

Parallel Chromatic MCMC with Spatial Partitioning

Abstract

We introduce a novel approach for parallelizing MCMC inference in models with spatially determined conditional independence relationships, for which existing techniques exploiting graphical model structure are not applicable. Our approach is motivated by a model of seismic events and signals, where events detected in distant regions are approximately independent given those in intermediate regions. We perform parallel inference by coloring a factor graph defined over regions of latent space, rather than individual model variables. Evaluating on a model of seismic event detection, we achieve significant speedups over serial MCMC with no degradation in inference quality.

Introduction

Probabilistic modeling is becoming an increasingly important framework for designing machine learning algorithms. However, scaling inference to large datasets is still a major challenge. Methods such as Markov chain Monte Carlo (MCMC) can require many steps to mix, especially for complex models on large datasets of practical interest. A natural solution is to exploit additional computational resources via parallel inference, but existing methods for parallel MCMC often depend on graphical model structure that may not be present in all problems.

This paper proposes a new approach for parallel inference in models with conditional independence relationships induced by spatial structure rather than a fixed graphical model. Our approach is inspired by an application to seismic event detection (Arora et al., 2010), and we use a toy model of seismic events and signals as the running example to illustrate the method. We partition a continuous latent space into regions, inducing a conditional independence structure on the *sets* of variables taking values in each region, rather than individual variables themselves. Motivated by Chromatic Gibbs Sampling (Gonzalez et al., 2011), we color the regions using an induced factor graph, and run inference in parallel within all regions of a given color. The spatial partitioning naturally allows for data subsampling, since the Markov blanket (Pearl, 1988) of each region involves only a local subset of observed data. We further improve mixing

by applying dynamic partitioning to avoid boundary effects between spatial regions. Although this paper focuses on an application to seismic event detection, the technique is applicable more generally to any model in which objects or latent structures are inferred from "local" surroundings in a large space. This potentially includes, among others, large-scale visual recognition, object detection from aerial or satellite imagery, radar tracking, modeling long time series such as speech or video, and simultaneous localization and mapping (SLAM).

We briefly review MCMC inference and graph coloring-based parallelization in the next section, before introducing the toy model of seismic events and illustrating our approach for parallel inference by spatial partitioning. Evaluating our approach with a comparison to serial MCMC, and to a naïve partitioning that does not preserve the correct stationary distribution, we observe significant speedups from parallelism without compromising the quality of the resulting inferences, as measured by precision, recall, and mean error in the location of recovered events.

Background

MCMC techniques are often applied to solve integration and optimisation problems in large dimensional spaces, including inference in probabilistic models. Most practically applied MCMC chains are constructed using the framework of the Metropolis-Hastings (MH) algorithm. An MH step of invariant distribution $\pi(x)$ and proposal distribution $q(x^*|x)$ involves sampling a candidate value x^* given the current value x according to $q(x^*|x)$. The Markov chain then moves towards x^* with acceptance probability $A(x, x^*) = \min\{1, \frac{\pi(x^*)q(x|x^*)}{\pi(x)q(x^*|x)}\}$, and otherwise remains at x (Andrieu et al., 2003). If the proposal distribution is symmetric, $q(x^*|x) = q(x|x^*)$, the acceptance probability reduces to $\alpha(x^*|x) = \min\{1, \frac{\pi(x^*)}{\pi(x)}\}$.

Gibbs sampling is the special case of Metropolis Hasting where the proposal distributions are the posterior conditionals. The acceptance probability of a Gibbs move is always 1 and thus all proposals are accepted.

Many methods have been proposed to parallelize MCMC inference. We build on the work of Gonzalez et al. (2011), which considers a Markov random field (MRF) with a k -coloring such that each vertex is assigned one of k colors

and adjacent vertices have different colors. Let k_i denote the variables in color i . Then the Chromatic sampler simultaneously draws new values for all variables in k_i before proceeding to k_{i+1} . The k -coloring of the graph ensures that all variables within a color are conditionally independent given the variables in the remaining colors and can therefore be sampled independently in parallel.

Algorithm 1 Chromatic Gibbs Sampler

```

1: Input: k-color MRF
2: for each of  $k$  colors  $k_i$ ;  $i \in 1 \dots k$  do
3:   for all  $X_i \in k_i$  in the  $i^{th}$  color do in parallel
4:     Execute Gibbs Update
5:      $X_j^{(t+1)} \sim \pi(X_j | X_{N_j \in k < i}^{(t+1)}, X_{N_j \in k > i}^{(t)})$ 

```

However, the Chromatic Gibbs sampler depends on a fixed factor graph structure and cannot exploit additional independences implied by spatial relationships between variables.

Generative Model of Seismic Signals

We motivate our proposed inference approach in a toy model of seismic event detection, which we use as a running example. Our model describes an unknown number of seismic events in a world with one spatial dimension, so that each event i is described by a tuple $e_i = (x_i, t_i)$ giving its scalar location and time, as well as additional latent variables including the arrival times of event signals at each of four stations. The signal observed at each station includes contributions from events as well as a background noise process. The inference problem is to recover the number of events, and their space-time positions, given noisy observed signals.

Event prior The number of events are generated from a Poisson prior $P(|e|) = \frac{(\lambda \cdot T)^{|e|} \exp(-\lambda \cdot T)}{|e|!}$, where λ is the event generation rate, T is the time span under consideration, e is the set of events, $|e|$ is the size of the set (number of events). The location of the event is uniformly distributed up to a maximum x_{max} . The time of the event is uniformly distributed between 0 and T .

Event Detections The arrival time a_{ij} of event i at station j is assigned by a Gaussian distribution with mean equal to $t_i + \frac{|x_i - x_{s_j}|}{v}$ where x_{s_j} is the station location and v is the velocity of the seismic wave ($v = 2$ in our experiments):

$$P(a_{ij} | x_i, t_i) = \mathcal{N}\left(t_i + \frac{|x_i - x_{s_j}|}{v}, \sigma^2\right)$$

The max travel time $\tau_{max} = x_{max}/v$ is defined as the time a seismic wave takes to travel from one end of the space to the other.

The signal generated by event i at station s_j lasts for a fixed duration t_s , covering the period $[a_{ij}, a_{ij} + t_s]$, and is sampled to be iid Gaussian, with larger variance than the background noise process.

Observed Signals The signal observed at each station is the sum of signals from arriving events, added together at the appropriate time offsets along with iid Gaussian background noise. Since each event signal is Gaussian, as is the

background noise, the observed signal is itself Gaussian conditioned on the arrival times a .

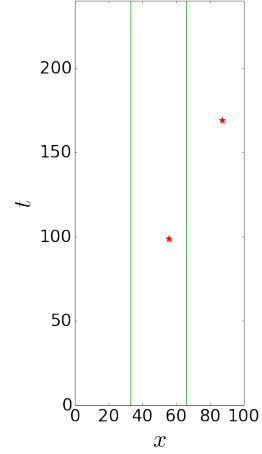


Figure 1: Example of two events located in space-time. Event e_1 can be described by $t_1 = 169$, $x_1 = 87$. Events e_2 can be described by $t_2 = 99$, $x_2 = 56$. Locations of detecting stations are shown by vertical lines, $x = 0, 33, 66$, and 100 .

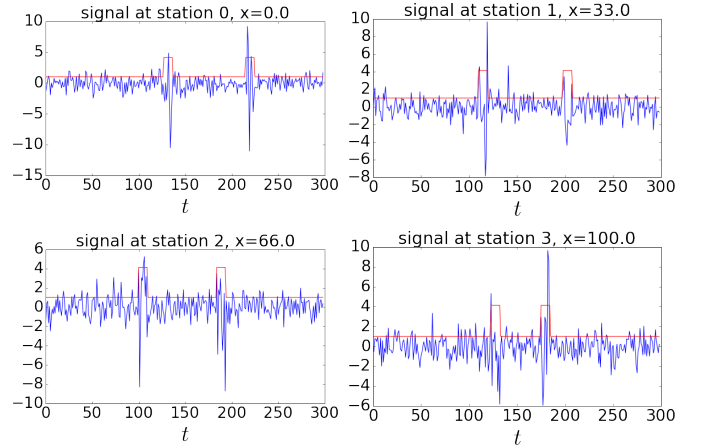


Figure 2: Signals (blue) generated by events in Figure 1, with arrival times dependent on the event–station distance. Red indicates the marginal standard deviation.

The overall probability of any set of events e and signals s can be written as

$$P(e, a, s) = P(e) \left(\prod_j^{|s|} \left(\prod_i^{|e|} P(a_{ij} | e_i) \right) P(s_j | a, e) \right)$$

$$P(e) = P(|e|) \prod_{i=1}^{|e|} P(x_i) P(t_i)$$

where the signal model $P(s_j | e, a)$ is a zero-mean, diagonal Gaussian density with variance at each timestep given by

the sum of variances from the noise process and any arriving signals. Figures 1 and 2 show a sample from this model, including space-time locations of two events and the signals they generate.

We use Metropolis Hastings to infer the posterior on events given observed signals. A serial inference algorithm for this model involves several types of MH proposals:

Birth and Death Moves: We propose birthing a new event from a uniform distribution, or killing an existing event.

Event Location Move: We propose moving a random event by a Gaussian offset in space and time.

Arrival Time Move: We propose changing the arrival time of a random event at a random station by a Gaussian offset.

Joint Events Move: The locations of nearby events are coupled by the observed signals, potentially including multiple joint modes from aliasing effects. We analyze the station geometry to jointly propose moving a pair of random events to another high probability mode (Figure 6).

Parallel Metropolis-Hastings Inference

Algorithm 2 Serial Metropolis Hastings

```

1: Input: world  $x$ 
2: for  $j$  in range( $k$ ) steps do
3:   pick a type of move, make proposal  $x^{cand}$ 
4:   accept move with probability:
        $\alpha(x^{cand} | x^{(j-1)}) =$ 
        $\min\{1, \frac{q(x^{(j-1)} | x^{cand}) \pi(x^{cand})}{q(x^{cand} | x^{(j-1)}) \pi(x^{(j-1)})}\}$ 
5:   if accepted then
6:      $x^{(j)} = x^{cand}$ 
7:   else
8:      $x^{(j)} = x^{(j-1)}$ 

```

Running inference on real seismic event signals serially can take a long time. In a more complicated model with real event signals from hundreds of stations all over the space, the inference process on a 24-hour period signals from roughly 100 stations will take weeks or even a month.

However, this model contains conditional independences that we can exploit for parallel inference. To a first approximation, the set of events occurring during a given time period $[t_1, t_2]$ can be inferred using only the signals from that time period. In fact it is also necessary to consider signals up to the maximum travel time τ_{max} following the time period in question, since an event at time t may not be detected until $t + \tau_{max}$. Since these additional signals may also include arrivals from events in the following period $[t, t + \tau_{max}]$, the posterior on events is coupled by the observed signals and does not decompose cleanly over adjacent time periods. However, the events in a given time period are independent of others *conditioned* on those occurring within the maximum travel time, i.e., conditioned on those events whose signals could directly “explain away” arrivals from the time period of interest.

Note that this independence is induced by the locations (x_i, t_i) of the inferred events, and is not present *a priori* in the factorization of the our model. Concretely, each timestep of signal depends *a priori* on every inferred event e_i , and in fact the exchangeability of events in our model introduces a labeling symmetry, so that the event variables e_i are strongly coupled by the observed signals. Therefore methods such as Chromatic Gibbs (Gonzalez et al., 2011) that exploit graphical model structure are not directly applicable.

Our contribution is to apply chromatic methods to an *induced* factor graph constructed from a spatial partition. We construct a factor graph by partitioning the continuous space and design parallel Metropolis Hastings algorithms that color these partitions and run inference on regions of the same color in parallel.

Partitioning the World

We divide the latent event space into regions corresponding to fixed time periods r^0, r^1, r^2, \dots . The region r^n describes a two dimensional space, with x ranging from 0 to x_{max} , t ranging from nl to $(n+1)l$ where l is the length of the region. We will also overload notation to let r^n refer to the set of events occurring in that region.

We considered partitioning by both space and time, instead of time alone; however, for this simple model there is no clear gain because the conditional independence is induced by the maximum travel time. A spatial partition would be natural if our model included attenuation, so that events are only detected by stations up to some maximum distance. In general it might be advantageous to consider more complex partitions across multiple latent dimensions, depending on the conditional independence structure of the model.

We also divide observed signals into time periods, written as s^1, s^2, \dots . The set of signals s^n describes signals collected by stations from $t = nl$ to $(n+1)l$.

When $l \geq \tau_{max}$, the Bayesian network of signals, regions is shown in the Figure 3.

Using the property of Markov blanket (Pearl, 1988), r^{n+2} are independent from r^n when conditioned on the Markov blanket of r^n , which contains r^{n+1} and s^n, s^{n+1} . Thus, we can bi-color the graph by assigning every two regions to the same color. With this coloring, regions of one color are mutually independent when conditioning on regions of other colors and the set of signals.

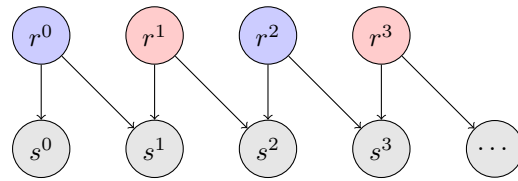


Figure 3: Factor graph when $l \geq \tau_{max}$, Bi-color

We could also decrease the region size to $\frac{1}{2}\tau_{max} \leq l < \tau_{max}$ and three-color the regions, but a two-coloring will always allow the most parallelism.

Naive Parallel Metropolis Hastings

As a simple baseline, we consider the Naive Parallel algorithm, which simply performs inference on all time periods in parallel, so that each block of signals s^n is explained by events r^n and also, independently, by events r^{n-1} occurring during the previous time period. As seen in Figure 4, this inference method breaks the factor graph and does not converge to the correct overall stationary distribution.

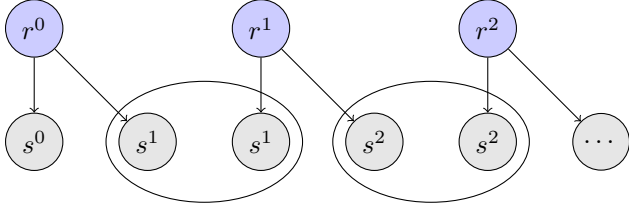


Figure 4: Factor graph of Naive Parallel Metropolis-Hastings

Algorithm 3 Naive Parallel Metropolis Hastings

- 1: Input: static world that are divided into regions of length τ_{max} , each region is written as x_m
 - 2: **for all** regions m **do in parallel**
 - 3: **for** j in range(k) steps **do**
 - 4: *pick a move randomly, make proposal* x_m^{cand}
 - 5: *accept move with probability:*

$$\alpha(x_m^{cand} | x_m^{(j-1)}) = \min\{1, \frac{q(x_m^{(j-1)} | x_m^{cand}) \pi(x_m^{cand})}{q(x_m^{cand} | x_m^{(j-1)}) \pi(x_m^{(j-1)})}\}$$
 - 6: **if accepted then**
 - 7: $x_m^{(j)} = x_m^{cand}$
 - 8: **else**
 - 9: $x_m^{(j)} = x_m^{(j-1)}$
-

For a world that has n regions, this approach can achieve up to n times speed up. However, because it imposes additional independence assumptions, it is likely to generate multiple events to explain a true event, especially when the true event is near the region boundary (Figure 7). To fix this problem, we design a Chromatic Metropolis Hastings algorithm.

Chromatic Metropolis Hastings

In our implementation, we assign l (region length) to τ_{max} and bi-color the world. The factor graph of this setting is in Figure 3. The actual coloring of this setting is in Figure 5.

Chromatic Metropolis Hastings with Static Coloring

The Chromatic Metropolis Hastings algorithm we designed is based on a static two colored world, which means the boundaries of each colored region and the order of region colors stay the same throughout the inference process. The method to color the world is as described above. Inference alternates between the two colors, running a sequence of MH updates on all red regions in parallel, then on all blue

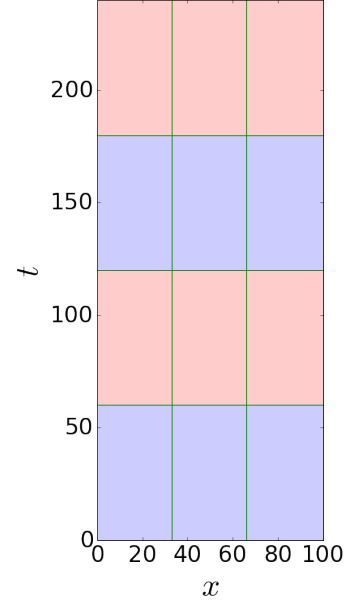


Figure 5: Coloring in our implementation, $l = \tau_{max}$

regions given the red regions, and so on. For a world with n regions, Chromatic Metropolis Hastings can get $\frac{n}{2}$ times speed up ideally.

Algorithm 4 Chromatic Metropolis Hastings with Static Coloring

- 1: Input: a two-colored world, initialize $X^{(0)}$ to be empty, $X^{(i)}$ will store region events after the completion of i^{th} full round inference on blue or red regions. $x^{(j)}$ stores temporary region events during a single round of inference.
 - 2: **while** $i < \text{max iterations}$ **do**
 - 3: **for** A in {blue, red} **do**
 - 4: **for all** regions m in color A **do in parallel**
 - 5: $x_m^{(0)} = X_m^{(i-1)}$
 - 6: **for** j in range(k) **do**
 - 7: *pick a move, make proposal* x_m^{cand}
 - 8: *accept move with probability:*

$$\alpha(x_m^{cand}, X_{\neg m}^{(-1)} | x_m^{(j-1)}, X_{\neg m}^{(-1)}) = \min\{1, \frac{q(x_m^{(j-1)}, X_{\neg m}^{(-1)} | x_m^{cand}, X_{\neg m}^{(-1)}) \pi(x_m^{cand}, X_{\neg m}^{(-1)})}{q(x_m^{cand}, X_{\neg m}^{(-1)} | x_m^{(j-1)}, X_{\neg m}^{(-1)}) \pi(x_m^{(j-1)}, X_{\neg m}^{(-1)})}\}$$
 - 9: **if accepted then**
 - 10: $x_m^{(j)} = x_m^{cand}$
 - 11: **else**
 - 12: $x_m^{(j)} = x_m^{(j-1)}$
 - 13: $X_m^{(i)} = x_m^{(k)}$
 - 14: $i = i + 1$
-

When doing inference using Chromatic Metropolis Hastings, all regions that run in parallel are independent from one another, conditioned on the contents of the remaining regions. Because birth and death moves allow for events to

be removed from one region and reborn in another, in principle, the joint chain is ergodic and has the correct stationary distribution. However, in practice the inability to move events across region boundaries leads to difficulties in mixing, especially for events whose posteriors straddle a boundary. And we are likely to wrongly infer near-boundary events to be in their neighbor regions. In addition, we cannot apply joint moves to pairs of nearby events separated by region boundaries, so they may get stuck in suboptimal modes. To fix these problems, we slightly modify the above algorithm and come up with the algorithm in the following section.

Chromatic Metropolis Hastings with Dynamic Coloring

After completing a round of inference on all blue and red regions, we select a random offset uniformly from $[0, l]$ and shift all region boundaries by this offset. We then run another round of inference moves, and repeat the procedure shifting the region boundaries after each epoch. This solves two major problems we have in Chromatic Metropolis Hastings with a static coloring—inferring near-boundary events and applying joint moves to pairs of events separated by region boundaries. In the below graph, let A', B' be the current event hypothesis and A, B the true events. With constant boundaries as in the graph, events A', B' are in different regions and cannot be jointly moved to the other high probability true state A, B . However, with dynamic coloring, the region boundaries change based on the random offset. Whenever A', B' are in the same region, the joint move can move them to the true state.

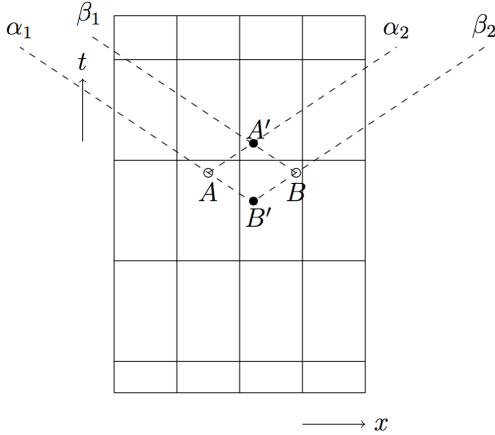


Figure 6: Local Maxima Mode and Another High Probability Mode

Experimental Results

We evaluate inference methods for our seismic model on synthetic data generated by sampling from the model’s distribution over possible worlds. We compute the accuracy of an inferred event hypothesis by comparing to the sampled ground truth. A bipartite matching is built between inferred and true events. We build the matching by looping through all true events and finding the closest inferred event to it. We

add an edge between them if they are less than 12 units apart in both time and distance. After an edge is added, the events on both sides of the edge are removed from the matching process, so that the degree of any event is at most 1. After the matching is built, we report three key quantities: location error (average distance between matched events), precision (percentage of inferred events that are matched) and recall (percentage of true events that are matched). It is desirable to have a method with low error, high precision and high recall, which means the inference is more accurate in terms of both location and number of events.

The setting of the world in our experiment is $x_{max} = 100, T = 240$. There are $n = 4$ regions in our experiment. $\tau_{max} < \frac{T}{n}$, thus allows bi-coloring. For Chromatic methods, we run 500 steps before switching to the other color. Naive parallel Metropolis Hastings algorithm can achieve a maximum 4 times speed-up. Chromatic Metropolis Hastings algorithm can achieve a maximum 2 times speed-up. The high-level distributed computing is built on PySpark. All the experiments are run on a quad-core computer.

Figure 8 shows location error, precision, recall, log probability versus time metrics for each algorithm. These metrics are generated based on the average result from 5 random worlds. From the metrics, we see that Chromatic Dynamic, Chromatic Static and Serial are achieving similar overall final performance but Serial converges slower than the other two methods. Naive Parallel converges fastest but it has the lowest precision and recall. The low error of Naive Parallel is also an artifact of its low recall (generates more events than actual). In the metrics, Chromatic Dynamic has a slightly better precision and recall than Serial algorithm because of random variation. This motivates us to do confidence interval analysis of the three methods on more random worlds.

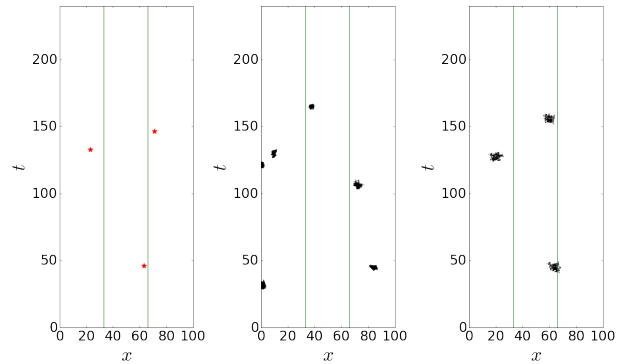


Figure 7: Red stars in the left sub-figure are actual events. The middle sub-figure shows 1000 samples of Naive Parallel Metropolis-Hastings. The right sub-figure shows 1000 samples of Chromatic Dynamic Metropolis-Hastings.

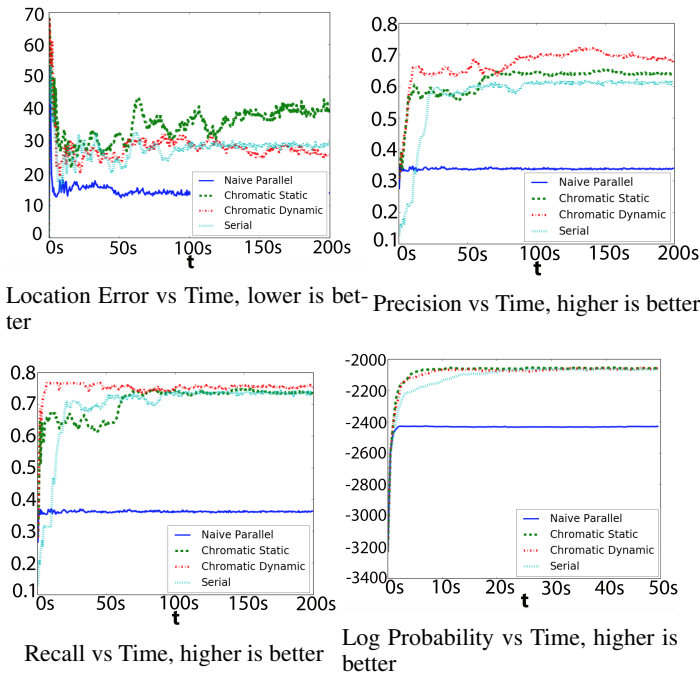


Figure 8

Using 50 possible worlds sampled from the model, we run each algorithm 5 times and measure the average precision, recall and error of our inference via the assumed ground truth. From metrics above, we can see that at 200 seconds, all algorithms have safely converged. So in this experiment, we run longer than 200 seconds for each algorithm to show the final performance after convergence. After we get 50 sets of results, we calculate the mean location error, mean precision and mean recall for each algorithm and generate error bars around the mean using 95% bootstrap confidence intervals (Efron and Tibshirani, 1993). See Figure 9. From the bar graph, we see that dynamic partitioning yields precision and recall statistically indistinguishable from serial MCMC, and significantly better than naïve parallelization.

Conclusion and Future Work

We have introduced a novel approach for parallel MCMC inference exploiting value-dependent conditional independence induced by spatial structure rather than a fixed graphical model. Evaluating on a simple model of seismic events and signals, we find that the Chromatic Metropolis Hastings using Dynamic Coloring can achieve $\frac{n}{k}$ times speedup while still maintaining a similar level of error, precision and recall as Serial Metropolis Hastings, where n is the number of spatial partitions and k the number of colors.

Future work involves extending this approach to production-scale models such as that of Arora et al. (2010), and to other applications involving spatial object detection and/or localization. Also of interest would be methods for automatically detecting spatial independence relationships of the form we exploit, given formal descriptions of model

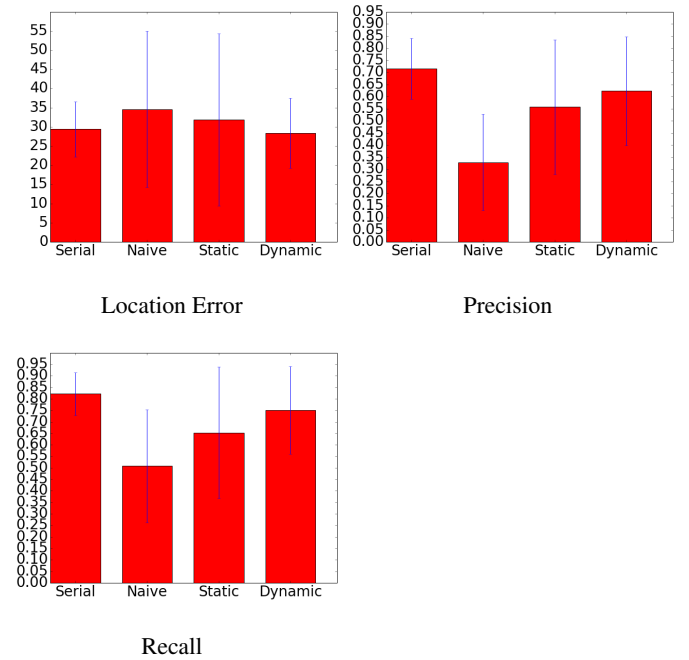


Figure 9: Bootstrap Confidence Interval of Location Error, Precision and Recall

structure, and selecting an appropriate partition of the latent space to maximize inference efficiency.

References

- Andrieu, C.; de Freitas, N.; Doucet, A.; and Jordan, M. I. 2003. An introduction to mcmc for machine learning. *Machine Learning* 50:5–43.
- Arora, N.; Russell, S. J.; Kidwell, P.; and Sudderth, E. B. 2010. Global seismic monitoring as probabilistic inference. In Lafferty, J. D.; Williams, C. K. I.; Shawe-Taylor, J.; Zemel, R. S.; and Culotta, A., eds., *Advances in Neural Information Processing Systems 23*. Curran Associates, Inc. 73–81.
- Efron, B., and Tibshirani, R. 1993. An Introduction to the Bootstrap.
- Gonzalez, J.; Low, Y.; Gretton, A.; and Guestrin, C. 2011. Parallel gibbs sampling: From colored fields to thin junction trees. In *Artificial Intelligence and Statistics (AISTATS)*.
- Pearl, J. 1988. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc.