A parallel Markov chain Monte Carlo method for calibrating computationally expensive models

J. Ray, L. Swiler, M. Huang and Z. Hou

Contact: *jairay@sandia.gov*

Sandia National Laboratories, CA & NM and Pacific Northwest National Laboratory.

SAND2016-4662C

Acknowledgments: Funded by DoE/SC/ASCR

Overview

- Aim
  - Construct a multi-chain Markov chain Monte Carlo (MCMC) method
  - Speed up the solution of statistical inverse problems which involve an expensive engineering or scientific model
- Motivation
  - Statistical inverse problems can be used to estimate model parameters from experimental data
    - Parameters estimated as PDFs; quantifies uncertainty in the estimate
- Technical challenges
  - MCMC requires $O(10^4)$ model invocations serially – difficult
    - Multi-chain might spread the sampling burden on *m* chains
  - Multi-chain MCMC is rare – little previous literature
    - Lots of theory & implementation on *single* chain MCMC

Outline

- Statistical inverse problem, specifically Bayesian
- How to solve them using MCMC
- What is adaptive MCMC – why needed and issues
- How to go parallel with MCMC
- Empirical data on correctness and savings on wall-clock time
- Results with calibration of the Community Land Model
  - Land component of the Community Earth System Model

Sandia National Laboratories

Statistical inverse problems - 1

- Consider a model that produces $\mathbf{y} = \mathcal{M}(\mathbf{x}; \mathbf{p})$
  - $\mathbf{p}$ are model parameters, $\mathbf{x}$ is input such as time, location etc.
  - They are unknown but we may have a prior belief $\pi(\mathbf{p})$ e.g., bounds on their values
- Consider observational data $(\mathbf{y}^{(obs)}, \mathbf{x})$
  - The simplest way to estimate $\mathbf{p}$ is via least-squares fitting
    - $\mathbf{y}^{(obs)} = \mathcal{M}(\mathbf{x}, \mathbf{p}) + \varepsilon, \varepsilon = \{\varepsilon_i\}, i = 1 \ldots N_{obs}$
    - Minimize $|| \varepsilon ||_2^2$ w.r.t. $\mathbf{p}$ i.e. minimize $|| \mathbf{y}^{(obs)} - \mathcal{M}(\mathbf{x}, \mathbf{p}) ||_2^2$
- Estimates of $\mathbf{p}$ so obtained provide no estimate of the uncertainty
  - In case there are multiple minima, you could get a wrong $\mathbf{p}$

Sandia National Laboratories

Statistical inverse problems - 2

- Consider a model for ε, e.g., $\varepsilon \sim N(0, \Gamma)$
  - i.e. there is a belief that for good values of **p**, the data – model mismatch will be near 0
- Then, for any **p**, one can compute and error and the likelihood L(: | :) of **p**, given observations $\mathbf{y}^{(obs)}$

$$\mathrm{L}(\mathbf{y}^{(obs)} \mid \mathbf{p}) \propto \exp\left(-\left(\mathbf{y}^{(\mathrm{obs})} - \mathcal{M}(\mathbf{x};\mathbf{p})\right)^T \Gamma^{-1} \left(\mathbf{y}^{(\mathrm{obs})} - \mathcal{M}(\mathbf{x};\mathbf{p})\right)\right)$$

- Bayes rule:

$$f(\mathbf{p} \mid \mathbf{y}^{(obs)}) \propto \mathrm{L}(\mathbf{y}^{(obs)} \mid \mathbf{p})\, \pi(\mathbf{p})$$

- The posterior density f(: | :) is arbitrary
  - Take samples from it and histogram samples
  - Done using MCMC

What is MCMC?

- A way of sampling from an arbitrary distribution
  - The samples recover the distribution (typically plot marginals via histograms)
- Efficient and adaptive
  - Given a starting point (1 sample), the MCMC chain will sequentially find the peaks and valleys in the distribution and sample proportionally
- Ergodic
  - Guaranteed that samples will be taken from the entire range of the distribution
- Drawback
  - Generating each sample requires one to evaluate the expression for the likelihood L(: | :)
  - Not a good idea if L(: | :) involves evaluating a computationally expensive model

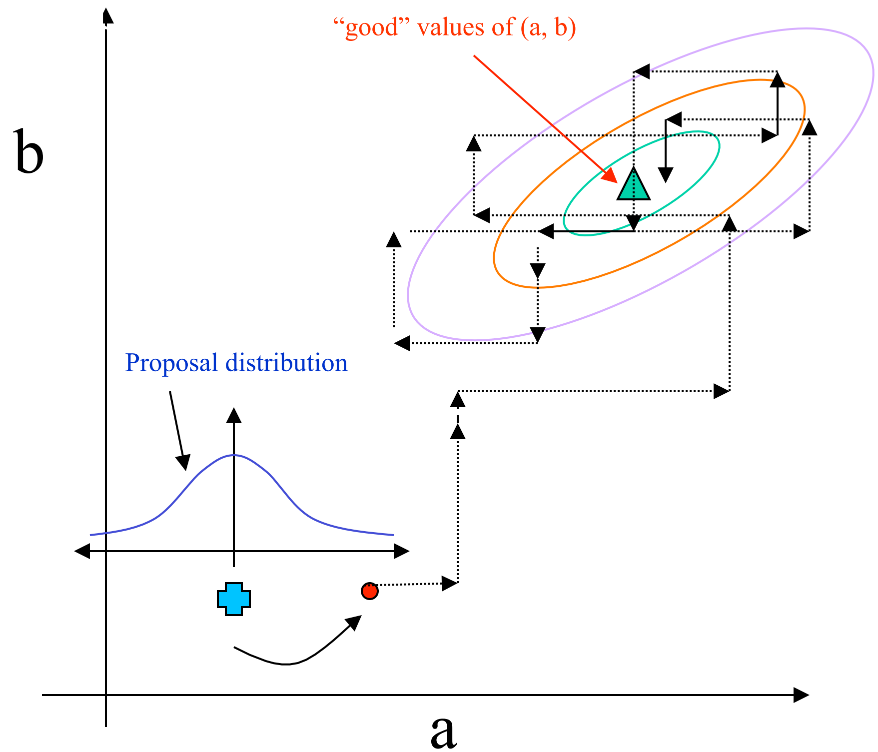
An example, using MCMC

- Given: (Y^{obs}, X), a bunch of n observations
- Believed: y = ax + b
- Model: $y_i^{obs} = ax_i + b_i + \varepsilon_i$, $\varepsilon \sim \mathcal{N}(0, \sigma)$
- We also know a range where a, b and σ might lie
  - i.e. we will use uniform distributions as prior beliefs for a, b, σ
- For a given value of (a, b, σ), compute "error" $\varepsilon_i = y_i^{obs} - (ax_i + b_i)$
  - Probability of the set (a, b, σ) = $\Pi \exp(- \varepsilon_i^2/\sigma^2)$
- Solution: π (a, b, σ | Y^{obs}, X) = $\Pi \exp(- \varepsilon_i^2/\sigma^2)$ * (bunch of uniform priors)
- Solution method:
  - Sample from π (a, b, σ | Y^{obs}, X) using MCMC; save them
  - Generate a "3D histogram" from the samples to determine which region in the (a, b, σ) space gives best fit
  - Histogram values of a, b and σ, to get individual PDFs for them
  - Estimation of model parameters, with confidence intervals!

Sandia National Laboratories

MCMC, pictorially

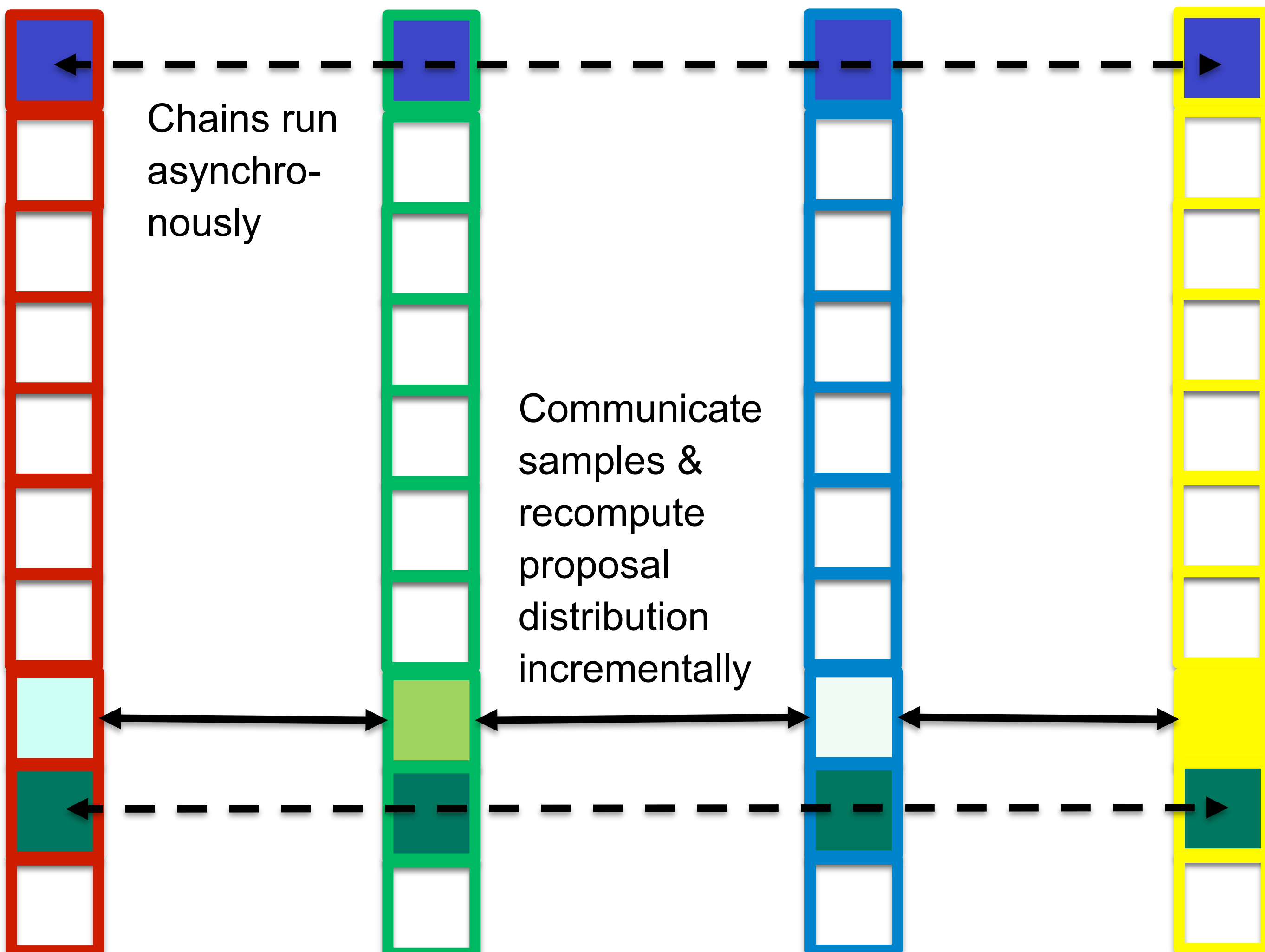
- Choose a starting point, $p_0 = (a_{curr}, b_{curr})$
- Propose a new a, $a_{prop} \sim \mathcal{N}(a_{curr}, \sigma_a)$
- Evaluate $\pi (a_{prop}, b_{curr} \mid \ldots) / \pi (a_{curr}, b_{curr} \mid \ldots) = m$
- Accept a_{prop} (i.e. a_{curr} <- a_{prop}) with probability min(1, m)
- Repeat with b
- Loop over till you have enough samples
- Two issues
  - Where do you start?
  - How do you choose a proposal distribution?



Multi-chain, adaptive MCMC

- **Problems with MCMC**
  - *Sampling cost:* Many samples needed; each sample leads to 1 model evaluation
  - *Poor proposals:* If proposal distribution is sub-optimal, most proposals will be rejected
  - *Bad start:* What's a good place to start
- **Solutions**
  - *Sampling cost:* Distribute sampling over *m* chains
  - *Poor proposals:* adaptive Metropolis-Hasting sampling
    - Periodically, use samples collected to compute a multivariate Gaussian approximation to f(: | :)
    - Inflate its variance and use it as a proposal
    - Only works if you have some samples to work with
  - *Bad start:* Have *m* chains start from an over-dispersed set of $\mathbf{p}_0$

Addressing sampling cost

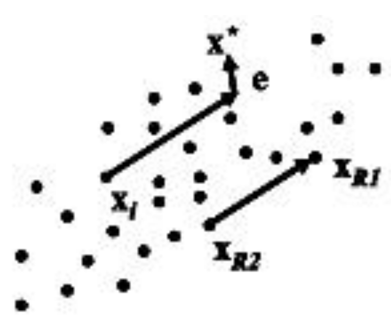


Generation i

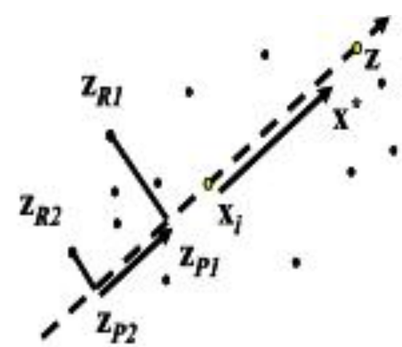
Each generation consists of
1. proposal generation
2. model run
3. accept/reject of proposal

Addressing bad starts

- When there aren't enough samples, how to make a good proposal distribution?
  - Use genetic algorithm (Differential Evolution) to collect a few good samples
  - Use parallel and snooker updates to construct proposals



Parallel



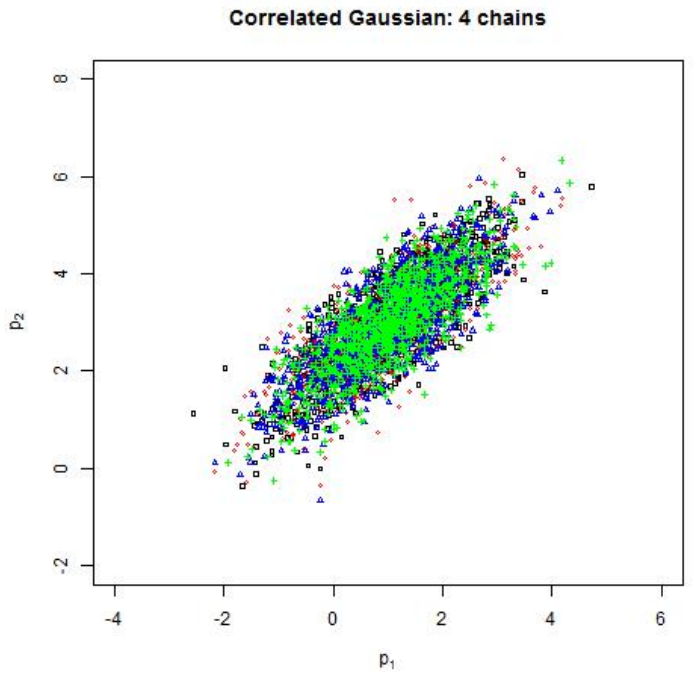
Snooker

  - Switch to adaptive Metropolis-Hastings when we have a few good samples

Pictures taken from: C. J. F. ter Braack and J. Vrugt, "Differential Evolution Markov Chain with snooker update and fewer chains", Statistical Computing, 2008

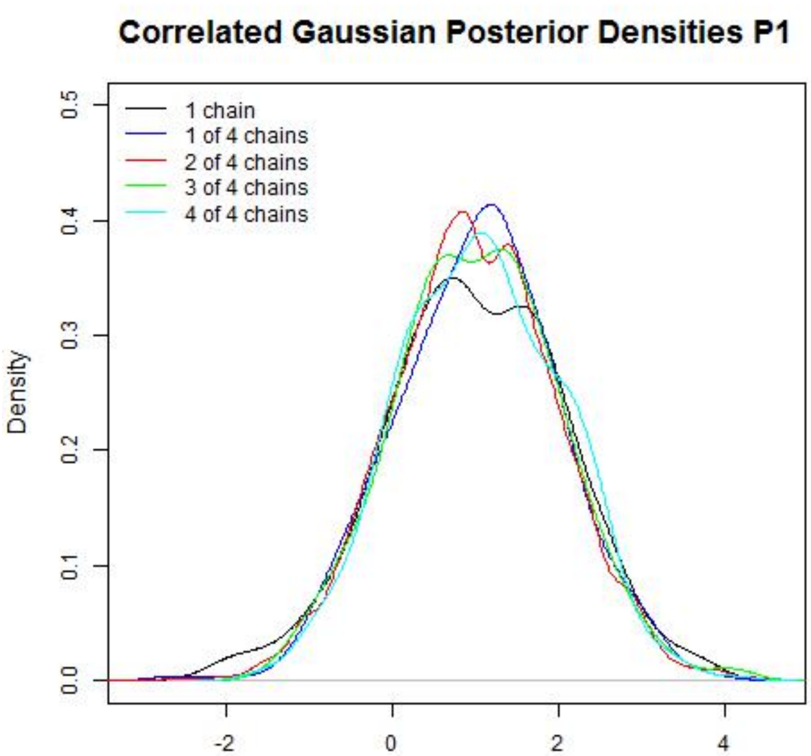
Performance

- Do multichain MCMC get us accurate PDFs with smaller wall-clock time than a 1 chain MCMC?
- Test
  - Pick a bivariate Gaussian with mean (1, 3) and correlation of 0.8
  - Run a 1-chain and 4-chain MCMC sampler on it
  - Explore region [-5, 8] x [-5, 8]
- Questions
  - Are the marginal distributions correct?
  - Are estimates of 5th, median and 95th percentiles correct?
  - We have analytical solutions

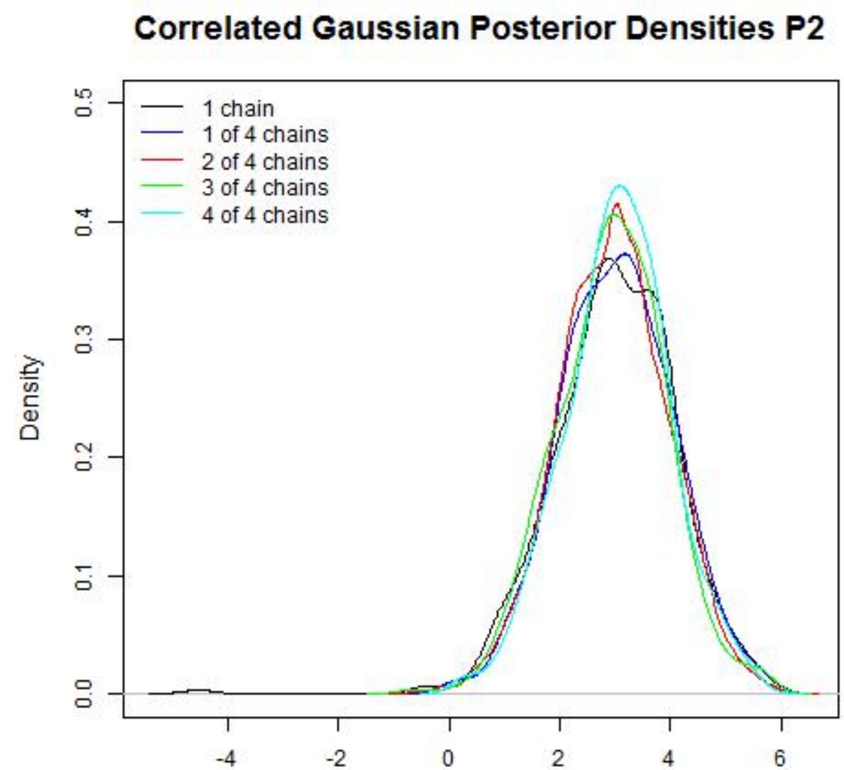


Samples colored by chain

Marginal distributions



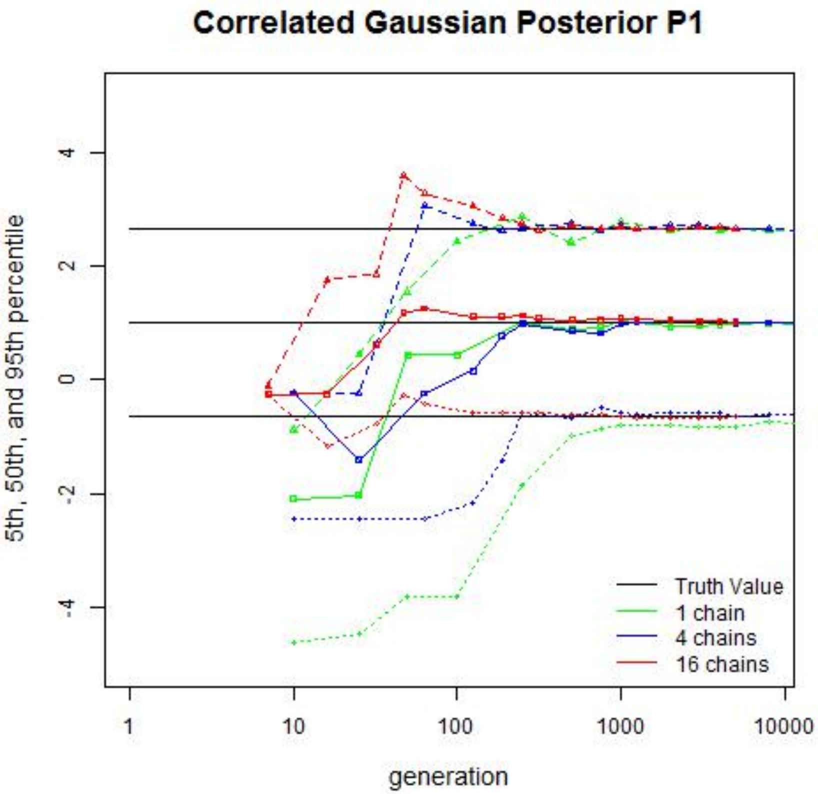
Variable p_1

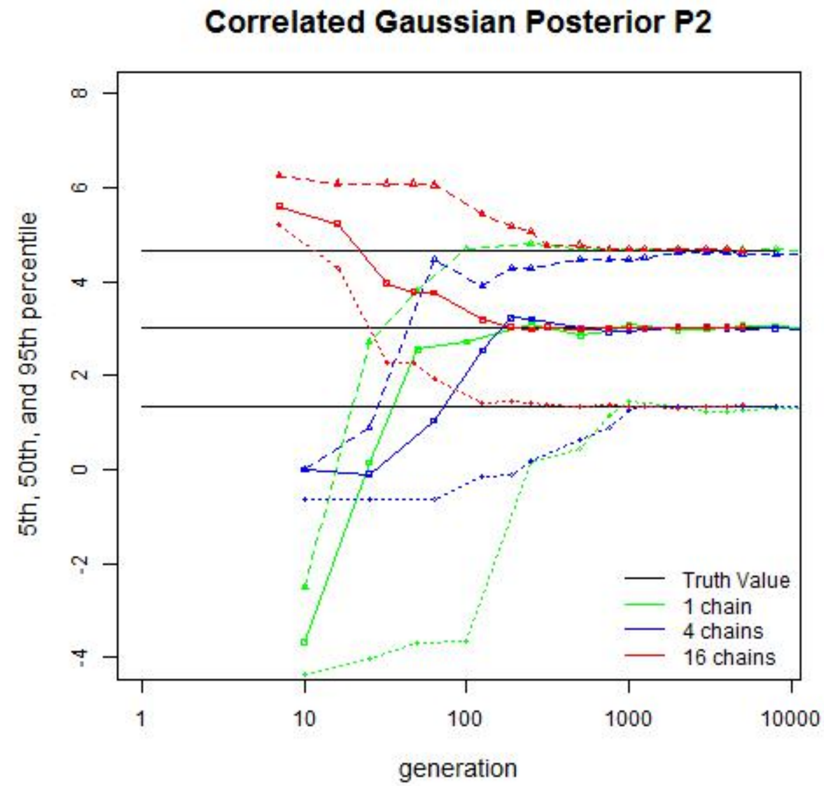


Variable p_2

- Chains from the 4-chain MCMC produce the same PDFs as the conventional 1-chain MCMC

Convergence

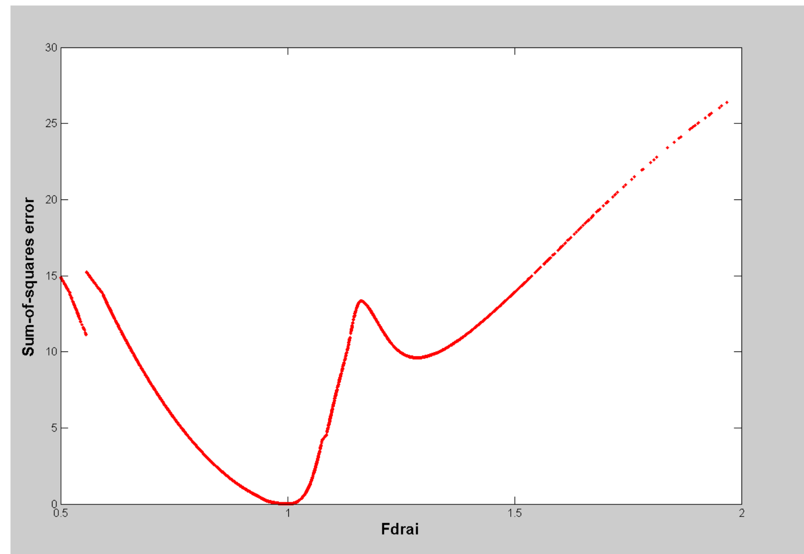




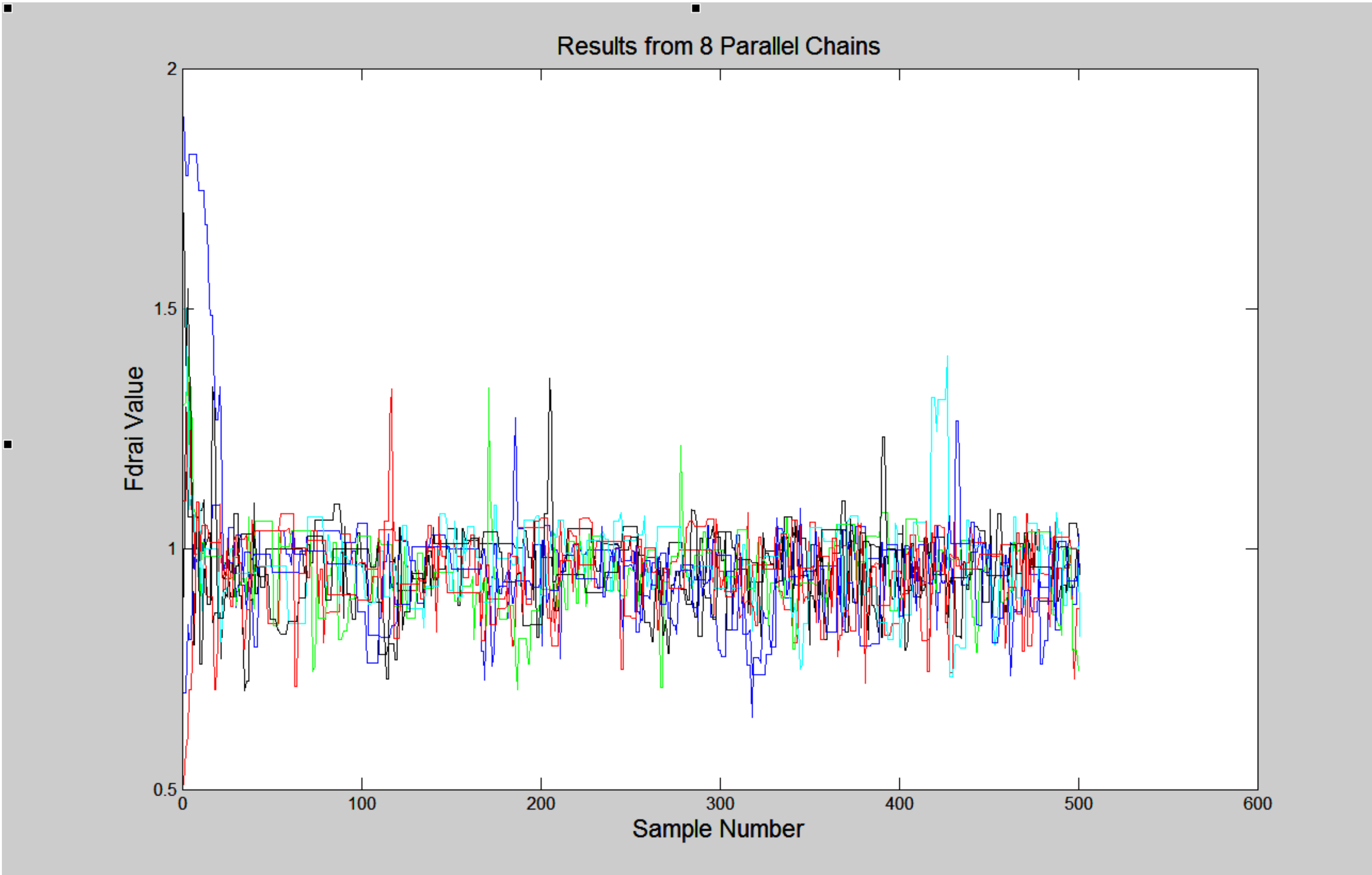
- Percentiles are computed by pooling together *g* generations of samples collected by *m* chains (i.e., *g* x *m* samples)
- 4-chain MCMC converges faster for tails of the PDF

Practical use – calibrate CLM

- CLM – Community Land Model
  - The land component of Community Earth System Model
  - Used in climate change simulations
  - Computationally expensive
  - Simulating 4 years for each grid-cell takes about 1 hour
- Our aim – check for correctness
  - Use multichain MCMC to calibrate CLM for 1 site (1 grid-cell)
    - ARM/Southern Great Plains site, 2003 meteorology
    - Use latent heat flux as observable ($\mathbf{y}^{(obs)}$)
  - Calibrate 1 CLM hydrological parameter (F_{drai}); synthetic data using F_{drai} = 1
  - Problem as a complex likelihood

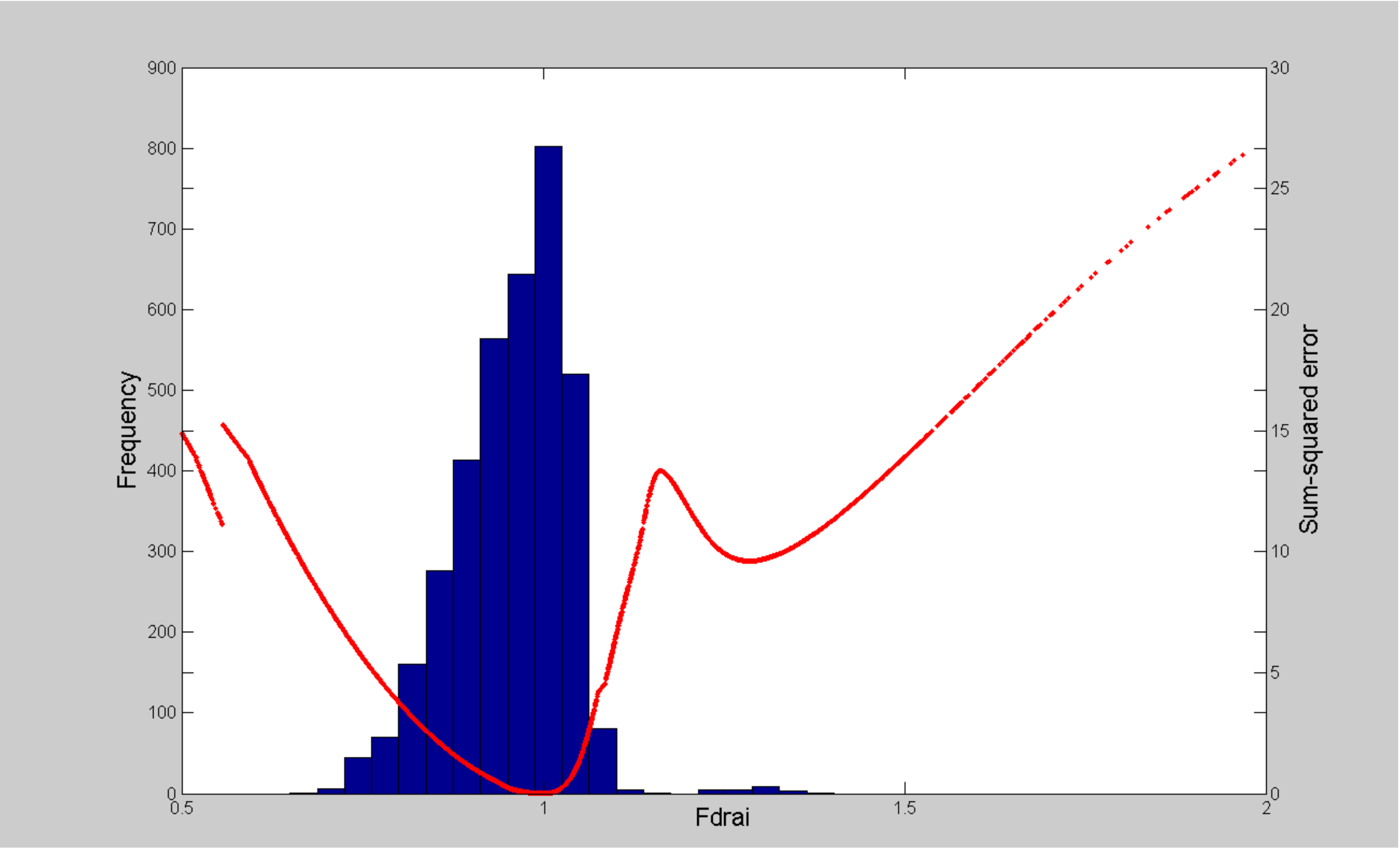


8-chain MCMC



- Chains have settled down to the same value of F_{drai}

PDF of F_{drai}

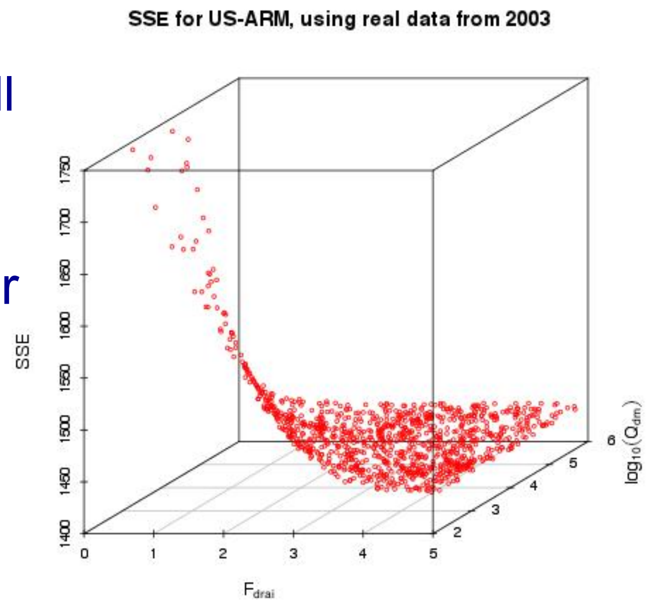


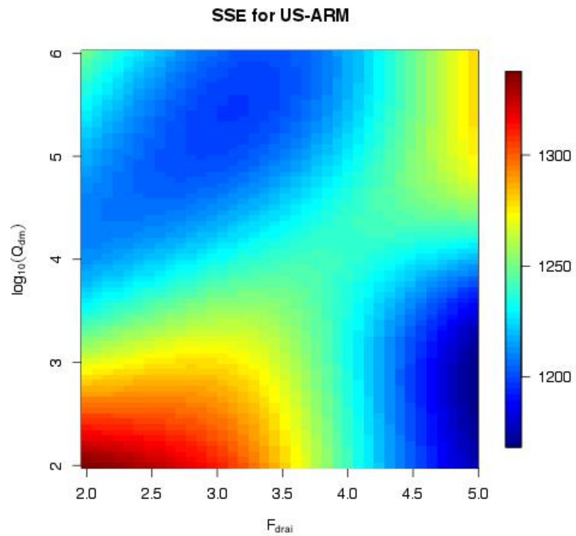
- Samples of Fdrai peak at 1, the correct position
- Cleanly misses local minima at 1.25 (few samples)

Sandia National Laboratories

CLM calibration with real LH observation

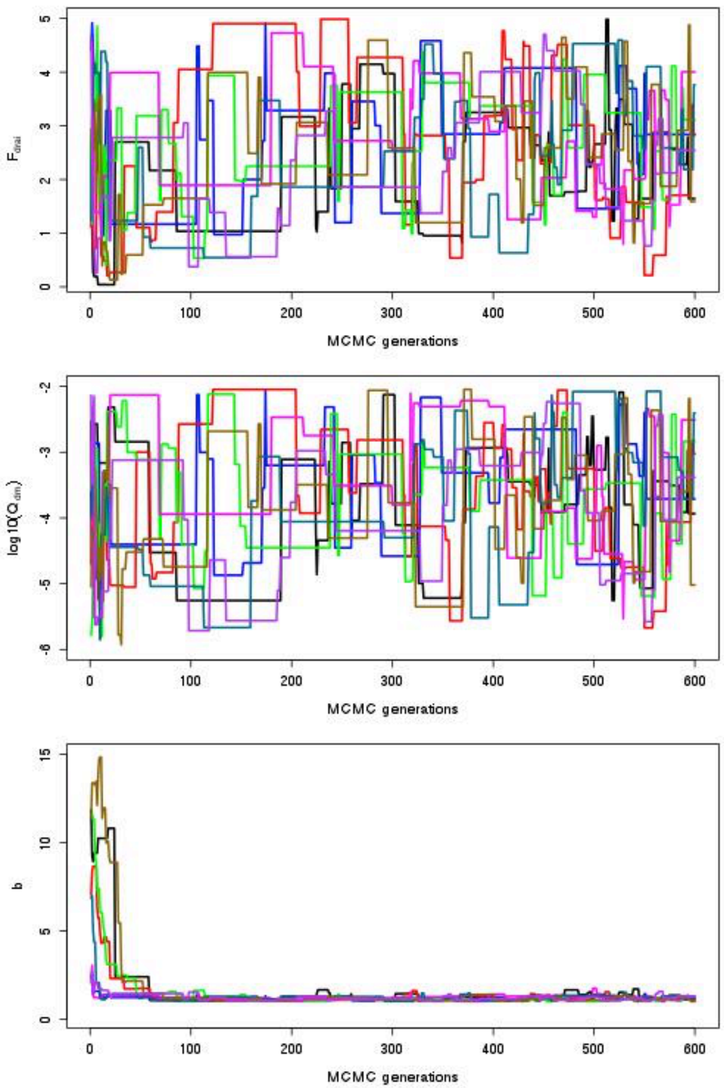
- Calibrate: F_{drai}, $\log(Q_{dm})$, b
  - 3 parameters, 12 observations – there will be uncertainty in the estimates
    - The PDF is required
- Use observations from ARM/SGS site for 2003
  - Observations are latent heat fluxes
  - Averaged to their monthly value





- The likelihood is flat near the minimum error point
  - The chains will wander

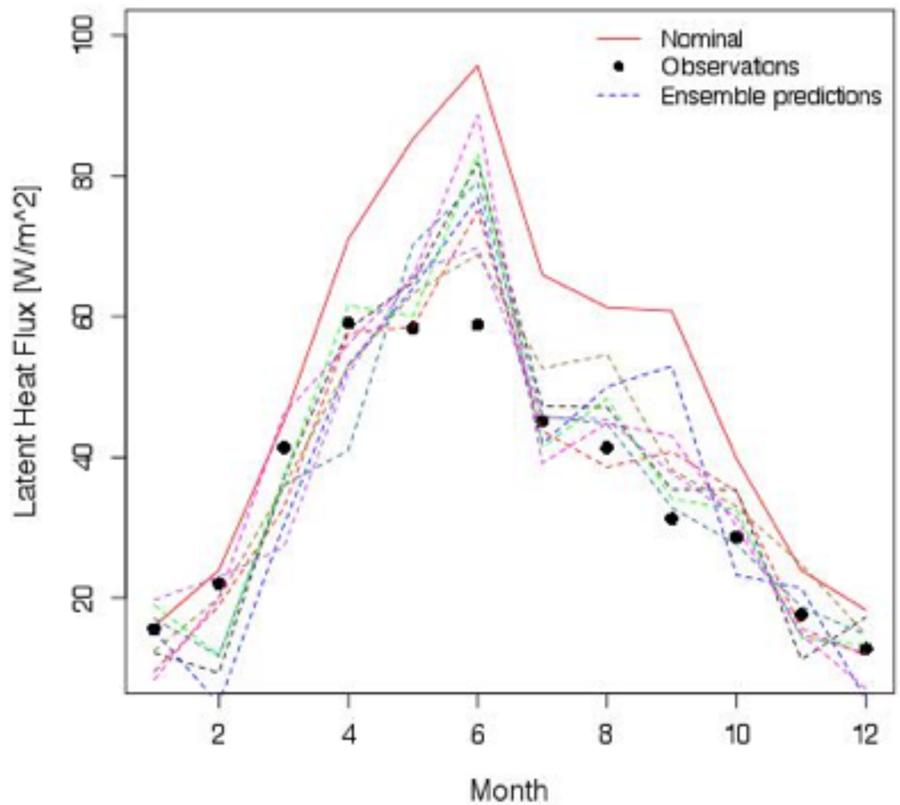
Evolution of the chains



- It's still running …
- The chain for b has converged
- The other chains are still wandering
- Far from convergence @ 600 generations

Predictions with samples

- Pick samples of (F_{drai}, $\log(Q_{dm})$, b) from chains
- Run CLM and produce predictions of LH
- Compare with predictions produced by default ("nominal") parameter values
- Compare with experimental data
- We're better than the predictions obtained with default parameter values



Conclusions

- We have a parallel multichain MCMC method implemented
  - It's being used to solve statistical inverse problems
    - Specifically, to calibrate computationally expensive models
  - Parameters are estimated as PDFs; captures uncertainty
- The multichain MCMC
  - Converges to true value of the parameters
  - Cuts down wall-clock time, especially when resolving tails of posterior distribution
- It is being used to calibrate the CLM
  - Has already been used to reconstruct moisture levels using GPR measurements
    - 10 parameters to be estimated, 20 chains
  - Can be applied to calibration of engineering models