Probabilistic Approaches for Communication Avoidance and Resilience in Exascale Simulations

SAND2013-6944C

Bert Debusschere†, Habib Najm, Khachik Sargsyan, Cosmin Safta

† **bjdebus@sandia.gov**
Sandia National Laboratories,
Livermore, CA

ExaMath13 Workshop – Aug 21-22, 2013 – Washington, DC

Supported by the US Department of Energy (DOE), Office of Advanced Scientific Computing Research (ASCR), Applied Mathematics Research (AMR) program

Outline

Sandia National Laboratories

Novel robust scalable solvers are needed

- Exascale architectures present many daunting challenges to application codes
- Scalability will be key issue
  - Need to run on millions of cores
  - Communication expensive compared to floating point operations
- Robustness against wide range of faults
  - Soft errors such as bit-flips introduce randomness
  - Loss of components due to hardware failures
  - Can not expect full machine to be up for any reasonable length of time
- Conventional approaches may not be effective
  - Time to save or restart from checkpoint may exceed MTBF (Mean Time Between Failures)

Sandia National Laboratories

Probabilistic Approaches for Communication Avoidance and Resilience

- Estimating values before they are available can let computations proceed
- Probabilistic generalization of accelerated Schwarz coupling for scalability and resilience

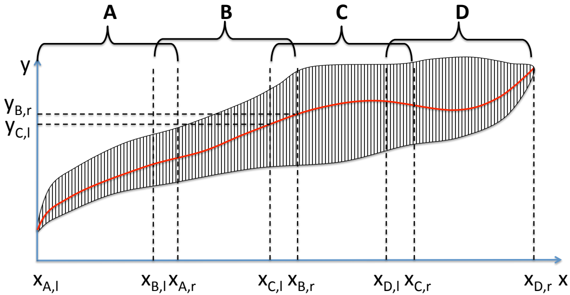
Sandia National Laboratories

Probabilistic Estimation of Information

- While waiting for information to become available, estimate its value, *e.g.* with Bayesian approaches
- Let computation proceed if sufficiently confident in the probabilistic estimate
- *E.g.* Residual of elliptic equation with domain decomposition
  - Norm of residual requires *all gather* operation
  - Instead of waiting for all subdomains to report back, estimate norm based on partial info and behavior in past iterations
  - Proceed if residual predicted to be below tolerance threshold with confidence

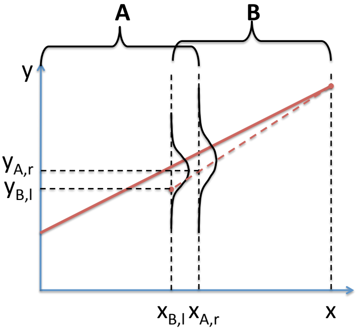
Sandia National Laboratories

Probabilistic generalization of Schwarz coupling



- Probability Density Function (PDF) represents current state of knowledge about solution
  - Uncertainty from incomplete convergence
  - Uncertainty due to noisy or failed operations
- Decompose into overlapping sub-domains with uncertain boundary conditions
- Targeted simulations to refine knowledge

Response surfaces between spatial boundaries provide concurrency and fault-tolerance

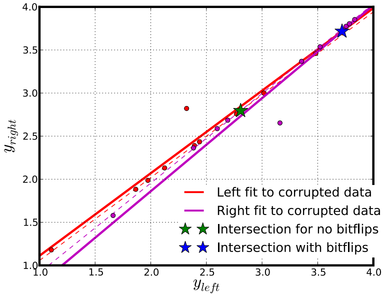


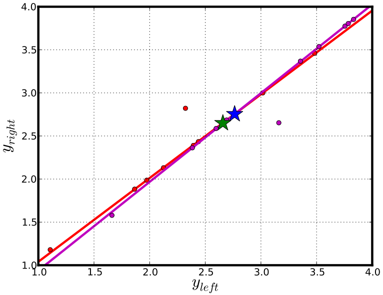
Faults show up as noisy or missing data

$$
\begin{aligned}
y_{A,r} &= f_{B,A}(y_{B,l}, \boldsymbol{\lambda}_B) \\
y_{B,l} &= f_{A,B}(y_{A,r}, \boldsymbol{\lambda}_A) \\
D &= \{y_{A,n}, y_{B,n}\}_{n=1}^{N} \\
p(\boldsymbol{\lambda} \mid D) &\propto p(D \mid \boldsymbol{\lambda}) p(\boldsymbol{\lambda})
\end{aligned}
$$

- Values of solution at boundaries are a function of each other
    - Intersection provides solution
- Infer response surfaces from subdomain samples

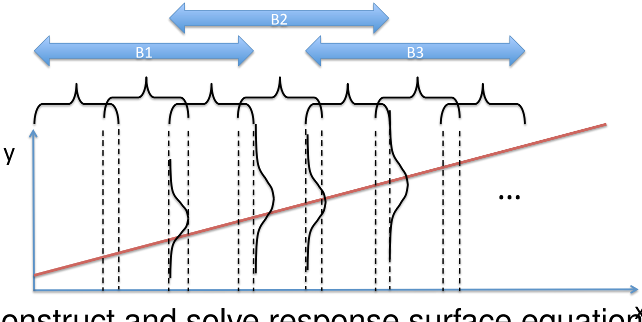
ℓ_1 noise model provides good noise rejection





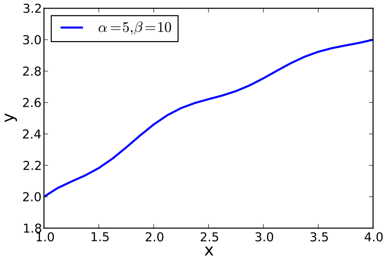
- ℓ_2 noise model (Gaussian) assumes equal noise in all data points
- ℓ_1 noise model (Laplacian) is better suited for case where most data is exact but some points have large errors
  - Analogy with compressed sensing: find solution with as few non-zero residuals as possible

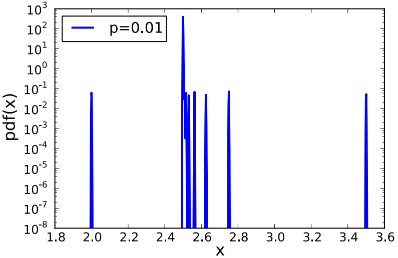
Iterative response surface approach



- Construct and solve response surface equations for blocks of subdomains
- Nested iteration loops provide course and fine grained concurrency
    - Inner sampling loop creates response surfaces
    - Outer sampling loop advances solution at block boundaries
- Aim is to reduce system-wide communication

1D non-linear differential equation

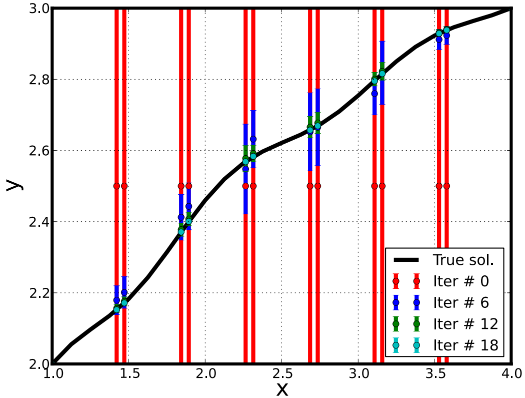




- Consider non-linear differential equation $(e^y)'' = \beta \sin(\alpha x)$, $\alpha = 5$ and $\beta = 10$
- Solution: $y(x) = \log\left(-\frac{\beta}{\alpha^2}\sin(\alpha x) + ax + b\right)$
- Subdomain solves perturbed with probability p
  - Flip randomly selected bit in 64 bit binary representation
  - Remove values outside [0,5] as outliers

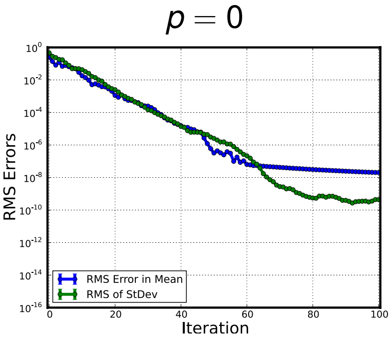
Sandia National Laboratories

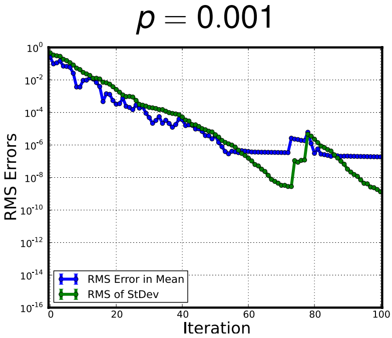
Successive iterations reduce uncertainty in the solution



- Initial guess is Gaussian $\mathcal{N}(2.5, 0.5^2)$
- Distribution moves and narrows towards exact solution

Convergence with and without bit-flips





- Fault-free system and system with bit-flips converge similarly
- Enhancements are planned to reduce perturbations from faults close to convergence

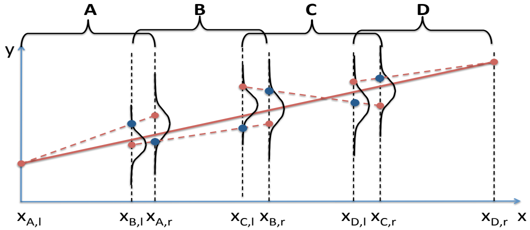
Conclusions and Ongoing Work

- Approach is designed to deal explicitly with faults
  - Soft errors such as bit-flips
  - Missing data due to communication issues or compute node failures
- Preliminary numerical examples show promising behavior in terms of convergence and fault tolerance
- Ongoing work
  - Rigorous numerical analysis of convergence behavior
  - Extension to higher-dimensional computational domains
  - Implementation on distributed system
  - Detailed scalability and effectiveness study on more complex problems

Sandia National Laboratories

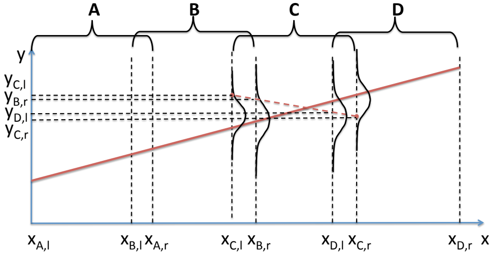
Extra Material

Brute force sampling approach



- Solve equation on subdomain for sampled values of uncertain solution at boundaries
- Each subdomain solve provides sample of solution at neighboring domain boundaries
- Ensemble of samples used to update PDF of solution at boundaries
- Analagous to additive Schwarz preconditioner
- Robust and scalable, but slow convergence

Response surface approach

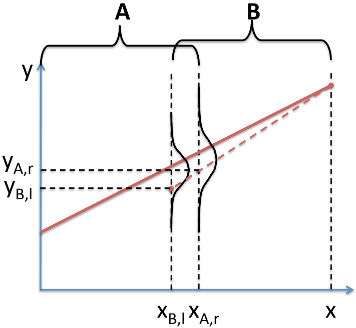


$$
\begin{aligned}
y_{B,r} &= f_{C,B}(y_{C,l}, y_{C,r}) \\
y_{D,l} &= f_{C,D}(y_{C,l}, y_{C,r}) \\
y_{C,l} &= f_{B,C}(y_{B,l}, y_{B,r}) \\
y_{C,r} &= f_{D,C}(y_{D,l}, y_{D,r})
\end{aligned}
$$

- Write values of solution at boundaries as a function of each other
- Infer response surfaces from subdomain samples
- Solution to resulting system of equations gives full solution
- Analogous to Aitken-like acceleration of Schwarz preconditioner [Garbey, SISC, 2005]

Sandia National Laboratories

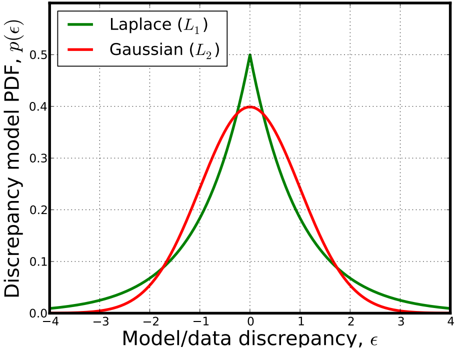
Bayesian inference of response surfaces provides fault tolerance



$$
\begin{aligned}
y_i &= \sum_{k=0}^{P} c_k \psi_k(y_j) \\
\boldsymbol{c} &= \{c_k\}_{k=0}^{P} \\
D &= \{y_{i,n}, y_{j,n}\}_{n=1}^{N} \\
p(\boldsymbol{c}|D) &\propto p(D|\boldsymbol{c})p(\boldsymbol{c})
\end{aligned}
$$

- Bit-flips show up as noise in function evaluations
- Other faults often show up as missing data
  - Communication delay
  - Failing nodes
  - Values rejected as outliers

Choice of noise model



- ℓ_2 noise model (Gaussian) assumes equal noise in all data points
- ℓ_1 noise model (Laplacian) is better suited for case where most data is exact with some deviations

Parameters of the algorithm

- D: Number of subdomains
- B: Number of blocks, as well as their structure $[A_i^{min}, A_i^{max}]_{i=1}^{B}$
- N_o: Number of outer samples at the left and right ends of each block
- N_i: Number of inner samples that enables response surface construction
- r: Order of response surfaces
- M: Number of iterations
- p: Probability of bit-flips
- R: Number of replica, identical runs, for more robust convergence results

Application to 1D non-linear differential equation

Consider a non-linear differential equation

$$(e^y)'' = \beta \sin(\alpha x), \tag{1}$$

for some constant parameters α, β. It clearly has the form $\mathcal{L}(y'', y', y) = h(x)$ with a non-linear operator $\mathcal{L}$. The solution of (1) has the form

$$y(x) = \log\left(-\frac{\beta}{\alpha^2}\sin(\alpha x) + ax + b\right), \tag{2}$$

for some values a and b found by matching the boundary conditions $y(s) = y_s$ and $y(f) = y_f$. For simplicity, we will operate in a range where such solution is real, i.e. the expression inside the logarithm in (2) is positive.

Note 1: One can achieve this by adjusting constants α and β, or by restricting the range of the boundary conditions y_s and y_f.
Note 2: In particular, for non-convergent methods, the boundary conditions often take values with which no real solutions exist.

The default values are $\alpha = 5$ and $\beta = 10$ - referred to as a 'smooth' problem.

Sandia National Laboratories