# OpenTURNS Developer training: first steps

Trainer : Sofiane Haddad Airbus sofiane.haddad@airbus.com

### Developers training



## OpenTURNS: first steps

Navigation in the source code

2 Library development

Module development

### Navigation in the source code

### The Uniform distribution

- Locate the class within the library source code;
- Follow its inheritance graph in order to explore the Bridge pattern;
- Locate the associated regression test;
- Execute the test;
- Locate its SWIG interface file and its associated Python module;
- Execute the associated python test.

## Library development 1/2

#### **Projects**

Weighted interpolation: create a new NumericalMathEvaluation that interpolate between points of a NumericalSample using weighted kernel interpolation.

$$f(\underline{x}) = \alpha \sum_{i=0}^{N-1} K(\underline{x} - \underline{X}^i) \underline{Y}^i$$
 (1)

where K is a given Distribution and  $\alpha$  is a normalization factor.

- OloudMesher: mesh generation over a cloud of points using kernel mixture, pca, rotation, then levelset mesher on an interval
- ClenshawCurtis 1-d integration: https://people.maths.ox.ac.uk/trefethen/cctalk.pdf + ClenshawCurtisProductExperiment
- TawnCopula/JoeCopula/MaxCopula 2-d copulas

## Library development 2/2

### **Projects**

- Extend archimedian copulas from 2-d to n-d (FarlieGumbelMorgenstern, FrankCopula, ClaytonCopula, AliMikhailHaqCopula)
- ArchiMaxCopula, see ExtremeValueCopula
- Extend ComposedDistribution to accept a list of n-d distributions
- Add a method to SobolSimulationAlgorithm to draw sobol indices (openturns/issues/1001)

## Module development

### **Projects**

- Convert the IntegralCompoundPoisson distribution python module into an OpenTURNS C++ module
- Create a RandomVector distributed uniformly over an n dimensional sphere/ball/simplex;