Generating conditional realizations of graphs and fields using Markov chain Monte Carlo

J. Ray

jairay [at] sandia [dot] gov
Sandia National Laboratories, Livermore, CA

Joint work with
A. Pinar, C. Seshadhri, B van Bloemen Waanders and S. A. McKenna,
Sandia National Laboratories

Sandia is a multiprogram laboratory operated by Sandia Corporation, a Lockheed Martin Company, for the United States Department of Energy's National Nuclear Security Administration under contract DE-AC04-94AL85000.

Statistical research in Sandia

- A significant effort, with multiple foci
  - Estimating risk of component/system failure in nuclear weapons
  - Statistical calibration of scientific (climate) and engineering (weapons) models
  - Also, propagation of parametric uncertainty through scientific / engineering models (i.e., research in sparse sampling methods)
  - Most "well-baked" methods deployed via DAKOTA (http://dakota.sandia.gov); LGPL license; widely used in academia and some industries
- Markov chain / random walk methods are employed in
  - Statistical inference of fields from sparse observations e.g., estimation of material properties from experimental data
  - Generation of networks (sparse matrices) conditioned on matrix properties

Outline of the talk

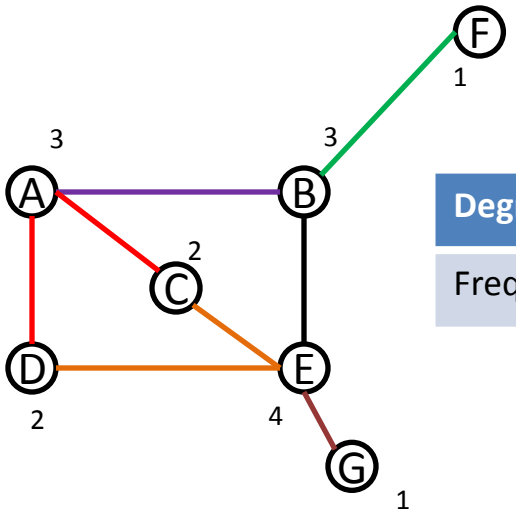
- Topic I: Generation of independent networks with prescribed properties using Markov chains
  - Motivation: generating “sanitized” versions of sensitive networks, for experimentation and study
  - Novelty: A collection of graphs which are independent, but which share a network property specified by the user
- Topic II: Statistical inference (inverse problem) of permeability fields from sparse observations
  - Motivation: Conditional construction of material property fields from sparse observations
  - Novelty: infer statistics of material structures too fine to be resolved by a grid

Topic I - Generation of independent graphs

- Aim: Generate a set of independent graphs that have the same joint degree distribution (JDD)
  - Given: A procedure that can rewire a graph without violating the prescribed joint degree distribution
- Motivation
  - Being able to generate synthetic graphs which are similar in some ways, and diverse in others, is necessary for experimentation and study
  - Many types of networks e.g., email traffic, critical infrastructure etc. have privacy and security concerns and cannot be handed out for study
  - Graph rewiring algorithms (graph models / generators) are common, but how to put them into practical use?

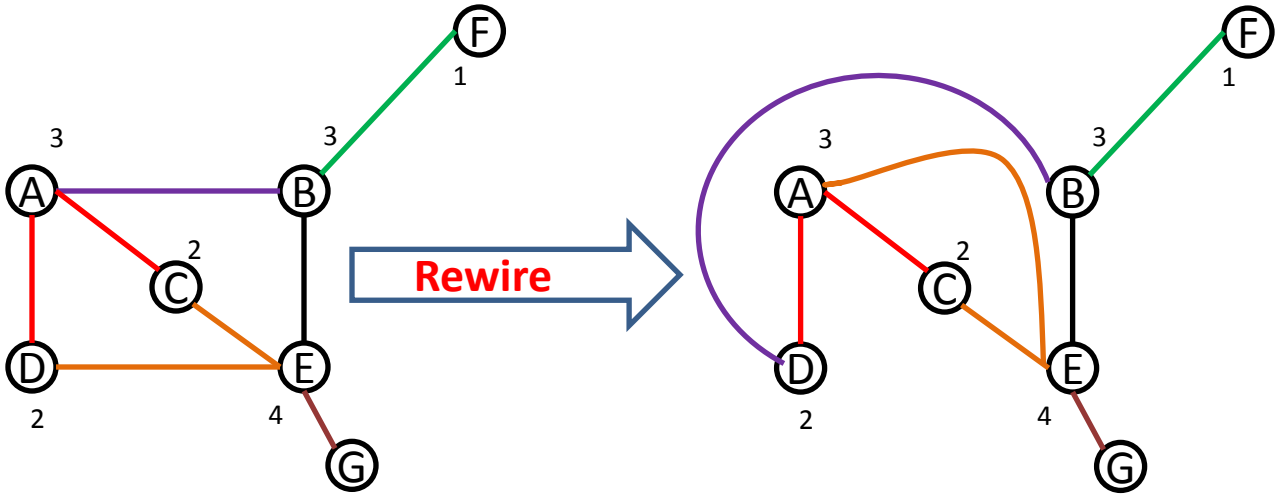
Definitions

- G(V, E)
  - |E| = # of edges
- Degree distribution
  - Histogram of vertex degrees
- Joint degree distribution
  - Joint distribution
- Rewiring
  - Reconnection of edges of a graph



Degree	1	2	3	4
Frequency	2	2	2	1

Degree distribution

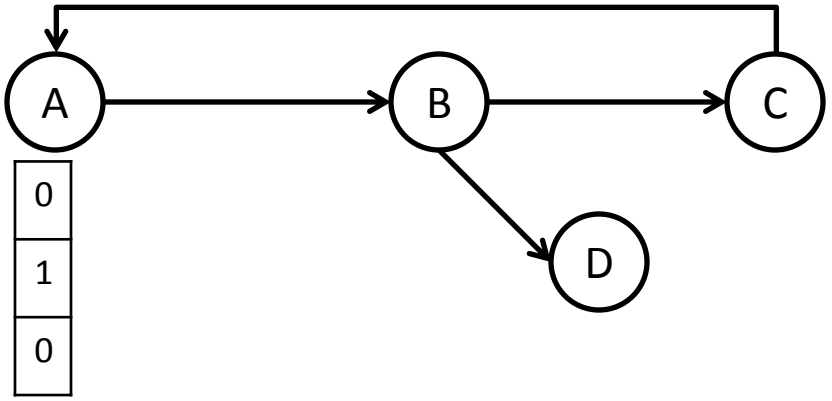


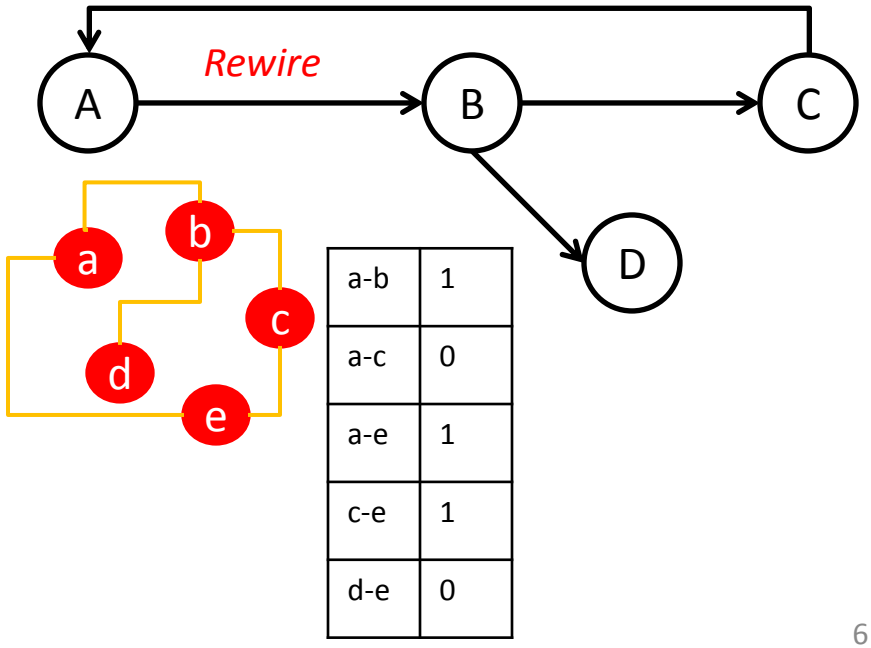
Degree	1	2	3	4
1	0	0	1	1
2	0	0	2	2
3	1	2	1	1
4	1	2	1	0

Joint degree distribution

Markov chain *of* graphs

- A Markov chain on discrete variables
  - Called random walk on a graph
- In our case, each state is also a graph
- In our talk, "graph" will refer to the state (red-and-yellow graph)
  - And not the graph on which the Markov chain runs (black-and-white graph)





Techniques for rewiring

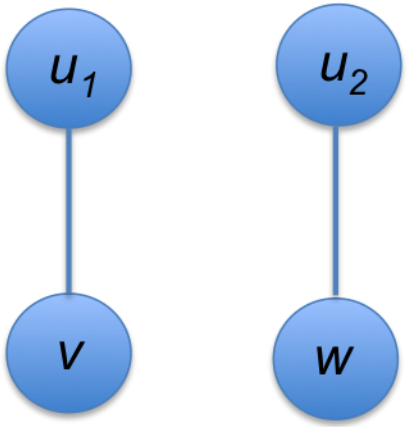
- Graph rewiring techniques exist
  - Preserve degree distribution or joint degree distribution
  - Applying this technique repeatedly leads to a set of samples from the uniform distribution of graphs (with the prescribed property)
- Shortcoming – the input to the procedure is a graph from the target distribution, not an arbitrary graph
  - The procedure generates a new sample, given an old sample.
  - Generally, the new sample is almost identical to the input – few graph edges change
  - The procedure produces a stream of *correlated* graphs
- Problem: How to get a stream of independent graphs?

How are independent graphs generated?

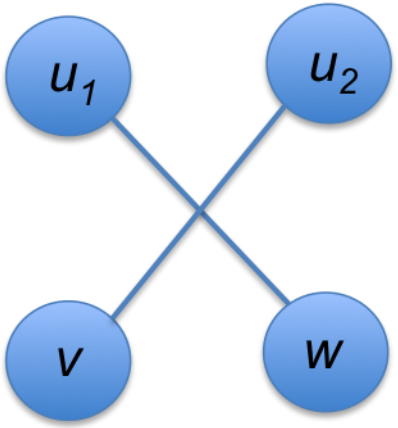
- Using Markov chains, we need to run *N* steps (to forget the starting point) before preserving the last one as a sample
  - What is *N*?
- Theoretical upper-bounds on *N* are huge
  - Practically, by choosing *N*, the number of MC steps to run arbitrarily
- We need a principled way of choosing *N*

The JDD-preserving rewiring technique

Step 1: Pick an edge (u_1,v), and pick one of its vertices, e.g., u_1



Step 2: Pick another edge *(u,w)*, such that $d(u_1)=d(u_2)$ *or* $d(u_1)=d(w)$



Step 3: Swap edges

- Stanton & Pinar, *ACM J. Expt. Algorithmics*, to appear
- Per invocation, only 1 pair of edges change
- Requires that the input graph obeys the prescribed JDD
- Problem of periodic edge appearance

Features of this chain

- Is a variant of a Markov chain Monte Carlo method
  - But there is no complicated likelihood expression
  - # of nodes, edges and JDD are preserved from graph to graph
- The posterior is a uniform distribution of graphs
- Consecutive graphs are very correlated
  - In fact, they only differ by 1 pair of edges
- In case the nodes of the graph are labeled
  - Each edge describes a binary time series $\{Z_t\}$, t = 1 … *N*
- To generate independent graphs, need to estimate *N* for which starting and ending graphs are “different”
  - i.e., the Markov chain converges to its stationary distribution

Mixing of the MCMC chain

- Stanton & Pinar analyzed the time-series $\{Z_t\}$, t = 1 … *K* of edges for mixing
  - K was a large number >> |E|
  - The autocorrelation of $\{Z_t\}$ decreased with lag, initially exponentially, and stabilized at a low "noise" level
  - Indicates that one could obtain independent samples by thinning a long chain, using a sufficiently large lag (set it equal to *N*)
    - But requires one to run the chain first and do the autocorrelation analysis
- Would ideally like a simple expression for *N*

Layout of the talk

- Is about estimating *N* that will lead to independent realizations
- Will create a closed-form expression for *N*
  - Exploits the fact that JDD is preserved
  - Assumes $\{Z_t\}$ for an edge is independent of others
  - Has a user-defined parameter
- Will check closed-form expression using a purely data-driven method
  - No use of JDD is made
- These are necessary, not sufficient, conditions for independence
- Will work on the time-series of edges $\{Z_t\}$

Model for estimating N – Method A

- Each edge can assume 2 states, {0, 1}
- Its evolution as $\{Z_t\}$ can be described with as a Markov chain with transition probabilities $\{\alpha, \beta\}$
- One can develop expressions for $\{\alpha, \beta\}$ using the fact that JDD is held constant
  - α scales as $1/|E|^2$; β scales as $1/|E|$; $|E|$ = number of edges in graph
  - Details in Ray, Pinar & Seshadhri, "Are we there yet?", arXiv:2012.3473
  - After *N* steps, the difference between stationary and realized distributions is ε

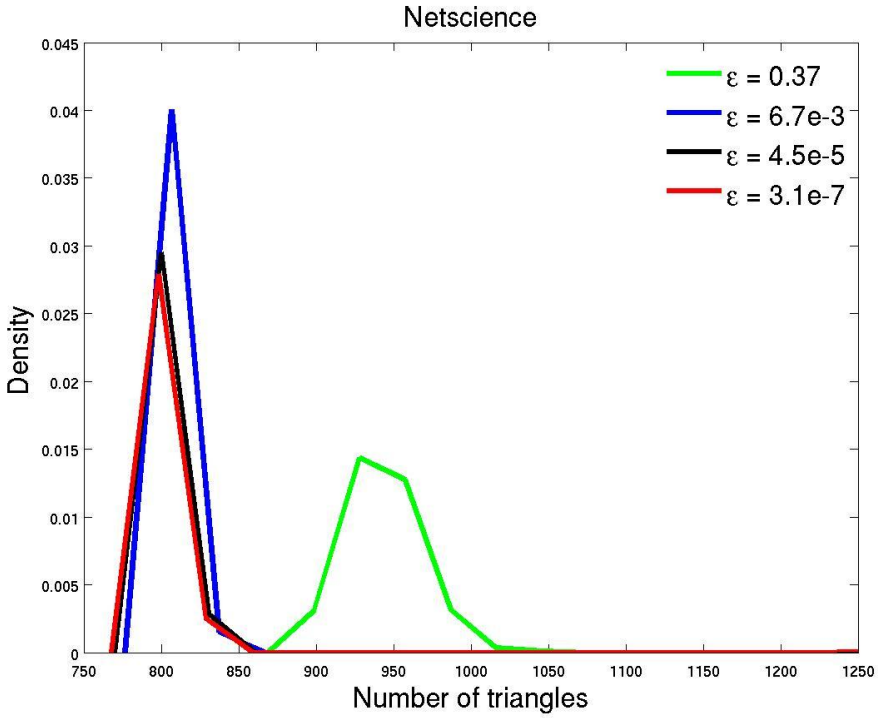
$$N = \frac{\ln(1/\varepsilon)}{\alpha+\beta} \leq |E| \ln\left(\frac{1}{\varepsilon}\right)$$

Estimating ε

- What ε should we use?
  - We are interested in the distribution of certain graphical parameters associated with a prescribed JDD
  - Max. eigenvalue of graph, diameter, # of triangles etc
- Pick various values of ε, and corresponding *N*
- Run *M* separate instances of the MCMC to generate *M* independent samples
  - Each chain runs N steps to "forget the initial graph" and the last sample is preserved
  - When the distributions stop changing with *N* (and have min variance) we have independent samples
- Check this with realistic graphs
  - Co-authorship in network science (|V| = 1461, |E| = 5484) and western states power network (|V| = 4941, |E| = 13,188)

Distribution # of triangles – co-authorship graph in network science

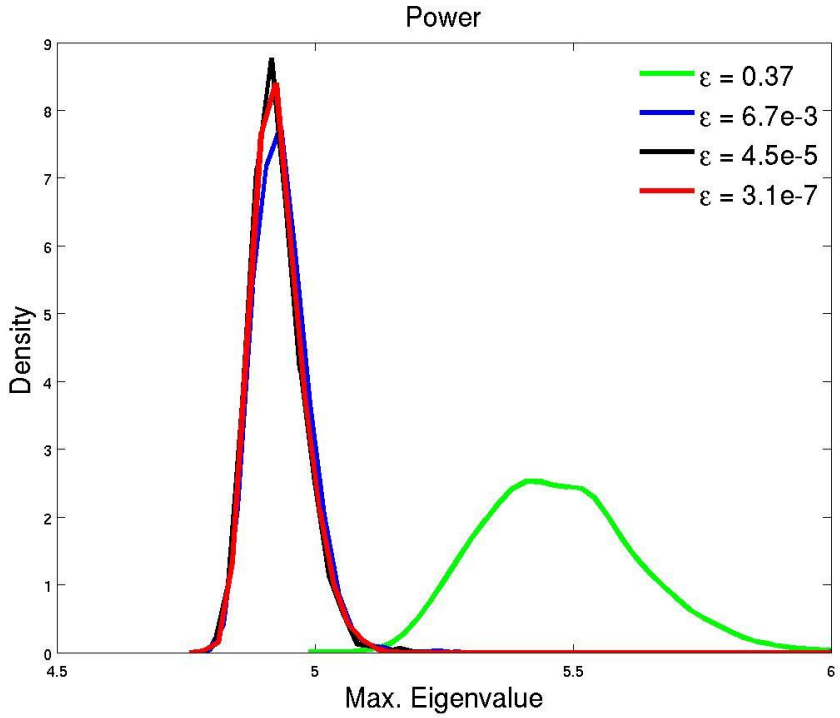
- $|V| = 1461$, $|E| = 5484$
- ε values correspond to $|E|$, $5|E|$, $10|E|$ and $15|E|$ MCMC steps
- Repeat 1000 times to generate 1000 graphs
  - Calculate # of triangles in each graph; plot distribution
  - Compare distributions (PDF) from each value of ε
  - Convergence?



$N = 10|E|$ seems to work

Distribution of max. eigenvalue – western states power grid

- |V|=4941, |E|=13188
- ε values correspond to |E|, 5|E|, 10|E| and 15|E| MCMC steps
- ε ~ 5e-5 (*N* = 10|E|) seems OK
- Henceforth, we'll use *N* = 10|E|



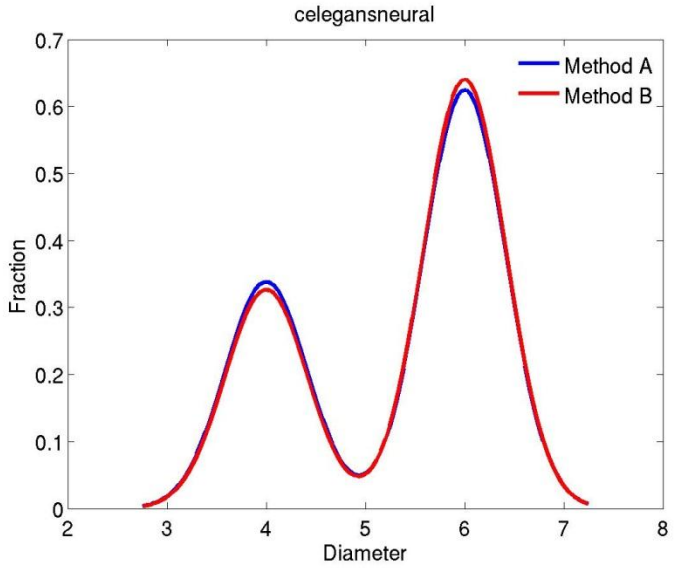
Checking the model (Method B)

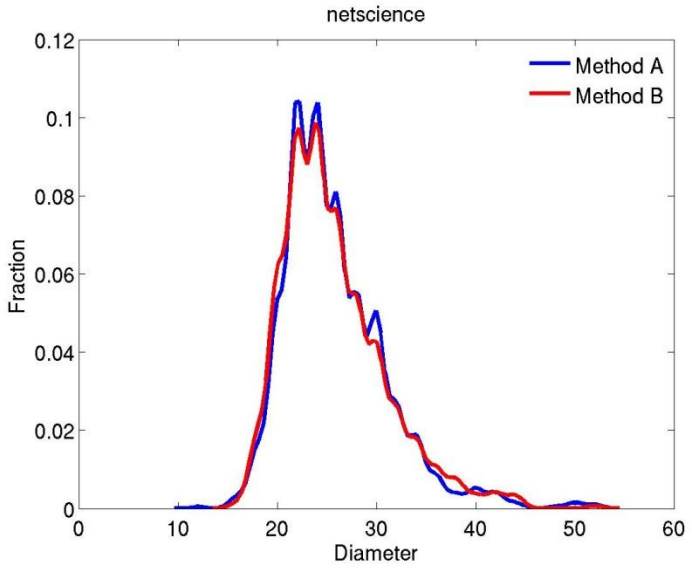
- The expression for *N* came from modeled values of α, β
  - These are approximate (e.g., assumption of independence of edges)
  - We can check by empirically calculating of α, β from the data $\{Z_t\}$
- We adopt the method in Raftery & Lewis, 1992
  - Run the MCMC very long, ~10,000-100,000|E| steps
  - Count the number of different types of transitions in $\{Z_t\}$
    - There are 4 different types of transitions
  - Do the counts resemble generation by a 1st-order Markov or independent process?
    - Usually, 1st-order Markov, since entries are correlated
  - Thin the chain, and repeat, till counts resemble generation by an independent sampler
  - The final thinning factor is an estimate of *N*

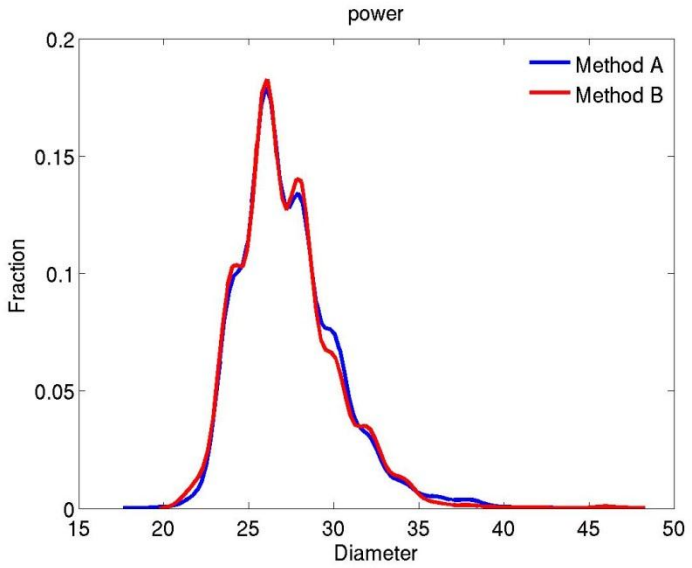
Markov or independent processes?

- How to decide if counts came from a 1st-order Markov or independent process?
  - Consider a complete 2x2 contingency table with data
    - They represent the number m_{ij} of transitions {(0,0), (0,1), (1,0), (1,1)} observed in $\{Z_t\}$
  - Log-linear models are used to model table data
    - 1st-order Markov process: $\log(m_{ij}) = u + u_{1(i)} + u_{2(j)} + u_{12(i,j)}$
    - Independent samples: $\log(m_{ij}) = u + u_{1(i)} + u_{2(j)}$
  - Using maximum likelihood, we can find expressions for the model parameters
    - Standard results in Bishop, Fienberg & Holland
  - Goodness of fits of models can be compared using BIC

Comparing diameter distributions

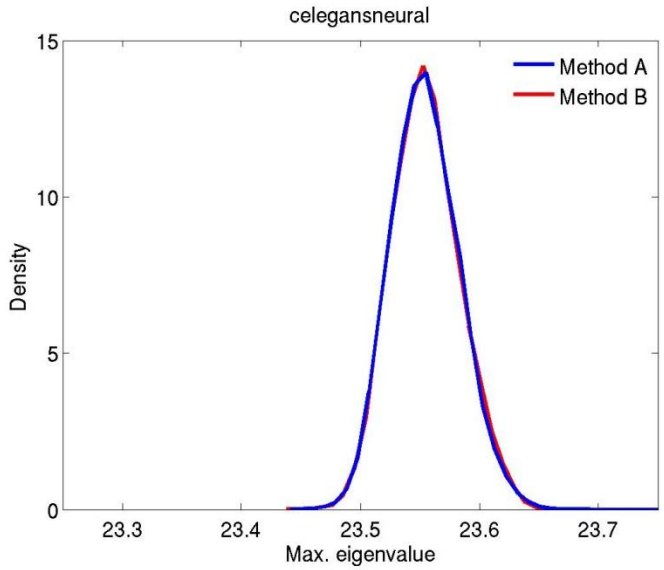


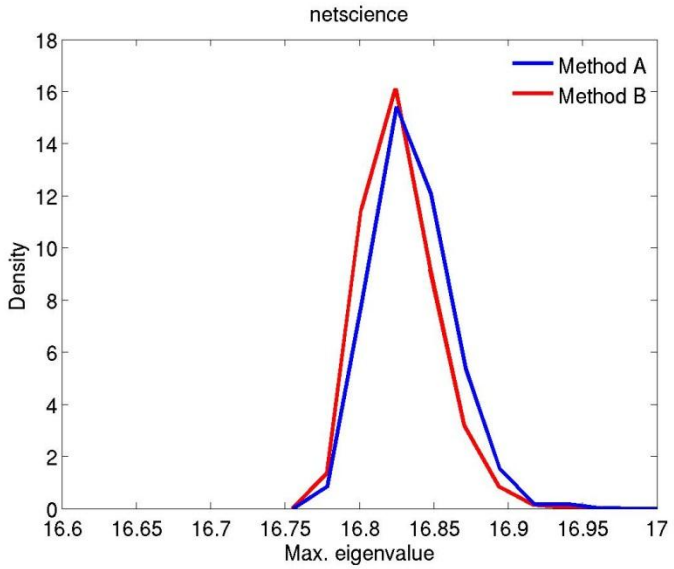


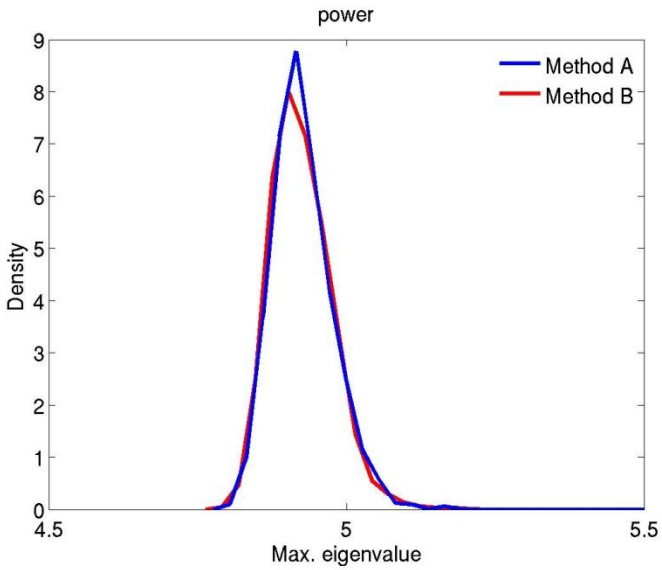


- C. Elegans, co-authorship network and Western States power grid
- $N = 10|E|$ MCMC steps for Method A
  - Seem to suffice for converged distributions

Comparing max. eigenvalues





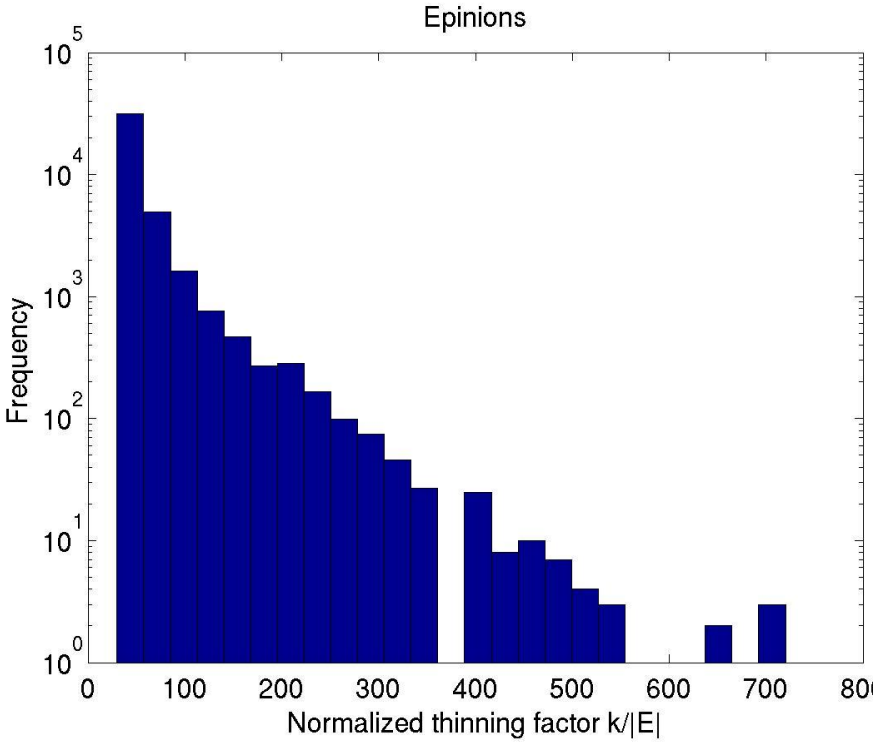


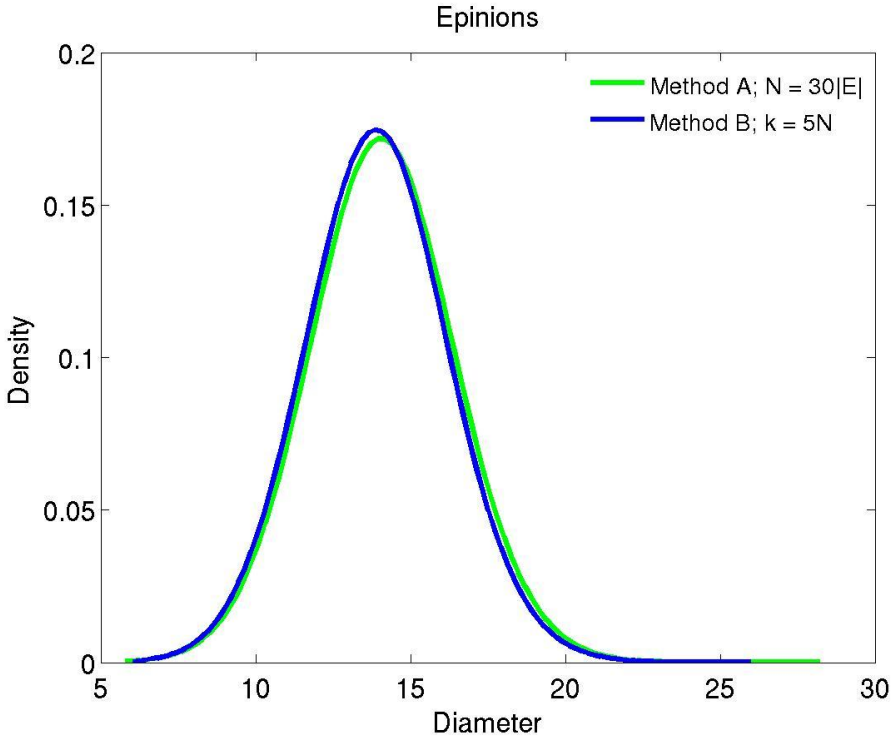
- C. Elegans, co-authorship network and Western States power grid
- N = 10|E| MCMC steps for Method A
  - Seem to suffice for converged distributions

Testing for large graphs

- Method B gets very expensive for large graphs
  - Only a few (10% of the edges) can be checked
  - Further, there are always a few edges that that take a long time to become de-correlated
- How important are such (few) correlated edges to the distributions?
  - How few is few i.e., how many edges need > 30 |E| steps to de-correlate?
  - Impact on distributions?
- Check with soc-Epinions1 graph
  - 75,000 vertices, ~400,000 edges
  - Applied Method B to 10% of the edges

Results for soc-Epinions1 graph





- About 95% edges converge in 30|E| steps
- The remainder makes a small difference in the distributions

Interim summary

- We see that running a MCMC chain 10|E| - 30|E| steps is sufficient to “forget” the starting graph
  - We have derived a simple model, which exploits our constant JDD requirement, to develop an expression for transition probabilities
  - We have checked it with a method that is data-driven
  - We find that in large graphs, about 5% of the edges may still be correlated after 30|E| steps
    - They do not make an appreciable difference in the distributions of graphical parameters in the set of graph samples collected.
- Similar results hold true when degree distribution is preserved

Ray, Pinar and Seshadhri, "Are we there yet? When to stop a Markov chain while generating random graphs", 9th Workshop on Algorithms and Models for the Web Graph, Halifax, Nova Scotia, Canada, June 22-23, 2012.

J. Ray, A. Pinar and C. Seshadhri, "A stopping criterion for Markov chains when generating independent graphs", arXiv:1210.8184[cs.SI]

Topic II – Conditional generation of random fields

- Aim: Given a material with spatially variable properties, estimate structural properties at all scales from *sparse measurements*
- Slight relaxation:
  - Need to know large-scale variations/structures accurately
  - Need to know statistics of the fine structures
- Given: measurements/observations which are impacted by both the fine & coarse structures
- Why? Materials with random & multiscale structures abound and cannot be imaged/measured at all scales
  - Geophysical materials are random & multiscale (geological strata, soil properties etc)
  - Mesoscale O(1μ) electrochemical & catalytic processes at fuel cell anodes
  - Material degradation/aging – e.g., “bubbles” in explosive “cook-off”

Challenges in estimation

- *Never enough data to infer fine & coarse scales simultaneously*
  - If possible to observe / image all scales, why bother to infer anything?
  - Corollary: inferences are always done with incomplete data
- Most inferential methods are iterative
  - Propose, compare with observations, reject/accept
  - Involve a forward model that links the objects of inference with the observables
- So even if a gigantic model resolving all scales is available, can't be used in a inferential setting (aka inverse problem)
  - Takes too long
  - Plus, never enough observables to inform the gigantic model's gigantic d.o.f
- *Net result: Inferences are always uncertain*
  - Due to the use of simplified models and incomplete observations
  - So how to capture the uncertainty?

Inference in a binary medium

- Given: A porous medium with 2 phases
  - A low permeability matrix
  - With fine, high-permeability inclusions
  - Inclusions are unevenly distributed in the domain
  - Domain is rectangular – 1.5 x 1.0
- Scale separation: Impose a 30 x 20 grid on domain
  - Inclusions are 1/10th the grid-block size
    - fine scale variable, δ
  - Each grid-block has an inclusion proportion (***F(x)***)
    - Resolved on the 30 x 20 mesh; coarse scale variable
- Impact: Permeability in a grid-block affected by both fine- and coarse-scale variables
  - $\boldsymbol{k} = \mathcal{K}_{eff}(\boldsymbol{F(x)}, \delta)$

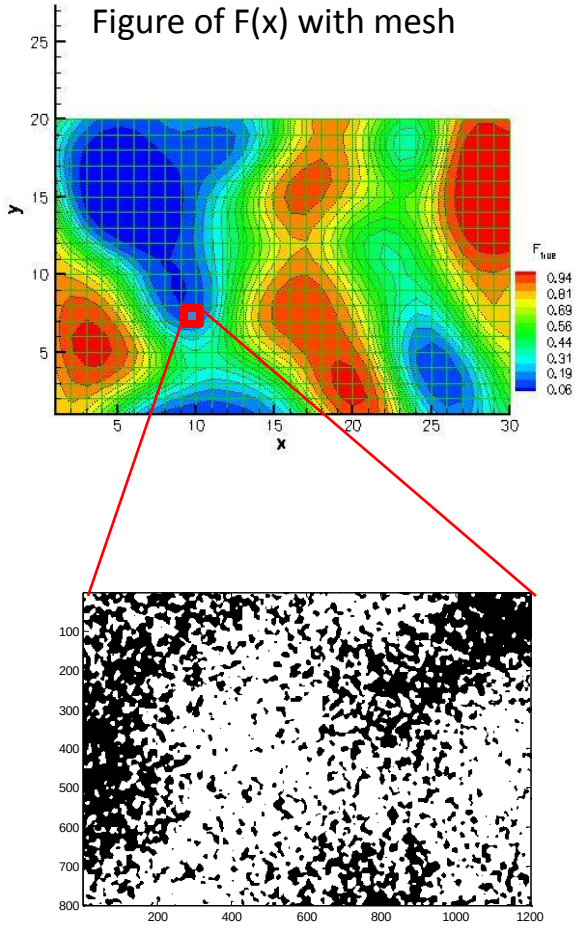
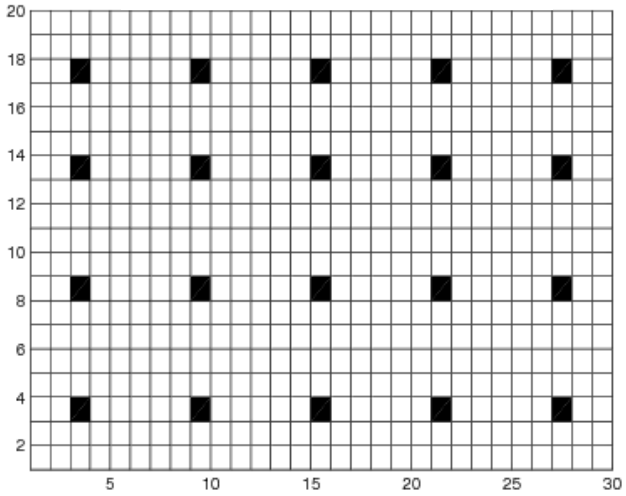


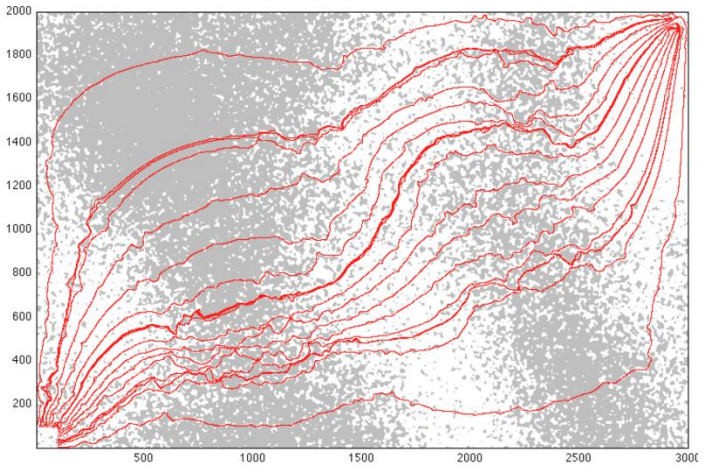
Figure of inclusions (white) in a grid-box

Informative observations

- Consider a set of 20 grid-blocks with sensors
  - $\{k^{obs}\}$ given info on $\{F, \delta\}$ at the sensors
  - OK for inferring structures > inter-sensor spacing
- Water-flood experiment for finer structures
  - What is this?
  - Inject water at one corner, pump it out at the diagonally opposite corner
  - Flow impacted by structures at all scales
  - Water breakthrough time at sensors $\{t^{obs}\}$ contain the integrated impact of multiscale structures
- Teasing out the contributions of the fine- and coarse-scale to $\{t^{obs}\}$ could allow inference of both scales
  - But how?

Location of sensors





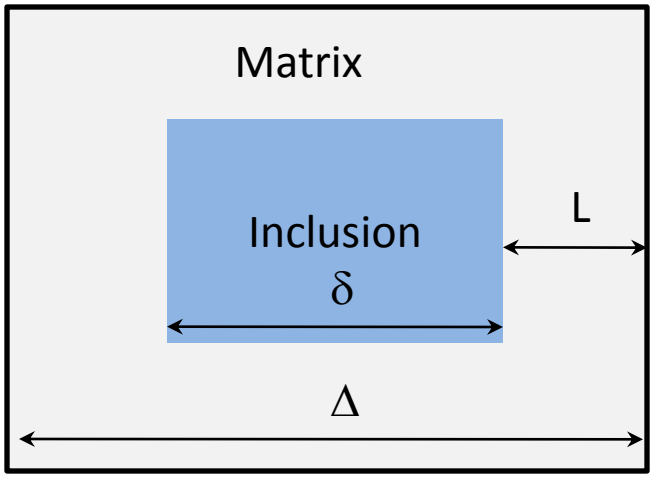
Picture of pathlines through the binary medium. Inclusions in white

Recap, and an idea for inference

- Permeability ***k(x)*** = $\mathcal{K}_{eff}$ (***F(x)***, δ)
  - But we don't know what the functional form of $\mathcal{K}_{eff}$ is
- Breakthrough time ***t*** = $\mathcal{M}$ (***k(x)***)
  - But we have only 20 measurements of **t**, **{t^{obs}}**
  - And 30 x 20 = 600 grid-blocks of unknown ***F*** and δ
- The idea
  - Model #1: Develop a "pointwise" model for ***k*** = $\mathcal{K}_{eff}$ (***F***, δ) in a grid-block
    - Subgrid model
  - Model #2: Develop a parameterized model for ***F*** to describe its spatial variation
    - Have a about 20 – 30 parameters in it – reduced order modeling of ***F(x)***
  - With 20 {k^{obs}} and 20 {t^{obs}}, should be able to infer all unknowns
    - 20-30 parameters for ***F(x)*** and one δ
- Caution
  - With 40 observations, none of these parameters will be estimated well
    - Fine, but how inaccurate are the estimations?

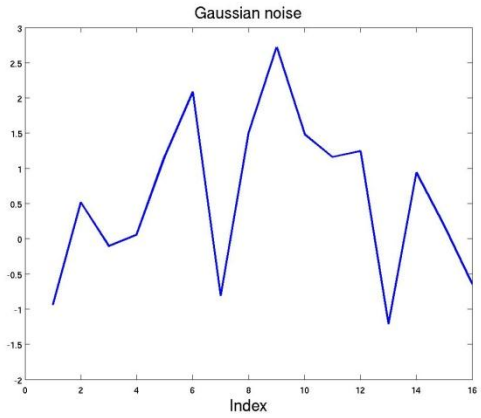
Model #1: subgrid model theory

- We need: $k = \mathcal{K}(F, \delta)$
- Knudby's theory, restricted to rectangular inclusions of size d
  - $k = \mathcal{K}_{\text{Knudby}}(F, \delta, L/\Delta)$
    - L = flow path in the matrix
- Problem: Our inclusions are arbitrarily shaped
- Questions:
  - Can we create a field of arbitrary inclusions, given F and δ?
  - Can we find L in such cases? Just the expected value.
  - Can we do so analytically, without actually creating a field and instantiating an inclusion-in-matrix field?
- Subgrid modeling, but solely geometric

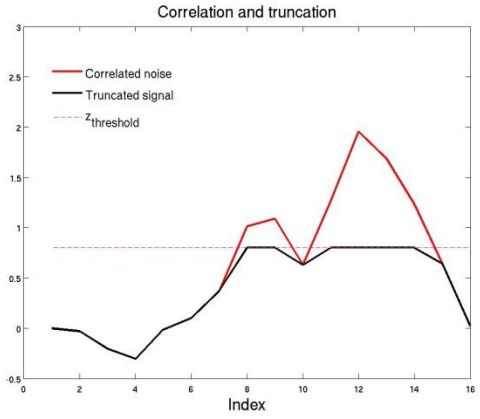


Subgrid geometric modeling

- Consider a grid-block divided into 100 x 100 *grid-cells*
- Initialize a 100 x 100 white-noise field
- Convolve with a Gaussian kernel with FWHM of δ
  - Creates a correlated field with correlation length δ
- Truncate at a level $z_{threshold}$
  - Flat sections are inclusions!
  - $Z_{threshold}$ decides the inclusion proportion ***F*** in the grid-block
- The theory of truncated pluriGaussian fields provides analytical expressions for expected values
  - Number of inclusions
  - Total area in the inclusions
  - These are explicit functions of *F* and δ



1d white noise field



Truncated, correlated field

Subgrid upscaling with Knudby

- If {***F***, δ} specified for each grid-block, we can analytically predict
  - Number of inclusions and total area of the inclusions
  - Ditto, area per inclusion
- Assume that the inclusions are round
  - Inclusion radius can be calculated
- Assume that the centroids of the inclusions are distributed per a Poisson point process
  - Expected value of inter-inclusion distance obtained
- Expected value of flowpath length in matrix *L* can be calculated
- Plug into $\mathcal{K}_{\text{Knudby}}$ and you're done
  - Not quite, but that's the rough outline of the subgrid model

S. A. McKenna, J. Ray, Y. Marzouk and B. van Bloemen Waanders, "Truncated multiGaussian fields and effective conductance of binary media", in *Advances in Water Resources* , 34:617-626, 2011.

Model #2: Reduced order modeling of ***F(x)***

- ***F(x)*** varies in space and is described on a 30 x 20 mesh
  - Don't want to infer all 600 values
  - But ***F(x)*** is smooth – can't we exploit this to make a lower-dimension model?
- Model ***F(x)*** as a 600 variate Gaussian
  - Smoothness guaranteed
  - Assume correlation function known (~ exp(- x^2)) i.e. covariance Γ of multiGaussian is known
- Any multiGaussian can be expanded in a Karhunen-Loeve series
  - We'll truncate at 30 terms
  - ***Φ (x; Γ)*** are called KL modes; w_i are the weights

$$F(x) = \sum_{i=1}^{30} w_i \sqrt{\lambda_i(\Gamma)}\Phi(x;\Gamma)$$

- Inferring ***F(x)*** means inferring w_i

Posing the inverse problem

- Given: $\{\boldsymbol{k}^{obs}, \boldsymbol{t}^{obs}\}$ at 20 sensors
- Models:
  - $\boldsymbol{F}$ = sum of KL modes with unknown weights w_i
  - $\boldsymbol{k} = \mathcal{K}_{eff}(F, \delta)$ – the subgrid model
  - $t = \mathcal{M}(\boldsymbol{k(x)})$ - Darcy flow model, solved using finite-difference method
- Infer weights w_i, i = 1 … 30 and δ
  - Develop distributions for these quantities, not point values
- Generating synthetic $\{\boldsymbol{k}^{obs}, \boldsymbol{t}^{obs}\}$
  - Start with a "ground-truth" binary medium on a 3000 x 2000 mesh
  - Push water through it and measure breakthrough times at 20 sensors – $\{\boldsymbol{t}^{obs}\}$
    - Done with MODFLOW, a Lagrangian code distributed by USGS
  - Superimpose a coarse 30 x 20 mesh
    - Pick out the grid-blocks with sensors
    - Solve a 1D flow equation in each and estimate effective grid-block permeability – $\{\boldsymbol{k}^{obs}\}$

Bayesian Inverse Problem

- Objects of inference, $\Theta = \{F(x), \delta\} = \{w_i, i = 1\ldots30, \delta\}$
- Bayesian inverse problem

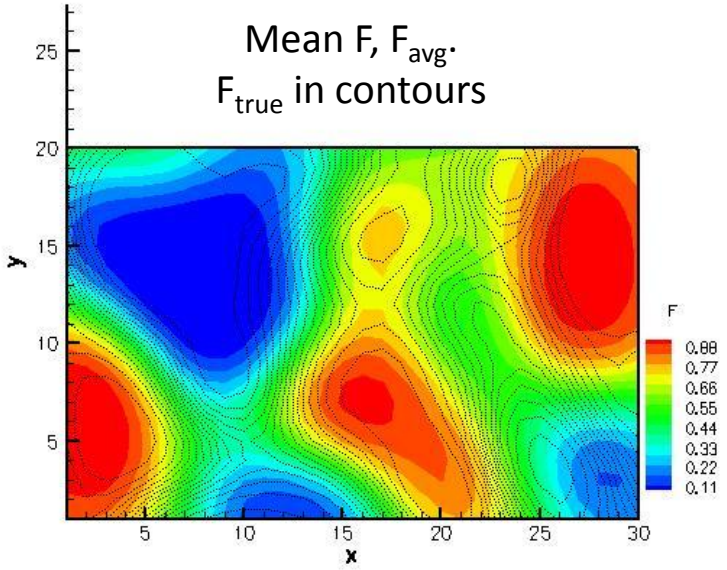
$$-2\log\pi(\Theta) \propto \frac{\{\boldsymbol{t}^{obs} - \mathcal{M}(\Theta)\}^2}{\sigma^2_{T}} + \frac{\{\boldsymbol{k}^{obs} - \mathcal{K}_{eff}(\Theta)\}^2}{\sigma_{K}{}^2} + \frac{\{\Theta - \Theta_p\}^2}{\sigma^2_{\Theta}}$$

  - $\mathcal{M}$, Darcy flow model to relate Θ to breakthrough times $\{\boldsymbol{t}^{obs}\}$
  - $\mathcal{K}_{eff}$, subgrid model to relate Θ to observed permeability at certain sampling points
  - Θ_p, prior beliefs regarding the values of Θ
  - $\sigma_{\{K, T\}}$, std. dev. of various measurement errors
- $\pi(\Theta)$ evaluated by Markov Chain Monte Carlo sampling
  - Particular algorithm called DRAM

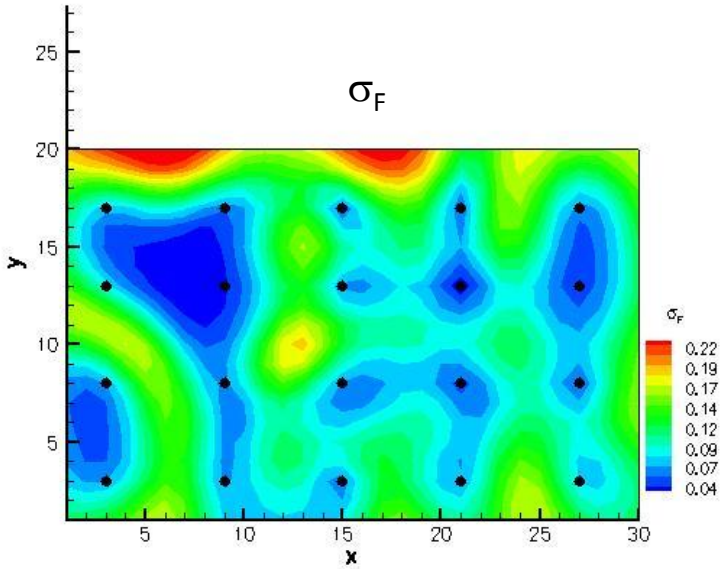
Results

- Get 10^4 samples of $\{w_i, \delta\}$
- From each $\{w_i, \delta\}$, develop 10^6 instances of ***F(x)*** and $\mathcal{K}_{eff}$(***F(x)***, δ)
- Take the mean & std dev of the 10^6 ***F(x)*** instances
- Take standard deviations tog

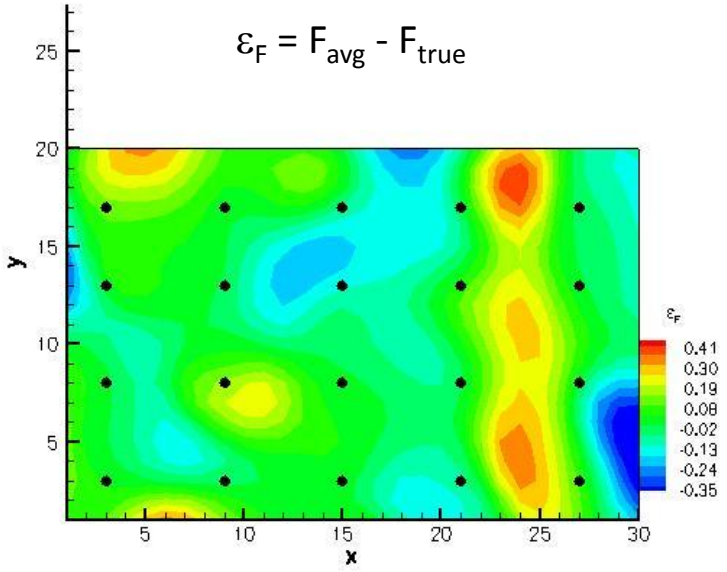
Mean F, F_{avg}.
F_{true} in contours



σ_F

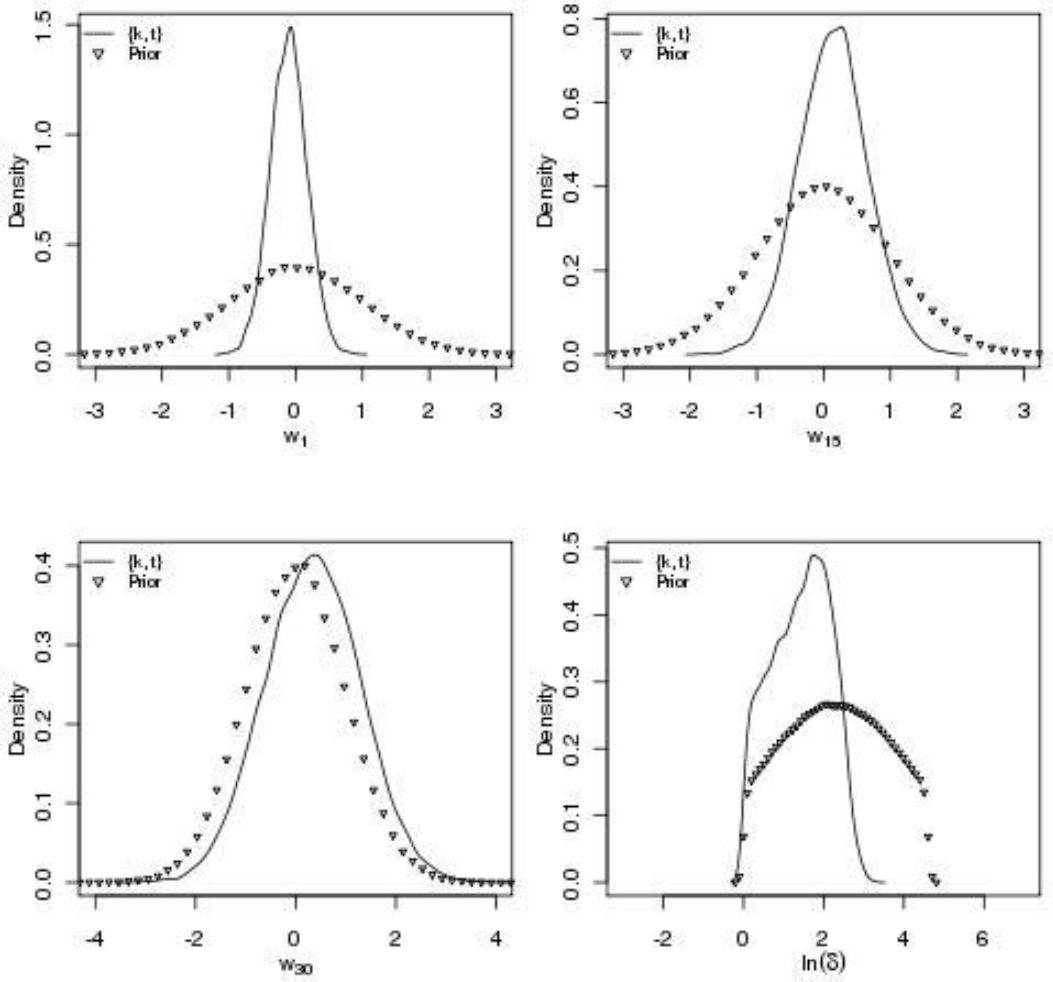


$\varepsilon_F = F_{avg} - F_{true}$



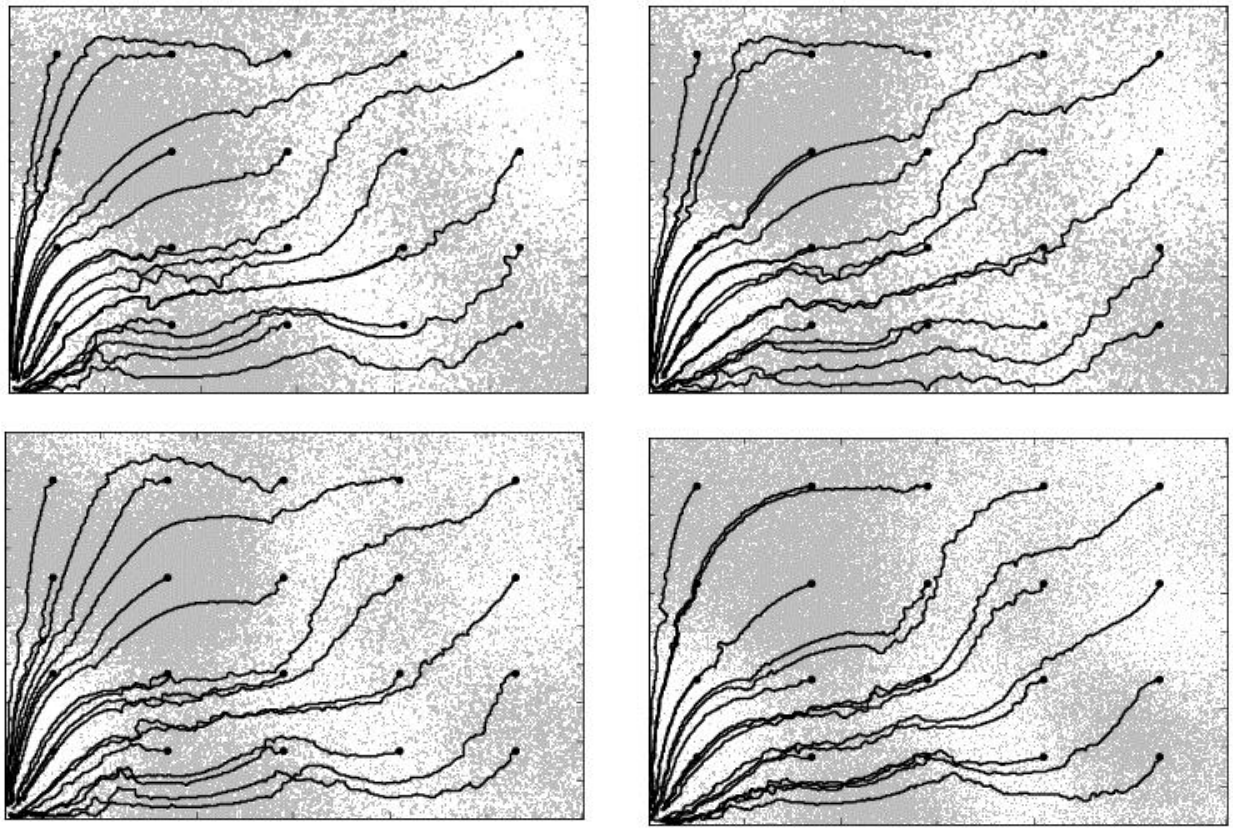
PDFS of $\{w_i, \delta\}$

- Use the 10^4 samples of $\{w_i, \delta\}$ to develop PDFs
- Take w_1, w_{15} and w_{30} as proxies for large, medium and small (but resolved) scale variations
- Inversions performed with $\{\boldsymbol{k}^{obs}\}$ only also plotted
- Takeaways:
  - Large-scale structures easy to infer
  - Gets harder as we get smaller
  - Doesn't apply to inclusions



Developing fine-scale realizations

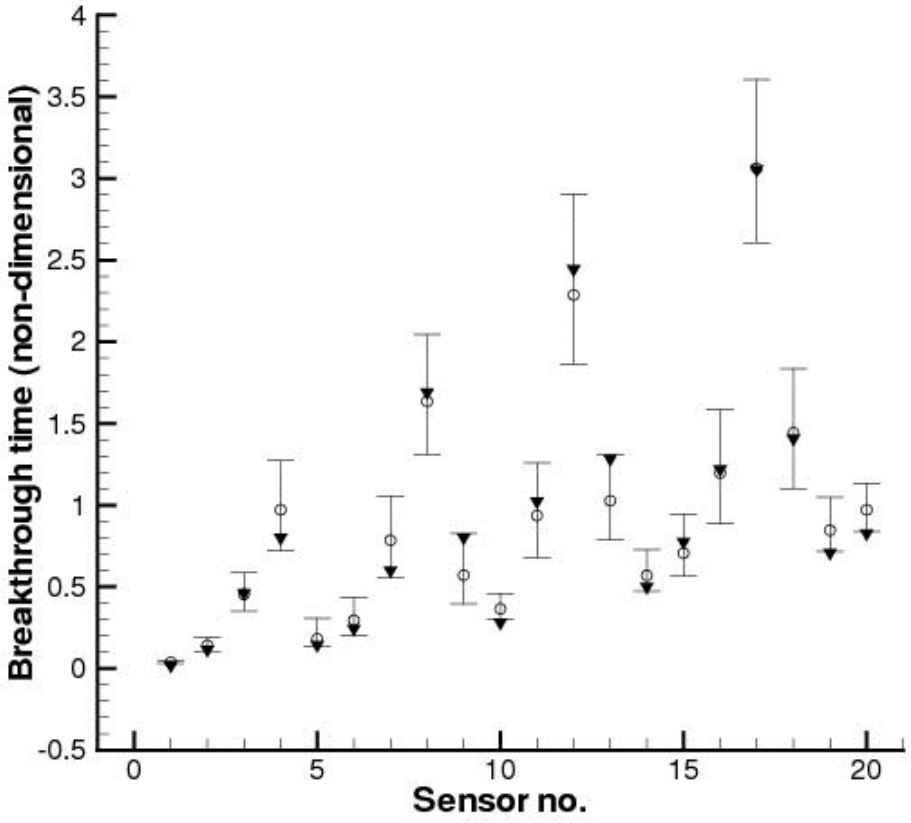
- The inferences can be used to develop fine-scale binary media



- Flow simulations can be used to obtain an ensemble of predicted breakthrough times at sensors

Posterior predictive checks

- Fine-scale binary media realizations (on a 3000 x 2000 mesh) can be used to calculate breakthrough times at 20 sensors
  - Did so with 1,000 realizations, not all 10^6 possible
  - Allowed us to plot 1st, 50th and 99th percentiles
  - Measurements plotted as references
- Why are some breakthrough times well predicted and others are not?



Interim summary

- One can use data to infer structures which one cannot resolve with a mesh
  - Require the use of a subgrid model, parameterized with subgrid structures
  - Requires proper data
  - Will only provide statistics of the subgrid structures
- In many cases, the subgrid structures may not affect the measurements sufficiently
- The inference can also quantify the uncertainty in the inference
- We may also be able to generate an ensemble of fine-scale structures which are consistent with the observations

J. Ray, S. A. McKenna, B. van Bloemen Waanders and Y. M. Marzouk, "Bayesian reconstruction of binary media with unresolved fine-scale spatial structures" in *Advances in Water Resources* , 44:1--19, 2012.

Conclusion

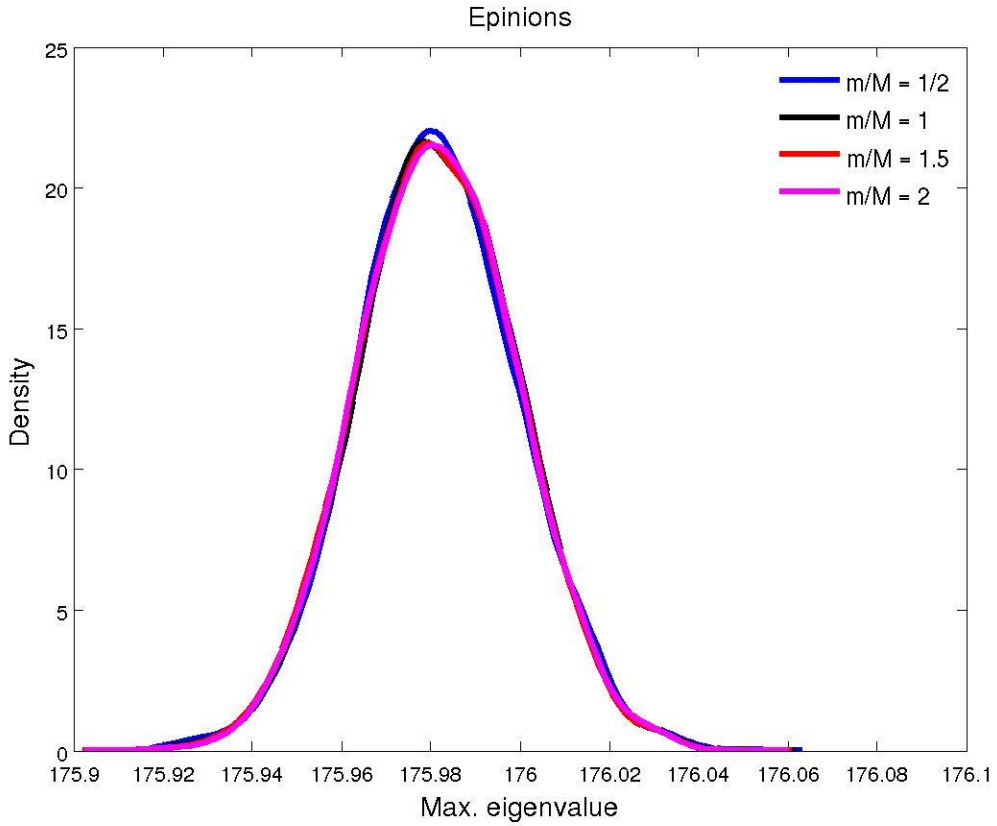
- Use of Markov chain Monte Carlo & Bayesian inference quite common in Sandia
  - Being used to calibrate climate models, material models of re-entry bodies, turbulence models etc.
  - Efforts to develop “parallel” MCMC methods
    - That amortize the sampling burden over N CPUs
  - Sparsity-enforced model reconstruction (“Bayesian LASSO”) common used for sensitivity analysis of climate models
    - Many putative parameters and not enough runs to do a proper sensitivity analysis
- Network construction
  - Models for constructing graphs (generative & rewiring models)
  - Sublinear algos for measuring graphical properties (e.g., estimate # of triangles in a graph)
  - New work on network tomography

Background

How many samples, *M*, to collect?

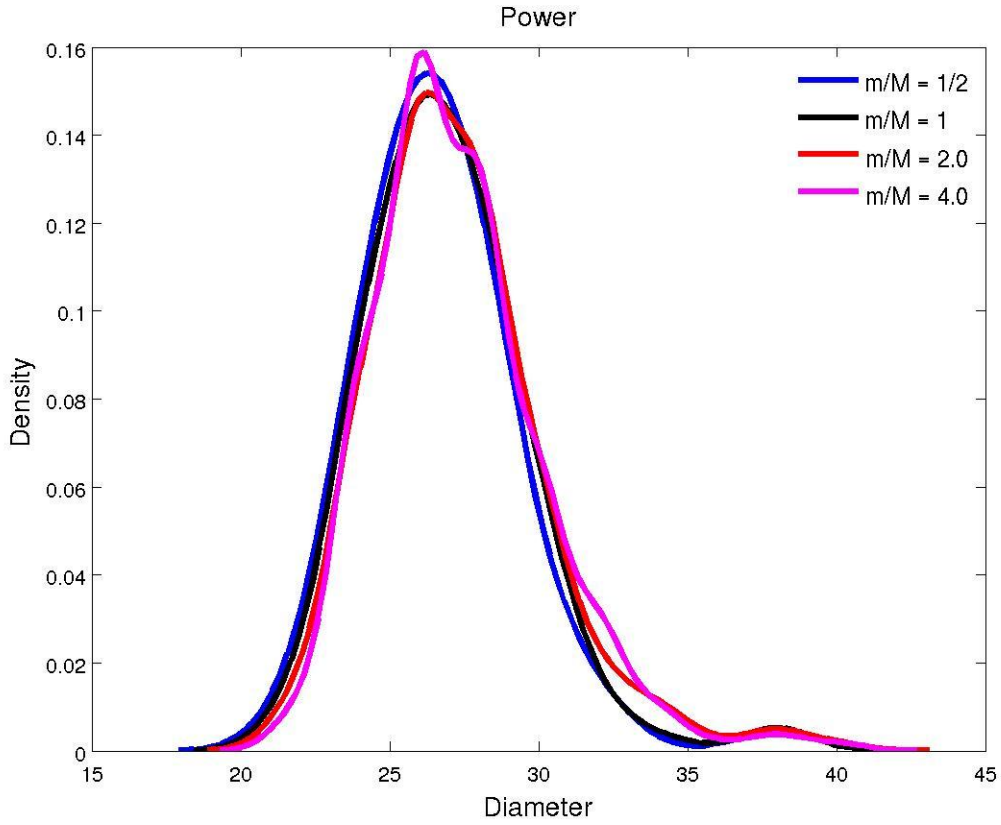
- We plot distributions of diameter, max. eigenvalues etc. empirically, from a generated set of graphs
  - Typically, we plot graphs with increase number of independent samples till we get convergence in the plots
- What is an approximate size of this sample set?
- We can track $\{Z_t\}$ and observe it converge to its mean
  - Does a particular accuracy in the estimate of mean provide a useful estimate of the number of samples to collect?
  - Estimate edge mean within 5% accuracy, with 95% confidence

How many samples to take? – soc-Epinions1 graph



- Epinions graph
  - Just because edge-means converge does not mean other graph properties do
  - Need about 2x more samples

How many samples to take? – Power graph



- Western states power graph
  - Need about 4x more samples

Checking via Gelman-Rubin statistic

- Both Methods A and B start with the same (real) graph
  - Is the agreement in distributions because of the starting point?
- Check: Generate 2 new starting graphs
  - Run a Markov chain for 10,000|E| steps, starting with a real graph, to get a new independent graph.
  - Start separate Markov chains from these 3 starting points
    - Samples are collected after 30 |E| MCMC steps
    - 300 samples collected
  - Monitor their convergence using G-R statistic

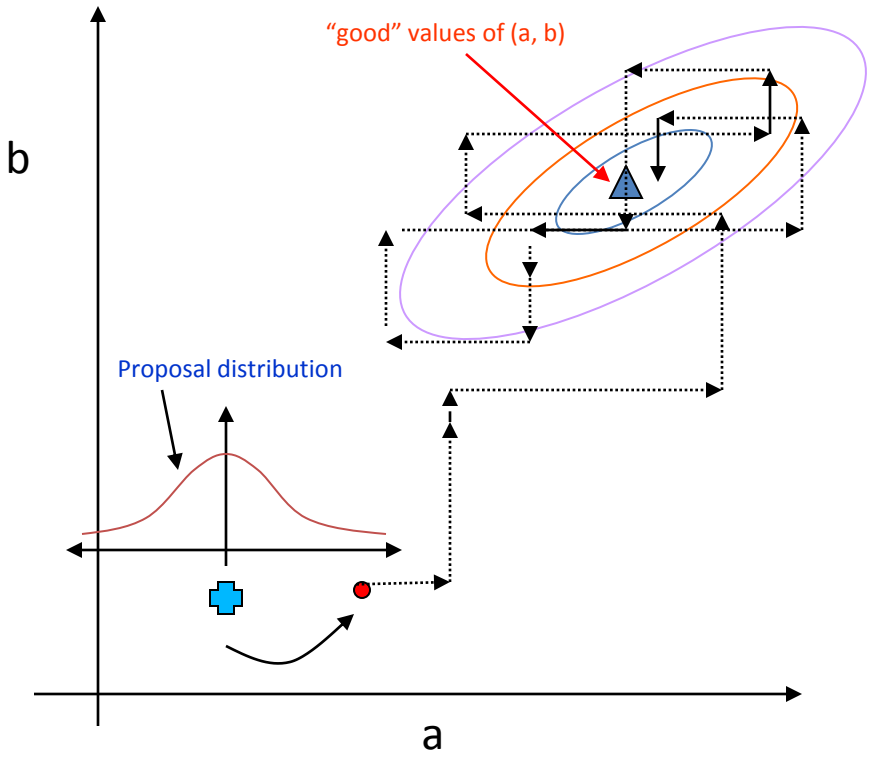
	C. Elegans	Netscience	Power grid	Soc-Epinions1
G-R statistic	1.05	1.02	1.006	1.06

What is MCMC?

- A way of sampling from an arbitrary distribution
  - The samples, if histogrammed, recover the distribution
  - Given a starting point (1 sample), the MCMC chain will sequentially find the peaks and valleys in the distribution and sample proportionally
  - Drawback: Generating each sample requires one to evaluate the expression for the density π
- An example
  - Given: (Y^{obs}, X), a bunch of n observations
  - Believed: $y = ax + b$
  - Model: $y_i^{obs} = ax_i + b + \varepsilon_i$, $\varepsilon \sim \mathcal{N}(0, \sigma)$
  - 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" $\varepsilon_i = y_i^{obs} - (ax_i + b_i)$
    - Likelihood of the set (a, b, σ) = $\Pi \exp(- \varepsilon_i^2/\sigma^2)$
  - Solution: $\pi (a, b, \sigma \mid Y^{obs}, X) = \Pi \exp(- \varepsilon_i^2/\sigma^2)$ * (bunch of uniform priors)

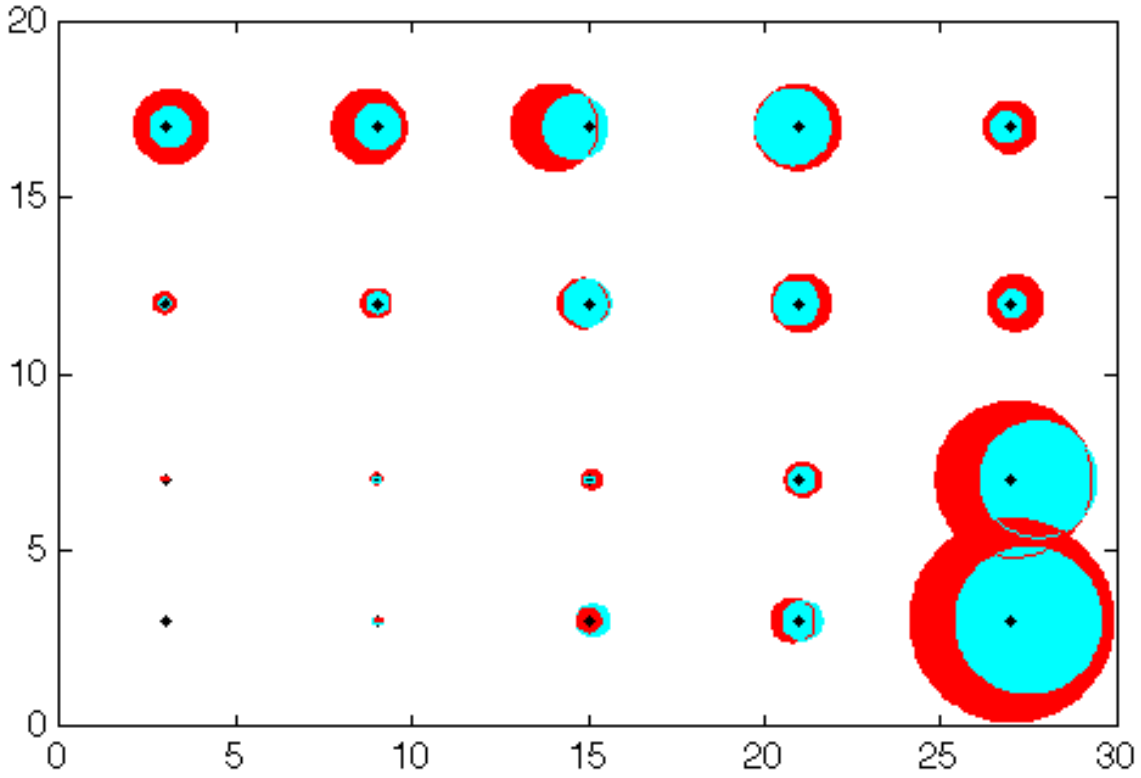
MCMC, pictorially

- 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
- Choose a starting point,
  - $P^n = (a_{curr}, b_{curr})$
- Propose a new a, $a_{prop} \sim \mathcal{N}(a_{curr}, \sigma_a)$
- Evaluate π (a_{prop}, b_{curr} | ...) / π (a_{curr}, b_{curr} | ...) = 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



Circle plots

- Sensor: Dots
- Circles
  - Red: PPC using reconstructions using just $\{k^{obs}\}$
  - Cyan: Using $\{k^{obs}, t^{obs}\}$
- Circle radius:
  - Prop to the 95% CI of breakthrough times
- Circle center offset:
  - Prop to diff between measured and mean pred. breakthrough



- Takeaway: Further the measurement from injector/producer, bigger the uncertainty in predictions from reconstructions. Two reasons
  1. Longer breakthrough times – the % uncertainty may not be large
  2. Smaller flow rates lead to less info gathered and bigger uncertainties