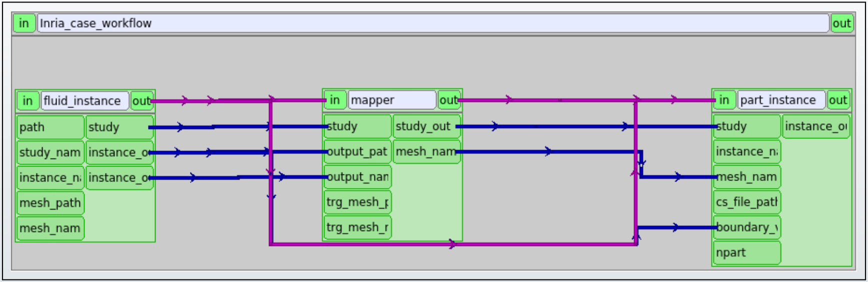
Salome

- SALOME is an open-source software that provides a generic Pre- and Post-Processing platform for coupled numerical simulation
- https://www.salome-platform.org/

SALOME enables to:

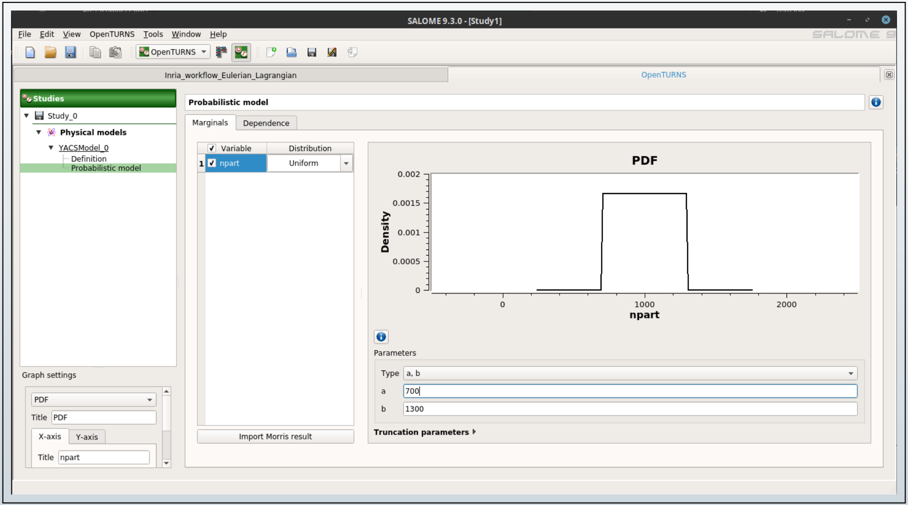
- create, modify, import and export, repair and clean CAD models
- generate, edit or check meshes; import and export mesh data
- perform computation using one or more external solvers (coupling)
- display computation results (scalar, vectorial data)

Salome - Yacsgen



Name	Type	Value
path	string	/user/adupre/home/Documents/wimmp/Inria_Workflow/...
study_name	string	HALF_PIPE_LAGR
instance_name	string	Eulerian
mesh_path	string	/user/adupre/home/Documents
mesh_name	string	half_pipe.med

Salome - OTGUI (OpenTurns)



Conclusion

Conclusion and perspectives

Conclusion:

- Implementation of the workflow is not trivial: high computational cost, several tools involved
- Meta-modeling is an essential tool
- Particle diameter and fluid velocity are the most infulencial variables
- The choice of the observable is a crucial step

Perspectives:

- Sensitivity analysis based on the entire distribution
- Uncertainty quantification on particle diameter
- Observable: particle size (agglomeration/fragmentation)
- VIMMP: Systematic tool to assess the question of uncertainty in material workflows

Thank you for your attention !

Special thanks to Pascale, Eric and Jean-Pierre for all the discussions.
Thanks Pascale and Eric for the help in the worflow implementation.

References

- **Code_Saturne:**
  - Frédéric Archambeau, Namane Méchitoua and Marc Sakiz. *Code Saturne: A Finite Volume Code for the computation of turbulent incompressible flows - Industrial Applications*. International Journal on Finite Volumes, 2004, 1.
  - https://www.code-saturne.org/cms/documentation
- **Salome:**
  - https://www.salome-platform.org/
- **OpenTurns:**
  - Michaël Baudin, Anne Dutfoy, Bertrand Iooss and Anne-Laure Popelin. *OpenTURNS: An industrial software for uncertainty quantification in simulation*. 2015.
  - https://openturns.github.io/openturns/latest/contents.html
- **Sensitivity analysis:**
  - Saltelli, A. *Making best use of model evaluations to compute sensitivity indices*. Computer Physics Communication, 2002, 145, 580-297