Parallel MCMC Recombination for Big Data Analysis

Michael Zhang, Daniele Schiavazzi, Lizhen Lin

The University of Texas at Austin
Department of Statistics and Data Sciences
College of Natural Sciences

UNIVERSITY OF NOTRE DAME
College of Science

Introduction

- Bayesian inference is usually intractable
- Popular inference algorithm–Markov chain Monte Carlo
- “Big data” motivated need for parallel algorithms

Bayesian Inference

- In Bayesian modeling we have a prior: $P(\theta)$
- And we have a likelihood:

$$P(X|\theta) = \prod_{i=1}^{N} P(X_i|\theta)$$

- Our posterior is:

$$P(\theta|X) = \frac{P(X|\theta)P(\theta)}{\int P(X|\theta)P(\theta)\,\mathrm{d}\theta}$$

The Metropolis-Hastings Algorithm

- Classic inference algorithm is Metropolis-Hastings (MH)
- MH algorithm is:
  1. Draw $\theta' \sim Q(\theta)$
  2. Let
$$r = \min\left\{1, \frac{P(X|\theta')P(\theta')Q(\theta^{(t)})}{P(X|\theta^{(t)})P(\theta^{(t)})Q(\theta')}\right\}$$
  3. Set $\theta^{(t+1)} := \theta'$ with probability r or $\theta^{(t+1)} := \theta^{(t)}$ with probability $1 - r$
- Exhibits detail balance

Problem Statement

- We cannot trivially parallelize MCMC
- The likelihood function, $\prod_{i=1}^{N} P(X_i|\theta)$, could be difficult to compute
- Our idea: parallelizable MCMC without excessive likelihood evaluation.

Divide-and-Conquer Inference

- Common parallel technique: Divide and conquer
  1. Divide the data randomly across P processors
  2. Perform MCMC independently
  3. Combine subposteriors
- This is *embarrassingly parallel*.

Divide-and-Conquer Inference

- Popular combination methods: Consensus Monte Carlo, KDE, geometric median, etc.
- Different variants of combining subposteriors
- Ours avoids excessive likelihood calculation

Local Polynomial Regression

- Suppose we want to approximate the log likelihood

$$\sum_{i=1}^{N} \log P(X_i|\theta) \approx \hat{\ell}(\theta)$$

- We fit local polynomial regression model to subposterior samples.
- Complexity is $O(MD^3)$

Parallel Sampler

- Our sampler works as follows:
  1. Sample subposterior $P(\theta|X^{(p)})$ for $p = 1, \ldots, P$ processors
  2. Pool subposteriors, calculate log likelihood
  3. Fit local regression model
  4. For $t = 1, \ldots, T$ iterations on master processor, accept new state with probability

$$r = \min \left\{ 1, \frac{\hat{\ell}(\theta')P(\theta')Q(\theta^{(t)})}{\hat{\ell}(\theta^{(t)})P(\theta^{(t)})Q(\theta')} \right\}$$

Proposal Distribution

- Choice of proposal distribution $Q(\theta)$ is important
- Our recommendation: Dirichlet process mixture of subposterior samples $G \sim \mathsf{DP}(\alpha, H) = \sum_{k=1}^{\infty} \pi_k \delta_{\phi_k}$:

$$\theta_m | z_m, \phi_{z_m} \sim P(\theta_m | \phi_{z_m}),\ z_m | \pi \sim \mathsf{Multinomial}(\pi),$$
$$\pi | \alpha \sim \mathsf{GEM}(\alpha), \phi_k \sim H.$$

  for $m = 1, \ldots, M$ subposterior samples
- Idea: propose samples with good estimates of log likelihood.

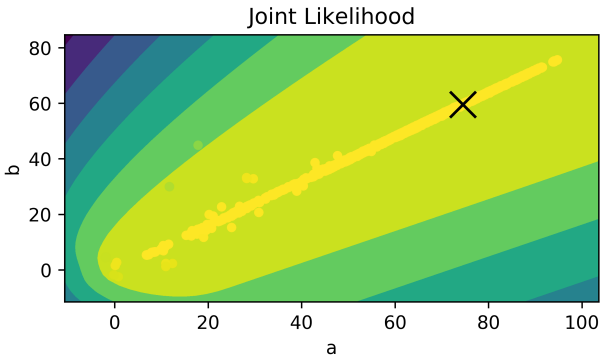
Numerical Tests: Logistic Regression

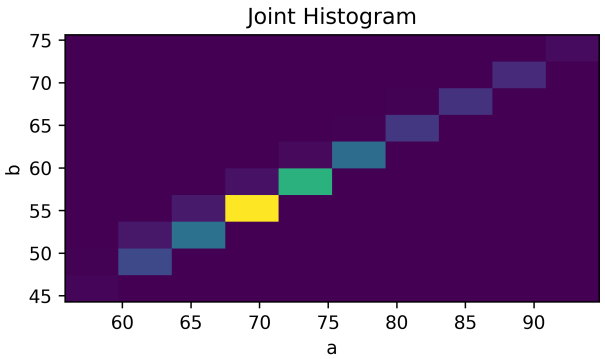
- Assume data generated from model

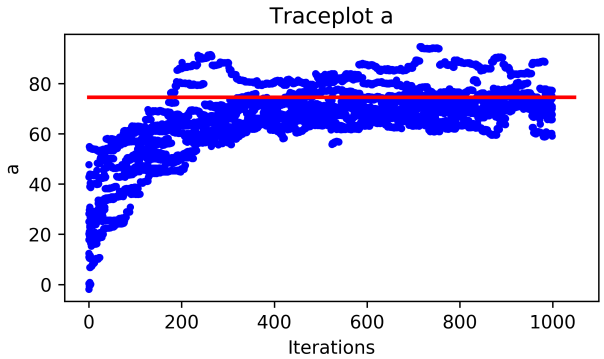
$$y_i \sim \text{Binomial}\left(z_i, \frac{1}{1+\exp\{a+bx_i\}}\right), \quad (a,b) \sim N(0, \tau^2 I)$$

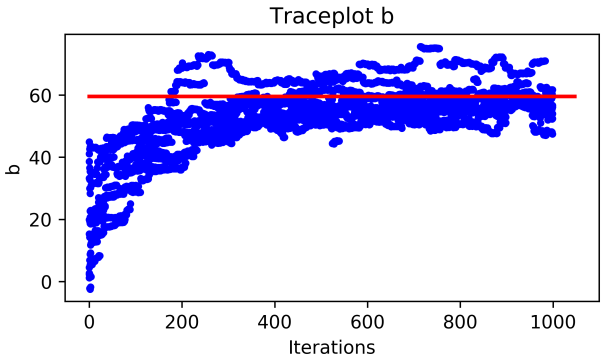
- N = 10,000 and P = 5

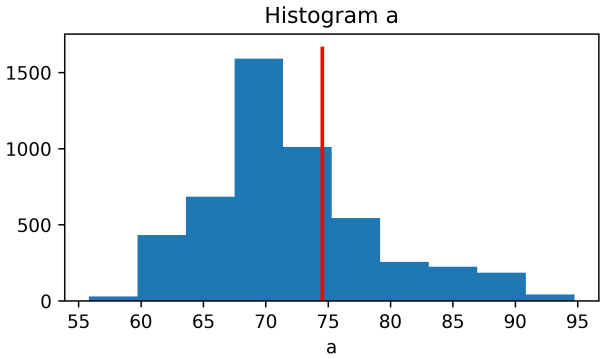
Numerical Tests: Logistic Regression, Subset Posterior

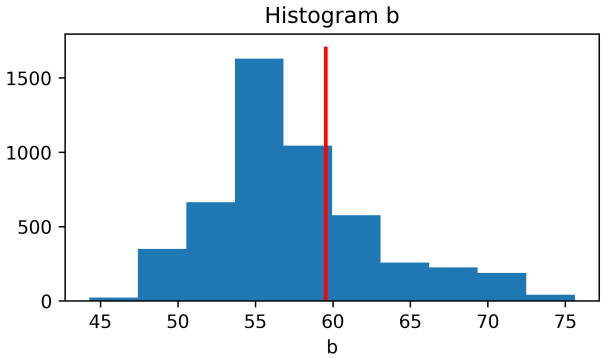




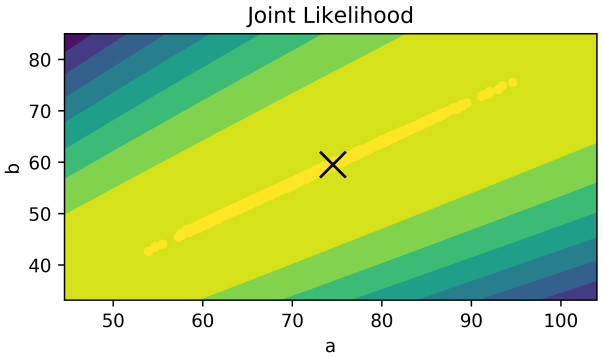


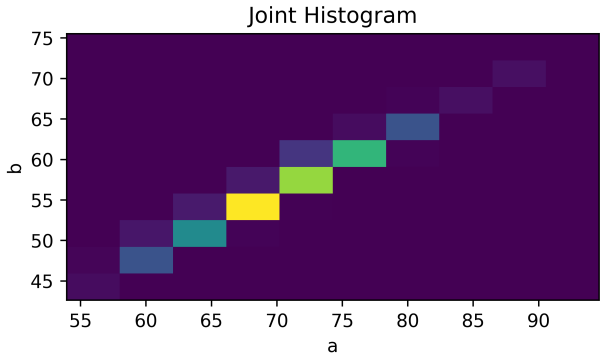


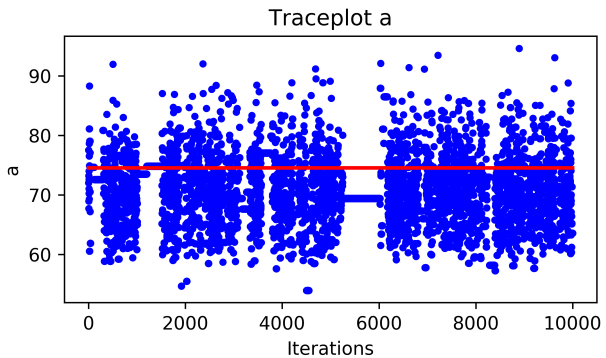


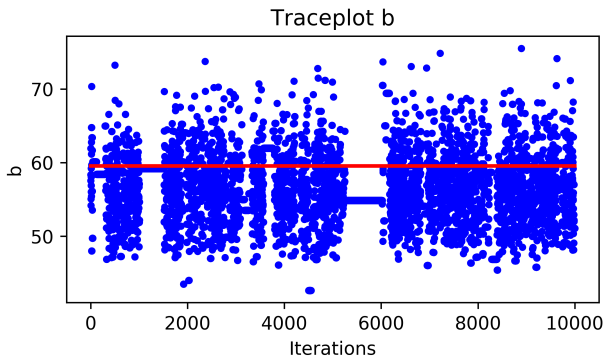


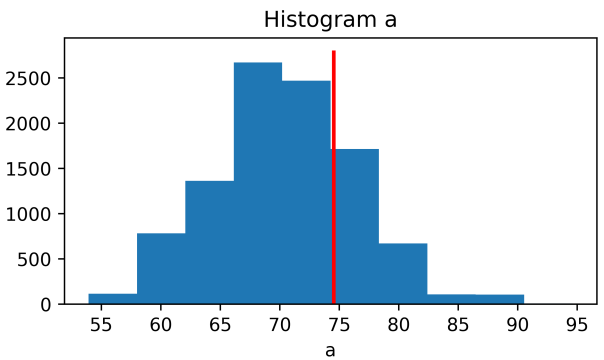
Numerical Tests: Logistic Regression, Final Posterior

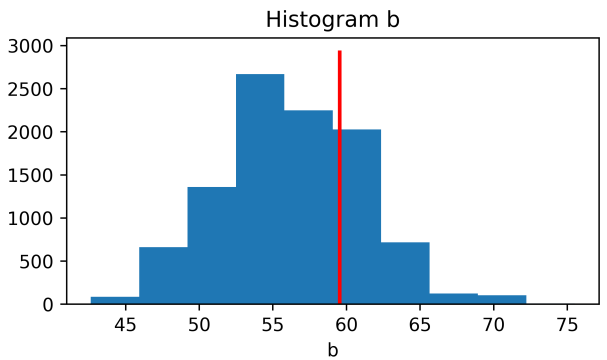




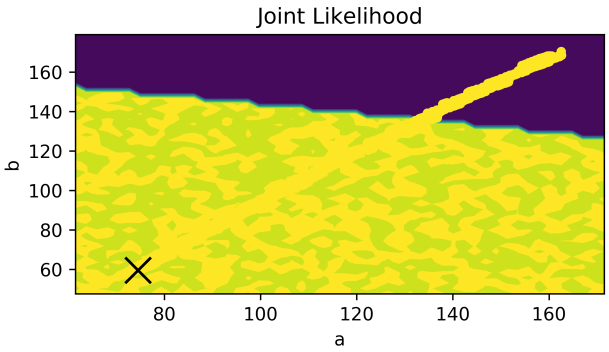


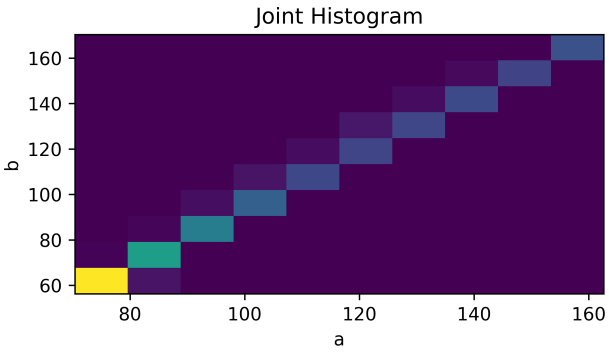


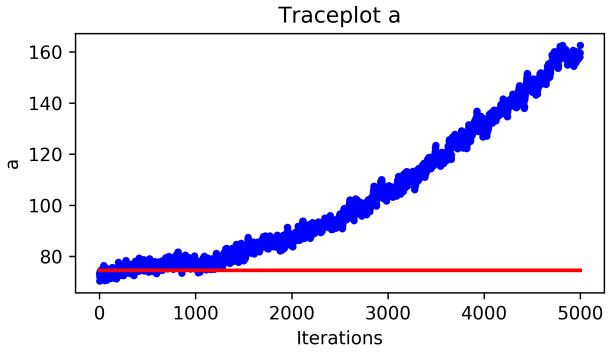


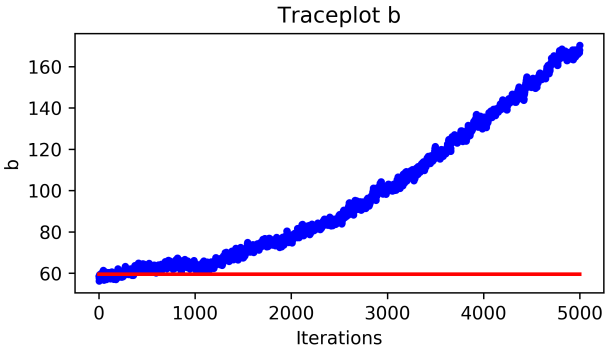


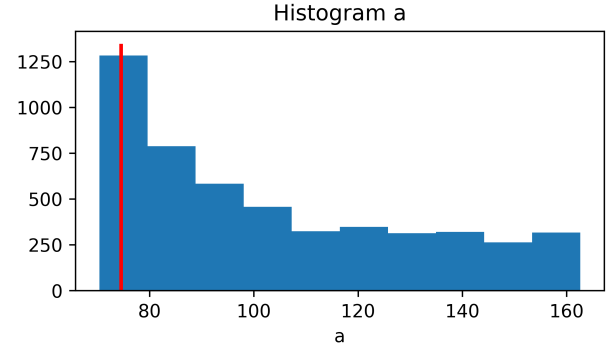
Numerical Tests: Logistic Regression, Proposal Distribution

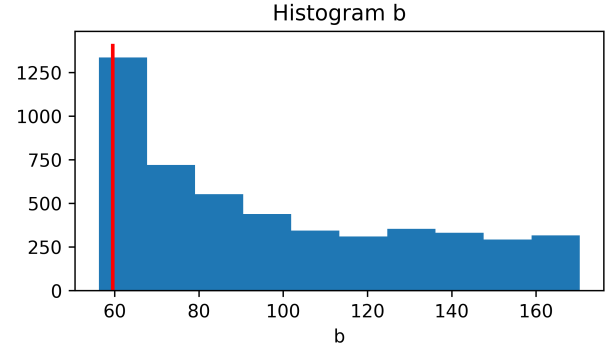




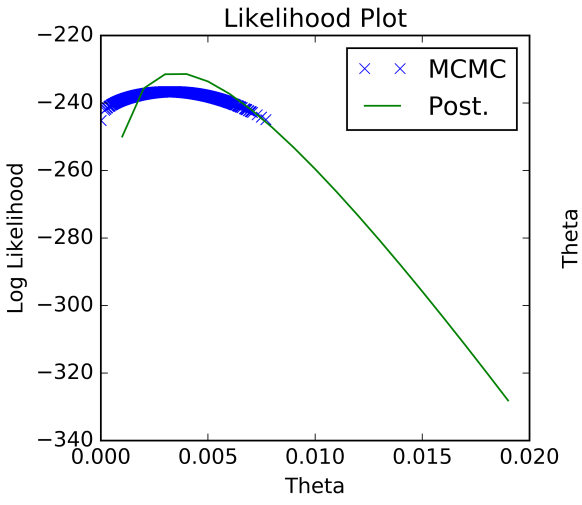


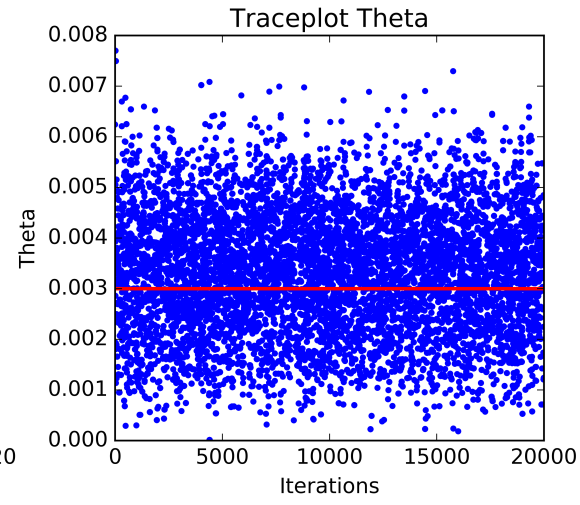


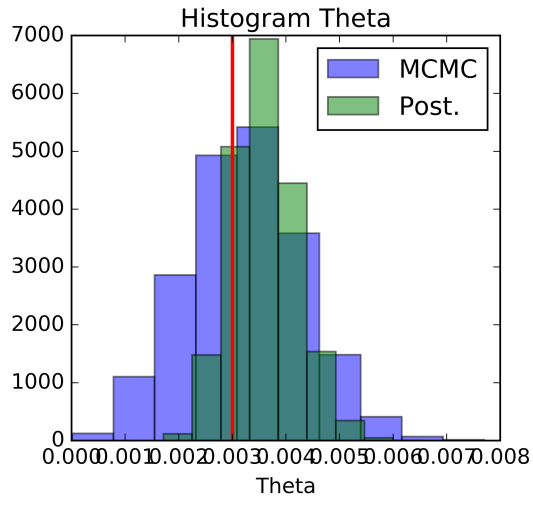




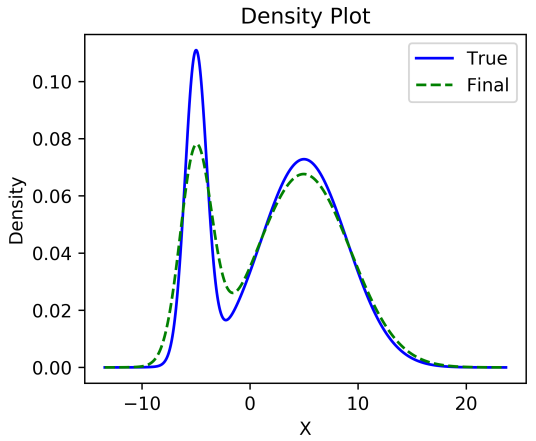
Numerical Tests: Rare Bernoulli

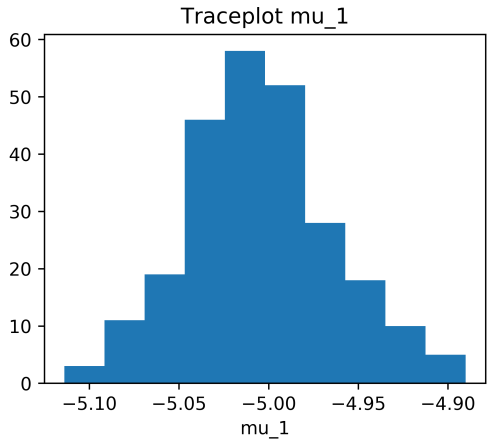


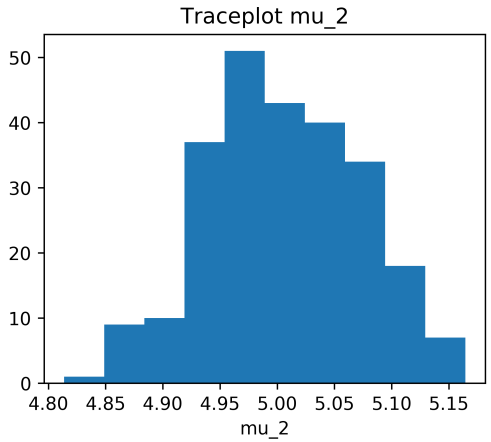


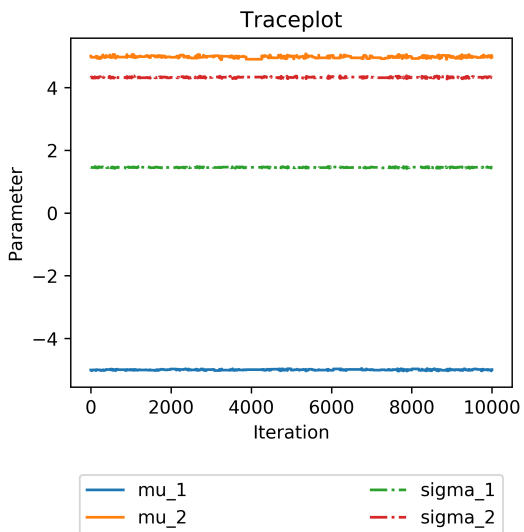


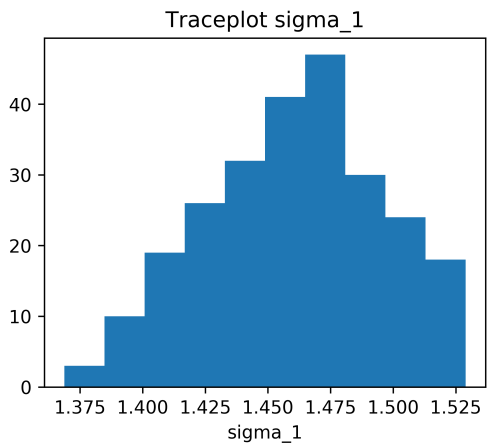
Numerical Tests: Bimodal Distribution

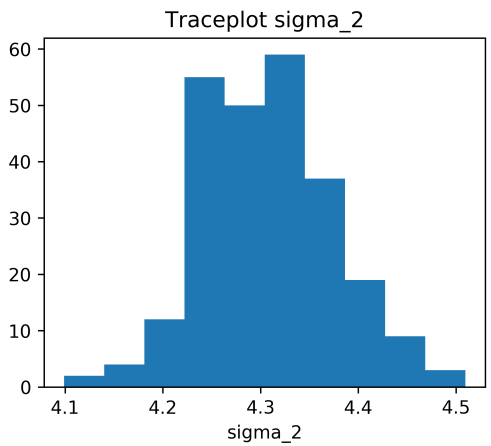






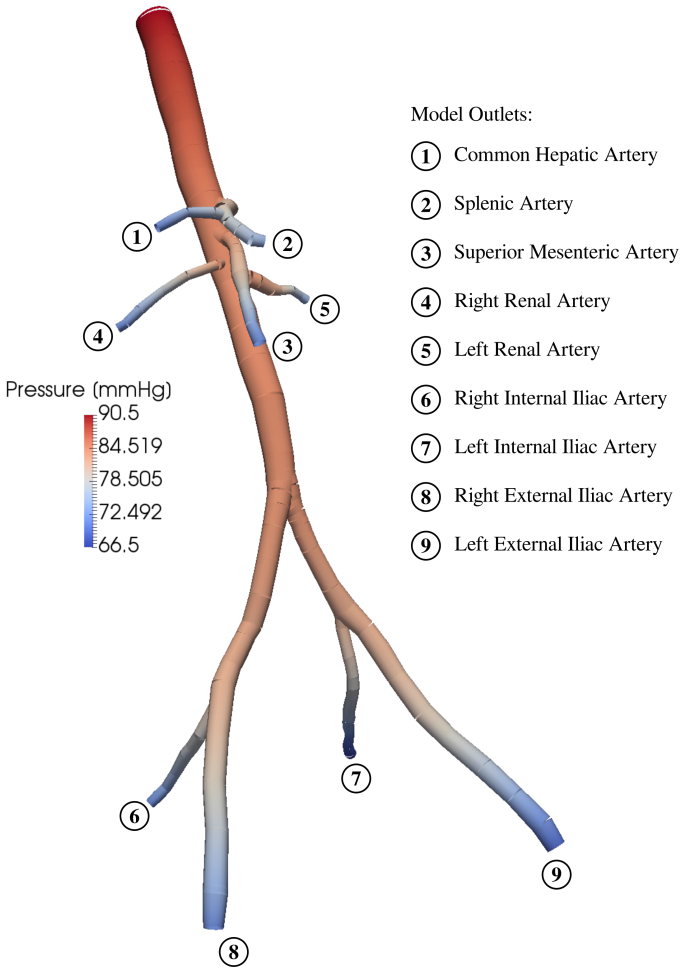






Future Work: Heart Blood Flow Model

- Interest in modeling blood flow dynamics in heart
- Parameter size is small but likelihood is expensive



Conclusion

- Parallel sampler idea: for expensive likelihoods with small parameter sizes
- Generally we need parallel samplers for “big data”
- Also want to quantify uncertainty of divide-and-conquer methods
- To be completed: Theorems and blood flow model