

Scaling the Science using Adaptivity and Uncertainty Quantification -- Case Study of Volcanic Hazard Analysis using HPC

"Parallelism and Adaptivity -- Marriages not Made in Heaven"

Abani K. Patra ...

Department of Mechanical and Aerospace Engineering
University at Buffalo, State University of New York, Buffalo, NY 14260
abani@eng.buffalo.edu

On leave
2007-08

Office of Cyberinfrastructure,
National Science Foundation

Project participation and funding

- ❑ Interdisciplinary research project funded by US National Science Foundation, ITR, CMG, EAR ...2001-?
- ❑ UB departments/people involved:
 - ❑ mechanical engineering: A Patra, A Bauer, T Kesavadas, C Bloebaum, A. Paliwal, K. Dalbey, N. Subramaniam, P. Nair, V. Kalivarappu, A. Vaze, A. Chanda
 - ❑ mathematics: EB Pitman, C Nichita, L. Le
 - ❑ geology (volcanology group): M Sheridan, M Bursik, E. Calder, B.Yu, B. Rupp, A. Stinton, A. Webb, B. Burkett
 - ❑ Geography (National Center for Geographic Information and Analysis): C Renschler, L. Namikawa, A. Sorokine, G. Sinha
 - ❑ CCR (UB Center for Comput. Research) M Jones, M. L. Green

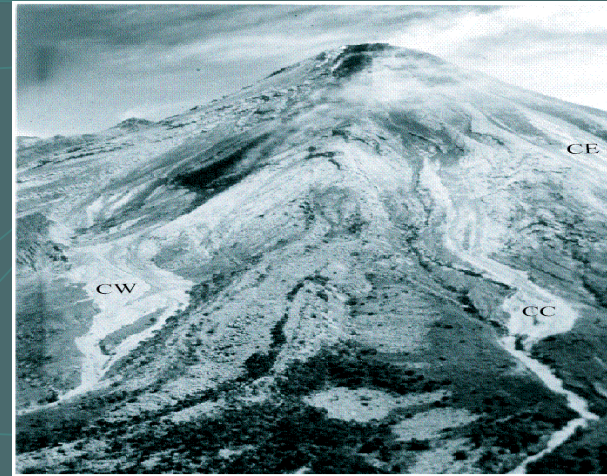
Outline of the talk

- Research Needs and a few difficulties
- Mathematical models used in TITAN2D and Numerical solvers
- Adaptive meshing, Load balancing, Parallel implementation
 - Performance Maintenance
- Uncertainty Quantification and Hazard Maps
 - Simulators and Emulators
 - Adaptivity and Bayes Linear Models
- “Real Time” \Leftrightarrow Parallel Construction of Hazard Maps

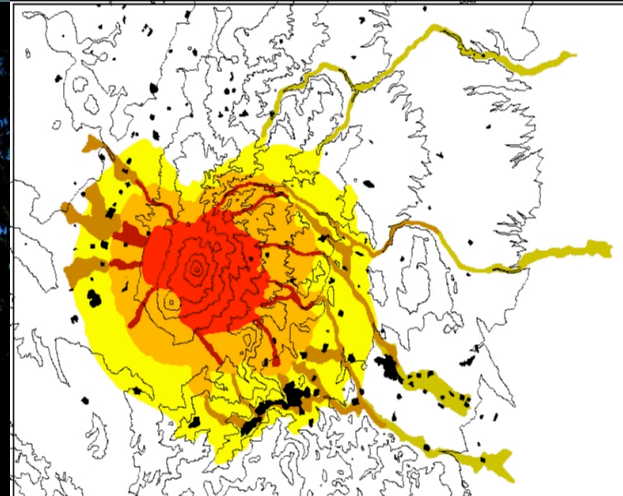
Geophysical Flows



Mt. St. Helens, USA



Volcan de Colima, Mexico



Hazard map at Pico de Orizaba -- hazard maps by Sheridan et. al. based on past flow data and expert intuition

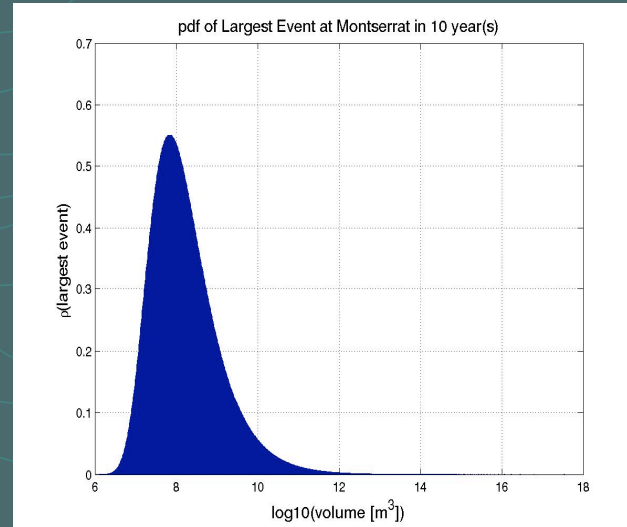
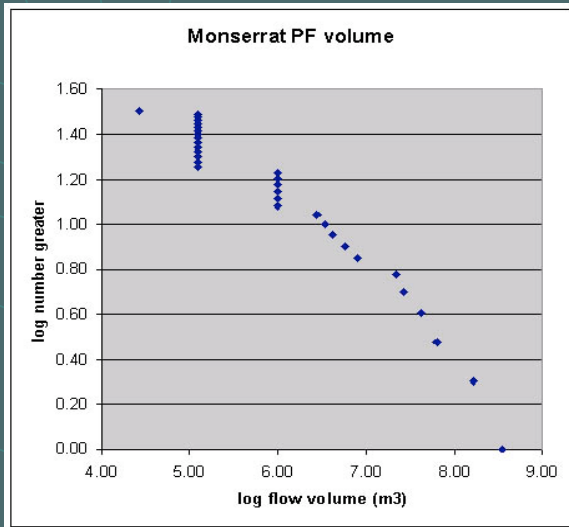
What do we need to know and What do we have?

- Q1: Given a location x and time T -- what is the hazard of a catastrophic event? e.g. $P(\text{flow} > 1\text{m in } T) \sim 0.000001?$
- Q2: Given a jurisdiction what is the hazard of an event in the next T time period of all locations?
- Models of the Physics of individual flows (PDE based)
- Data on past events -- detailed and precise for some aspects, sparse and poor for most aspects
- Expert belief and intuition
- Methodology for quantifying uncertainty

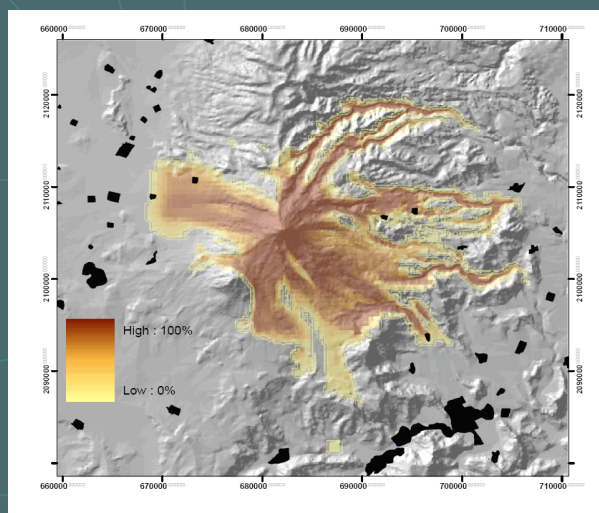
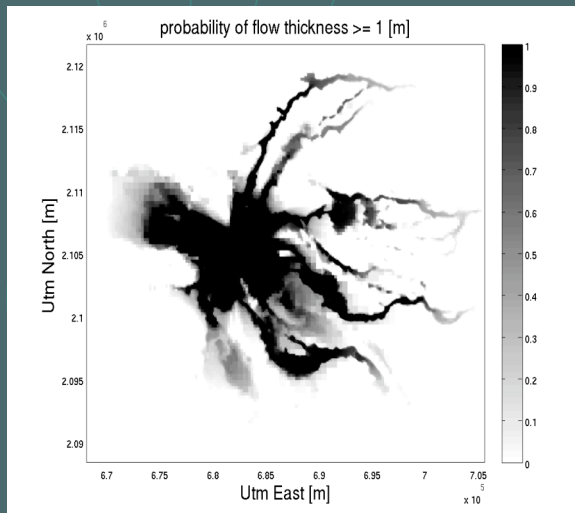
What do we need to know and How do we get it?

- Q1: Given a location x and time T -- what is the hazard of a catastrophic event? e.g. $P(\text{flow} > 1\text{m in } T) \sim 0.000001$?
- Q2: Given a jurisdiction what is the hazard of an event in the next T time period of all locations?
- Approach 1: Given a simulator with “well defined input data uncertainties” -- use well chosen ensemble (Latin Hypercube, Quadrature driven ...) to propagate uncertainty and use simple expectation computations to make hazard map. [*Dalbey et. al. 2008, J. Geophys. Res.*]
- Approach 2: Given a location and sparse data create estimates of predictions and associated uncertainty using Bayesian methodology by using simulator to create emulator and use emulator in appropriate statistical methodology. [*Bayarri et. al. in review, Dalbey et. al. in prep.*]

Hazard Map Construction



Historical flow data and expert belief converted into recurrence probability of largest events



Probability of flow exceeding 1m for initial volume ranging from 5000 to 10^8 m³ and basal friction from 28 to 35 deg at Colima and Pico de Orizaba

a few difficulties

- Complex unpredictable physics
 - Flows are hazardous mixture of soil, rocks, clasts with interstitial fluid present -- many models Johnson '70, Savage - Hutter '89, Iverson '97, Pitman-Le '05 ... -- complex physics is still not perfectly represented

$$U_t + F(U)_x + G(U)_y = S(U)$$

$$U = (h, hv_x, hv_y), F(U) = (hv_x, hv_x^2 + 0.5kg_z h^2, hv_x v_y)$$

$$S_x = g_x h - hk \operatorname{sgn}\left(\frac{\partial v_x}{\partial y}\right) \partial_y (g_z h) \sin \phi_{\text{int}} - \frac{v}{|v|} \left[g_z h \left(1 + \frac{v}{rg_z}\right) \right] \tan \phi_{\text{bed}}$$

h: flow depth; hv: depth averaged momentum; g : gravity; ϕ : friction

Numerics

- High order slope-limiting upwinding two dimensional Godunov solver, second order predictor-corrector in time
 - Toro, 1997, 2001, Cockburn 2001, Hartmann and Houston 2003, Patra et. al. 2005, 2006.

- Runge Kutta Discontinuous Galerkin Formulation –
 - Patra et. al. 2006 Comp. Geosc

- Drying and Wetting Areas: System of equations loses strict hyperbolicity near the front (where $h=0$). Need front tracking algorithms
 - solve exactly the Riemann problem in the primitive variables near the front (ref. Toro-"Shock Capturing Methods for Free Surface Flows"-2001)

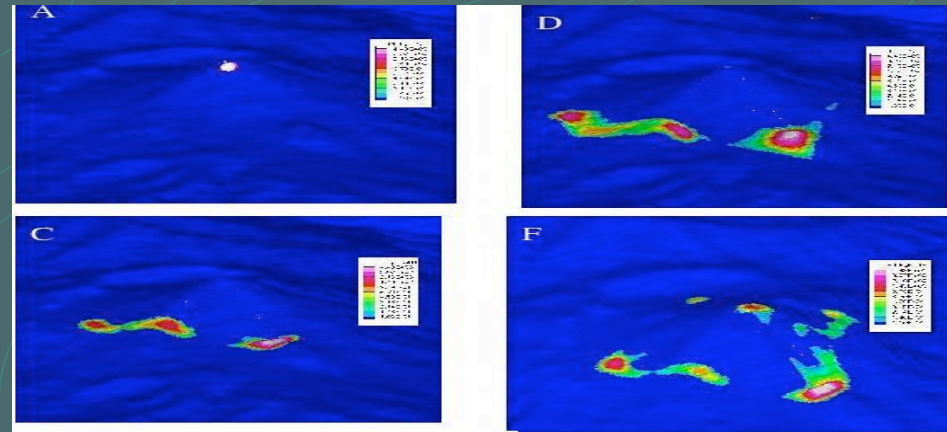
a few difficulties

● Uncertain Inputs based on sparse data

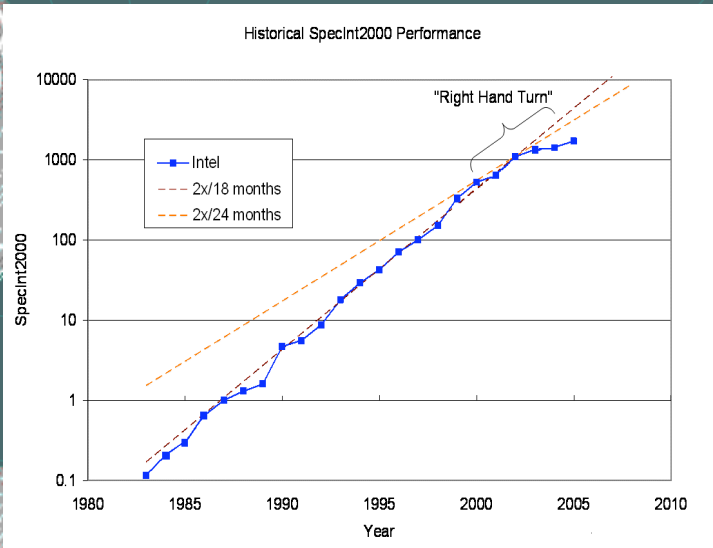
1. ϕ_{bed}
2. ϕ_{int}
3. Initial location
4. Initial volume
5. Initial velocity
6. Terrain elevation

● Expensive simulators

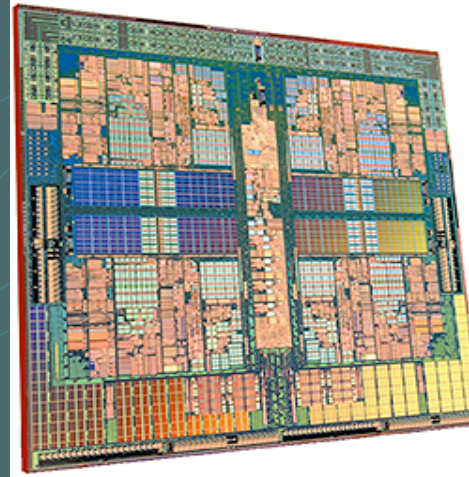
- Ensemble computations needed for hazard map constructions are **EXPENSIVE!**
- Single calculation -- 20 minute on 64 proc => Monte Carlo type computation needs 217 days
- Data dependence issues -- parallel efficiency beyond 64 cores is limited



a few recent difficulties!

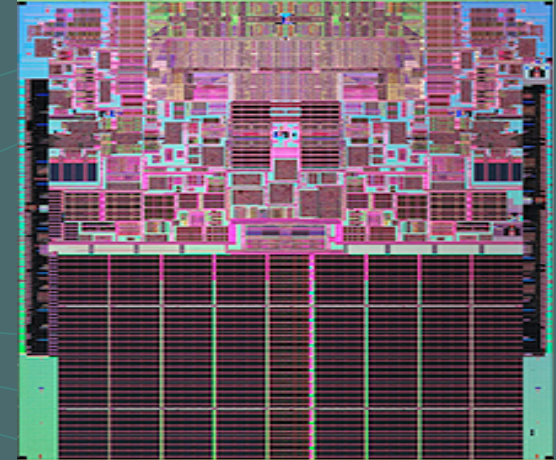


AMD Phenom



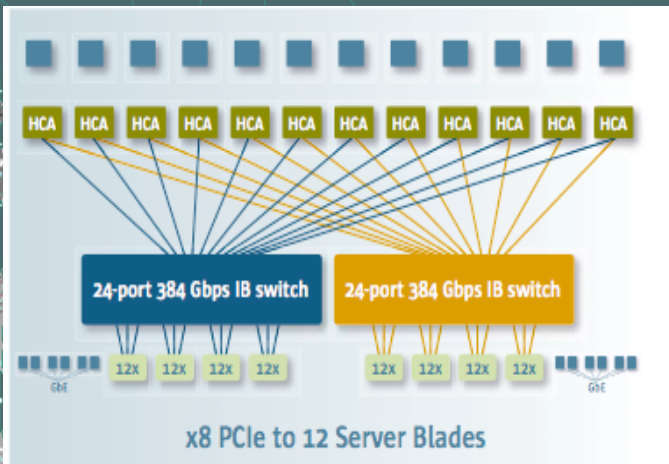
http://www.amd.com/us-en/assets/content_type/DigitalMedia/43264A_hi_res.jpg

Intel Woodcrest

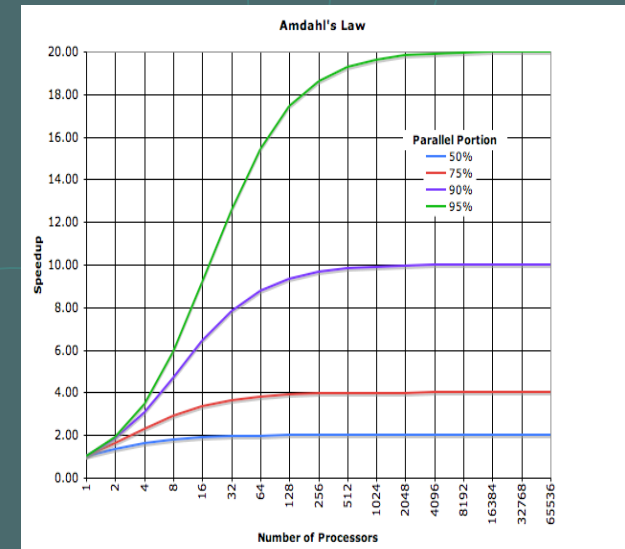


<http://www.intelstartyourengines.com/images/Woodcrest%20Die%20Shot%202.jpg>

Single Thread Performance



Heterogeneous, Hierarchic, Computer Architectures -- $O(100K)$ Compute cores, I/O nodes, communications subsystems, accelerators, vector units ...



Adaptivity and Parallelism

Computational Cost of simulations is a big obstacle in meaningful use of physical model based statistical methods

Adaptivity to minimize simulation cost of each instance

Adaptivity is crucial in accurate front capture

Adaptivity used in construction of the emulators

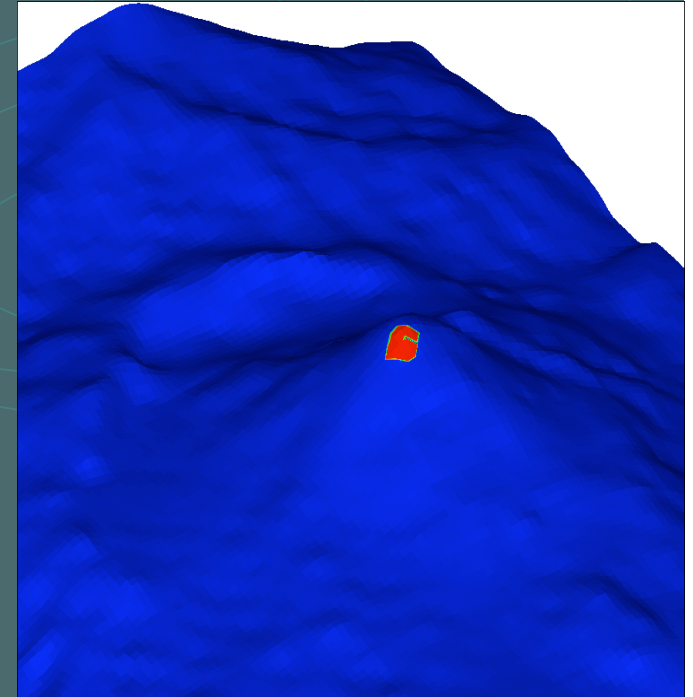
Parallelism needs to be used to maximize throughput

Motivation for Mesh Adaptivity

- ❑ Flow path cannot be predicted
- ❑ Flow at any time covers **less than 20%** of entire run-out region
- ❑ Capturing flow boundaries correctly

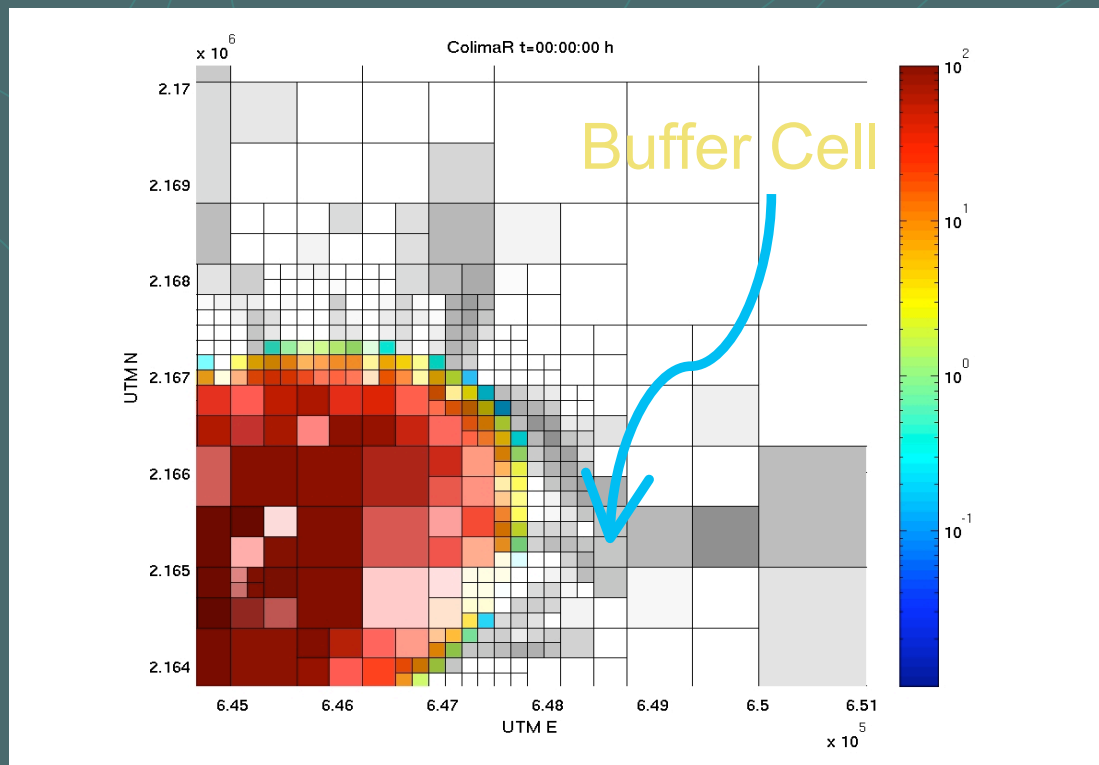
$$v \equiv \frac{hv}{h}; h \rightarrow 0; v \rightarrow \infty$$

- ❑ Mesh refinement every 10 time steps
 - ❑ Refine the top 20% of elements with the largest change in cell fluxes
 - ❑ Refine ahead of the front of the pile to capture front
 - ❑ Flow needs to enter only smallest cells
 - ❑ Mesh coarsening when the pile height is very small



Adaptivity for Wet Dry Front Capture

- 3 types of cells -- empty, full, buffer layer at front



Load-Balancing and Data Management

Load-balancing constraints

- ❑ Minimize interprocess communication
- ❑ Minimize load-balance time
- ❑ Minimize objects assigned to new processes (incrementality)

Introduced integrated data management using Hilbert Space Filling Curve (SFC) based indexing of objects (cells and nodes), distributed Hash tables and SFC based mesh partitioning

Patra et. al, '94,'01,'03,'05

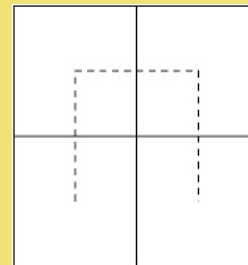
Space-Filling Curves for Load-Balancing

The Space-Filling Curves load-balancing algorithm is basically dividing up a weighted line

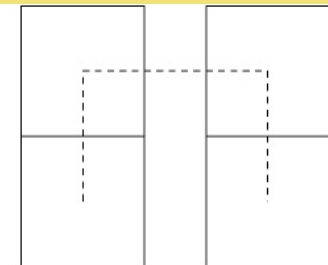
- Need to assign some type of computational weight to each load-balancing object

For dynamic load-balancing, need incremental partitions

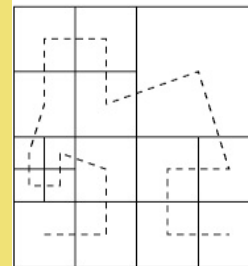
- Space-Filling Curves does this intrinsically



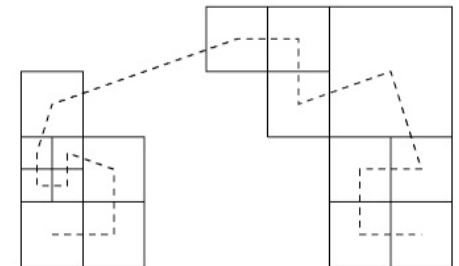
Initial Mesh



Initial Partition



Refined Mesh



Refined Partition

Data Driven Model Based Dynamic Load Balancing

- 3 types of cells -- empty, full, buffer layer at front -- use weighed partitioning using SFC with 3 different weights
- Performance Model Based: goal is to minimize communication time
 - Collect timing data for all MPI calls and total wall clock
 - Use previous 100 time steps data and least squares to obtain weights that minimize MPI time

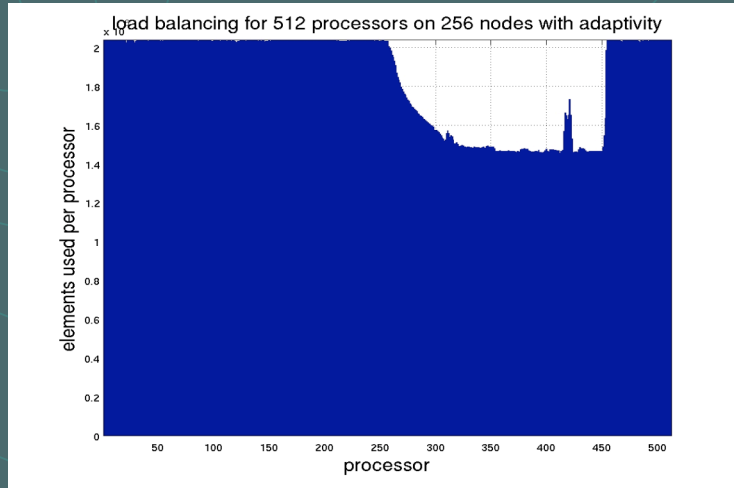
$$t_c \propto w_f \sum_i N_{f,i} + w_b \sum_i N_{b,i} + w_e \sum_i N_{e,i}$$

$$t_c + t_w = \text{const}; \quad -t_w = t_c - \text{const}$$

$$-t_w = Aw; \quad w = -(A^T A)^{-1} (A^T t_w)$$

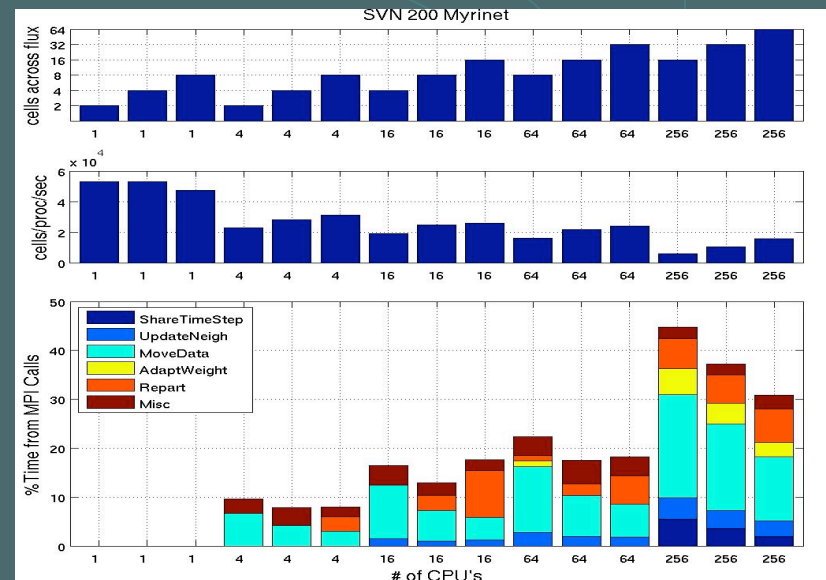
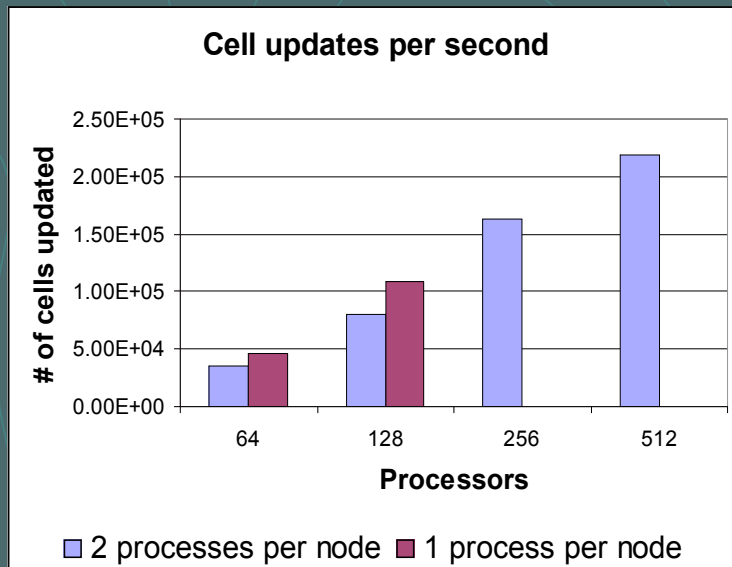
t_c : compute time; t_w : MPI time, $w=(w_f, w_b, w_e)$: weights

Computational Efficiency



Time step synchronization based on one time step previous data

Efficiency drops rapidly after 256 procs!



A way out!

Scale the science -- not the code!

→ Scale the construction of the hazard map!

Accelerate quantification of uncertainty

Monte Carlo/ Quasi Monte Carlo -- “too expensive!”

Stochastic Galerkin -- “hard to use with hyperbolic systems”

Bayesian Emulation with adaptive construction -
method of choice!

A Serendipitous Benefit

- Erratic results for 1 processor
 - Processor dies after two days
 - Experiment where one processor was overloaded with another process running at high priority
 - Dynamic load balancing moves most cells out of processor as weights adjust

Difficulties with MC, LHS & SG?

MC

- **Error = $\sigma/N_{MC}^{0.5}$**
- **$N_{MC} = 10^6$ required for 3 sig. fig. of accuracy**
- **$N_{MC} = 10^6$ only feasible if run time is small!**

LHS

- **Non-trivial to generate good high dimensional sample designs**
- **Not valid for quite as many problems as plain MC, ok for input uncertainty**

SG

- **For polynomial nonlinear systems, work is binomial in # of random dimensions and degree of polynomial series**
- **Non-polynomial nonlinear systems = infinite degree polynomial systems**
- **Truncating polynomial series adds error **each** time step**
- **Implementation can be prohibitively difficult depending on system of equations**

Bayes Linear Emulator

Statistical model of a process, can be built from multiple sources of different fidelity (coarse/fine simulator output, expert belief, experiments, etc.) data

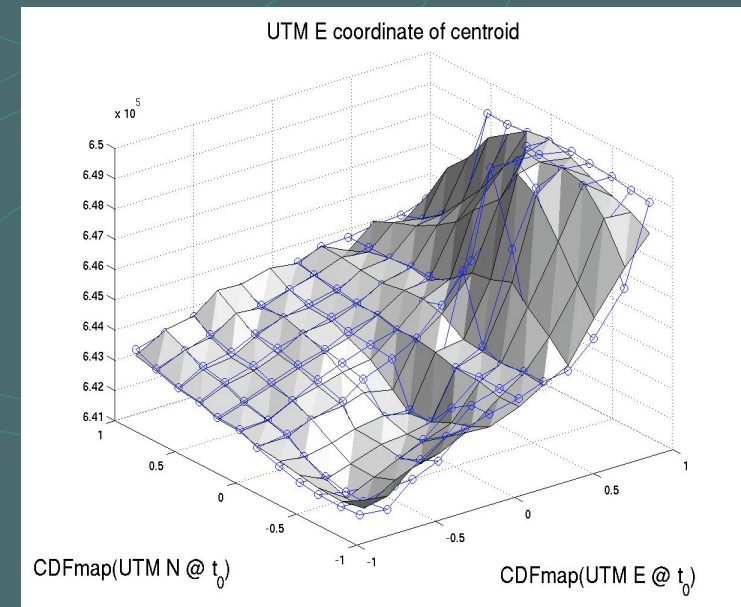
$$\begin{aligned} s(\mathbf{x})_{BL} &= \mu(\mathbf{x}) + \varepsilon(\mathbf{x}) \\ &= g_i(\mathbf{x})\beta_i + \varepsilon(\mathbf{x}) \end{aligned}$$

Emulator is essentially a “best fit” of the data $\mu(\mathbf{x})$ plus a pointwise model of error $\varepsilon(\mathbf{x})$ usually Gaussian

Unlike full Bayesian this only needs Expectations and Variance to be specified.

Acts as a fast surrogate for simulator & can be used by MC or LHS to generate very robust statistics AND

CAN BE CONSTRUCTED IN PARALLEL!



Bayes Linear Emulator

A spatial correlation of “error” with solution at known points is used to (non-linearly) adjust/correct the emulator’s mean and variance, which measures uncertainty

$$E(s_{BL}(x) | s_y) = E(s_{BL}(x)) + Cov(s_{BL}(x), s_y) Var(s_y)^{-1} (s_y - E(s_{BL}(y)))$$

$$Var(s_{BL}(x) | s_y) = Var(s_{BL}(x)) - Cov(s_{BL}(x), s_y) Var(s_y)^{-1} Cov(s_y, s_{BL}(x))$$

$$Cov(s_{BL}(x), s_y) \approx Cov\left(\sqrt{\varepsilon}(x), \sqrt{\varepsilon}(y)\right) = \sigma r(x-y)$$

$$r(x-y) = \exp\left(-\sum_{i=1}^{N_{in}} \theta_i (x_i - y_i)^2\right)$$

$$\varepsilon(y) = s_y - E(s_{BL}(y))$$

Error covariance model must be **positive definite**

Interpretation of Bayes Linear Emulator

- $C = \{X_1, X_2, \dots\}$ X_1, X_2 : possibilities
- X : vector space is span $[C] = \{c_0 X_0 + c_1 X_1 \dots\}$
- Inner product and Norm

$$(X, Y) = \text{Cov}(X, Y) \quad \|X\| = \sqrt{\text{Var}(X)}$$

Let $\{C\} = \{B_1, B_2, \dots, D_1, D_2 \dots\}$

P_D orthogonal projection from $[B]$ to $[D]$ i.e. $E_D(X) = P_D(X)$

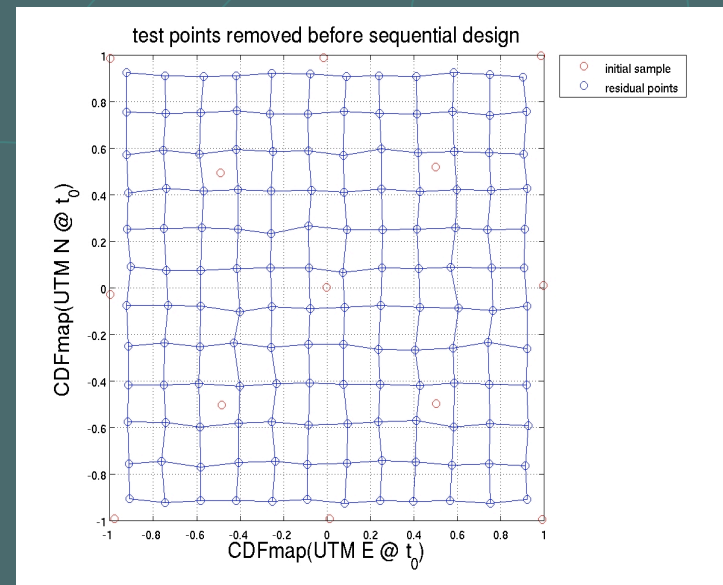
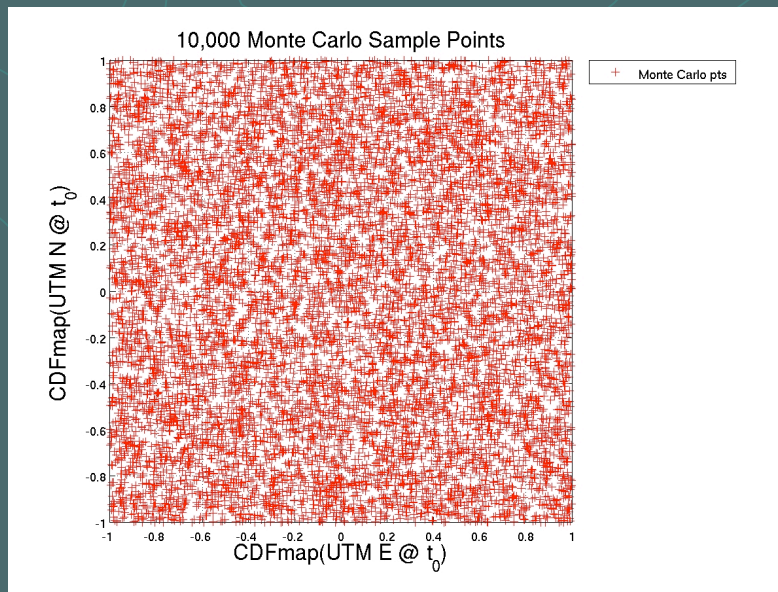
...

Bayes Linear Sequential Design

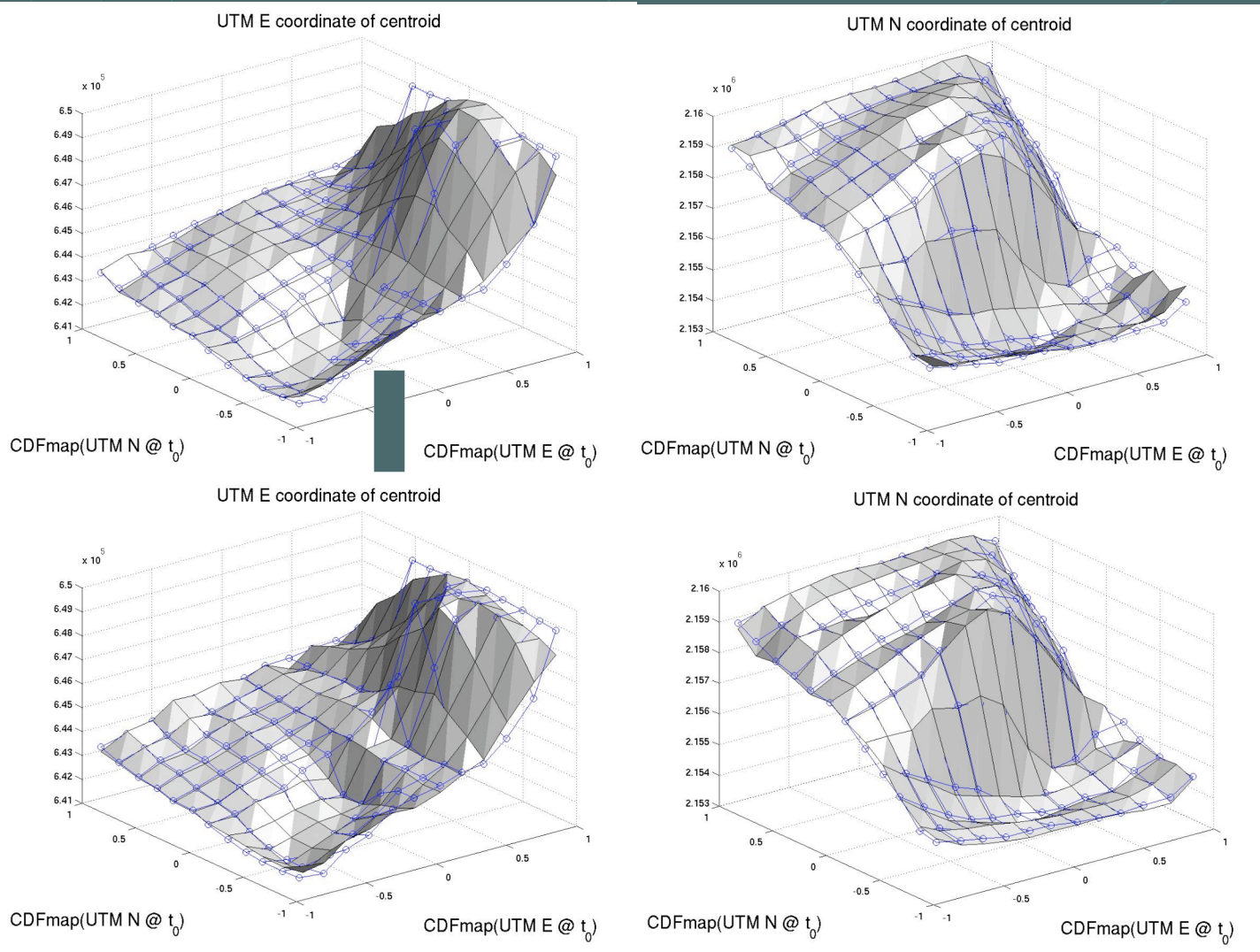
- From a set of known sample points/runs construct a Bayes linear emulator
 - Evaluate the emulator at every member of a set of candidate points; simulate at the candidate with largest adjusted Variance
 - Rebuild emulator to include the new sample; if you have enough samples, increase the order of polynomial basis functions
- Repeat steps 2 & 3 as needed
- But how do we run this in parallel?

Results: Geophysical Mass Flow

Starting location normally distributed about peak of Colima Volcano: $\sigma_{\text{East/North}} = 150$ [m]



Results: Geophysical Mass Flow

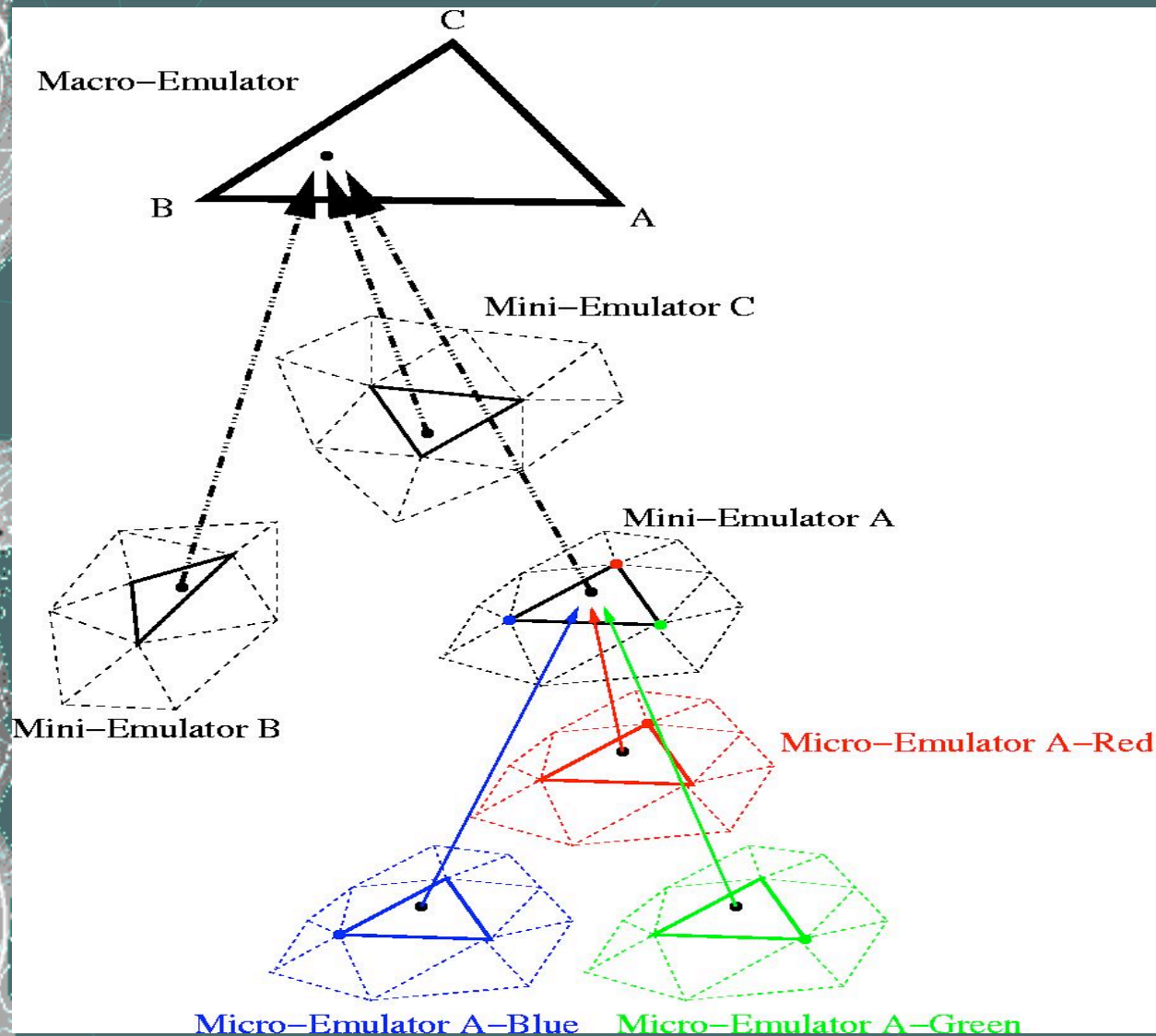


Results: Geophysical Mass Flow

Results Summary

Method	# of runs	Polynomial Order	UTM E RMS Error	UTM N RMS Error	Total RMS Error
Multi-BLM LHS	72	7	625.6	602.7	868.7
Multi-BLM Seq Des	72	7	476.1	292.0	558.5
Single-BLM with Multi-BLM Seq Des	72	7	525.8	254.9	584.1

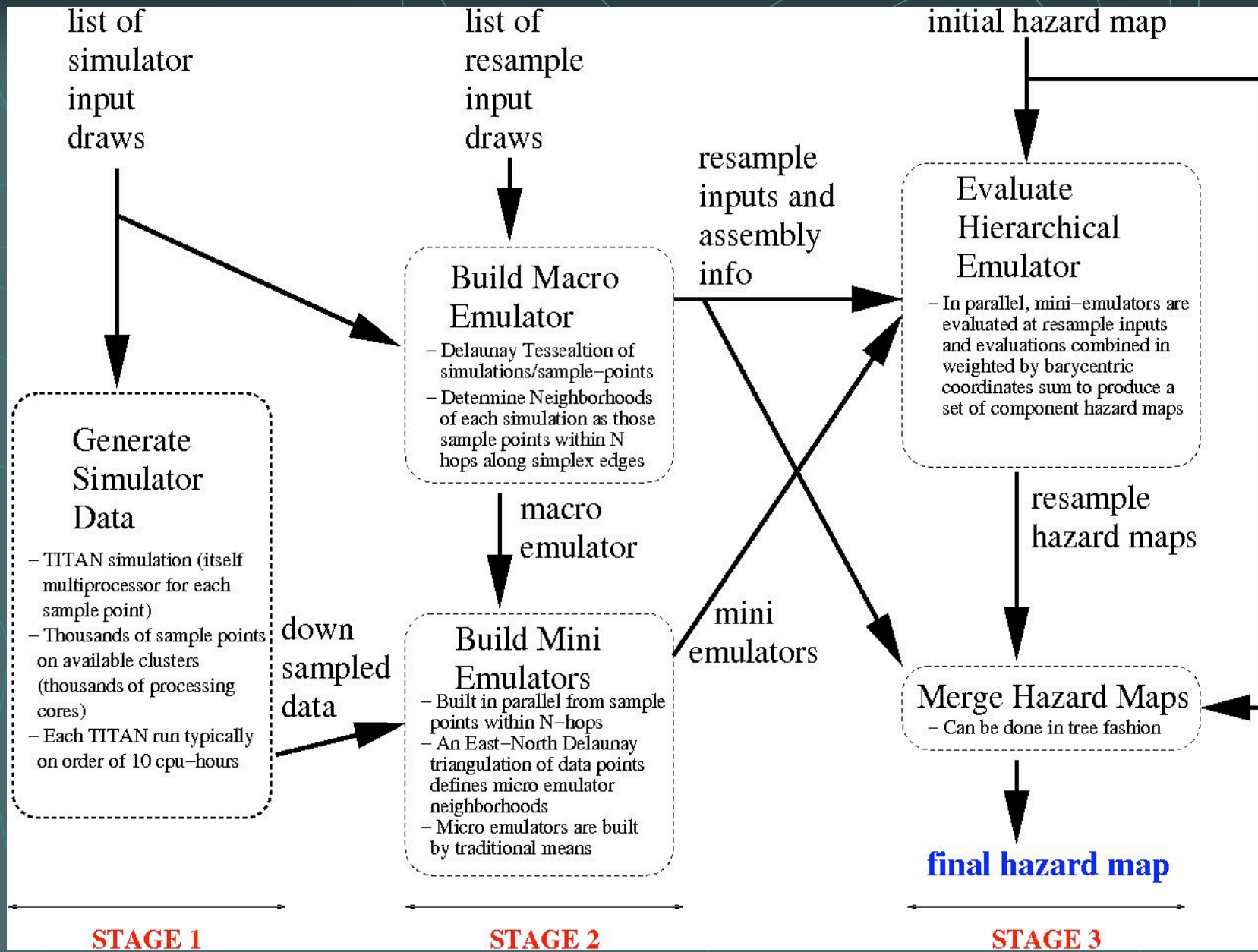
Piecewise Multi-level Emulator construction



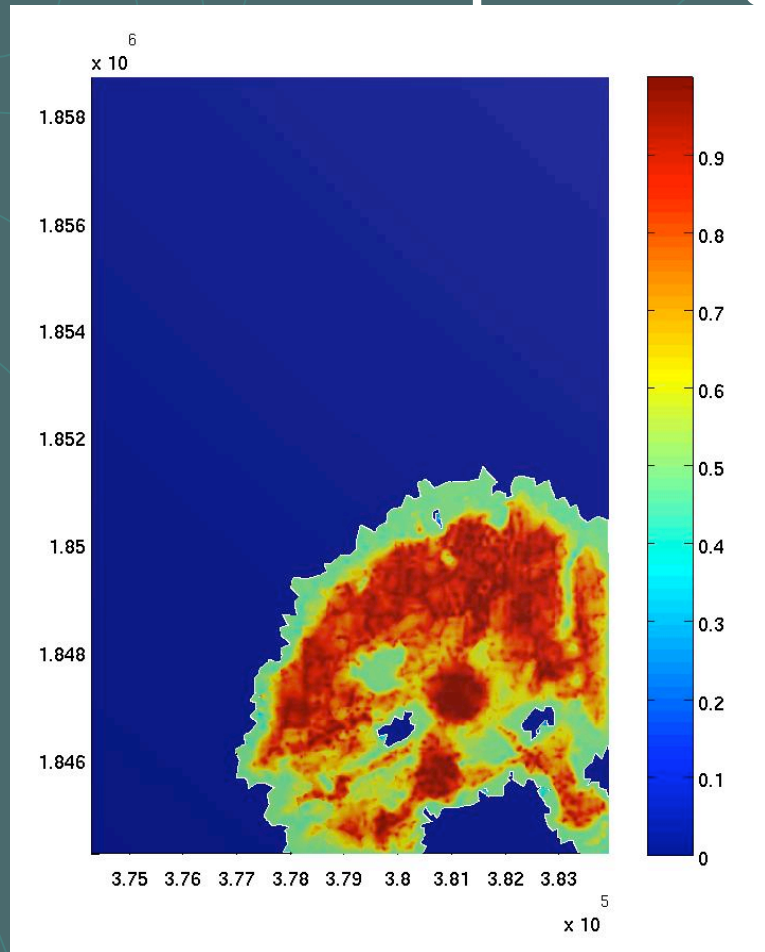
Piecewise Multilevel construction allows parallel computation of emulator!

Use correlation lengths to “cut off” contributions to a point from data at a distance

Hierarchical parallelism -- should be a good match for multi-core ...

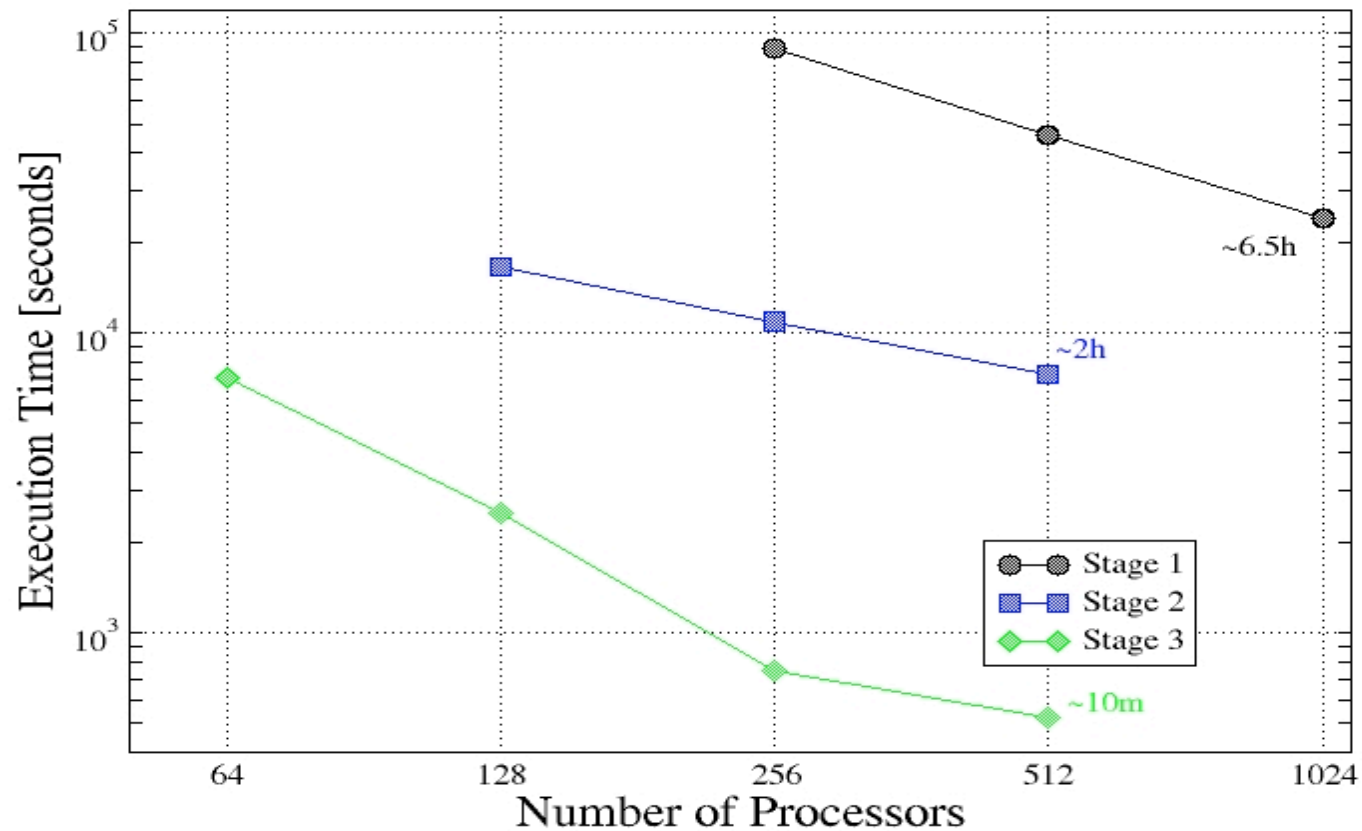


Hazard Map Using Emulators



Hazard Map at Montserrat generated by using simulator evaluations to construct multi-level emulator which is then sampled to generate the probability of flow exceeding a critical threshold in the next T time period.

128 procs running ensembles of 4 proc. simulator runs -- evaluation of hazard map by using emulator so constructed takes ~ 1 day.



Performance speedup of three stages of the hazard map workflow: Stage 1 is generation of direct simulation inputs, Stage 2 is emulator construction, and Stage 3 is emulator evaluation (only Stage 3 needs to be redone to produce a new hazard map based on the range covered by the initial direct simulations)

Current work

- Parallel adaptive construction of emulator and MCMC process to generate hazard maps
- Efficient access to GIS data from multiple cores
- Multi-threaded implementations ...