

Algorithms for Uncertainty Quantification

Lecture 11: Software for UQ

ST 2018

Tobias Neckel
Scientific Computing in Computer Science
TUM



TUM Uhrenturm

Repetition of Previous Lecture

- random fields
 - motivation
 - application examples
 - formal definition
- stochastic processes
 - definition
 - example: Brownian motion
- approximation of random fields
 - the Karhunen-Loève expansion
 - example: approximation of the Wiener process

Concept of Building Block:

- Time: \approx 80 minutes
- Content
 - coding from scratch vs. using existing code
 - UQ software vs. UQ packages
 - the world of UQ software: what software is available?
 - survey on selected UQ software

Concept of Building Block:

- Time: \approx 80 minutes
- Content
 - coding from scratch vs. using existing code
 - UQ software vs. UQ packages
 - the world of UQ software: what software is available?
 - survey on selected UQ software
- Expected Learning Outcomes
 - The participants can list a variety of different software packages or libraries dealing with UQ aspects.
 - They are able to indicate advantages or target use cases for those implementations that are listed in more detail (i.e. in red in the overview).
 - For different use cases, the participants can indicate possible software and are able to give arguments for their choice.

Coding from scratch vs. Using Packages

from scratch

Coding from scratch vs. Using Packages

from scratch

- no dependency on external packages/libraries (and their changes/bugs/...)
at least: decision, on which to rely on
- Computer Science people like it :-)
- context UQ: “Monte Carlo is so easy”

Coding from scratch vs. Using Packages

from scratch

- no dependency on external packages/libraries (and their changes/bugs/...)
at least: decision, on which to rely on
- Computer Science people like it :-)
- context UQ: “Monte Carlo is so easy”

reusing existing codes

Coding from scratch vs. Using Packages

from scratch

- no dependency on external packages/libraries (and their changes/bugs/...)
at least: decision, on which to rely on
- Computer Science people like it :-)
- context UQ: “Monte Carlo is so easy”

reusing existing codes

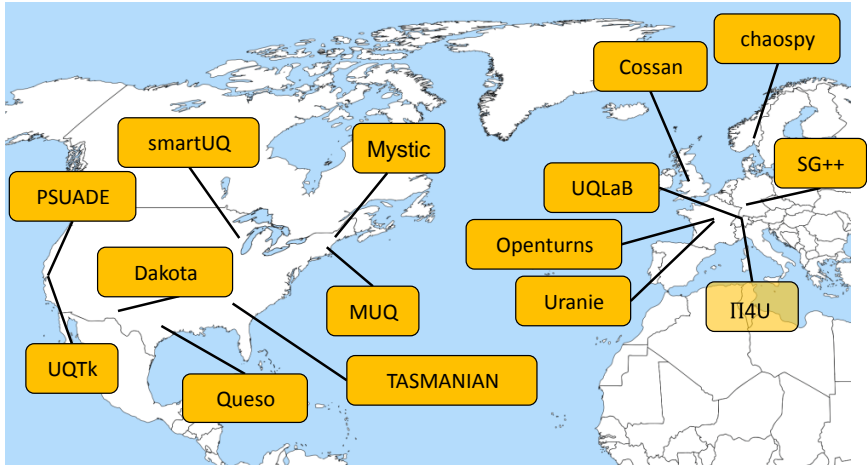
- avoid reinventing the wheel over and over again
- rely on work of others
 - faster time to solution (less debugging, testing, ...)!
 - quality of packages?
 - flexibility/portability supported?
 - maintainability of (own) code?
 - API fix or not very flexible
 - license issues?

UQ Software vs. UQ Packages

difference between software and small code packages:

- size of developing group:
single Ph.D. candidates \Leftrightarrow many life-time positions
- goal of implementations:
specific tasks \Leftrightarrow many methods/approaches, for large user base
- ease of use:
direct start with (almost) no docu \Leftrightarrow a lot of docu (to read)
- quality assurance:
manual tests (if any) \Leftrightarrow software engineering aspects, user support
- ...

World of UQ Software

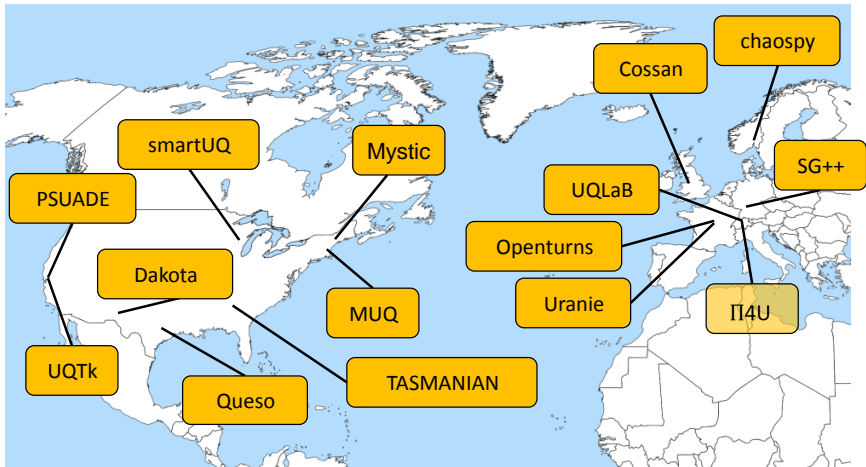


source of world map: <http://en.wikipedia.org>

SIAM conference UQ 18, minisymposia 88, 102, 115, and 128: Software for UQ, Tobias Neckel & Dirk Pflüger
 see also https://www5.in.tum.de/wiki/index.php/SIAMUQ18_-_Slides_Minisymph_Software4UQ

Discussion of Evaluation!

World of UQ Software – Reloaded



source of world map: <http://en.wikipedia.org>

SIAM conference UQ 18, minisymposia 88, 102, 115, and 128: Software for UQ, Tobias Neckel & Dirk Pflüger
 see also https://www5.in.tum.de/wiki/index.php/SIAMUQ18_-_Slides_Minisymph_Software4UQ

List of UQ Software

alphabetical; for sure: incomplete!

- Chaospy
- Dakota
- MUQ library
- Mystic
- NASA UQTools
- OpenCossan (matlab)
- Openturns
- P4U
- Promethee
- Psuade
- Queso
- SG++
- SmartUQ
- TASMANIAN
- UQLab (matlab)
- UQ Toolkit (UQTk)
- Uranie
- ...

List of UQ Software – Details

alphabetical; for sure: incomplete!

- Chaospy
- Dakota
- MUQ library
- Mystic
- NASA UQTools
- OpenCossan (matlab)
- Openturns
- P4U
- Promethee
- Psuade
- Queso
- SG++
- SmartUQ
- TASMANIAN
- UQLab (matlab)
- UQ Toolkit (UQTk)
- Uranie
- ...

for some: more detailed aspects

Aspects of Chaospy

URL	https://github.com/jonathf/chaospy
developing organisation	Jonathan Feinberg (SIMULA / private)
existing since	2014?
main purpose	forward UQ (PCE, sampling)
user support	mail support, documentation, tutorials
type of SW	framework/library, "full" license
programming language	Python
interface language support / API	Python
unique selling point	high-level Python, fast prototyping, PCE pseudospectral approach

Aspects of Dakota

URL	https://dakota.sandia.gov/
developing organisation	Sandia National Laboratory
existing since	1994
main purpose	UQ/optimisation for HPC applications
user support	FAQs, maillists, trainings, manuals & tutorials
type of SW	toolkit, LGPL license
programming language	C++
interface language support / API	text input files
unique selling point	large user base, very mature, HPC aspects

Aspects of MUQ

URL	http://uqgroup.mit.edu/software
developing organisation	UQ Group (Marzouk), MIT
existing since	2011
main purpose	MCMC, PCE, KL, optimisation
user support	Q&A, example page, interactive test sessions, doxygen
type of SW	collection of tools, "full" license
programming language	C++
interface language support / API	Python, C++
unique selling point	advanced UQ algorithms

Aspects of Openturns

URL	http://openturns.org/
developing organisation	4 companies: Airbus, EDF, IMACS, Phimeca Engineering
existing since	2005
main purpose	forward UQ, data analysis
user support	manuals, examp., mailing lists, user days
type of SW	library, open source LGPL license
programming language	C++
interface language support / API	Python, symbolic
unique selling point	larger user base?

Aspects of Π 4U

URL	http://www.cse-lab.ethz.ch/index.php
developing organisation	CSE lab, ETH
existing since	2014?
main purpose	Bayesian inversion for large-scale applications, sampling, optimisation
user support	tutorial
type of SW	framework, GPL license
programming language	C/C++
interface language support / API	TORC library
unique selling point	inversion, (heterogeneous) HPC platforms

Aspects of Queso

URL	http://libqueso.com/
developing organisation	UT Austin
existing since	2011?
main purpose	forward/inverse UQ for parallel applications
user support	maillists, doxygen
type of SW	collection of algorithms & functionalities, LGPL license
programming language	C++
interface language support / API	C++ library
unique selling point	inversion, Software Engineering aspects; collaboration with Dakota

Aspects of SG++

URL	http://sgpp.sparsegrids.org/
developing organisation	University of Stuttgart, TUM
existing since	2007?
main purpose	sparse grid toolbox for interpolation, quadrature, PDEs, data mining & machine learning, UQ, optimisation
user support	manuals, coding days, doxygen
type of SW	library, BSD-like license?
programming language	C++
interface language support / API	C++, Python, Java, MATLAB
unique selling point	sparse grids (spatially adaptive! combination technique)

Aspects of TASMANIAN

URL	http://tasmanian.ornl.gov/
developing organisation	Oak Ridge National Laboratory
existing since	2013?
main purpose	high-dimensional integration, interpolation & parameter calibration
user support	manuals
type of SW	toolkit (libraries)
programming language	C++
interface language support / API	C++, Python
unique selling point	sparse grids (combination technique)

Aspects of UQTk

URL	http://www.sandia.gov/UQToolkit/
developing organisation	Sandia National Laboratory
existing since	2005+X?
main purpose	forward/inverse UQ
user support	mailing lists, doxygen, manuals
type of SW	collection of libraries, LGPL license
programming language	C++
interface language support / API	C++, Python
unique selling point	high-level UQ, fast prototyping, many methods

Aspects of Uranie

URL	https://sourceforge.net/projects/uranie
developing organisation	CEA, France
existing since	2013?
main purpose	data analysis, UQ, optimisation
user support	mail support
type of SW	(L)GPL
programming language	Python, C++, C
interface language support / API	CINT, Python
unique selling point	many use cases, support for HPC platforms

Literature:

- References and manuals of software websites
- Overview of status at conference SIAM UQ 2018,
https://www5.in.tum.de/wiki/index.php/SIAMUQ18_-_Slides_Minisymp_Software4UQ

Summary

Software for UQ

- coding from scratch vs. using existing code
- UQ software vs. UQ packages
- the world of UQ software: what software is available?
- overview on selected UQ software
 - maturity/features
 - preparation for use case decisions