A Grid Computing environment for Design and Analysis of Computer Experiments

Yann Richet\textsuperscript{1}, David Ginsbourger\textsuperscript{2}, Olivier Roustant\textsuperscript{3}, Yves Deville\textsuperscript{4}

\textsuperscript{1} Radioprotection and Nuclear Safety Institute, France
\textsuperscript{2} Institute of Geology and Hydrogeology, University of Neuchatel, Switzerland
\textsuperscript{3} Graduate School of Engineering, Saint-Etienne, France
\textsuperscript{4} Statistical consultant, Chambéry, France

Great thanks for \texttt{Rserve} package and support: Simon Urbanek
Overview

Few words about Research and Industry

Computer Experiments framework

PROMETHEE Grid Computing environment

Real world example

Summary
Overview

Few words about Research and Industry

Computer Experiments framework

PROMETHEE Grid Computing environment

Real world example

Summary
Few words about Research and Industry

Reasons to work together
  Industry needs
    increase productivity
    overtake competitors
  Applied research needs
    industrial applications
    funding
Few words about Research and Industry

Reasons to work together

Industry needs
increase productivity
overtake competitors

Applied research needs
industrial applications
funding

Resiliency against partnership

Industry needs
short term RoI
efficient productive integration over existing practice

Applied research needs
"formal bridge" between theory and application
mid / long term & continuous partnership
Few words about Research and Industry

A well-suited partnership **DICE Consortium**
http://www.dice-consortium.fr (Deep Inside Computer Experiments)

Industrial partners

Research partners
Few words about Research and Industry

A well-suited partnership **DICE Consortium**
http://www.dice-consortium.fr (Deep Inside Computer Experiments)

Industrial partners: applications and testing
"orthogonal" high tech fields:
  automotive, oil, aerospace, nuclear plants & safety
shared funding: 40 000 € / year.partner

Research partners: **scientific and software deliverables**
  supplementary skills
  contractual contribution and goals
  hold scientific organization (PhD, postdoc, ...)

---

**DICE Consortium**
Few words about Research and Industry

A well-suited partnership DICE Consortium
http://www.dice-consortium.fr (Deep Inside Computer Experiments)

Industrial partners: applications and testing
"orthogonal" high tech fields:
  automotive, oil, aerospace, nuclear plants & safety
shared funding: 40 000 € / year.partner

Research partners: scientific and software deliverables
  supplementary skills
  contractual contribution and goals
  hold scientific organization (PhD, postdoc, ...)

Finite term project
  3 years long & every 6 month meeting focus on advances
software deliverables to be released as OSS (GPL/LGPL) in the end
scientific deliverables to be released in ~ public domain in the end
Overview

Few words about Research and Industry

Computer Experiments framework

PROMETHEE Grid Computing environment

Real world example

Summary
Overview

Few words about Research and Industry

Computer Experiments framework

PROMETHEE Grid Computing environment

Real world example

Summary
Computer Experiments framework

Computer code
Used as an unknown function
(Maybe) heavy CPU cost
Represents any existing simulation solver: finite-elements, Monte Carlo, ...
Fortran, C, close source, ...

Input variables
Environment, control or simulation variables
Scalar, vector, time sequences, ...

Output variables
Interest values
Scalar, vector, time sequences, ...
Computer Experiments framework

Input variables ➔ Computer code ➔ Output variables

From math. tools ... Design of experiments ... Surrogate modeling
DiceDesign, lhs, stats, ... DiceKriging, DiceEval, tgp, ...
Computer Experiments framework

From math. tools ...
Design of experiments
Surrogate modeling

DiceDesign, Ihs, stats, ...
DiceKriging, DiceEval, tgp, ...

... To engineering issues
Sensitivity analysis
Uncertainties propagation
Optimization
Inversion

DiceScreening, sensitivity, ...
DiceMRM, Ihs, boot, ...
DiceOptim, ...
...?
Computer Experiments framework

Software continuous integration: input / code / output
Wrap "Computer code" as a [R] function
  support computing environment (remote exec, network, grid load, ...)
  integrate parallel capabilities of algorithms (primary issue !)
Computer Experiments framework

Software continuous integration: input / code / output

Wrap "Computer code" as a [R] function
  support computing environment (remote exec, network, grid load, ...)
  integrate parallel capabilities of algorithms (primary issue !)

Integrate [R] within grid computing environment
  language interface & objects mapping [R] / {Java, C++, C#, Python, ...}
  sequential access to algorithms ( ask(...) & tell(...) )
Overview

Few words about Research and Industry

Computer Experiments framework

PROMETHEE Grid Computing environment

Real world example

Summary
Overview

Few words about Research and Industry

Computer Experiments framework

PROMETHEE Grid Computing environment

Real world example

Summary
PROMETHEE
Grid Computing environment

Software overview
Engineering through "Computer Experiments"
  Allows engineer to easily apply "brute" factorial design ...
  ... then induces to formalize its model and goals in a DoE approach
Frequently needs for supplementary features (through dedicated code plugin)
PROMETHEE
Grid Computing environment

Software overview
Engineering through "Computer Experiments"
  Allows engineer to easily apply "brute" factorial design ...
  ... then induces to formalize its model and goals in a DoE approach
  Frequently needs for supplementary features (through dedicated code plugin)

Distributed computing
  Compatible with larger set of CPU boxes:
    server, workstation, grid, cluster, ... and even (Windows) office desktop !
  Easy dynamic merge of heterogeneous power
Software overview

Engineering through "Computer Experiments"

Allows engineer to easily apply "brute" factorial design ...

... then induces to formalize its model and goals in a DoE approach

Frequently needs for supplementary features (through dedicated code plugin)

Distributed computing

Compatible with larger set of CPU boxes:

server, workstation, grid, cluster, ... and even (Windows) office desktop !

Easy dynamic merge of heterogeneous power

Application fields agnostic software

Any ASCII I/O software is compatible

All algorithms selectable for any computing software
PROMETHEE

Grid Computing environment

Software overview

Engineering through "Computer Experiments"
  Allows engineer to easily apply "brute" factorial design ...
  ... then induces to formalize its model and goals in a DoE approach
  Frequently needs for supplementary features (through dedicated code plugin)

Distributed computing
  Compatible with larger set of CPU boxes:
  server, workstation, grid, cluster, ... and even (Windows) office desktop !
  Easy dynamic merge of heterogeneous power

Application fields agnostic software
  Any ASCII I/O software is compatible
  All algorithms selectable for any computing software

Extendability & wrapping
  Basic (Groovy-DSL scripting) and extended (Java) plugins for computing code
  Basic ([R]) and extended (Java::Rserve or Java::* ) plugins for algorithms
PROMETHEE
Grid Computing environment

Network integration overview
PROMETHEE
Grid Computing environment

[R] tech. overview
[R] used as a script engine for dataset parameterizing
PROMETHEE
Grid Computing environment

[R] tech. overview

[R] used as a script engine for dataset parameterizing
[R]/Rserve used as an API inside Java DoE algorithm plugin

```r
R.voidEval("km" + currentiteration + "" + hcode + " <- km[y~" + kmodel + ","
+ "optin.method='gen',"
+ "penalty = NULL," + "covtype=" + covtype + "" + nuggetnoise_str
+ "design=X" + currentiteration + "" + hcode + "","
+ "response=Y" + currentiteration + "" + hcode + "","
+ "control=list(" + control_km + ")")");

REXP exists = R.eval("exists('km' + currentiteration + "" + hcode + "')")
if (exists == null || ![exists.asInteger() == 1]) {
    return new Status(Decision.DESIGN_OVER, "No km object built.");
}

R.saves(new File(_repository, "km" + (currentiteration) + "" + hcode + "".Rdata"), (currentiteration) + "" + hcode);

R.voidEval("EGO" + currentiteration + "" + hcode + " <- max_qEI.CL(model=km" + currentiteration + "" + hcode + "" + "npoints=" + batchSize + ","
+ "L=c(" + liar + ") + (search_min ? " : "-" +) + "Y" + currentiteration + "," + hcode + "$y)" + liar_noise + ","
+ "lower=c(" + ASCII.cat("", min) + "),"
+ "upper=c(" + ASCII.cat("", max) + "),"
+ "control=list(" + control_ego + ")
+ (expertfunction != null ? ",weight.EI=" + expertfunction : "") + ");

/*REXP*/ exists = R.eval("exists('EGO' + currentiteration + "" + hcode + "")
if (exists == null || ![exists.asInteger() == 1]) {
    return new Status(Decision.DESIGN_OVER, "No EGO object built.");
}
```
PROMETHEE Grid Computing environment

[R] tech. overview

[R] used as a script engine for dataset parameterizing
[R]/Rserve used as an API inside Java DoE algorithm plugin
[R] DoE algorithm plugin
Overview

Few words about Research and Industry

Computer Experiments framework

PROMETHEE Grid Computing environment

Real world example

Summary
Overview

Few words about Research and Industry

Computer Experiments framework

PROMETHEE Grid Computing environment

Real world example

Summary
Real world example

Criticality safety assessment

- Computer code: Monte Carlo neutrons simulator
- Output variables: neutron multiplication factor (scalar ~1)
- Input variables: many hypothesis as independent scalar code input parameters
- Engineering issue: find optimization (max) of output over hypothesis range
Criticality safety assessment

**Computer code:** Monte Carlo neutrons simulator

**Output variables:** neutron multiplication factor (scalar ~1)

**Input variables:** many hypothesis as independent scalar code input parameters

**Engineering issue:** find optimization (max) of output over hypothesis range

Old practical method (2 years ago)

- Hierarchical (user's prior) selection of ~3 input variables
- By-hand remote code launching (over interactive shell)
- Iterative & orthogonal maximization search (<20 points of calculation)
Real world example

Criticality safety assessment

- Computer code: Monte Carlo neutrons simulator
- Output variables: neutron multiplication factor (scalar ~1)
- Input variables: many hypothesis as independent scalar code input parameters
- Engineering issue: find optimization (max) of output over hypothesis range

Old practical method (2 years ago)
- Hierarchical (user's prior) selection of ~3 input variables
- By-hand remote code launching (over interactive shell)
- Iterative & orthogonal maximization search (<20 points of calculation)

Within Computer Experiments paradigm (PROMETHEE & R::DiceOptim / DiceKriging)
- No input variable ignored (no expert prior necessary)
- Automatic remote code launching & output parsing
  - Global maximization of output (may support >1000 points of calculation)
Real world example
Real world example

```
* Prologue pour 60 cm de béton

** TYPE 10 BIET @pas_cm / 2. | #.##### | @pas_cm / 2. | #.##### | @((60.0 + 2 * (1.5 + H.Cont / 2.)) | #.#####)
  VOLU 10 0 10 1. 0.0. @2 * ((1.5 + H.Cont / 2.) | #.#####)
  R21T 111100

* Interieur de la salle (bruit de densité variable)

** TYPE 11 BIET @pas_cm / 2. | #.##### | @pas_cm / 2. | #.##### | @((2.0 * (1.5 + H.Cont / 2.)) | #.#####)
  VOLU 11 10 11 @air_broui(d.broui(d.broui(d.broui(0)))) | 0.0. @2 * ((1.5 + H.Cont / 2.) | #.#####)

* Puits (acier 3 mm)

** TYPE 20 CYLZ @0.5 + Ep.Puits / 10. | #.##### | @1.5 + H.Cont / 2. | #.#####)
  VOLU 21 11 20 4. 0.0. @1.5 + H.Cont / 2. | #.#####)
  VOLU 22 12 20 4. 0.0. @4.5 + 2 * H.Cont / 2. | #.#####)

* Interieur conteneur (air)

** TYPE 30 CYLZ 6.5 @H.Cont / 2. | #.#####)
  VOLU 31 21 30 2. 0.0. @3.0 + H.Cont / 2. | #.#####)
  VOLU 32 22 30 2. 0.0. @5.0 + 3 * H.Cont / 2. | #.#####)

* Conteneur (acier 2.5 mm)

  VOLU 41 31 40 4. 0.0. @3.0 + H.Cont / 2. | #.#####)
  VOLU 42 32 40 4. 0.0. @6.0 + 3 * H.Cont / 2. | #.#####)

* Interieur conteneur (air)

** TYPE 50 CYLZ @(H.Cont / 2. | #.##### | @(H.Cont / 2. | #.#####)
  VOLU 51 41 50 2. 0.0. @3.0 + H.Cont / 2. | #.#####)
  VOLU 52 42 50 2. 0.0. @6.0 + 3 * H.Cont / 2. | #.#####)

* Fissure

** TYPE 60 CYLZ @(H.Cont / 2. | #.##### | @(dilcyl_cm(mPu.nax_kg,4d.Pu02) | #.#####)
```

```
Real world example
Real world example
Real world example
**Real world example**

<table>
<thead>
<tr>
<th>d_boulsc</th>
<th>d_FuO2</th>
<th>Status</th>
<th>Results</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.0552279181119280907</td>
<td>0.609450717202726</td>
<td>neutrosec-3: 4.4755 (06:30:46)</td>
<td>[mean_0F: 0.6351, sigma_0F: 0.9E-1]</td>
</tr>
<tr>
<td>0.42989818765058193</td>
<td>0.210020403942498</td>
<td>neutrosec-5: 4.6716 (06:30:46)</td>
<td>[mean_0F: -0.2866, sigma_0F: 0.7E-1]</td>
</tr>
<tr>
<td>0.8515932746145013</td>
<td>1.2342007517097087</td>
<td>neutrosec-4: 4.3378 (06:30:46)</td>
<td>[mean_0F: 0.37687, sigma_0F: 0.9E-1]</td>
</tr>
<tr>
<td>0.47786815941431814</td>
<td>2.7307404074808952</td>
<td>neutrosec-5: 4.1623 (06:30:46)</td>
<td>[mean_0F: 0.48104, sigma_0F: 0.6E-1]</td>
</tr>
<tr>
<td>0.75725785452618062</td>
<td>0.8738007003001.257</td>
<td>neutrosec-3: 4.6000 (06:30:46)</td>
<td>[mean_0F: 0.530765, sigma_0F: 0.6E-1]</td>
</tr>
<tr>
<td>0.77340199889888884</td>
<td>3.2489485173198572</td>
<td>neutrosec-4: 4.4755 (06:30:46)</td>
<td>[mean_0F: 0.61211, sigma_0F: 1.7E-1]</td>
</tr>
<tr>
<td>0.15940560818319386</td>
<td>2.8911014898930265</td>
<td>neutrosec-4: 3.3306 (06:30:46)</td>
<td>[mean_0F: 0.59163, sigma_0F: 0.7E-1]</td>
</tr>
<tr>
<td>0.85094001054127552</td>
<td>1.813208053323284</td>
<td>neutrosec-5: 4.6710 (06:30:46)</td>
<td>[mean_0F: -0.48950, sigma_0F: 0.7E-1]</td>
</tr>
<tr>
<td>0.20000000406680395</td>
<td>1.2190449242425010</td>
<td>neutrosec-5: 4.6000 (06:30:46)</td>
<td>[mean_0F: 0.47256, sigma_0F: 0.9E-1]</td>
</tr>
<tr>
<td>0.0</td>
<td>0.0</td>
<td>neutrosec-5: 4.1623 (06:30:46)</td>
<td>[mean_0F: 0.6471, sigma_0F: 0.9E-1]</td>
</tr>
<tr>
<td>1.0</td>
<td>0.0</td>
<td>neutrosec-3: 3.3306 (06:30:46)</td>
<td>[mean_0F: 0.59163, sigma_0F: 0.7E-1]</td>
</tr>
<tr>
<td>0.0</td>
<td>3.5</td>
<td>neutrosec-5: 4.6000 (06:30:46)</td>
<td>[mean_0F: -0.48950, sigma_0F: 0.7E-1]</td>
</tr>
<tr>
<td>1.0</td>
<td>3.5</td>
<td>neutrosec-5: 4.6716 (06:30:46)</td>
<td>[mean_0F: 0.67024, sigma_0F: 0.6E-1]</td>
</tr>
<tr>
<td>0.04500724876512460</td>
<td>2.60595187027531</td>
<td>running@neutrosec-3: 3.44823</td>
<td>Running calculation</td>
</tr>
<tr>
<td>0.21818560910901874</td>
<td>3.4398304585997464</td>
<td>running@neutrosec-3: 3.50893</td>
<td>Running calculation</td>
</tr>
<tr>
<td>0.1150214498566585</td>
<td>0.533852821727392</td>
<td>running@neutrosec-3: 3.30010</td>
<td>Running calculation</td>
</tr>
<tr>
<td>0.4131551401812902</td>
<td>0.8305360300903537</td>
<td>running@neutrosec-2: 3.47787</td>
<td>Running calculation</td>
</tr>
<tr>
<td>0.0521406105018179</td>
<td>2.4835083440109271</td>
<td>running@neutrosec-5: 3.45718</td>
<td>Running calculation</td>
</tr>
<tr>
<td>0.43041353099026530</td>
<td>3.777738400007727</td>
<td>intact</td>
<td>?</td>
</tr>
<tr>
<td>0.73433029202606555</td>
<td>4.004459749642715</td>
<td>intact</td>
<td>?</td>
</tr>
<tr>
<td>0.92594205093995111</td>
<td>2.100906522020333</td>
<td>intact</td>
<td>?</td>
</tr>
<tr>
<td>0.3427231062030111</td>
<td>2.604397592931983</td>
<td>intact</td>
<td>?</td>
</tr>
</tbody>
</table>

**Results**

<table>
<thead>
<tr>
<th>Data set</th>
<th>Size</th>
<th>Pts</th>
<th>State</th>
</tr>
</thead>
<tbody>
<tr>
<td>neutrosec-6</td>
<td>1</td>
<td>8</td>
<td>running</td>
</tr>
<tr>
<td>neutrosec-5</td>
<td>1</td>
<td>8</td>
<td>running</td>
</tr>
<tr>
<td>neutrosec-3</td>
<td>1</td>
<td>8</td>
<td>running</td>
</tr>
<tr>
<td>neutrosec-3</td>
<td>1</td>
<td>8</td>
<td>running</td>
</tr>
<tr>
<td>neutrosec-3</td>
<td>1</td>
<td>8</td>
<td>running</td>
</tr>
<tr>
<td>neutrosec-2</td>
<td>1</td>
<td>8</td>
<td>running</td>
</tr>
</tbody>
</table>

**Time spent:** 00:02:14, remaining(est.): 00:02:14, cases: 1322
Real world example

Size = 18
Maximum value is 0.67957 (sd=0.7E-4)
for
dPuO2 = 0.9
dPuO2 = 3.5

Next expected maximum value may be 0.436672361632858 (sd=0.102211865913429)
Improvement sequence is 0.0393452555916935

dPuO2 = 3.5
dbrui.scale = 0
Real world example
Real world example

Calculators pool

Optimum

Size = 22
Maximum value is 0.67957 (sd=0.7E-4)
for
d.PuO2 = 3.5
d.broui.scale = 0.0

Next expected maximum value may be 0.4037555564175635 (sd=0.05129131185748822)
Improvement sequence is 0.03634525559116935 0.015487465658755436

\[ d\text{.PuO2} = 3.5 \]

\[ d\text{.broui.scale} = 0.0 \]
Real world example

Size = 26
Maximum value is 0.54493 (sd=0.7E-4)
for
d.broui.scale = 0.05184126226735635

Next expected maximum value may be 0.330565836167595 (sd=0.0375218360226355)
Improvement sequence is 0.0363482855316935 0.015482485656755436 0.044748414478939784
Real world example
Real world example
Real world example

```plaintext
Size = 58
Maximum value is 0.95954 (d=-9.6E-4)
for
d.broui.scale = 0.09327404717601559
d.PuO2 = 1.01104293735437

Next expected maximum value may be 0.34113579499640373 (d=0.006657586035673094)
Improvement sequence is 0.0383435536916315 0.015487485656735436 0.04474444478333764 2.76385839951097E-4 0.01585271212596177 0.0

d.PuO2 = 1.01104293735437  d.broui.scale = 0.09327404717601559
```

![Graph with data points and curves]

- **d.PuO2 = 1.01104293735437**
- **d.broui.scale = 0.09327404717601559**
Real world example
Real world example
Real world example
Overview

Few words about Research and Industry

Computer Experiments framework

PROMETHEE Grid Computing environment

Real world example

Summary
Overview

Few words about Research and Industry

Computer Experiments framework

PROMETHEE Grid Computing environment

Real world example

Summary
Industry benefits: a five years leap
Better/stronger day-to-day eng. conclusions

Face new challenges: harder eng. issues *now* reachable

New abstract & formalized approach of old engineering practices

Research support
Lot of new industrial applications

Lot of feedback on algorithms, underlying hypothesis, *ideas*

New [R] users ...

... and a bit of (wholesome) money :(
... thanks to integration of a flexible (technology, license & community)

research software: [R]

a disruptive (re-think true needs, use true resources)

industrial software: PROMETHEE

... available for free at http://promethee.irsn.fr