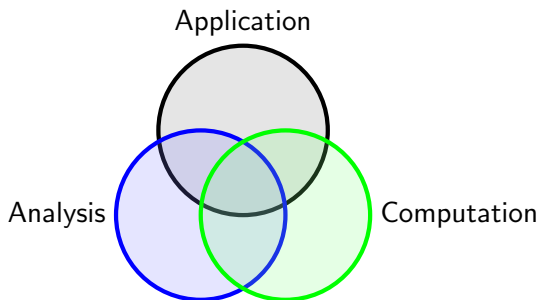# Software for Parallel Global Optimization

David Bindel

6 Jul 2017

The Computational Science & Engineering Picture

- MEMS
- Smart grids
- Networks
- Systems

- Linear algebra
- Approximation theory
- Symmetry + structure
- Optimization

- HPC / cloud
- Simulators
- Solvers
- Frameworks

Work with Chris now touches most of these (modulo choice of apps)!

A brief (and biased) history

- **Summer 1997**: My first exposure to high-dimensional interpolation (Bayesian Transformed Gaussian approach)

  … a long interlude …

- **Late 2010**: Shoemaker asks Birman about parallel global optimization; Birman introduces Bindel.

- **Dec 2010**: Proposal to NSF: "Parallel Global Optimization Algorithms with Asynchrony, Adaptive Re-Planning, and Response Surfaces for Costly Simulations."

- **Aug 2011**: Notification of award from NSF.

A brief (and biased) history

- **Phase 1 (2012–2013)**:
  - 2012: Initial parallel code with Yilun Wang, Jungmin Yun, and Jesseon Chang as CS MEng + summer project.
  - 2013: Julie Müller extends code + gets running on Yellowstone.
- **Phase 2 (2014–present)**:
  - Bindel rewrites "plumbing" layer POAP (Summer 2014);
  - Eriksson writes PySOT (Summer 2015);
  - Paul West adapts for cloud (Spring 2017).

The Challenge

Goal: Framework for surrogate-based optimization

- Serial or parallel (synchronous or *asynchronous*)

- Incorporating bounds or partial evaluations

- Variable run time

- Early terminations and partial failures allowed

... and ideally not too hard to program.

POAP: Plumbing for Opt with Asynch Parallelism

- Three main components

  1. A **strategy** to propose new evaluations.

  2. A set of **workers** that carry out evaluations.

  3. A **controller** that orchestrates workers.

- The controller is also responsible for

  1. Accepting/rejecting proposals by the strategy.

  2. Controlling and monitoring workers.

  3. Informing the strategy object of relevant events.

- Can naturally compose strategies.

- Workers and strategy communicate via the controller.

```python
from poap.strategy import FixedSampleStrategy
from poap.strategy import CheckWorkStrategy
from poap.controller import ThreadController
from poap.controller import BasicWorkerThread

# samples = list of sample points ...

controller = ThreadController()
sampler = FixedSampleStrategy(samples)
controller.strategy = CheckWorkerStrategy(controller, sampler)

for i in range(NUM_WORKERS):
    t = BasicWorkerThread(controller, objective)
    controller.launch_worker(t)

result = controller.run()
print('Best result: 0 at 1'.format(result.value, result.params))
```

POAP Built-Ins

- Controllers for multi-threading, MPI, simulated-time execution, etc.

- Adapters for several parallel programming styles (serial and bulk evaluation calls, promise objects)

- Adapters for controlling/combining strategies (merge, retry, max eval)

- Tests and test support (including the `ChaosMonkeyStrategy`)

PySOT: Python Surrogate Opt Toolbox

- Collection of surrogate optimization strategies for POAP.

- Comes with several optimization test problems.

- Easy to extend (we hope?!), several users already.

PySOT: Main components

1. Optimization problem specification.

2. Initial experimental design (LHD/SLHD, Full Factorial, Box-Behnken)

3. Surrogate models (RBF, MARS, kriging, polynomial, combined)

4. Adaptive sampling methods (DyCORS, SRBF, DDS, Gutmann, GA and gradient optimizers on surrogate)

5. Optimization strategies (serial/synchronous/asynchronous, continuous and integer variables, etc)

6. The PySOT GUI

Things to play with!

- Dated but still good: talk by David E at CMWR 2016
  https://people.cam.cornell.edu/~dme65/presentations.html

- POAP

    - Blurb: http://www.cs.cornell.edu/~bindel/blurbs/poap.html

    - GitHub: https://github.com/dbindel/POAP

    - PyPI: https://pypi.python.org/pypi/POAP

- PySOT

    - Blurb: http://www.cs.cornell.edu/~bindel/blurbs/pysot.html

    - GitHub: https://github.com/dme65/pySOT

    - PyPI: https://pypi.python.org/pypi/pySOT

- More planned for this summer as well!

Thanks, Chris!

I'm looking forward to much fruitful collaboration to come.

P.S. How many collaborators named "David" have you had?
           (Albonesi, Bindel, Eriksson, Onstad, Pimentel, Ruppert, ...)